Poler Vision: Uplifting the Pole Sport Community with Artificial Intelligence

Oakleigh Weekes
WEE18678532

18678532@lincoln.ac.uk

School of Computer Science
College of Science
University of Lincoln

Submitted in partial fulfilment of the requirements for the
Degree of BSc(Hons) Computer Science

Supervisor: Dr James Brown

May 2022

Acknowledgements

This project would not have been possible without the cooperation of some wonderful members of the poling community, which via social media sites and in person, supplied relevant imagery necessary for the training of neural network models. I wish to especially thank Odette Ross of 'Polez Apart Fitness', Boston, Lincolnshire for allowing dedicated time per session for capturing pole images. I would also like to thank Bernice Bradley for actively volunteering for move demonstration every week. I appreciate every participant who has taken part in image capture in these classes, as these photos formed much of my dataset. I would like to show gratitude for the work of Dr Carolina Are, who has researched the under-explored niche of the censorship of pole dancers and women on social platforms. I welcome Dr Sandra Avila of the University of Campinas, Brazil for supplying me with the NPDI pornography dataset which would allow for the training of a censorship algorithm. Finally, I would like to acknowledge the aid of Dr James Brown, my supervisor, for supporting a project that could have faced stigma amongst conservative views.

Abstract

Pole (Sport/ Dance/ Fitness) is a modern blend of art and fitness, consisting of choreographed lifts and spins centred around a chrome, steel, brass or powder-coated vertical pole. It could make use of more technology to aid the community, being a vibrant collection of instructors, performers, and fitness enthusiasts. This project comprises of literature review, created solutions, and discussion for possible applications of artificial intelligence in pole sport. Included is an exploration into a pole-move recognition solution, a more inclusive censorship algorithm, and pole injury prevention. These could solve key problems in the community such as minimal communication around the hobby, stigma, and some serious accidents, respectively. It was found that pose recognition methods are an effective way to classify staple pole sport moves using the K-Nearest-Neighbour algorithm. Using binary classification methods, it was concluded that censorship algorithms need to be inclusive of pole sport media to avoid their unfair suppression, and certain pole dance expression is essentially non-pornographic. Using a blend of image processing, object detection and pose recognition it may be possible to create an intelligent injury prevention solution. However, the creation of this artefact requires further user study, and will need to incorporate the expertise of experienced pole dancers and medical professionals, whilst prioritising safety in development and use.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Pole Sport: A brief Overview

Pole, also known as 'pole dance', 'pole fitness' or 'pole sport', is a form of aerial and floor art consisting of choreographed movements along with strength and flexible poses centred around a spinning or static vertical pole. Very few studies have been made surrounding the sport, but with 11.2 million posts tagged with '#poledance' on Instagram (2022), and a wealth of located studios on the 'PolePedia' (2020) map, it is evident that pole dancers are an ever-growing population.

The origins of pole-related dance can be traced back to the 12th century. 'Mallakhamb' is an ancient Indian sport that has variations such as pole, rope, hanging, cane, and floating platform with the first three being performed competitively. Pole mallakhamb concentrates around a thick wooden pole, with sharp, agile movements akin to gymnastics and martial arts, and unlike modern pole sport, had a mostly male uptake (Sharma, Choubey 2016). Following traditions, Chinese pole is a sport that also carries through to present day, consisting of multiple poles up to twenty feet high. These are usually silicone or powder coated as performers are usually clothed, thereby needing a more adhesive surface (Pole Fit Freedom, 2019). Again, men seemed to dominate the performing percentage until present years.

It is peculiar how such traditional performances which celebrated strength and the human form, transformed into a form of erotica. For a time, pole was affiliated to a subculture in which men entered clubs with the intention of viewing women display their bodies to potentially evoke arousal (Whitehead, Kurz 2009). It can be argued whether women in these facilities are trained to perform the likes of traditional pole movements by mounting the pole or more so conform to the likes of provocative erotic dance.

Modern pole acknowledges the history of the artform, and usually incorporates elements of all previous counterparts, with techniques such as Chinese climb and body flow incorporated into choreography depending on style. Modern pole can be seen as fitness (Nicholas et al. 2019), has connotations of empowerment, and with the inclusivity of all genders and body types, is becoming a possible gateway towards body confidence and positivity. It is strange, however, that the potential of the current sport becoming an Olympic one splits opinion both inside and outside the pole community, becoming a separate debate (Weaving 2020).

Figure 1 The development of pole-related performance.

Pole Mallakhamb (Kaplish 2018); Chinese Pole (Wikipedia Commons 2005); The strip club perception (Dombrowski 2011); Modern Pole.

1.2 Solving Problems with Computer Vision

In recent years, pole has seen mobile application aid the community, such as ‘Not the Mainstream’s (2022) ‘Polelearn’ and ‘Pole Dance Companion’, which supply movement guides, paid and unpaid, and have over 11,000 combined downloads on the Google Play store. This project aims to both expand pole technology and the research domain, focusing on how computer vision solutions could improve the quality-of-life of a pole dancer.

Pole is a growing artform gaining popularity due to more progressive viewpoints (Fennel 2020), and as such, the language is not as easily accessible to beginners. To search for moves and poses online, polers must know the name of them. Some moves have multiple names, and some are recently choreographed. To unify the space between knowing the pole move and being able to search for tutorials for personal improvement, the first part of this research involves creating a move recognition solution. For what ‘Shazam’ (2022) is to music, and ‘Google Lens’ (2022) is for your environment, the ‘Point your Pose’ application aims to learn how body landmark positions pertain to certain pole moves and can return this information live to the user.

Figure 2 Google Lens being used to identify flower species

(Martinez 2020).

Polers have a complicated relationship with social media algorithms, as user’s pole progression posts are often removed from platforms as deemed “inappropriate content”, or fall victim to the ‘Shadowban’, thereby not being shown or discoverable on those applications (Are, 2021). It is agreed that garments which show more skin allow for better pole grip (Baldin, Menegucci 2017), and so for some difficult moves, covering up is not an option. This project proposes a computer vision model which can generalize the censorship of explicit content without bias against pole dance imagery and video.

Pole can be enjoyed in a relatively safe environment if certain conditions are put in place. However, injuries both minor and major can potentially occur due to the risk of falls from height (Dittrich et al. 2020). The final part of this project is an exploratory discussion on the use of computer vision-based anomaly detection techniques and how they could be applied to pole sport to help prevent injury.

Chapter 2

Literature Review

2.1 Applied Pose Detection & Recognition

Pose recognition has been previously carried out on recreational activities aside from pole dance. Dohyung et al. (2017) proposed a method for classifying Korean pop dances from skeletal joint data pinpointed using a Kinect. Classification of moves has had similar success in that of folkloric dance sequences (Protopapadakis et al. 2018), aeta dance (a Filipino cultural dance) (Mindoro, Festijo, de Guz-man 2021), and is known for improving yoga stances (Thar, Winn, Funabiki 2017).

Pose recognition incorporates two tasks: detection of the human and their features, and classification of their stance. A classic form of capturing data to train the former is through using motion capture. A method proved successful is the use of spatio-temporal templates, which consist of several silhouettes computed at different instances which correspond to a particular view, taken by many cameras surrounding a model (Dimitrijevic, Lepetit, Fua 2006). Multi-silhouette templates are then corresponded against portions of the input sequence, using "chamfer matching", the process of matching objects to their templates via their boundaries (Davies E.R. 2018).

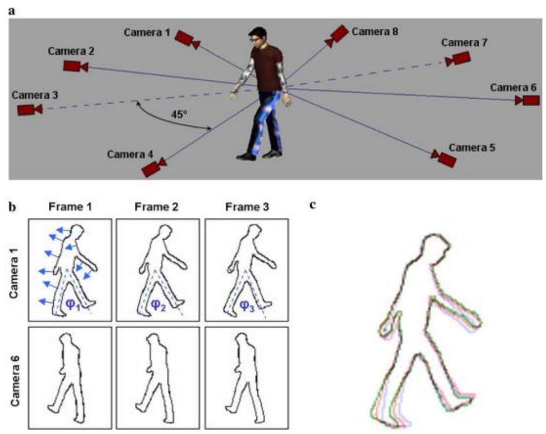


Figure 3 Creating spatio-temporal templates (Dimitrijevic, Lepetit, Fua 2006, 131).

Dimitrijevic et al.'s method of collecting training data can be mimicked by capturing images for each pole move directional angle, by taking 360-degree videos of the poler and slicing them into relevant frames.

BlazePose is described as "a lightweight convolutional neural network architecture for human pose estimation that is tailed for real-time inference on mobile devices" (Bazarevsky et al. 2020, 1). It uses an encoder-decoder type network architecture to predict heatmaps for all joints of one individual, followed by another encoder which regresses to the coordinates of all joints. It works on the basis that the person's face is always somewhat visible, as a face detector is used as a proxy for the person detector. The topology of 33 landmarks found is a superset of the COCO (Lin et al. 2015) topology, allowing it to be consistent with its relevant datasets and inference networks. BlazePose is ideal for a pole-move recognition solution due to being lightweight in nature and being designed for mobile applications.

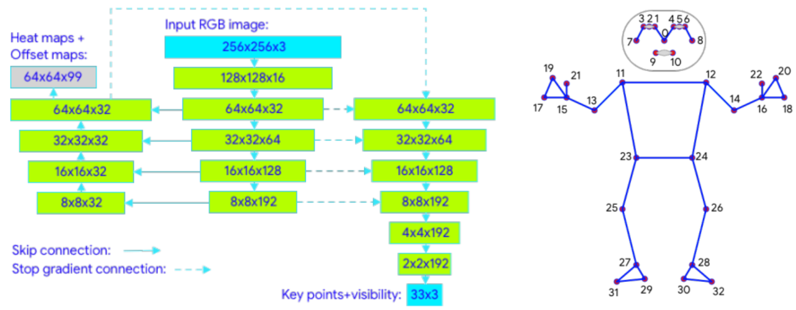


Figure 4 BlazePose Network Architecture (left) and topology of landmarks (right) (Bazarevsky et al. 2020, 2)

The second task in pose recognition is classification. In the case of BlazePose, a csv file can be returned with each landmark's x and y location (with 3 degrees of freedom) in a 2-D plane, and its level of visibility. These features, or a derived form (e.g., pair-wise distances) can be used to train a model such as Random Forest (Rogez et al. 2008) or K-Nearest-Neighbour (Google Developers 2022), to recognise future similar poses.

2.2 Social Media & The Poler

It is difficult to understand how the "black box" image and video censorship algorithms of social media platforms work. According to Instagram's Community Guidelines (2022), they do not allow nudity on their platform, including some "digitally-created" content that shows sexual intercourse, genitals, close-ups of fully nude buttocks and female nipples, except for the context of breast-feeding, birth-giving, after-birth moments, and health-related situations. They also allow for nudity in photos of paintings and sculptures. Are (2021) explores this under-researched area by examining how the censorship algorithm is still inadequate in its configuration, preventing the likes of freelancers, artists and for the most part women from reaching new audiences on their Instagram platform. This has resulted in backlash that has sparked campaign (EveryBody Visible 2019).

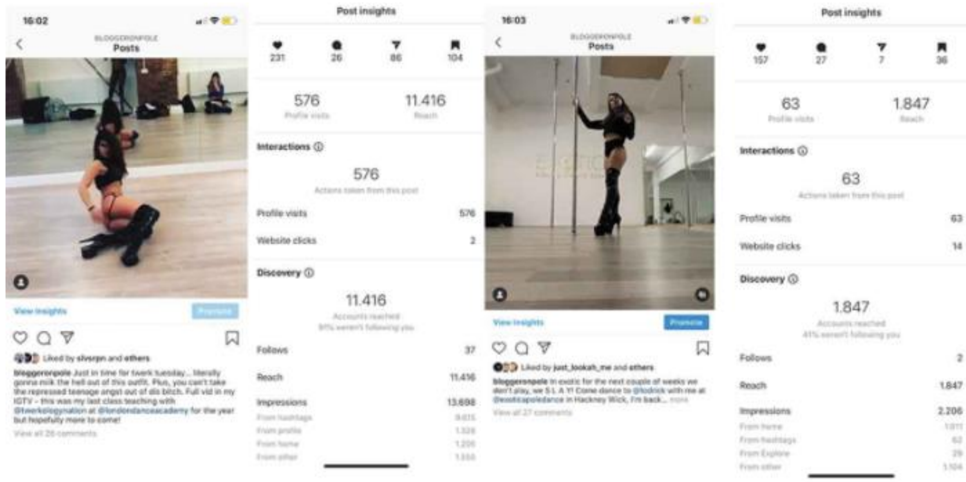


Figure 5 Are's (2021) Instagram account's decline in engagement, 2018 versus 2020.

Figure 6 The 'Shadowban' Cycle (Are 2021)

Social Media has accepted that content moderation is required “to protect one user from another, or one group from its antagonists, and to remove the offensive, vile, or illegal – as well as to present [platforms’] best face to new users, to their advertisers and partners, and to the public at large” (Gillespie 2018, 5). Due to the ever-scaling platforms which process huge volumes of information, and the complexity of the human language (such as sarcasm or jest), platforms have moved away from using many human content moderators to focus on an algorithmic artificial-intelligence approach (Cobbe 2021).

The use of machine learning, such as neural networks, has proven successful in content moderation, and is mostly trained on “to-be-censored” data generated by users on different sites and shared with other social networks (Cambridge Consultants, 2019). However, Are’s (2021) work suggests that this technique does not generalise well to imagery and video correlated to pole dance where movement may resemble erotic behaviour, and more skin is shown to aid grip (Baldin, Menegucci 2017). The correlation with pole and skin visibility means that base nudity cannot be used as a feasible method of censorship, rendering algorithms such as Ap-apid’s (2005) nudity detection based on skin-coloured pixels obsolete in this case.

In attempting to create a more inclusive image & video moderation solution, a pornography classification system will be created which will be assessed for its accuracy on Pole Sport media. Avila et al.’s (2013) pornography database has had a great impact in similar fields of research and so acts as an efficient dataset source to classify pole media against.

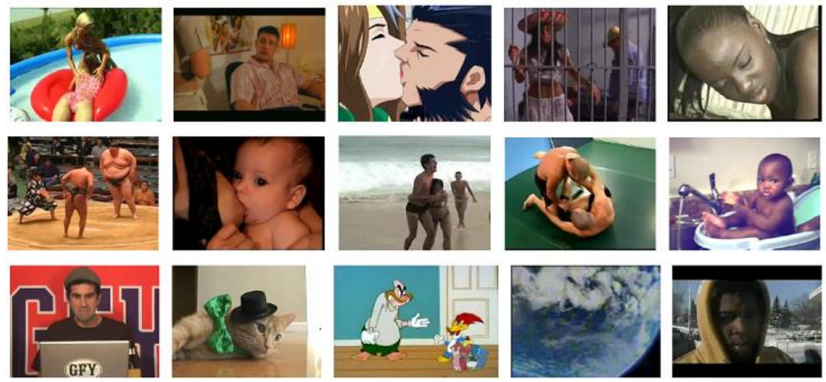


Figure 7 Sample of Avila et al. (2013) Pornography Dataset, including neutral "easy" [bottom row] and "difficult" (for a classifier to learn) images [middle row], and pornographic images (frames from video) [top row].

Previous pornography classification methods have included BOSSA (Bag of Statistical Sampling Analysis), which is an image representation technique that extends the "Bag of Words" approach, keeping more information during the pooling step of summarizing vectorial information in imagery (Avila et al. 2011). Deep learning classifiers have also been used such as "Ensemble-ConvNet", a fusion of scores from both Anet and GNet, which tends to outperform BossaNova (Moustafa 2015). More recent studies such as Wehrmann et al. (2018) use a CNN as a feature extractor ("representation learning") and a long short-term memory (LSTM) to perform video classification, known collectively as the ACORDE (Adult Content Recognition with Deep Neural Networks) method, and reaches classification accuracies of up to 95.3%.

Table 1 Sample of results of Wehrmann et al. (2018) Video Classification with ACORDE using Avila et al. (2013) pornography database, compared with other classification methods.

Method	Accuracy (%)	AU(ROC)
BossaNova-BRISK (Caetano et al. 2014)	88.6 ± 2	0.960
AGbNet	94.1 ± 2	–
ACORDE-152	95.3 ± 1	0.990

2.3 Cushioning the Fall

Both pole training and performance come with the risk of serious injury, including damage to the head, neck, and spine, due to possible falls from heights usually of up to three metres (Dittrich et al. 2020). Reports have found that over 85% of a sample of female polers have admitted to sustaining some sort of injury during their career, which also includes that to the wrists and shoulders, possibly due to moves such as the cupid, handspring, and extended butterfly, which may lead to overloading the musculoskeletal system (Szopa et al. 2022).

Although preventative methods exist for injury prevention, such as taking part in warm-ups (Naczk, Kowalewska, Naczk 2020), placing floor mats (Dittrich et al. 2020), and gaining prior experience (Lee, Lin, Tan 2020), it would be interesting to explore whether Computer Vision techniques could help prevent accidents in real-time.

Figure 8 A typical pole "crash mat".

Even though the use of a monitoring device is deemed less favourable, object detection has been used to spot hazards to prevent falls in the elderly (Gruszczyński, Stefańczyk 2021). A monitoring device may be more accepted in an environment where it exists for the sole duration of a pole lesson or practice. Anomaly detection could be tuned to recognise stance and the local environment surrounding the pole, such as the way Fang et al. (2018) uses an object detector SSD and CNN classifier to judge whether a steeplejack has been made adequately safe in construction environments, to prevent falls from height. This could possibly be tuned to recognise unsafe positions on the pole and alert the sole instructor to aid, as they usually juggle surveying multiple students in a lesson. This could possibly be tuned for sensitivity depending on the poler's experience level, as the risk of injury is lower for the experienced in certain moves (Lee, Lin, Tan 2020).

Chapter 3

Methodology

3.1 Project Management

Project management mainly consisted of meetings and discussions between Oakleigh Weekes (the project developer/ researcher) and supervisor Dr James Brown. Meetings during development mostly took place via ‘Microsoft Teams’ (Microsoft 2022). These were scheduled weekly whilst development took place to discuss project direction. These occurred irregularly once artefacts for the first two sections of the project were complete.

What was at first a software development-based project quickly transitioned into a predominantly research-based one. As per the Gantt chart (**APPENDIX A**), the project originally aimed to create a rounded pole move recognition solution, i.e., fully developed ‘Point your Pose’ application ready for release. However, it was discovered that toolsets could be used to create and test the solution quickly, as they worked unexpectedly well. Furthermore, it was found that there are very few research materials available on pole sport, let alone technology surrounding it. This fuelled the project to take a more research-based approach, aiming to provide research on technologies that could positively impact pole sport in a holistic fashion.

On this discovery, the project evolved iteratively, akin to agile software development, where new requirements were constantly being established to further improvement of the quality-of-life of polers as much as possible through research means. The aims of the project became that to become reproducible, in the hopes that further research work and product development will arise on its conclusion.

Both prepared for and unseen risks did arise, but these were mostly handled by following the contingency plan (see **APPENDIX B** for full table).

Table 2 Contingency plan sections 1 & 4 involving risks that were met.

Risk no.	**Risk Details**	**Likelihood**	**Assessed Impact**	**Mitigation Action**
1	External pole studios become un-cooperative with image collection	Medium	Medium	Ensure many studios and communities are contacted. Increase reliance on local studio's image collection
4	Google Colab becomes dysfunctional	Low	High	Migrate project to Jupyter Notebooks on personal computer.

External pole studios and polers contacted seemed unresponsive to pole move image donation (risk 1). Mitigation action was followed, increasing reliance on the local studio to provide imagery. This did not have an impact on data quality for the first part of the project (what this plan was concerned with), however, diversity was forfeited which could influence the censorship CNN model (**Chapter 5**).

Google Colab became somewhat dysfunctional, as per risk 4. Due to reliance on GPUs for model training, the software halted their use. This prompted a personal payment of £8.10. This was in replacement of following the mitigation action (migrating to Jupyter Notebooks). Migrating the data locally and installing libraries, configurations and dependencies to Jupyter Notebooks would have taken time. The model configuration was nearing its end, so it became wise to instead forfeit a low payment for the sake of an earlier conclusion. Risk 4 was also forged with the possibility that Google Colab became completely dysfunctional, not that it may be met by a temporary unexpected paywall.

3.2 Toolsets and Machine Environments

The toolsets and environments used will be advocated for the ‘point your pose’ and ‘inclusive censorship’ elements of this project. Pole injury prevention is a theoretical exploration of requirements for a solution.

Python (v3.6.9, 2022) is the primary language used for the creation of artefacts for this project. It comes with a wealth of tools and libraries for both machine learning and computer vision use cases. It allows for higher level configuration of models and easy pre-processing of directories, data, and images. Important libraries include PIL (v7.1.2), Pillow (v8.1.0, 2022), OpenCV (v4.5.1.48, 2022), Matplotlib (v3.3.4, 2022), NumPy (v1.19.3, 2022) and MediaPipe (v0.8.3, 2022). Matplotlib v3.1.0 was used for analysing pose detection outliers due to dependencies.

Google Colab (2022) was the notebook-style python IDE for running pose detection, pose classification experiments, censorship algorithm training and testing. It allowed access to efficient GPUs to speed up training of models, and most libraries are pre-configured and pre-installed. It allowed mounting Google Drive to pull in necessary data. Unfortunately, the caveats included time-outs for long training periods, and GPU usage was halted entirely due to heavy use. An upgrade to Colab Pro was needed for a month to finish development. Deletion of unused variables was required to save available memory before model training.

MediaPipe (2022) is a selection of cross-platform customisable toolsets for live and streaming media. It covers the likes of face detection, object detection, and pose detection (‘BlazePose’), which is this project’s focus. Testing its efficiency on detecting pole moves is simple, as a template Google Colab template is provided by the developers to kickstart development. It returns a concise CSV file of pole move features that can be used in a machine learning model to classify poses. NumPy was used to manipulate data matrices, with Pillow, OpenCV and Matplotlib used interchangeably for their own benefits in image display.

Table 3 Sample from BlazePose output csv showing the first landmark’s coordinates and visibility for test images.

Image	Class	X	Y	Visibility(Z)
supermanTest8.jpg	superman	636.3015	1033.743	-979.199
supermanTest9.jpg	superman	500.1328	852.4138	-903.591
butterflyTest0.jpg	butterfly	726.3889	993.9302	-35.3989
butterflyTest1.jpg	butterfly	821.7746	1144.244	-265.843

Scikit-learn (2022) is an accessible, open-source python machine learning library with efficient tools for data analysis. It is used for comparing for the best model to classify pole moves given BlazePose’s output landmark features. The Scikit modules used in this experiment cover k-nearest-neighbour, random forest, support vector machine, logistic regression, and artificial neural network (multilayer perceptron). It is also used for metric calculations such as accuracy and AUC (‘area under curve’). Keras (v2.8.0, 2022) is used to test whether a basic custom convolutional neural network can classify these moves using the whole images as opposed to the derived landmarks.

Android Studio (2022) was used as the IDE running on a local desktop PC for creating the sample application for showing ‘Point your Pose’ in action. It supports MLKit (2022), a mobile development machine learning package built by Google developers. It supports both Android and iOS devices. MLKit is written in both Java and Kotlin. The Java version was chosen for development due to it being more established with a larger development community. If the application were to be developed further outside of this project however, Kotlin may be favoured due its lightweight nature and more compact code (Ardito et al. 2020).

Simple black box testing for functionality occurred on a simulated Google Pixel 4 API 30 smartphone running Android 11.0, x86 CPU and disk size of 10GB. The built sample application was hosted on this through the functionality of Android Studio and Gradle (2022). Test images were run through the simulated application using MLKit’s ‘StillImageActivity’ to classify new pole move images, brought onto the phone through Google Drive. Testing live pole move footage was carried out through downloading the built application through the IDE as a .apk file. This was installed on a Huawei P30 Pro model VOG-L09 running android version 10 with 8GB RAM. Through selecting MLKit ‘LivePreviewActivity’, the moves could be classified in real-time in classification mode.

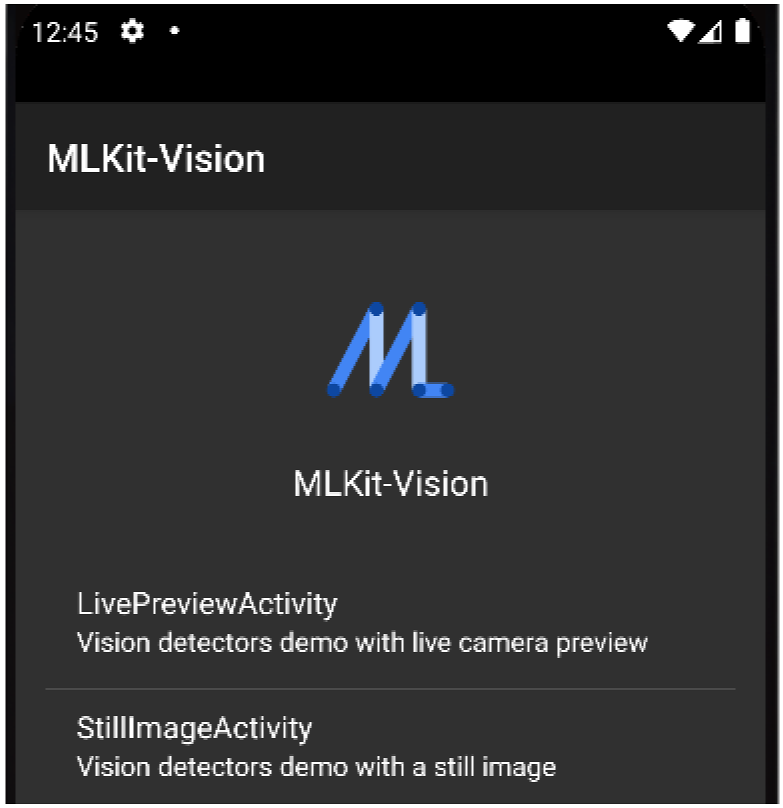


Figure 9 *MLKit template application running on a simulated Google Pixel 4 smartphone showing options used for testing sample application.*

Censorship model analysis shared many of the libraries used in pose detection and classification experimentation. Python's OS library was relied on for organising directory data splits. Keras was used yet again for model definition, bringing in the InceptionV3 (Szegedy et al. 2015) structure. A newer CNN model structure could have been implemented, however, the aim of this part of the project is to examine how easy it is to differentiate pole-related media from explicit media using neural network methods. It is also unknown which model structure modern social media platforms use to flag image content. Pandas (v1.3.5, 2022) and Seaborn (v0.11.2, 2022) python libraries worked together for verbose display of confusion matrices. OpenCV's 'vidcap' module was used for pulling apart video media for censorship classification of individual frames.

Chapter 4

Point your Pose

4.1 Research Methods

Is pose detection a suitable form of smart move recognition for pole sport? This study involves experimentation using acquired body landmarks as features and training a machine learning solution to test if it can accurately predict moves in unseen footage. Different machine learning models are compared and the best one is used in a solution which strives to answer the question.

The Dataset

This research required the use of a pole sport dataset which was gathered for this purpose. 362 images were collected, pertaining to three distinct pole dance moves; 142 ‘butterfly’, 116 ‘scissor-sit’ and 104 ‘superman’ images. 15 ,13 and 12 of these images respectively were donated from external sources (including some aesthetically filtered images). The rest were collected from a local studio by taking a subset of frames from 7 butterfly, 8 scissor-sit and 6 superman videos. These videos were taken by rotating around the poler in a 360-turn perpendicular to the pole. 10 ‘out-of-sample’ test videos were taken on separate days to the training data, including a mix of the three moves: 4 ‘butterfly’, 3 ‘scissor-sit’ and 3 ‘superman’. Test videos were converted into 31, 29 and 15 respective frames (using manual video ‘snapshotting’ via a phone camera’s functionality).

Figure 10 Images of a 'scissor-sit', 'butterfly' and 'superman' pose.

Figure 11 'Superman' pose frames at various angles from the same video.

Where diversity in images lacked in locally sourced data (taken at a familiar studio), variety was increased through different volunteers and outfits. However, white women made up proportionally 80% of the dataset. It is believed that this will not affect the accuracy of this artefact which is based purely off body landmark data.

Outlier Removal

MediaPipe's BlazePose Google Colab notebook creates a landmark features dataset but also performs a validation to remove outliers. Samples are removed if they are significantly different from others in the dataset (have an increased landmark maximum distance) and are found by using the K-nearest-neighbour algorithm. After outlier removal, the training set includes 129 'butterfly' samples, '90' scissor-sit samples and 91 'superman' samples.

Figure 12 Samples removed from pose training dataset due to BlazePose K-NN validation.

As can be seen from the images of the samples removed (Fig 12), BlazePose finds it difficult to determine landmarks where the face is not at least in partial view. This could be due to the fact it uses a face-detector as a proxy (Bazarevsky et al. 2020). 'Variations' of moves also occur in pole, whereby signature moves are slightly changed based on alterations such as slight leg or arm position. Other factors could be BlazePose viewing people in the studio mirror, or body not in full view.

Figure 13 Cropped sample outliers where people are in a background mirror, and full body is not in view.

Outlier removal resulted in the number of samples used in the following table:

Table 4 The numbers of images captured, and samples used for training and testing after outlier removal.

Pole Pose	**Training images including outliers**	**Training Samples**	**Testing images including outliers**	**Testing Samples**
Butterfly	133	129	31	31
Scissor-sit	107	90	29	29
Superman	98	91	15	12

The 310 training samples and 72 test samples were brought into k-nearest-neighbour, random forest, support vector machine, logistic regression, and artificial neural network (multilayer perceptron) classifiers to analyse for the most accurate.

Data Splitting and Evaluation Metrics

The 310 samples were originally split further into 70% for training and 30% for testing. It was noticed that these test samples could give rise to unjust accuracy readings due to a form of data snooping. As samples are taken randomly from frames in a video, a subsequent frame from a video frame in the 'training' set could leak into the 'test' set. This led to further analysis on the 72 independent 'out of sample' test samples.

Accuracy and confusion matrices were used to compare and visualise model efficiency, whilst also used for the censorship classification, and are discussed further in that section (5.1). The model with the highest accuracy result was to be brought into the classifier mobile application.

4.2 Design, Development and Evaluation

Dataset Collection & Ethics

A dataset source was needed for training the application model. Due to pole dance being a niche interest, there were no openly available databases under a creative commons license. The application required an image collection of staple pole moves; the 'butterfly', 'scissor-sit' and 'superman', which also narrowed the searching space.

GDPR regulation (ICO 2018) was followed, along with the data ethics framework (CDDO 2020). It was made known that images will be stored in Google Drive, and as such, the 'Google Cloud Privacy Notice' (Google 2022) applies to the dataset.

Two consent forms were made, one for online submissions, and the other for in-person data collection (**APPENDIX C**). An information sheet was drafted to explain image usage (**APPENDIX D**). The online version was created via Google Forms (2022). It included a consent and information section that required to be acknowledged before submission (**APPENDIX E**). The submission section included a separate section for each 3 moves, with a respective example picture demonstrate the pose (**APPENDIX F**). There is a known limitation that under the 'free' Google Forms tier, a maximum of 15GB of image responses can be stored according to one form.

An advertisement was created to include an example prototype video clip and links to both the project proposal and the consent form. The simulated prototype was a simple pose digital image embossed with a digitally-drawn skeleton mesh, with a classification pop-up (**fig 14**). Initially there was a callout for images in two Facebook groups. The first was "You know You're A Poler When…" which is a popular private community of 33,500 members. Unfortunately, the post only garnered 22 likes and 8 comments, which was deemed a low number of impressions. The other was "Everything POLE and Aerial", a private group of 19,000 members which also encompass 'hoop' and 'silk' aerial artists. This group's post attracted more engagement with receiving 40 likes and 4 comments, which was peculiar considering the lower member count. The difference in reach is difficult to explain due to the mystery of private social media algorithms and would require deeper knowledge of the game theory of marketing tactics. Over the two posts, little responses resulted in 40 total images collected. After finishing the prototype artefact using all the data, a follow-up post was made with gained 99 likes and 33 comments, which could possibly aid in artefact improvement. The remaining (majority of) data was collected from the local pole studio.

Figure 14 Sample video animation (frame) shown on callout for pole imagery in Facebook groups.

For in-person video collection, participants were given a paper consent form to sign and be collected, and a copy for them to keep. The information sheet was also to be included. Oakleigh, the researcher, was at hand to answer questions. For both online and in-person submissions an email was given in case of further questions or a wish to withdraw images from the dataset. Participants could also wish to receive the findings of these related studies (via email for online submissions). The use of the images remained transparent to the participants, and further consent was obtained if they wished for their images to be included in this publication.

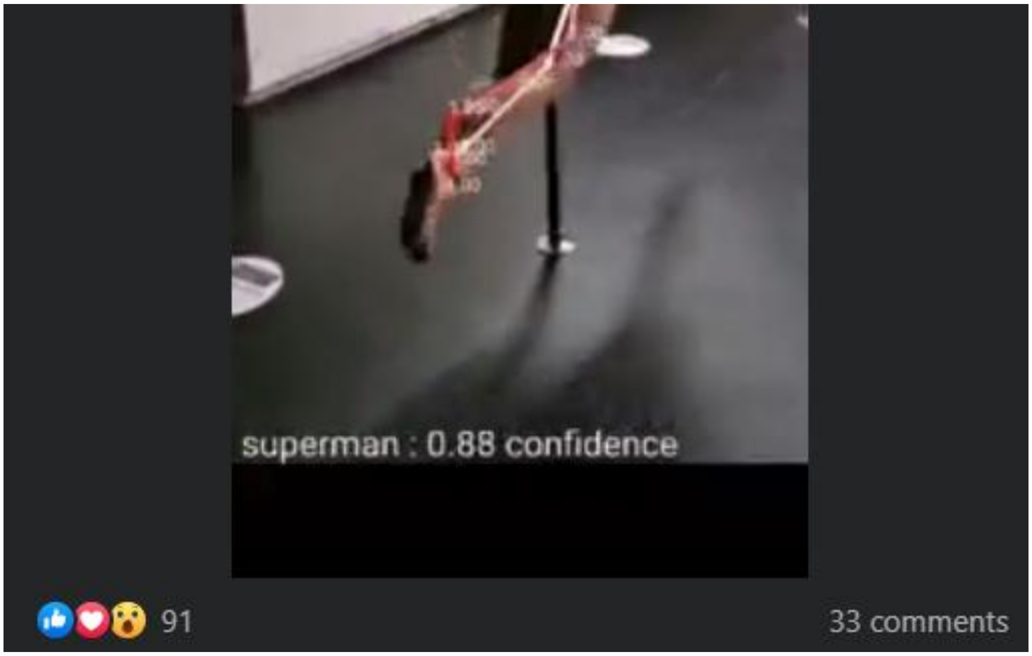


Figure 15 Follow-up 'artefact' group post showing higher interaction from polers.

Training Comparison Models

Blazepose was used to convert the images to a set of landmarks (**table 3**). These were passed into machine learning algorithms to assess for the most accurate.

A custom deep-learning convolutional neural network (defined in Keras) was first assessed for accuracy, using the images as opposed to the landmark data. It was trained with 255 images: 100 'butterfly', 82 'scissor-sit' and 73 'superman'. To create more data, an image data generator was used.

```
training_datagen = ImageDataGenerator(
      width_shift_range=0.2,
      height_shift_range=0.2,
      shear_range=0.2,
      zoom_range=0.2,
      fill_mode='nearest'
)
```

Figure 16 Image data generator modifiers.

The model used a training and validation batch size of 30 and was compiled with a categorical cross entropy loss function with a RMSprop optimizer. It consisted of a simple four convolution and pooling structure followed by a 512-neuron hidden layer (all with ReLU activation function). A 3-neuron layer with a SoftMax function was used to output the probabilities of the 3 pose classes.

```
model = tf.keras.models.Sequential([
    # This is the first convolution
    tf.keras.layers.Conv2D(64, (3,3), activation='relu', input_shape=(2
40, 240, 3)),
    tf.keras.layers.MaxPooling2D(2, 2),
    # The second convolution
    tf.keras.layers.Conv2D(64, (3,3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2,2),
    # The third convolution
    tf.keras.layers.Conv2D(128, (3,3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2,2),
    # The fourth convolution
    tf.keras.layers.Conv2D(128, (3,3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2,2),
    # Flatten the results to feed into a DNN
    tf.keras.layers.Flatten(),
    # 512 neuron hidden layer
    tf.keras.layers.Dense(512, activation='relu'),
    tf.keras.layers.Dense(3, activation='softmax')
])
```

Figure 17 Custom CNN definition for pose classification.

The Model was trained for 100 epochs and validated with 72 (20%) images. It was tested with the remaining 35 (10%).

The other models were trained using landmark values. A series of Random Forests were trained, each with a minimum samples per leaf of 10. Forests of sizes 5, 10, 50, 100, 1000 and 10000 were assessed. K-fold cross-validation was used in determining the most accurate to assess for prediction, with 10 shuffled splits of a random state of 1. It was found that the 50-tree forest was most accurate with a cross-validation score of 0.873.

A support vector machine was trained. A multi-layer perceptron classifier was also analysed with 500 maximum iterations, an activation of 'logistic' with a stochastic gradient descent solver. A logistic regression model was assessed. These were all trained with 70% of the data.

For assessment of K-nearest-neighbour, the best 'k' value was first searched. 'K' values between 1 and 9 were examined on models trained with 70% of the training data. Through analysis of the error, models of K = 3, 4 and 5 were brought forward for comparison.

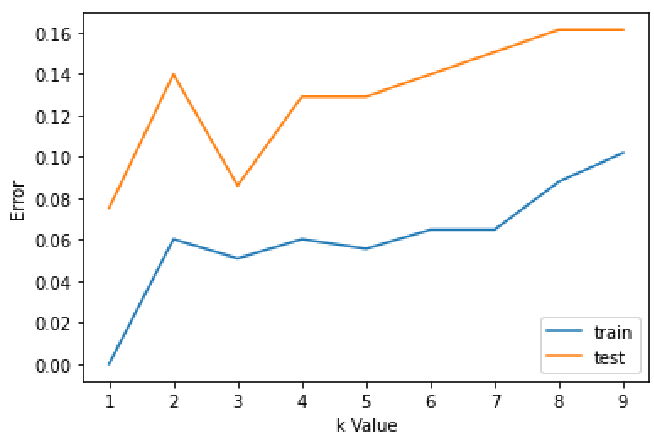


Figure 18 Error of models for K-values, showing the lowest training and test error for K = 3,4,5

The 3 ‘K’ values were cross-validated using 10 K-folds with 10 shuffled splits of a random state of 1. ‘K = 4’ was found to be the most accurate out of the three with a cross-validation accuracy score of 0.906.

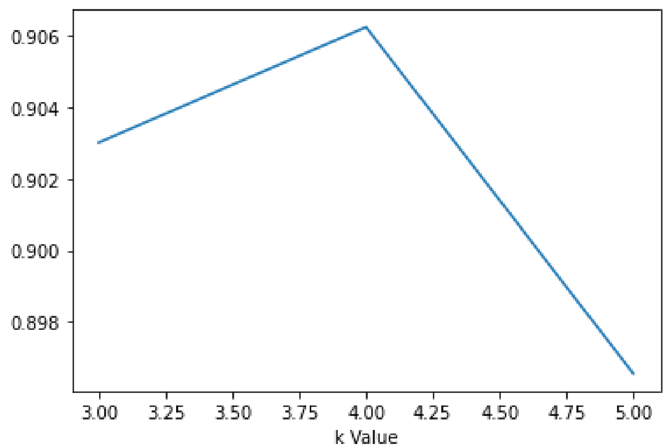


Figure 19 Accuracy values per K value, showing ‘k=4’ as the most accurate.

Results

Most of the models were tested on a 30% subset of the data, except for the CNN (10%). It was found however, that a form of data snooping could occur as data consisted of a shuffled set of video frames. A subsequent frame from the training set could fall into the test set, skewing the results. On this discovery, the models were also tested on 75 frames from completely independent captured video.

The CNN achieved a poor accuracy of 0.4. With the little amount of data, this did not increase no matter the number of epochs trained.

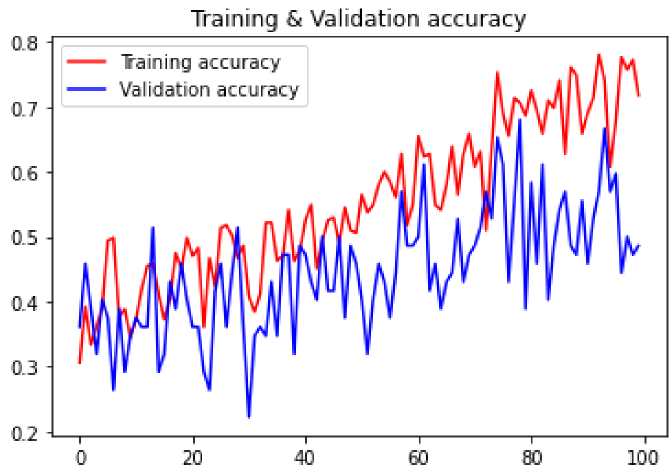


Figure 20 Graph showing noisy accuracy results with no substantial increase in CNN validation accuracy between epochs 80-100.

The remaining accuracy values are as follows:

Table 5 Pose recognition model accuracy results when trained from BlazePose landmark data.

Machine Learning Algorithm	**Test Split Accuracy**	**Independent Test Accuracy**
50-Tree Random Forest	0.86	0.66
Support Vector Machine	0.87	0.65
Multilayer Perceptron Network	0.75	0.81
Logistic Regression	0.86	0.72
K-Nearest-Neighbour (K=4)	0.91	0.86

Application

MLKit uses the most accurate model, the K-Nearest Neighbour algorithm, to predict the classification of poses. It develops on this by using the distances between landmarks as opposed to using the raw values. Firstly, it removes possible classifications where certain joints are bent the incorrect direction (removing by ‘max distance’), before classifying based on mean distance (**APPENDIX G**).

The Google Development team released a ‘Vision-Quickstart’ template application (Google MLKit 2022) which allows for kick-started development. The template is built for counting push-ups (learning two classes, up and down). However, it can be edited by supplying the app asset folder with the returned BlazePose pole classes CSV and updating the ‘POSE_SAMPLES_FILE’ and ‘POSE_CLASSES’ variables in the ‘PoseClassifierProcessor’ to match the three moves.

The application provides an added benefit when it comes to classifying live video. It uses ‘smoothing’ to classify segments of the video based on a window of a certain number of frames. This instance uses 10 frames as a window, and the algorithm supplied by the ‘vision-quickstart’ can be seen in **APPENDIX H**.

The solution is built and tested using black box methods. Using the ‘stillImageActivity’ option (**fig 9**) in the mobile phone simulation, still images are uploaded to be classified. The application successfully classified all images.

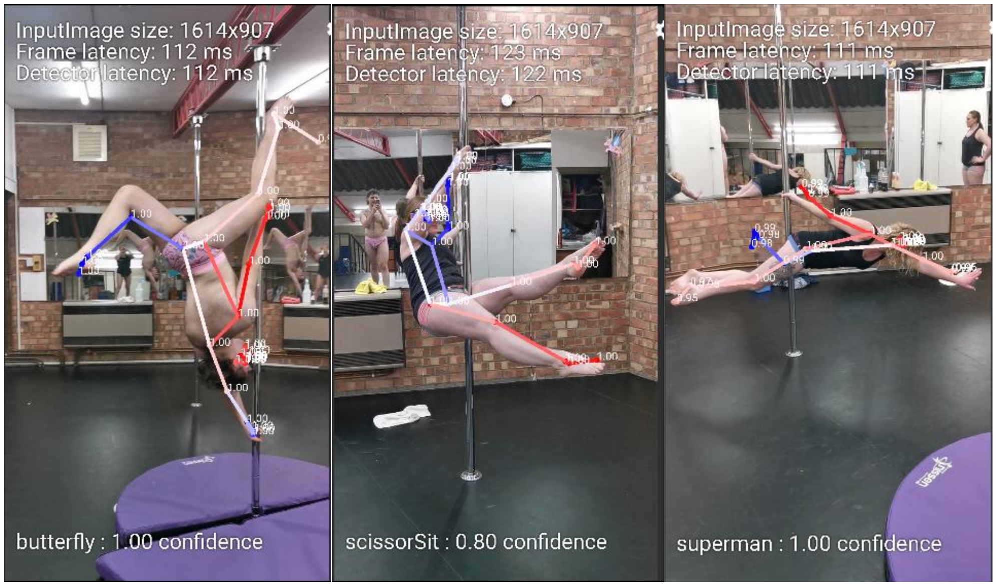


Figure 21 Output from black box testing unseen pole pose images in 'point your pose' application, showing correct classifications of high confidence.

The '.apk' file was downloaded and ran on a real smartphone to test how it classifies live video data, using classification mode in the 'livePreviewActivity' option. Used in the studio, it was exceptionally accurate at classifying moves.

Figure 22 Output from black box testing live video in 'point your pose' application, showing correct classifications of high confidence.

However accurate, there were a couple of flaws spotted in the design found on testing live video. Firstly, the application classifies constantly, and so when the poler is standing, it is classified as a move (usually a low confident 'butterfly'). A remedy for this could be only classifying when a certain threshold is reached (which also disregards mid-move poses) or classifying 'standing' as a pose in itself.

Another observation was that the application classifies the move as 'butterfly' the second the poler inverts upside-down. This could be due to the large difference of landmark position compared to that of scissor-sit and superman. This classification problem may be solved by training the model with more moves, which would hopefully decrease the model's confidence of scoring all inverts as 'butterfly'.

Figure 23 The application classifying a basic invert and standing as 'butterfly'.

Evaluation

K-nearest neighbour is the most accurate machine learning algorithm for classifying pole dance moves through BlazePose landmark identification. Care needs to be taken when using frames from videos as independent samples, due to the risk of test splits containing data correlated with that in the training set. When working with small amounts of data, using landmarks to classify poses is more efficient than using deep learning methods, with regards to time and resources used in training.

Pose detection and recognition is an effective form of classifying pole dance moves. BlazePose is a lightweight solution for creating a pole move recognition application. The fact that it uses a face detector as a proxy has little effect on the end product's classification efficiency when the face is not fully in frame, and the body is partially obscured by the pole. There is room to research if other tools are more efficient, such as 'OpenPose' (Cao et al. 2019).

As the amount of moves to classify increases, the application may run less efficiently due to an increased complexity, and this needs to be studied. If this seems to be the case, the solution could be to utilise cloud resources. The landmark data could be stored in the cloud, for querying live unseen data with K-NN (Nodarakis et al. 2017). The forfeit would be the requirement of a stable phone internet connection as interpretation would no longer happen on-device.

In pole dance, variations of moves occur. These are staple poses where a limb or two may be in a slightly different position. The accuracy of spotting these differences needs to be assessed, and whether a form of hierarchical classification could solve this issue.

An extension of the application could be that of classifying poses of other aerial artforms, such as ‘hoop’ or ‘silks’. Internet URLs could be shown to the user on classifying a seen pole move, to link to tutorials on how to perform them. Finally, a progression system could be made to certify that a poler has performed a move on witnessing it at classification.

Chapter 5

Pole versus Pornography

5.1 Research Methods

How easy is it for a deep learning system to distinguish between pole dance and explicit imagery? Tests are compared when an InceptionV3 model is trained with different sets of image data. The experiment tests classification on generic non-explicit image data, pole image data and pornography frames. The experiment also assesses for the most accurate classification of pole dance and pornographic video clips using the created model.

The Dataset

Two datasets are used in this experiment. The first is Avila et al.'s (2013) 'NPDI Pornography Database'. It contains 80 hours of 400 pornographic and 400 non-pornographic videos, covering several genres and containing diverse ethnicities. Non- pornographic images comprise of 'easy' and 'difficult' classifications. This is in the context of the difficulty to classify between these and porn (for a machine learning model, or potentially the human eye). Data is supplied in both frame and video format.

Figure 24 An image of a car (non-pornographic 'easy' image) and an image of wrestling (non-pornographic 'difficult' image).

Pornographic data comprises of typical videos seen on adult websites, covering genres such as 'gay', 'straight', 'lesbian' and 'cartoon'. The data not only covers imagery of sexual acts, but also scenes of porn with no nudity, such as the video prelude to a pornographic storyline.

Figure 25 A pornography prelude image and an image of sexual intercourse from the dataset (censored for publication purposes).

The dataset used for pole imagery is the same as that carried over from the 'Point your Pose' experiment. However, it is important to note the limitations associated with the use of this dataset for this experiment. Unlike the former, where raw landmark figures are used for model training, this experiment uses the individual pixels which are inputted as features into a convolutional neural network. This means that lack of diversity could have an impact on model bias (Fazelpour, De-Arteaga 2022). It is worthy to note that if this solution were to be used out-of-the-box, it could breach terms of ‘fair artificial intelligence’ (Feurriegel, Dolata, Schwabe 2020). ‘Fair AI’ means that all end-users have been considered on the design of a machine learning model, which leads to better accuracy results for the population, and is why companies should strive for these solutions (Microsoft 2022; Google 2022; Gayhardt et al. 2022). The lack of diversity is due to the lack of poler image supply (**Table 2,** ‘Risk 1’), and so if this experiment were to be replicated, it would be advised to collect imagery from more studio/ home environments and to include a higher proportion of polers of diverse ethnicities and people of colour. This censorship algorithm may work unfairly on pole media containing these individuals.

Binary labels were given to the dataset classes. The non-explicit class was labelled ‘0’ whilst explicit was labelled ‘1’.

Dataset Splits

Prior to training, data was randomly sampled then randomly shuffled. The data is split: 70% for training, 20% for validation and 10% for testing. Note that this is binary classification, therefore where the model was trained on ‘easy’ and ‘difficult’ non-pornographic imagery, these fell under the same training label as an equal mix of both. Where pole imagery was trained with this, *all* the pole training data was used, and the testing pole data consisted of frames from completely independent footage. The following table shows the resulting image count (full table including results in **APPENDIX I**):

Table 6 The number of images used for training and testing the binary censorship model.

Model No.	**Trained on**	**Sample**	**Training Images**	**Validation Images**	**Testing Images**	**Pole Testing Images**
1	Pole; Porn	362; 362	253; 253	72; 72	37; 37	75
2	Easy; Porn	3000; 3000	2100; 2100	600; 600	300; 300	362
3	Difficult; Porn	3000; 3000	2100; 2100	600; 600	300; 300	362
4	Easy & Difficult; Porn	(1500; 1500); 3000	2100; 2100	600; 600	300; 300	362
5/6/7	Easy & Difficult & Pole; Porn	(1319; 1319); 3000	2100; 2100	600; 600	300; 300	75
8/9	Easy & Difficult & Pole; Porn	(1569; 1569; 362); 3500	2450; 2450	700; 700	350; 350	75

Analysis Metrics

The evaluation metrics used were (1) accuracy, (2) sensitivity and (3) specificity. Where TP = True Positives; TN = True Negatives; FP = False Positives and FN = False Negatives:

$$\text{Accuracy} = \frac{Correctly\ Classified\ Images}{Number\ of\ Classified\ Images} \quad (1)$$

$$\text{Sensitivity} = \frac{TP}{TP + FN} \quad (2) \quad \text{Specificity} = \frac{TN}{TN + FP} \quad (3)$$

Accuracy score was calculated using Sci-kit's '*accuracy_score()*' while sensitivity & specificity were calculated using the values (TP/TN/FP/FN) harnessed from Sci-kit's confusion matrix. These metrics were the main factors in deciding the best hyperparameters.

Sensitivity and specificity are important in the context of censorship as the aim is to eliminate the amount of misclassified 'explicit' images whilst minimising the amount of misclassified 'non explicit'. Ideally the objective is to have high values for both, but the priority is sensitivity, as in a social media setting having undetected pornographic images posted is most detrimental to both the company's perception (Gillespie 2018) and the livelihoods of younger viewers (Massey, Burns, Franz 2021).

The binary classification model produces a set of predictions between '0' and '1' for the test set. A discrimination threshold needs to be set to distinguish which prediction belongs to which category. This was set at 0.5 as a baseline. For the most accurate models, a ROC curve was produced to show separation between the classes. The AUC of this was analysed and a discrimination threshold was chosen based on this, that maximised sensitivity. These metrics were produced using the Scikit-Learn library metrics module (with the AUC calculation using the trapezoidal rule). Finally, confusion matrices were used to quantify the quality of predictions visually. This was carried out by creating a heatmap (using seaborn) from a Pandas data-frame converted from a Sci-kit confusion matrix.

Video Analysis Algorithms

The best model was saved to be brought into video analysis. Videos were analysed by classifying each individual frame using the model with a discrimination threshold of 0.5. They were also analysed separately on taking classification of every 5^{th} frame (therefore using less data). Five completely independent pole videos were analysed, encompassing different settings (e.g. clothing, perspective, ultra-violet light). Eight pornographic videos were analysed, two from each of four genres. For every video, a summarised classification value was produced using each of six different calculations: mean classification, median classification, weighted beginning, weighted centre, weighted end, and an average of the last three.

A ‘weighted’ frame in this case means that a classification of ‘explicit’ has more impact on the video’s average summary classification. Non-weighted frames that have a higher ‘explicit’ probability have a reduced impact on the summary classification. This is calculated linearly based on array index, the array being a series of frame classification probabilities making up a video clip.

For the beginning-weighted algorithm, explicit frame predictions are penalised the further away they are from the beginning of the video. i.e., If a prediction was 0.2 at position 0 of a 10-frame video, it would have a 0.2 - (0.2 * 0/10) value, 0.2 (unchanged). However, if the prediction was 0.2 at position 7, it would have a 0.2 – (0.2 * 7/10) value, or 0.06.

```
#Favour explicit classifications at the beginning of a video
def begWeighted(predictions):
  total = 0 #sum to be averaged
  for i in range (0,len(predictions)):
    #weighted prediction = prediction - (prediction * i/n predictions).
     Approaching the end, explicit prediction is penalized more.
    total += ((predictions[i] - (predictions[i] * i/len(predictions))))
  return total/len(predictions)
```

Figure 26 Code for beginning-weighted video classification summary algorithm.

For the end-weighted algorithm, explicit frame predictions are penalised the further away they are from the end of the video. The end-weighted algorithm calculated in a similar fashion except the array of frame predictions is looped through backwards. If a prediction was 0.4 at position 9 of a 10-frame video, it would have a 0.4 - (0.4 * 0/10) value, 0.4 (unchanged). However, if the prediction was 0.4 at position 1, it would have a 0.4 – (0.4 * 8/10) value, or 0.08.

```
#Favour explicit classifications at the end of a video
def endWeighted(predictions):
  total = 0 #sum to be averaged
  j = 0 #index increasing from 0

  for i in range (len(predictions)-1,-1,1):
    #i = index going backwards
    #weighted prediction = prediction - prediction * i/n predictions.
    Approaching the beginning, prediction is penalized more.
    total += ((predictions[i] - (predictions[i] * j/len(predictions))))
    j+=1
  return total/len(predictions)
```

Figure 27 Code for end-weighted video classification summary algorithm.

For the centrally weighted algorithm, ‘explicit’ predictions are penalised more based on their distance away from the centre of the video. If a prediction was 0.2 at position 4 of a 10-frame video, it would have a 0.2 - (0.2 * √ (((4+1)/10) -

$0.5)^2$) value, 0.2 (unchanged). However, if the prediction was 0.2 at position 7, it would have a 0.2 - (0.2 * $\sqrt{}$ (((7+1)/10) - $0.5)^2$) value, or 0.14. Before subtracting the weighting from the prediction, the weight is squared and rooted to prevent minus figures.

```
#Favour explicit classifications at the centre of a video
def midWeighted(predictions):
  total = 0
  for i in range (0,len(predictions)):
#weighted prediction = prediction - prediction*sqrt(((i+1)/n - 0.5)**2)
Further from the center of predictions, explicit prediction is penalise
d more
    weight = sqrt((((i+1)/len(predictions)-0.5)**2))
    weight = predictions[i] - predictions[i] * weight
    total += weight
  return total/len(predictions)
```

Figure 28 Code for centrally weighted video classification summary algorithm.

5.2 The Design of an Inclusive Censorship Algorithm

Dataset Collection

The pole image dataset used in this study was that carried over from the ‘point your pose’ application. New pole videos were sourced from local studio members to examine the model’s efficiency and best video summary metric. These videos were more varied than the images collected from the previous study, including various lighting, pole moves and camera movements. Avila et al.’s (2013) NPDI pornography dataset was accessed by contacting Prof Dr Sandra Avila of the University of Campinas by email (**APPENDIX K**). It is used according to the terms of the signed license agreement, which requires adequate citation and academic usage only. The limit of published frames from the dataset has been adhered to in this dissertation.

Model Configuration

The models used shared a fixed set of static hyperparameters. An inceptionV3 model was defined in Keras with two added dense layers of 64 neurons, using a ReLU activation function. The output layer consisted of 1 neuron with a sigmoid activation to output the class probability. The model was compiled with an Adam optimiser and binary cross entropy loss function, logging accuracy as a metric. Images were fed into the model using training and validation generators which converted the images to 240 x 240 dimensions. Static hyperparameters for all model configurations can be seen in the following table:

Table 7 Static parameters that remained uniform for all model configurations.

Static Parameter	**Setting**
Model	Inceptionv3 + (2 layers * 64 neurons)
Training Batch Size	32
Validation Batch Size	32
Image Dimensions	240 x 240
Class Mode	Binary
Loss Function	Binary Cross Entropy
Optimisation Function	Adam
Epochs	100 (with call-back)

Five Models were trained until a stopping criterion of 90% training accuracy was reached. When the best training data combination was found, this was increased to 93% but concerned validation accuracy. This percentage was chosen due to a negligible difference in scoring metrics past this point.

Results

Model & Image Classification

The first five model configurations were concerned with finding the best combination of training data. After 90% training accuracy was reached, these were the results of testing on 'standard' test data and pole-specific test data. 'Standard' data refers to non-pole non-pornographic and pornographic images. This is akin to a naive social media censorship dataset (see **APPENDIX I** for full configurations).

Table 8 'Standard' and 'Pole-Specific' classification resulting metrics when models are trained until 90% training accuracy.

Model Output No.	Training Combination	Standard Test Accuracy	Standard Sensitivity	Standard Specificity	Pole Accuracy	Pole Sensitivity	Pole Specificity
1	Pole; Porn	0.514	0.027	1	0.679	0.027	1
2	Easy; Porn	0.825	0.656	0.99	0.386	0.656	0.162
3	Difficult; Porn	0.913	0.857	0.97	0.396	0.857	0.014
4	Easy & Difficult; Porn	0.857	0.97	0.743	0.482	0.97	0.077
5	Easy & Difficult & Pole; Porn	0.628	0.62	0.637	0.696	0.62	1

Training a model on only little pole data against pornographic imagery leads to inaccurate test results. Accuracy of 0.514 for classifying explicit images against the pole test split nears random guessing when it comes to binary classification. Accuracy of 0.679 when classifying pornography against independent pole imagery is also low. Specificity is 1, as all pole images are classified correctly, however approximately no porn images are flagged as such, leading to extremely low sensitivity values. In this scenario, the model is highly biased. This may be due to both the model not being trained on a general dataset, and not having been trained on many images (506). This model configuration would not serve a general social media censorship solution well due to its bias, although could potentially be used in a pole-only media application if training was further explored. With the low amount of pole imagery acquired, this is not feasible for this experiment.

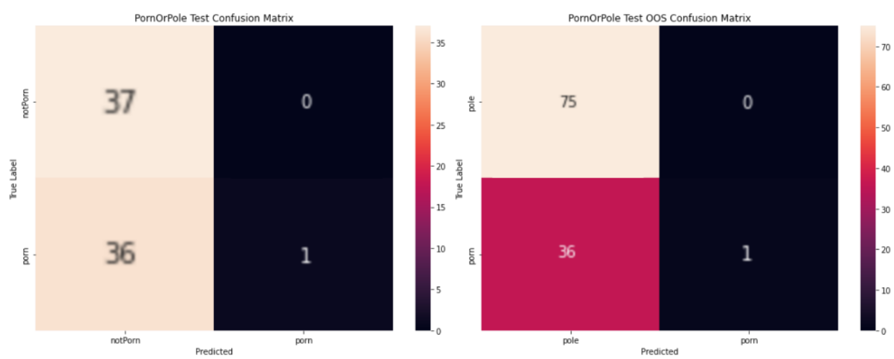


Figure 29 Confusion matrices showing poor classifications of the pole test data split (left) and held-out independent pole test data (right) against pornography.

When trained with 'easy' neutral data against pornography, 'standard' accuracy is higher at 0.825. 'Standard' sensitivity remains low at 0.656, but this is an increase in the previous model by 0.629, thus more pornography is classed correctly. Nearly all neutral images are classed correctly, but flagging explicit content remains unsatisfactory. In addition, much pole imagery is classified incorrectly, with an accuracy of 0.386 and an extremely low specificity of 0.162. This model is not poler inclusive.

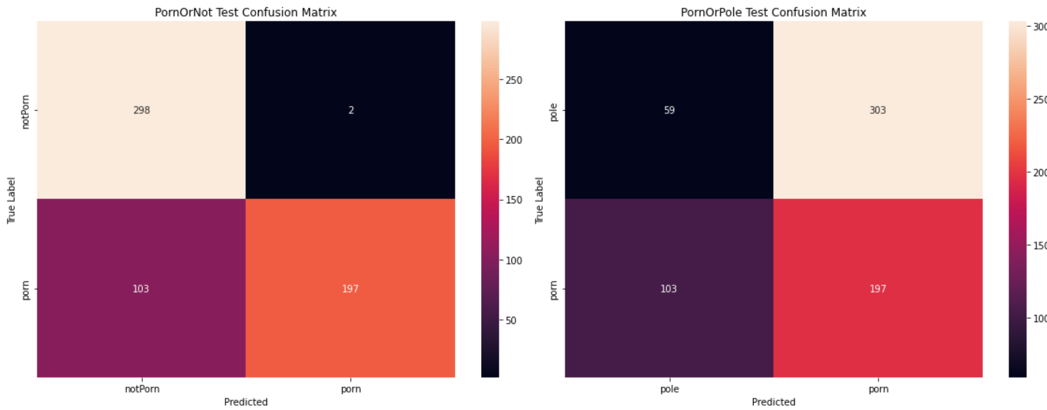


Figure 30 Confusion matrices showing classifications of 'easy' neutral versus explicit data (left) and pole against explicit data (right).

Training the model on 'difficult' versus porn, there is a noticeable result regarding classifying the 'standard' data, with an accuracy of 0.913, sensitivity of 0.857 and a specificity of 0.97. Many more neutral and pornographic images are classified correctly. Unfortunately, pole specificity is the lowest yet at 0.014, making the model exclude polers exceedingly.

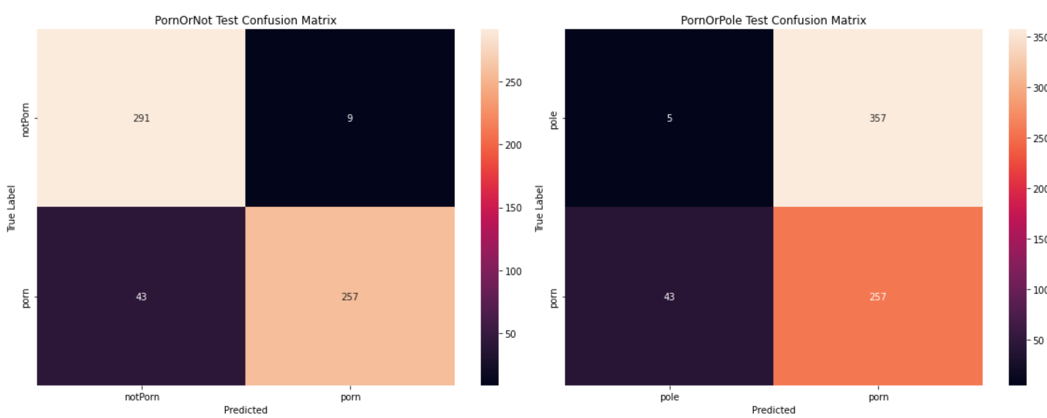


Figure 31 Confusion matrices showing classifications of 'difficult' neutral versus explicit data (left) and pole against explicit data (right).

A model was then trained on mixed neutral imagery against pornography. It led to a drop in 'standard' accuracy to 0.857, but increased sensitivity to 0.97. The algorithm flags a large majority of the pornographic imagery as such. There is a drop in standard specificity to 0.743, however, which flags more neutral imagery as explicit than the previous model. The low pole specificity remains evident at 0.077. This model tends to over-censor from a bias towards pornography. Analysing the confusion matrices along with the pole accuracy of 0.482 proves inadequacy.

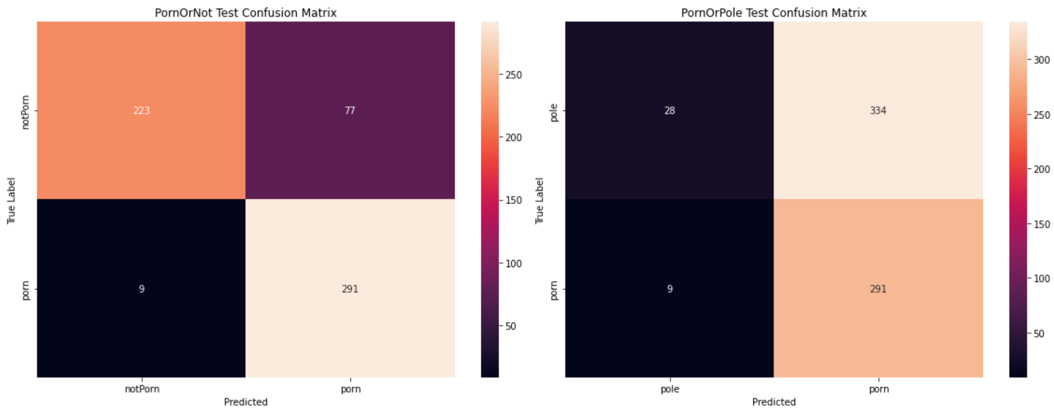


Figure 32 Confusion matrices showing classifications of mixed neutral versus explicit data (left) and pole against explicit data (right).

The best trained model was found to be model configuration 5, which was trained on mixed neutral images and ‘pole’ images against pornographic imagery. The standard metrics sit below those of the previous model, allowing more pornography to be flagged incorrectly. However, the accuracy regarding pole censorship is higher, at 0.696, while pole specificity is the highest at 1. Every pole image is classified correctly. It is understandable that this is still unacceptable as a censorship solution, but due to acknowledging pole imagery, can be improved upon by means other than data mixing.

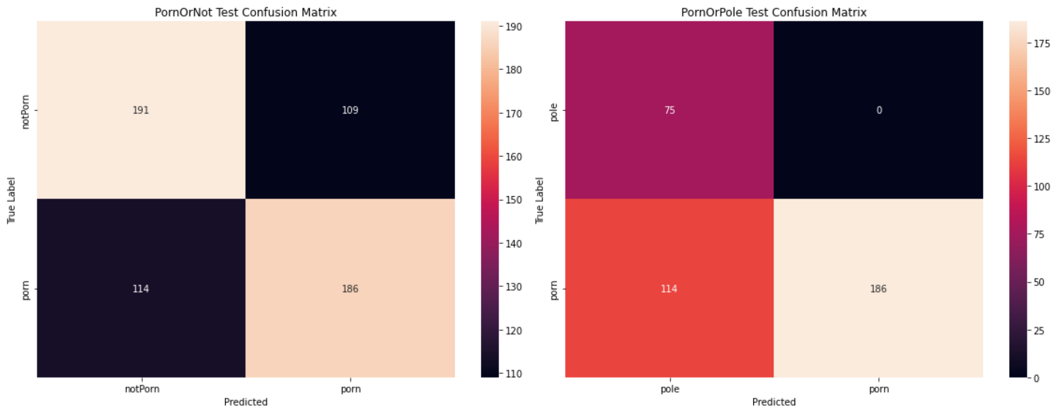


Figure 33 Confusion matrices showing classifications of mixed neutral and pole data versus explicit (left) and pole against explicit data (right).

To improve upon model 5, several factors are changed. Model 6 is trained until a call-back of 93% validation accuracy is achieved, rather than training accuracy. This substantially improves all metrics for standard and pole-specific testing, while keeping pole specificity at 1 (hence classifying all pole images correctly).

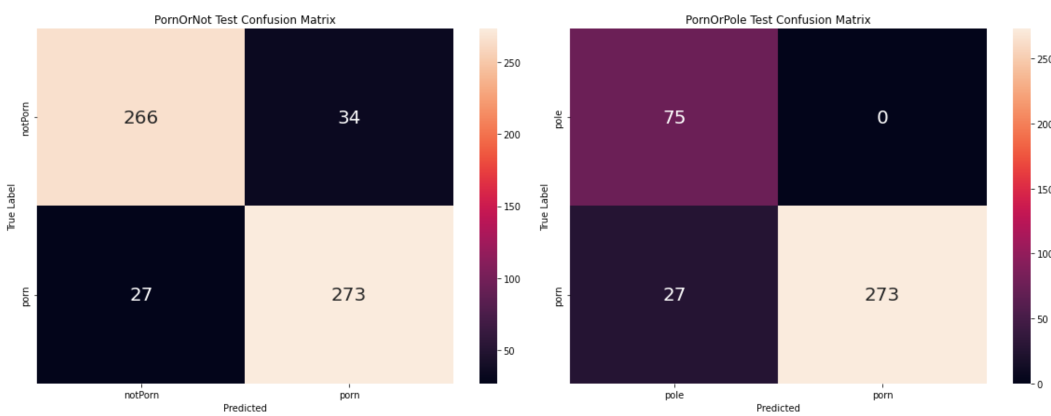


Figure 34 Confusion Matrices showing how a higher validation call-back improved resulting classification.

Model output 6 still misclassified 27 pornographic images incorrectly, which could still be considered detrimental to a social media company if a sensitivity of 0.91 is deemed unsatisfactory. The method of prioritising explicit censorship was to configure the discrimination threshold based on ROC (Receiving Operating

Characteristic) curve analysis. So far, images were classified based on probabilities under and over 0.5 (the 'threshold').

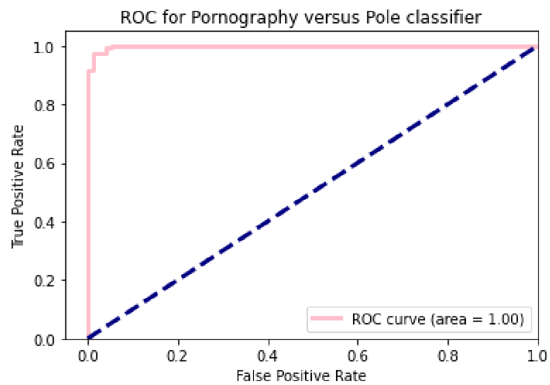


Figure 35 ROC curve produced for model configuration 6.

The curve showed that there was already a plausible separation between the two classes, with an AUC value of 0.998. Based on the graph, the discrimination threshold was shifted from 0.5 to a minimal 0.03 (model output 7). This change led to misclassified pornography decreasing from 27 to 3 (increasing sensitivity to 0.99). The forfeit was increasing misclassified pole images to 3 (decreasing pole specificity to 0.96).

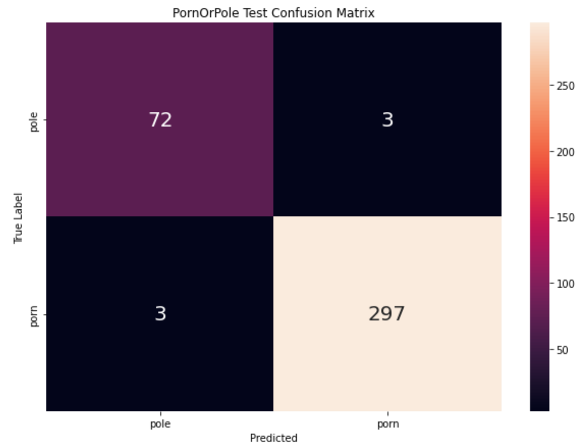


Figure 36 Confusion Matrix for test pole data after shifting the discrimination threshold from 0.5 to 0.03.

Table 9 'Standard' and 'Pole-Specific' classification resulting metrics when models are trained with 4200 images until 93% validation accuracy.

Model Output No.	**Standard Test Accuracy**	**Standard Sensitivity**	**Standard Specificity**	**Discrimination Threshold**	**Pole Accuracy**	**Pole Sensitivity**	**Pole Specificity**
6	0.898	0.91	0.887	0.5	0.928	0.91	1
7	---	---	---	0.03	0.79	0.99	0.96

One final model configuration was examined, through a simple increase in training images from 4200 to 4900. This increased all metric scores bar standard specificity which dropped a negligible amount. The AUC rose to a highly satisfactory 0.999. Once again, scrutinous analysis of the ROC curve led to the discrimination threshold shifting from 0.5 to 0.004 on this occasion.

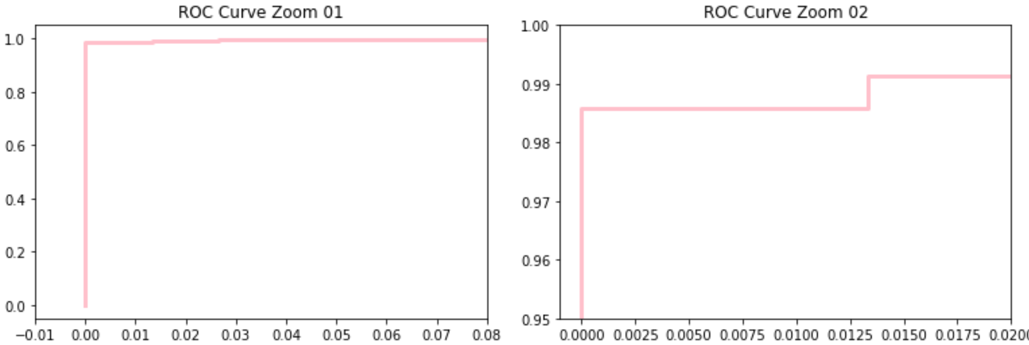


Figure 37 Meticulous examination of ROC curve for model 7.

This led to a sensitivity score of 1, classifying all pornography correctly. The forfeit was a reduction in pole specificity to 0.88. On reaching this satisfactory result, the model was saved to be brought into video classification.

Table 10 'Standard' and 'Pole-Specific' classification resulting metrics when models are trained with 4900 images until 93% validation accuracy.

Model Output No.	**Standard Test Accuracy**	**Standard Sensitivity**	**Standard Specificity**	**Discrimination Threshold**	**Pole Accuracy**	**Pole Sensitivity**	**Pole Specificity**
7	0.917	0.983	0.851	0.5	0.986	0.98	1
8	---	---	---	0.004	0.79	1	0.88

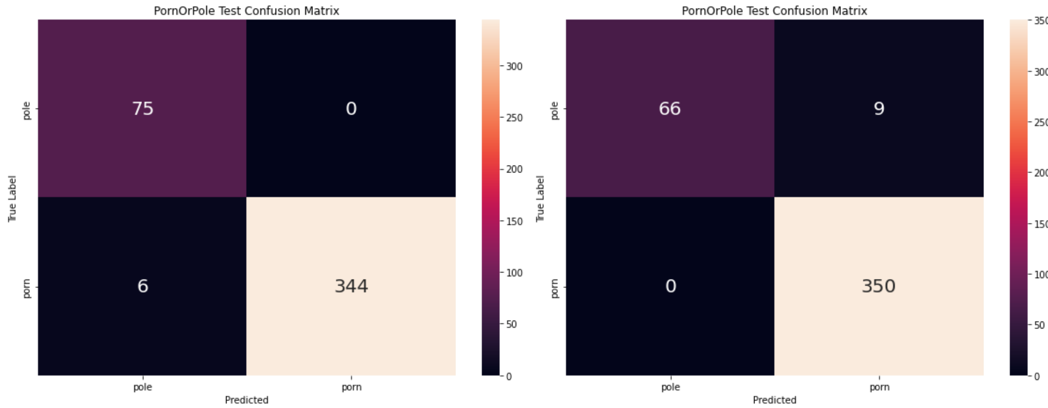


Figure 38 Confusion matrices produced from the 4900 image-trained model on predicting test pole data before (left) and after (right) discrimination threshold shift (Model output 7 & 8).

Video Classification Analysis

The aim of the video classification task is to summarise whether video contents contain pornographic material. A secondary goal is to minimise resources used, as social media processes big data continuously. To accomplish the former, five pole videos and eight pornography clips were split into frames and passed through six algorithms, covering frame classification central tendency, including linearly weighted means. The resource-conservative goal was accomplished by analysing these again but on every 5th frame. The video categories and their corresponding number of frames are listed in the following table. Note that acronyms such as ‘WsM’ mean ‘woman having sex with man’. The best metric is one that returns a score nearest to ‘0’ for pole media and nearest ‘1’ for pornography. The full results table can be seen in **APPENDIX J**.

Table 11 Video category, Frame Count and Best Metric.

Video Category	**No. Frames All; Every 5th**	**Best Metric All; Every 5th**	**Correct Classification? (Discrimination Threshold = 0.5)**
Pole standard video	363; 60	Beginning weighted; Beginning weighted	yes
Pole video - x2 speed and rotating around poler	772; 128	Median; Median	yes
Pole video – far perspective	1082; 180	Beginning weighted; Beginning weighted	yes
Pole video – moving from poler to poler	511; 85	Median; Median	yes
Pole video – ultraviolet light	1627; 271	Median; Median	yes
Porn – WsM 1	814; 135	Median; Median	yes
Porn – WsM 2	658; 109	Median; Median	yes
Porn – MsM 1	73; 12	Median; Median/Mean	yes

Porn – MsM 2	274; 45	Median/Mean; Median/Mean	yes
Porn – WsW 1	402; 67	Mean; Mean	no
Porn – WsW 2	177; 29	Median; Median	yes
Cartoon – WsM 1	414; 69	Median; Median/Mean	yes
Cartoon – WsM 2	39; 6	Median/Mean; Median/Mean	yes

On a glance at the results, it may seem that the Median frame's assigned label is the best fit for classifying the data. It has the nearest probability to the true label for most of the videos. The disadvantage is that it only takes one frame into account, meaning that a large portion of the video is ignored, which could contain explicit material. Utilising the mean value of frame classification considers all the frames and has the same number of correct classifications as the median, and so would be the most efficient metric out of the two.

Table 12 Scoring of video classification algorithms.

Central Tendency/ Algorithm	**No. Correct Classifications (Discrimination Threshold = 0.5)**
Mean	22/26
Median	22/26
Beginning-Weighted	20/26
Central-Weighted	24/26
End-Weighted	20/26
'Fully'-Weighted	23/26

On further analysis, a centrally weighted algorithm labels 24 out of 26 videos correctly, compared to 22 for mean and median. It assumes the core of the video contains the most important information. This could be true for a pole video, where the poler is on the pole during this period. In addition to the mean, all frames are considered. Yet, the centrally weighted algorithm has less certain probabilities, meaning it would be more reactive to change classifications if the discrimination threshold were to shift from 0.5.

Classifying every 5^{th} frame from a video significantly reduces the amount of data analysed. For instance, for the ultraviolet pole video it reduces the frames from 1627 to 271. Although much data is lost, the classification does not change for most of the videos. It does for the third pornography video, where the 'full-weighted' algorithm increases the probability from 0.474 to 0.5, changing to a correct classification on analysing every 5^{th} frame. This essentially means that in this study this formula is more accurate on inspecting less frames. The drawback of interpreting less is the same as that of the median, where lost data could contain pornographic frames. This is the forfeit for using less processing resources. A balance needs to be had in reducing the periods where malicious frames could be inserted into a 'clean' clip. A proposed solution is randomising a reasonable number of frames classified by the algorithm in every uploaded video.

Examining Misclassified Video

It is interesting to examine two videos that were often classified incorrectly using the saved model and the 0.5 discrimination threshold. Both the mean and the median predicted the class of the "pole standard video" incorrectly, whilst the centrally weighted algorithm predicted a threshold-nearing '0.467' for the full video. This may be due to the central individual wearing a colourful leotard (not in the training set), and a reflection of few people in the background, rendering the frames 'noisier'. As pole training images were passed from the pole move detection dataset, images tended to be less noisy to prioritise BlazePose detecting a main individual.

A pornographic video ‘Porn – WsW 1’ was misclassified by all metrics. On visual analysis, no intercourse takes place, and in the 402 frames partial breast and nude buttock are visible. Two people are wrestling in oil. Arguably, this misclassification is not detrimental, however would be a case decided by an ethical department in a media company.

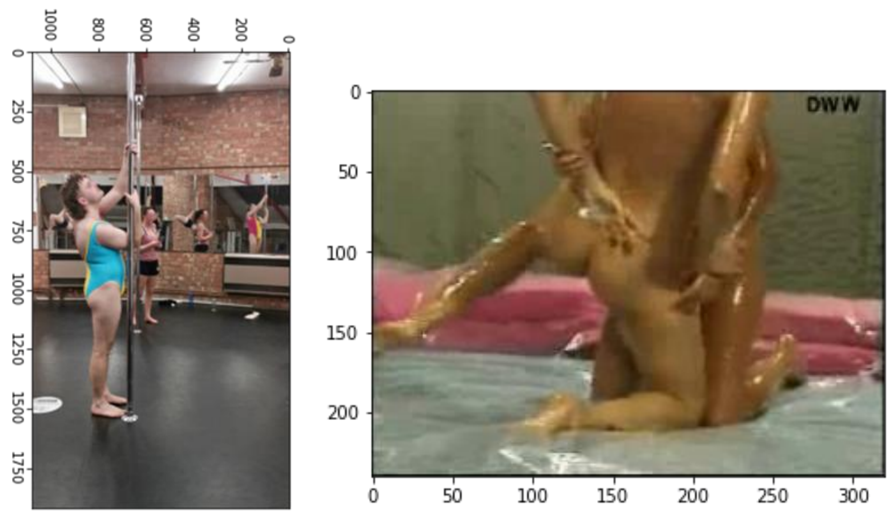


Figure 39 A frame from a misclassified ‘pole standard video’ and one of nude buttocks from ‘porn – WsW 1’.

Evaluation

Several conclusions can be drawn from analysis of the results of this experiment. Iterative development of a derived form of InceptionV3 gave rise to a model which was 100% sensitive in censoring general images and 88% specific in censoring explicit material against pole dance. This was trained using only 362 pole images along with 3,138 neutral images against 3500 pornography frames. Utilising a validation accuracy call-back, an appropriate training image size, and tuning the discrimination threshold is important for improving a binary censorship flagging solution.

Using a correct summary function is necessary for accurate video classification. In the case of pole dance versus pornography, a centrally weighted mean of frames gives rise to an accurate classifier. Inspecting a subset of frames is an efficient way of saving processing time, but at the cost of losing data. Unassessed frames could potentially be exploited by uploaders, and so a measure needs to be put in place to mitigate this.

This study attempts to answer an important ethical dilemma. Pole dance media assessed in this experiment is inherently non-pornographic, as shown by the separation between predicted classes when trained with a small pole dataset. This challenges the stigma faced by the sport in the modern era.

It is understood that social media platforms may use a multilabel classifier, incorporating other flags such as violent content. This study was made under the knowledge that pole dance is deemed erotic based on certain viewpoints, and that is why it is censored as such. Further studies need to be made to assess whether it resembles other kinds of to-be-censored data. Platforms have a wealth of image data accessible to them, with pole images being accessible through a simple search of a hashtag. These companies need to train their algorithms in an inclusive fashion, which does not leave polers, or other communities, shadow-banned or silenced. This could be in the form of further iterative training on current models. If they are knowingly excluding this population, then there is a need for action to be had from an ethical standpoint.

It is to be acknowledged that machine learning models can be inaccurate to some completely unseen data. If this model were to be integrated into an application, it would be wise to use it as a flagging solution. Images flagged as explicit could be either appealed against by the user, or automatically sent to the relevant team to be assessed by the human eye. Similarly, pornographic content misclassified by the model can be reported by a user, for it removed and used to improve upon the current censorship solution.

At the time of writing, Instagram has introduced new measures to tailor the application towards its members. A section on Instagram (Mosseri 2022) has appeared named 'following'. Previously, Instagram has only retained a feed which uses a recommendation system to show the user what it deems appropriate. This new section could show a user's 'followed' polers more, however, is not set to default, and the user experience needs to be assessed on having to access this tab separately. Instagram (2022) has recently enforced the requirement of being aged 13 years old at time of account creation. It now prevents current users from accessing the platform until their age is known. It is yet to be seen if this will have an impact on shadow-banning and censorship.

Chapter 6

Exploring Pole Sport Injury Prevention with Vision

6.1 Research Methods

Is it possible to prevent pole injury by using vision anomaly detection methods? This exploratory study lays the groundwork for designing a solution, taking abstract methods from other realms of computer vision research and combining them with current practices in relation to the sport. Requirements engineering in context involves defining, documenting, and maintaining the requirements of producing software. Creating a machine learning system is a slight shift from conventional development, including both ethical requirements (explainability, freedom from discrimination, specific legal requirements) (Vogelsang, Borg 2019) and those relating to the model training with specific hyperparameters and associated hardware.

To acquire detailed insight into the creation of an injury prevention solution, it is necessary to understand 'the poler'. For full requirements solicitation, therefore, it would be necessary to perform interviews and/or surveys as a qualitative approach to research. Alternatively, this research is in the form of a comparative literature and software review, with a visual prototype drafted based on findings.

6.2 Current Practices

Although the pole itself needs to stay in its current metal/ powder-coated, sleek form for performance, there exist external material methods of increasing safety. A 'crash mat' is a thick foam-padded floor mat used to cushion landings in sports such as gymnastics, stunts in movies, and falls in the elderly, and are commonly used to soften falls in pole sport. They are used as an injury mitigation technique rather than a preventative measure. A report by the health and safety executive (2019) investigated their efficiency in different employment sectors, from construction to recreational scenarios (obstacle courses from height). The review concluded that decelerations provided by these "soft landing systems" did reduce the chances of serious injury when installed properly and the body lands directly on them. If they were to be effective in pole sport these cases must be met. However, injury is still possible, including sprains, contusions, and some bone fracture. No studies have been made on their effectiveness in mitigating or preventing pole sport injury. Furthermore, they are not seen as a necessity in the pole community on purchase of a pole, as they are not advertised on leading pole selling websites in the UK (such as 'X-Pole' (2022)). They are used based off anecdotal information in pole classes in an attempt to mitigate serious injury and boost confidence in learning new moves.

Wrist braces, also known as "supports", are commonly used in pole sport to try to reduce wrist pain and injury in load bearing moves and poses. There is evidence to suggest the use of a wrist brace reduces the chance of injury in wrist-intensive activities (Moore et al. 1997). Trials in gymnastics indicate that they could be used in sport to prevent injury so long as they are used beforehand in situations where wrist pain is felt (Trevithick, Mellifont, Sayers 2020). There are currently no studies regarding their impact on pole sport safety. However, they are sometimes used when performing invert moves (with the wrist under more strain). It could be argued that pole is a derivative form of gymnastics, being an aerial art.

Figure 40 An image of wrist supports.

Supervision during instructive classes is a non-material injury-preventing measure. However, this is only applicable to the learning environment, as poling alone or performing in competitions does not garner instructor surveillance (although the latter has those at hand to aid if safety concerns arise). Instructors take part in 'spotting' (Nicholas et al. 2019) a poler to perform a new move, whether watching or physically supporting their body for guidance or safety reasons. An instructor during a group class usually must manage multiple polers, and so this is not always possible for the entirety and general overall surveillance is provided to 'spot' those that seem at a safety risk. Yet again, no studies have been made into the importance of spotting and risks involved in managing multiple pole students.

6.2 Requirements

Computer vision's anomaly detection methods could make way for a modern pole injury prevention system. The following are a collection of requirements for a solution. These incorporate technologies used in both pose detection and image classification, used in the previous move recognition and explicit classification artefacts. The theoretical requirements need to be established, which include:

1. A collection of safety measures: What is considered to be dangerous?
2. A monitoring system: How will a program accept data?

3. Space classification: How will a model analyse the poler's surroundings?
4. Detecting Form: What moves are considered dangerous?
5. Flagging to a relative authority: How will the poler access aid before injury?
6. Hardware: What devices are needed to fulfil this solution?

6.3 Design

Obtaining Safety Measures

In order to create a solution that can identify a situation which is unsafe, this environment needs to be defined. In this scenario, a safe environment is one where there is ample room to move around the pole without colliding with obstacles.

Obstacles around the poler can be material items. These items may be 'crash' mats, or furniture jutting into the pole space. These can be of more concern in a home environment not purpose-built for the poler. Other polers could also be counted as obstacles, which also increases the amount of possible injury. However, other polers are not to be confused with those that are 'spotting' the poler in question, and the system will need to differentiate between the two.

The pole itself can be considered an obstacle if not erected properly. For context, there are three main types of poles; stage, fixed by attachment and fixed by pressure. Stages are freestanding poles that usually remain stationary due to the weight and area of the base. They can shuffle based on factors such as the weight and motion of the poler, and the material on which they are stood. Attachment-mounted poles are usually connected to the ceiling or beam with a ball mount, allowing for no pole sway. A possible way in which these poles would detach would be from poor installation. Pressure mounted poles are built to be tightly fixed between ceiling and floor and may be popular among home polers due to their easy installation, easy disassemble, cost and discrete qualities. However, these poles need to be checked for rigidity often, as changes to the environment can affect their tightness against surfaces. Safety is also relative to the quality of installation. The injury prevention solution needs to maintain that the pole remains perpendicular to the floor and in the centre of picture for the environment to remain safe. It is necessary to acknowledge the existence of ‘floating’ pole as an edge case, but these are used by experienced pole dancers.

Figure 41 Images of a stage pole, attachment fitted pole and floating pole.

The last required measure is safe poler form. This varies according to the experience of the poler; the more experienced they are, the more movement they can perform safely (Lee, Lin, Tan 2020). ‘Inverting’ (the poler going upside-down) is considered an intermediate move and should be flagged as ‘unsafe’ for a beginner poler. The smaller number of limbs or amount skin attached to the pole, and the more height involved, the more risk a certain move is to a poler. The increase of severity of these factors decreases poler safety regardless of the level of experience and needs to be acknowledged by the system.

Monitoring System

A monitoring system needs to be in place to stream video data to the model. A recording device could be suspended in each pole’s region. This could be costly. Another method could be situating a sole camera in full view of the studio floor. This would increase the complexity of the solution, by either needing to split the incoming frames into each pole section before classification or having a well-trained model which could consider the whole setting. For the rest of this research, it will be assumed that the solution works with separate individual poles.

There are ethical problems surrounding the use of ‘always-on’ monitoring devices. The way in which their video data is managed would need to be made transparent to polers. Even though the data needs to be streamed to a model for it to classify, it should not need to be stored anywhere afterwards. However, it may be useful to give polers the options to send their data to application development, whereby it can be used to further train and improve the model. Polers should be surveyed on their thoughts on being recorded during a session. The camera system could double as a recording device to capture their choreography during a session, which may make them feel as if the solution is benefitting them through progression logging.

Spatial Awareness

The model would need an element to assess whether the space is safe. The first concern is the pole, whereby it must be in the centre of the view and standing at a 90-degree angle to the floor, pointing straight to the ceiling. This could be found by using classical image processing methods, provided that live imagery is captured with a level camera. This follows the pattern of converting the image frame to binary, extracting objects of interest using contours (and central location), then finding the orientation of the pole using a means of principle component analysis (OpenCV 2022).

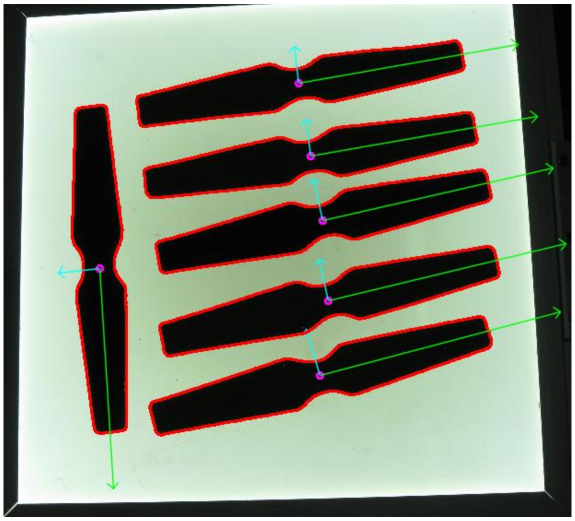


Figure 42 Obtaining the orientation of objects using principal component analysis (OpenCV 2022).

The poling space needs to be analysed to detect whether there is ample area to move around the pole without collision with walls or objects. Camera devices such as the Kinect have a laser scanner which can analyse depth. These can take distance readings, to ensure there is ample room at each side of the pole. Robots uses packages such as 'laserscan' ('MichalDrwiega' 2021) in ROS to avoid objects, and so these principles can be taken into pole obstacle detection. Similarly, the Kinect used this technology to identify clear space to play motion tracking games on the Xbox (Park et al. 2012). To avoid detecting the instructor 'spotting', the system could be switched off (due to adequate supervision) or avoid detection in the immediate area adjacent to the pole.

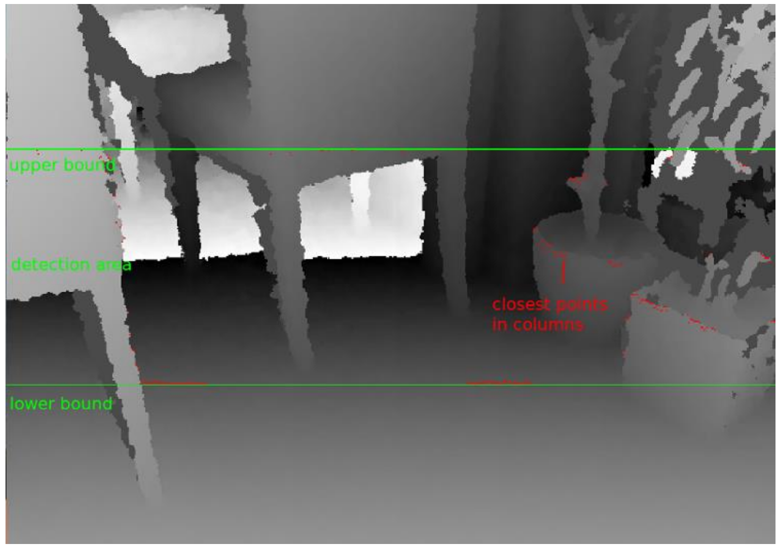


Figure 43 ROS laserscan used to identify close objects within specified bounds (‘MichalDrwiega’ 2021).

Detecting Form

As seen in chapter 4, it is possible to classify pole moves using pose estimation. This can be used to flag more intermediate moves, such as the ‘butterfly’, as ‘dangerous’ to a beginner poler. Move ‘risk’ can then be tuned based on the poler’s level of expertise, to avoid risking injury (Lee, Lin, Tan 2020).

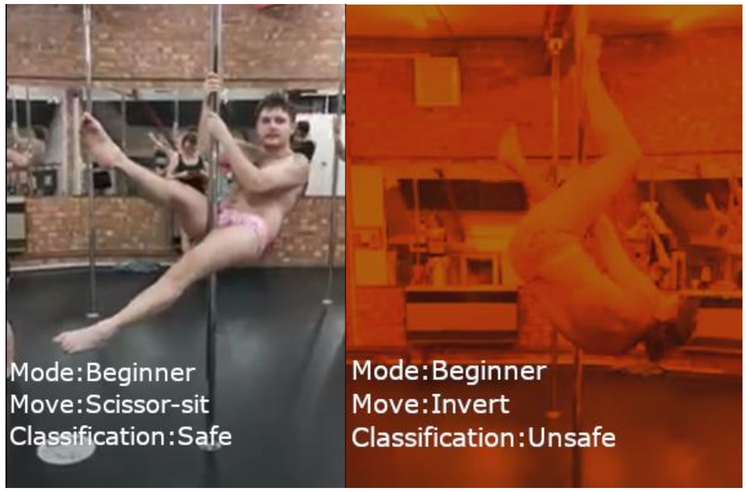


Figure 44 Move classification used as a form of risk determination (simulated).

On a more granular level, regardless of poler experience, 'risk' can be decided by how far away from the pole the poler is during a move, and how many attachments the poler has to it, as polers require skin contact to adhere safely (Baldin, Menegucci 2017). Landmarks can be obtained through pose detection methods (such as 'OpenPose' and 'BlazePose'), and distances can be calculated between them. In theory, points on the pole could be considered 'joints' or 'landmarks' to be used in these calculations. This would also allow the model to know when items such as wrists are lower down on the pole, indicating that the move is more load bearing on them, involving higher risk (Szopa et al. 2022).

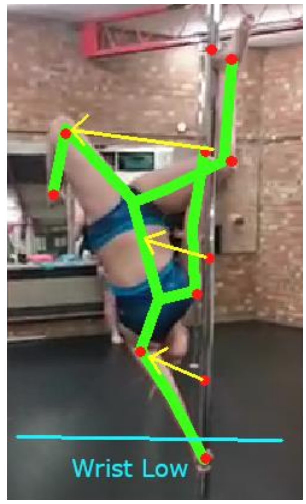


Figure 45 Distances (yellow) of pole landmarks to body landmarks, and the position of wrists (simulated detection).

Gaining training data will be more difficult for ‘risky’ poses, as participants used for capturing data would need to be experienced in complex moves. An alternative could be capturing landmark points from simulated data (Nishi, Miura 2017) which would also decrease the risk of injury in training dataset collection. The ‘risk level’ needs to be established based on the factors of distance and wrist position, which could be assigned by a group of experienced polers and healthcare professionals. It could be added as a label to ‘pose’ sample data features, as a float indicating ‘risk’ on a normalised scale of 0-1.

Risk level would not be needed to be organised for every possible set of landmark positions, as a regression system could be used to infer the risk level given a set of established samples (Diaz-Quijano 2012). A model could then be used to determine the risk level of live poses based on seen samples and can classify as ‘dangerous’ based on a set threshold, in a binary fashion. This approach changes the problem from one based off classification alone to one of regression.

Flagging anomalies & Receiving aid

Once a situation is flagged as ‘dangerous’ within live video for a certain number of frames, an authority needs to be told, namely the instructor during a pole lesson. A device which causes some form of indication, such as noise (Beyea 2007) or light (Morgan et al. 2020), could cause the poler distraction, causing a fall (Chi, Chang, Ting 2005). A safer approach could be indicating the location of the ‘dangerous’ pole on a monitor that the instructor is aware of or having an earphone which relays this information directly to them. This allows them to be at hand to ‘spot’ if needed.

When the program is running at home (where an instructor is not at hand to help), the solution would act more so as an injury signaller. When the model notices a danger, it can send the relevant frames to a next of kin or guardian, to function as a call for aid. Even if no injury occurs from a fall, or a ‘missed’ fall, the poler can use this video data to learn to prevent the danger from occurring in the future.

Hardware & Software Tools

The hardware and software necessary to conduct the studied requirements are as follows:

Software

1. An IDE and relevant programming language efficient at working with data (e.g. Python).
2. Modelling tools (e.g., blender ‘Pose’ (2022)), a system to generate more training images of dangerous poses.
3. OpenCV (2022) to perform principal component analysis.
4. Libraries to perform spatial awareness (e.g. OpenCV and Kinect SDK (Microsoft 2022)).
5. A pose detection library (such as OpenPose or BlazePose).
6. A programming library that can perform regression (e.g. Scikit-Learn).

7. A camera with a laser sensor and 'internet of things' capabilities to send data (video and laser output) to a computer.
8. A cloud service that can perform serverless computing by sending flagged video data to the relevant personnel.

Hardware

1. A camera to record pole landmark data.
2. A CPU/GPU that can learn and save a risk regression model based off landmark data and professional input.
3. A computer than can evaluate and run the injury prevention software.
4. A Bluetooth earpiece that can connect to the computer to receive information on 'dangerous' flagged poles.

6.2 Visual Prototype

The following visual prototype depicts the setup for notifying an instructor of danger based on a pole flagged with 'risk'.

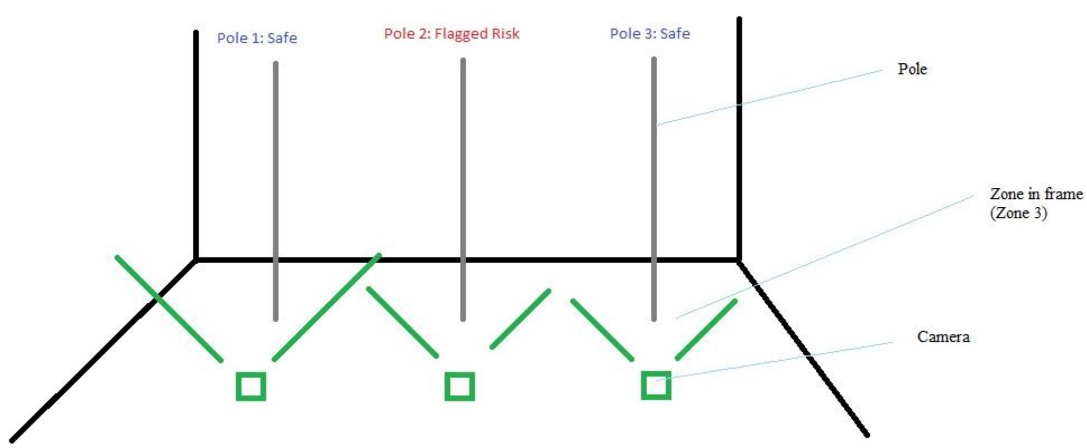


Figure 46 Pole studio setup with system flagging a pole as a risk (prototype).

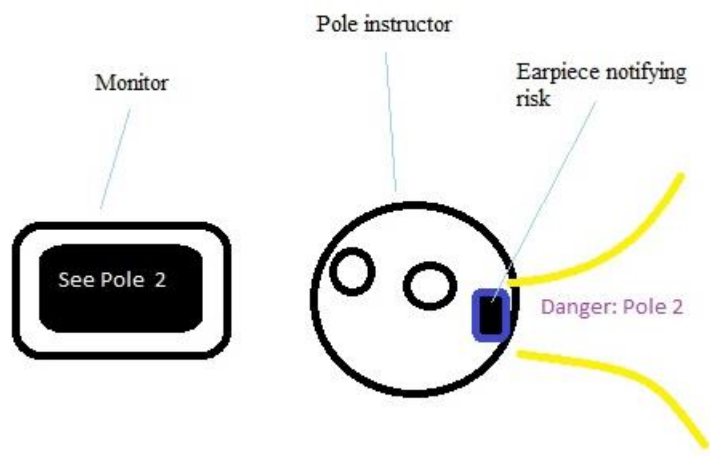


Figure 47 A pole instructor receiving information on a poler at risk (prototype).

6.2 Evaluation

In order to create an injury preventative pole sport solution using vision, a blend of image processing, object detection and pose detection may be incorporated. Development needs to be inclusive of the wishes of the poler, and experienced aerial artists and medical professionals should be collaborated with to ensure risks are defined correctly. Data to train models can be collected using generative means to avoid injury in the creation of the artefact. Regression methods can be used to infer risk seen in real-time data, but the solution will need rigorous safe testing.

The proposed solution was drafted using research of literature and software. To ensure the artefact is viable, further studies are needed on the poler's view of being recorded on video, and the possible uptake on this software. These could be carried out using qualitative approaches such as surveys and focus groups. The ethics surrounding the solution needs to be recognised, both in training a model with dangerous positions, and avoiding over-reliance in its use. Like that of machine-learning methods of medical diagnosis, this model should not have full responsibility for human safety and should act as an aid to spot risks before they happen (Russell, Norvig 2016 p1036), by influencing the pole instructor to stay alert. Accountability remains in the hands of the instructor. A safe testing environment needs to also be proposed whereby people are at minimal risk of injury.

Chapter 7

Conclusions

It is conclusive that integrating artificial intelligence with pole dance includes computer science domains such as vision, machine learning and cloud computing, whilst also incorporating other fields such as data science, user experience and ethics. Involving technology in pole sport is an interdisciplinary task but can positively affect the community by providing a variety of useful tools.

Using pose detection with the K-Nearest-Neighbour algorithm is an accurate method of recognising staple pole moves. The system can easily be incorporated into a mobile application to ease the sharing of pole names and instructive resources. BlazePose is an accurate pose detection library, but it needs to be noted that the occlusion of a face in a piece of training data does affect the quality of the landmarks sample. This artefact invites research in studying the classification of move variations.

The censorship algorithm study showed that certain pole imagery is inherently non pornographic, shown by the large separation between pole and explicit classes. Training a model using generic non-pornographic data helps differentiate explicit content from a smaller amount of non-pornographic specific data. Including pole imagery in the training of a model is key to reduce the chances of pole dance imagery being incorrectly flagged as explicit by social media systems. Video assessment requires a dynamic approach, using the correct metric to summarise content, and may need to utilise a more progressive discrimination threshold than image classification. Reducing assessed video frames can give rise to similar classification and uses less resources but leaves room for loophole exploitation. Therefore, censorship maintains the need to be a collaboration between a platform's users and the moderation team.

Further studies need to me made using a larger diverse pole dataset to determine if all forms of pole dance in different settings are inherently non-pornographic. Working with social media companies is the solution to making their particular algorithmic system more inclusive. The likelihood of pole resembling other censored topics such as violent and graphic imagery also needs to be researched.

Groundwork has been provided for the creation of an injury prevention system in pole dance. It may require approaches such as object detection, regression, and classification. The input of experienced polers and medical professionals needs to be included in the design of the artefact. The solution should aid instructors to become more alert of dangers to students and not relieve themselves of responsibility. Further user experience studies need to be carried out to analyse the full uptake of this system, including factors such as cost and risk to health.

The advances made in this research could possibly be combined into a sole pole-specific social application; one that can perform move recognition, image portfolio logging (without censorship), and examination of the studio environment for hazards. This work can also be extended to forms of aerial arts such as hoop, silks, and types of gymnastics. Research that reproduces and improves on these results and artefacts is welcomed, especially in the case where it will holistically uplift the pole sport community.

Chapter 8

Reflective Analysis

The development of this project was an interesting one as I originally did not plan it to take the form of three parts. The Gantt chart created on project proposal only covered the ‘point your pose’ application, and time was overestimated substantially. I found it difficult to balance time between the remainder of the project, and perhaps creating a new Gantt chart and project plan would have made that easier.

Communication went smoothly whilst working on this project. I had a good rapport with my supervisor Dr James Brown, who pushed me to work hard to reach this project’s potential at every meeting. I was lucky to have contact with Dr Sandra Avila (**APPENDIX K**) to receive the pornography dataset promptly for my research. I have made good connections with Dr Carolina Are on basing my censorship study off her research, and I hope this work is carried forward.

The pole image data collection was a challenge. I had to dedicate time for this collection each week after pole classes, but luckily, I had good collaboration with the instructor who understood my research. I should have made more of a consideration when moving the data from use in the pole move app to the censorship study. A more varied dataset would have proved useful, and it may have been easier to collect due to not requiring the obtainment of specific moves. It would have strengthened the ‘pole dance is not pornographic’ argument by increasing diversity in training data. However, it would pose a new challenge as I would have to rely on areas such as Facebook groups for collection again, which did not prove that successful previously. However, after showing group members the results of the recognition application, this may make future interaction fruitful.

My research methodology enhanced throughout these studies. The aim of the project was to make each part reproducible, but there is space for improvement. For instance, using random state and cross-validation for all trained models would have given metrics that could be accurately reproduced, and randomising data partitions and keeping a log of which exact ones were used in training would help study replication. Some workflow aspects can be improved upon; streamlining data splits outside of the colab notebooks may prove more robust and would not impact the RAM limit of a particular notebook session.

In my opinion, pole has had a hugely positive impact on my life, with both the fitness and social impact greatly improving both my physical and mental health, as well as my social and body confidence. Dr Carolina Are's (2021) work raises a valid point in that it is important to acknowledge bias when researching a domain that is inclusive of one's personal experience (Mitra 2010). Observations about studios and pole dance behaviours are based on somewhat anecdotal information, but this is due to this lack of global study in the area. Censorship has personally affected my pole social media posts; however, this study follows a data science approach as opposed to an autoethnographic one, offering a new perspective on the case of censorship. My study is one effort to help dismantle the stigma associated with pole and all its forms. I hope that my research is built upon by both polers and non-polers alike, of diverse gender and social backgrounds.

References

Android Studio (2022) Available from: https://developer.android.com/studio [Accessed 10th April 2022].

Ap-apid R. (2005). An Algorithm for Nudity Detection. https://www.researchgate.net/publication/249767252_An_Algorithm_for_Nudity_Detection [accessed 15th February 2022].

Ardito L., Coppola R., Malnati G., Torchiano M. (2020) Effectiveness of Kotlin vs. Java in android app development tasks, Information and Software Technology, 127, 106327. Available from: https://www.sciencedirect.com/science/article/pii/S0950584920301439?via%3Dihub [Accessed 10th April 2022].

Are C. (2021) The Shadowban Cycle: an autoethnography of pole dancing, nudity and censorship on Instagram, *Feminist Media Studies*. Available from: Full article: The Shadowban Cycle: an autoethnography of pole dancing, nudity and censorship on Instagram (tandfonline.com) [Accessed 8th April 2022].

Avila A., Thome N., Cord M., Valle E. and Araújo A.D.A. (2011) BOSSA: Extended BoW Formalism for Image Classification. In: *18th International Conference on Image Processing (ICIP)*, pp.2966-2969, Brussels, Belgium.

Avila A., Thome N., Cord M., Valle E., and Araújo A.D.A (2013) Pooling in Image Representation: The Visual Codeword Point of View. *Computer Vision and Image Understanding* (CVIU), 117(5), pp.453-465.

Baldin AEC. and Menegucci F. (2017) Ergonomic apparel pole for practicing sports: thermal comfort as a design requirement, *Projetica* (Portuguese), 8(2), pp.113-126.

Bazarevsky V., Grishchenko I., Raveendran K., Zhu T., Zhang F. and Grundmann M. (2020) BlazePose: On-device Real-time Body Pose tracking. *Google Research*, https://arxiv.org/pdf/2006.10204.pdf [accessed 14th February 2022].

Beyea, S.C. (2007) 'Noise: a distraction, interruption, and safety hazard', *AORN Journal*, 86(2), 281+. [online] Available from:

https://link.gale.com/apps/doc/A167775475/AONE?u=anon~59945730&sid=googleScholar&xid=5608dee2 [accessed 12 May 2022].

Blender (2022) Pose, Blender Manual. Available from: https://docs.blender.org/manual/en/latest/sculpt_paint/sculpting/tools/pose.html [Accessed 12 May 2022].

Caetano, C., Avila, S., Guimaraes, S. and Araújo, A.D.A. (2014) Pornography detection using bossanova video descriptor. In: *IEEE 2014 22nd European Signal Processing Conference* (EUSIPCO), pp.1681-1685.

Cambridge Consultants (2019) Use of AI in online content moderation. 2019 *Report Produced on Behalf of OFCOM*.

Cao Z., Hidalgo G., Simon T., Wei S. and Sheikh A. (2019) OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(1), pp. 172-186.

Central Digital & Data Office (2020) Data Ethics Framework, *Gov.uk*. Available from: https://www.gov.uk/government/publications/data-ethics-framework/data-ethics-framework-2020 [Accessed 9th May 2022].

Chi C-F., Chang T-C. and Ting H-I. (2005) Accident patterns and prevention measures for fatal occupational falls in the construction industry, *Applied Ergonomics*, 36(4), pp.391-400.

Cobbe J. (2021) Algorithmic Censorship by Social Platforms: Power and Resistance. *Philos. Technol*, 34, pp.739–766. https://doi.org/10.1007/s13347-020-00429-0 .

Davies E.R. (2018) Surveillance. In: *Computer Vision Principles, Algorithms, Applications, Learning*. 5th edition. Academic Press via Elsevier Inc.

Diaz-Quijano F. (2012) A simple method for estimating relative risk using logistic regression, *BMC Medical Research Methodology*, 12(14). [online] Available from: https://link.springer.com/article/10.1186/1471-2288-12-14#citeas [Accessed 12 May 2022].

Dimitrijevic M., Lepetit V. and Fua P. (2006) Human body pose detection using Bayesian spatio-temporal templates. *Computer Vision and Image Understanding*, 104, pp.127-139.

Dittich F, Beck S., Burggraf M., Busch A., Dudda M., Jäger M. and Kauther MD. (2020) A small series of pole sport injuries. *Orthopedic Reviews*, 12(3), pp.8308.

Dohyung K., Dong-Hyeon K. and Keun-Chang K. (2017) Classification of K-Pop Dance Movements Based on Skeleton Information Obtained by a Kinect Sensor. *Sensors*, 17(6), pp.1261.

Dombrowski Q. (2011) Stripper Giantess, *Flikr*. Available from: Stripper giantess | Quinn Dombrowski | Flickr [Accessed 8th April 2022].

EveryBody Visible (2019) https://everybodyvisible.com/ [accessed 15th February 2022].

Fang Q., Li H., Luo X., Ding L., Luo H. and Li C. (2018) Computer vision aided inspection on falling prevention measures for steeplejacks in an aerial environment. *Automation in Construction*, 93, pp.148-164.

Fazelpour A., De-Arteaga M. (2022) Diversity in sociotechnical machine learning systems, *Big Data & Society*, 9(1).

Fennel, D. (2020) Pole Sports: Considering Stigma. *Sport, Ethics and Philosophy*. Available from: https://doi.org/10.1080/17511321.2020.1856914 [Accessed 14th February 2022].

Gayhardt L., Gronlund C.J., Quintanilla L., 'atikmapari' (GitHub), Arya H. (2022) Machine learning fairness (preview), *Microsoft Docs*. Available from: https://docs.microsoft.com/en-us/azure/machine-learning/concept-fairness-ml [Accessed 22 April 2022].

Gillespie, T. (2018) *Custodians of the internet: platforms, content moderation, and the hidden decisions that shape social media*. New Haven: Yale University Press

Google (2022) Google Cloud Privacy Notice. Available from: https://cloud.google.com/terms/cloud-privacy-notice [Accessed 9th May 2022].

Google (2022) Google Forms. Available from: https://www.google.co.uk/forms/about/ [Accessed 9th May 2022].

Google (2022) Responsible AI Practices, *Google AI*. Available from: https://ai.google/responsibilities/responsible-ai-practices/?category=fairness [Accessed 22 April 2022].

Google Colab (2022) Available from: https://colab.research.google.com/ [Accessed 10th April 2022].

Google Developers (2022) Pose Classification Options. *MLKit Guides*, https://developers.google.com/ml-kit/vision/pose-detection/classifying-poses [accessed 14th February 2022].

Google Lens (2020) https://lens.google/ [Accessed 14th February 2022].

Google MLKit (2022) Vision-Quickstart. Google Samples Github (MLKit). Available from: https://github.com/googlesamples/mlkit/tree/master/android/vision-quickstart [Accessed 11th May 2022].

Gradle (2022) Gradle Build Tool. Available from: https://gradle.org/ [Accessed 11th April 2022].

Gruszczyński D., Stefańczyk M. (2021) Active fall prevention: robotic vision in AAL, *arXiv*, https://arxiv.org/abs/2103.09298 .

Health and Safety Executive (2019) Evidence review of the effectiveness of soft-landing systems for preventing injury from falls when working at height. Available from: https://www.hse.gov.uk/research/rrpdf/rr1145.pdf [Accessed 19th April 2022].

Information Commissioner's Office (2018) Guide to the General Data Protection Regulation, *Gov.uk*. Available from: https://www.gov.uk/government/publications/guide-to-the-general-data-protection-regulation [Accessed 9th May 2022].

Instagram (2022) Community Guidelines. https://help.instagram.com/477434105621119/ [accessed 14th February 2022].

Instagram (2022) How do I report a child under the age of 13 on Instagram? *Instagram Help Centre*. Available from: https://help.instagram.com/517920941588885 [Accessed 5th May 2022].

Java (2022) https://www.java.com/en/ [Accessed 10th April 2022].

Joanna N.C., McDonalds K.A., Peeling P, Jackson B, Dimmock J.A., Alderson J.A. and Donnelly C.J. (2019) Pole Dancing for Fitness: The Physiological and Metabolic Demand of a 60-Minute Class, *The Journal of Strength and Conditioning Research*, 33(10), pp.2704-2710.

Kaplish L. (2018) Sun salutations and yoga synthesis in India. Wellcome Collection. Available from: https://wellcomecollection.org/articles/WnBAsSoAACsA5tuj [Accessed 8th April 2022].

Keras (2022) Available from: https://keras.io/ [Accessed 10th April 2022].

Kotlin (2022) https://kotlinlang.org/ [Accessed 10th April 2022].

Lee J.Y., Lin L. and Tan A. (2020) Prevalence of pole dance injuries from a global online survey. *The Journal of Sports Medicine and Physical Fitness*, 60(2), pp.270-275.

Lin T., Maire M., Belongie S., Bourdev L., Girshick R., Hays J., Perona P., Ramanan D., Zitnick C.L. and Dollár P. (2015) *Microsoft COCO: Common Objects in Context*. arXiv:1405.0312. https://arxiv.org/abs/1405.0312, [accessed 14th February 2022].

Martinez J (2020) Plant Identification, *Flikr*. Available from: https://www.flickr.com/photos/inannabintali/49917674616 [Accessed 8th April 2022].

Massey, K., Burns, J. and Franz, A. (2021) Young People, Sexuality and the Age of Pornography, *Sexuality & Culture* 25, pp.318–336.

Matplotlib (2022) Available from: https://matplotlib.org/ [Accessed 11th April 2022].

MediaPipe (2022) Pose Classification. Available from: https://google.github.io/mediapipe/solutions/pose_classification.html [Accessed 10th April 2022].

MichalDrwiega (2021) laserscan_kinect Package Summary, ROS (Robot Operating System) Wiki. Available from: http://wiki.ros.org/laserscan_kinect [Accessed 12 May 2022].

Microsoft (2022) Kinect SDK 2.0. Available from: https://www.microsoft.com/en-gb/download/details.aspx?id=44561 [Accessed 12 May 2022].

Microsoft (2022) Microsoft Responsible AI principles in practice, *Microsoft AI*. Available from: https://www.microsoft.com/en-gb/ai/responsible-ai?activetab=pivot1%3aprimaryr6 [Accessed 22 April 2022].

Microsoft (2022) Microsoft Teams. Available from: https://www.microsoft.com/en-us/microsoft-teams/group-chat-software [Accessed 11th April 2022].

Mindoro J.N., Festijo E.D. and de Guzman M.T.G (2021) A Comparative Study of Deep Transfer Learning Techniques for Cultural (Aeta) Dance Classification utilizing Skeleton-Based Choreographic Motion Capture Data. In: *International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*, pp.74-79.

Mitra R (2010) Doing Ethnography, Being an Ethnographer: The Autoethnographic Research Process and I. *Journal of Research Practice* ,6 (1), pp.1–21.

Moore M.S., Popovic N.A., Daniel, J.N., Boyea S.R. and Polly JR, D.W. (1997) The effect of a wrist brace on injury patterns in experimentally produced distal radial fractures in a cadaveric model. *The American Journal of Sports Medicine*, 25(3), pp.394-401.

Morgan P., Macken B., Toet A., Bompas A., Bray M., Rushton S. and Jones D. (2020) Distraction for the eye and ear, *Theoretical Issues in Ergonomics Science*, 21(6), pp.633-657.

Mosseri, A (2022) Control your Instagram Feed with Favorites and Following. *Official Instagram Blog*. Available from: https://about.instagram.com/blog/announcements/favorites-and-following [Accessed 4th May 2022].

Moustafa M.N. (2015) Applying Deep Learning to Classify Pornographic Images and Videos. In: *7th Pacific-Rim Symposium on Image and Video Technology (PSIVT)*, Auckland, New Zealand.

Naczk M., Kowalewska A. and Naczk A. (2020) The risk of injuries and physiological ben-efits of pole dancing. *The Journal of Sports Medicine and Physical Fitness*, 60(6), pp.883-888.

Nishi K. and Miura J. (2017) Generation of human depth images with body part labels for complex human pose recognition, *Pattern Recognition*, 71, pp.402-413.

Nodarakis N., Rapti A., Sioutas S., Tsakalidis A.K., Tsolis D., Tzimas G., Panagis Y. (2017) (A)kNN Query Processing on the Cloud: A Survey. *Lecture Notes in Computer Science*, pp.26–40. Available from: http://dx.doi.org/10.1007/978-3-319-57045-7_3 [Accessed 14 May 2022].

Not The MainStream (2022) The Pole Dance Companion. https://www.notthemainstream.net/poledance-companion/?cookie-state-change=1644827378684 [Accessed 14th February 2022].

NumPy (2022) Available from: https://numpy.org/ [Accessed 11th April 2022].

OpenCV (2022) Available from: https://opencv.org/ [Accessed 11th April 2022].

OpenCV (2022) Introduction to Principal Component Analysis (PCA). *OpenCV Tutorials*. Available from: https://docs.opencv.org/4.x/d1/dee/tutorial_introduction_to_pca.html [Accessed 12 May 2022].

Pandas (2022) Available from: https://pandas.pydata.org/ [Accessed 11th April 2022].

Park, J.-H., Shin, Y.-D., Bae, J.-H. and Baeg, M.-H. (2012) 'Spatial Uncertainty Model for Visual Features Using a KinectTM Sensor', *Sensors*, MDPI AG, vol. 12, no. 7, pp. 8640–8662.

Pillow (2022) https://pillow.readthedocs.io/en/stable/# [Accessed 11th April 2022].

Pole Fit Freedom (2019) The History of Pole Dancing: Where it all Began. *Pole Fit Freedom*. Available from: https://www.polefitfreedom.com/history-of-pole-dancing/. [Accessed 8th April 2022]

PolePedia (2020) Global Pole Dance Studio Map. Available from: https://polepedia.com/pole-studio-map/ [Accessed 18th May 2022].

Protopapadakis E., Voulodimos A., Doulamis A., Camarinopoulos S., Doulamis N. and Miaoulis G. (2018) Dance Pose Identification from Motion Capture Data: A Comparison of Classifiers. *Technologies*, 6(31), pp.1-16.

Python (2022) https://www.python.org/ [Accessed 10th April 2022].

Rogez G., Rihan J., Ramalingam S., Orrite C. and Torr P.H.S (2008) Randomized trees for human pose detection. In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp.1-8.

Russell S., Norvig R. (2016) Philosophical Foundations. In: *Artificial Intelligence: a Modern Approach*, EBook, Global Edition, Pearson Education, Limited, Harlow. Available from: ProQuest Ebook Central. [Accessed 12 May 2022].

Scikit-Learn (2022) Machine Learning in Python. Available from: https://scikit-learn.org/stable/index.html [Accessed 10th April 2022].

Seaborn (2022) Available from: https://seaborn.pydata.org/ [Accessed 11th April 2022].

Sharma R., Choubey D.K. (2016) Sport of Mallakhamb : A Traditional Game of Indian Culture. In: *Indian Journal of Physical Education, Sports and Applied Sciences*, 6(1), pp.22-32.

Shazam (2022) Available from: https://www.shazam.com/gb [Accessed 14th February 2022].

Szegedy C., Vanhouke V., Ioffe S., Shlens J. and Wojna Z (2015) Rethinking the Inception Architecture for Computer Vision. *arXiv:1512.00567*. Available from: https://arxiv.org/abs/1512.00567 [Accessed 11th April 2022].

Szegedy C., Vanhouke V., Ioffe S., Shlens J., and Wojna Z. (2015) Rethinking the Inception Architecture for Computer Vision, *arXiv:1512.00567*. https://arxiv.org/abs/1512.00567.

Szopa, A., Domagalska-Szopa M., Urbańska A., and Grygorowicz M. (2022) Factors associated with injury and re-injury occurrence in female pole dancers. *Scientific Reports*, 12(33). https://doi.org/10.1038/s41598-021-04000-5 .

Thar M.C., Winn K.Z.N. and Funabiki N. (2019) A Proposal of Yoga Pose Assessment Method Using Pose Detection for Self-Learning. In: *International Conference on Advanced Information Technologies* (ICAIT), pp.137-142.

Trevithick B, Mellifont R, Sayers M (2020) Wrist pain in gymnasts: Efficacy of a wrist brace to decrease wrist pain while performing gymnastics, *Journal of Hand Therapy*, 33(3), pp.354-360.

Vogelsang, A. and Borg, M. (2019) Requirements engineering for machine learning: Perspectives from data scientists. In: *2019 IEEE 27th International Requirements Engineering Conference Workshops (REW)*, pp. 245-251, IEEE.

Weaving C. (2020) Sliding Up and Down a Golden Glory Pole: Pole Dancing and the Olympic Games. *Sport, Ethics and Philosophy*, 14(4), pp.525-536.

Wehrmann J., Simões G.S., Barros R.C. and Cavalcante V.F. (2018) Adult Content Detection in Videos with Convolutional and Recurrent Neural Networks. *Neurocomputing*, 272, pp.432-438.

Whitehead K., Kurz T. (2009) 'Empowerment' and the Pole: A Discursive Investigation of the Reinvention of Pole Dancing as a Recreational Activity. In: *Feminism & Psychology*, 19(2), pp.224-244.

Wikipedia Commons (2005) Chinese Pole Dance Image. Available from: File:Chinese Pole Dance.jpg - Wikimedia Commons [Accessed 8th April 2022].

X-Pole (2022) Leaders in Pole & Aerial Fitness. Available from: https://x-pole.co.uk/ [Accessed 19th April 2022].

Word Count

12,491

Appendix

A

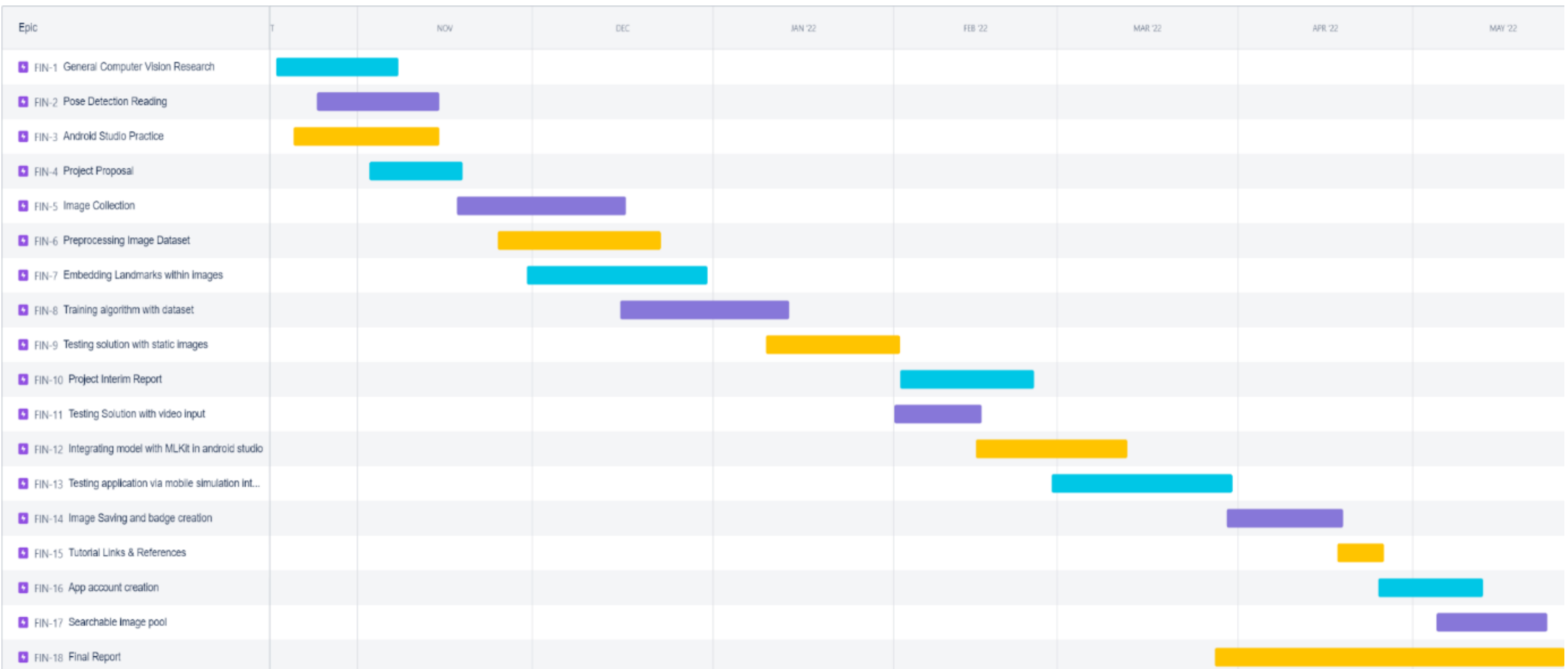


Figure 48 Gantt chart for 'point your pose' application overestimating the time taken to develop a basic solution.

B

Table 13 Original Contingency plan involving risk analysis on project proposal.

Risk no.	**Risk Details**	**Likelihood**	**Assessed Impact**	**Mitigation Action**
1	External pole studios become un-cooperative with image collection.	Medium	Medium	Ensure many studios and communities are contacted. Increase reliance on local studio's image collection.
2	Image collection device (mobile camera) breaks	Low	Medium	Ask polers to record/ take pictures with their own device.
3	A poler wishes to withdraw from the project	Medium	Medium/High (depending on images collected from the individual)	Remove Images associated with the poler from the project, along with their personal details. Obtain more data from other polers.

4	Google Colab becomes dysfunctional	Low	High	Migrate project to Jupyter Notebooks on personal computer.
5	Mediapipe/ MLkit become proprietary software.	Low	High	Seek other forms of pose recognition resources, such as OpenPose.

C

Ethics reference:

Participant Identification Number for this study:

CONSENT TO PARTICIPATE IN RESEARCH

Title of Project: "Poler Vision: Uplifting the Pole Sport Community with Artificial Intelligence"

Name of Researcher: Oakleigh Weekes

Name of Participant:

Please initial box

1. I confirm that I have read the information sheet dated 29/10/2021 (version 1) for the above study. I have had the opportunity to consider the information, ask questions and have had these answered satisfactorily. ☐
2. I understand that my participation is voluntary and that I am free to withdraw at any time without giving any reason, without my legal rights being affected.

   I understand that should I withdraw then the information and imagery I have given us up to this point will be deleted/destroyed. However, once the anonymised data set has been created it may not be possible to remove my anonymised data from the analysis. Images will only be published if further consent is given (Q5), and otherwise will only be used anonymously in training & testing machine learning algorithms. ☐
3. I understand that individuals from the University of Lincoln may look at research data collected during the study, to ensure that the study is conducted appropriately. I give permission for these individuals to have access to my records; I understand that my personal details shall be kept confidential. ☐
4. I would like to receive a summary of the results of the study ☐
5. I consent to allowing images of myself taken as part of the research to be included in the published dissertation (Optional) ☐
6. I agree to take part in the above study. ☐

___________________	29/10/2021	___________________
Name of Participant	Date	Signature
Oakleigh Weekes	29/10/2021	___________________
Name of Person taking consent	Date	Signature

Figure 49 Consent sheet distributed when capturing data in-person

D

Information Sheet - 29/10/2021

1. This study involves examining the use cases of artificial intelligence in pole sport.

2. Videos collected will be converted into image frames to be used in this study.

3. Images collected are used to train a machine learning algorithm. This means that a digital system, a "deep neural network", learns patterns in images, and can use this information to generalise predictions on new, unseen images.

4. Images collected will be stored in a dedicated private Google Drive, to be brought into a series of machine learning programs stored in the cloud.

5. Consent can be withdrawn at any point. Images can be removed simply before anonymisation of the image dataset. However, a best attempt can be made to remove your images if you wish after dataset anonymisation.

6. Images will not be made public unless given consent to do so for the purpose of the published dissertation. If published, faces will be masked by a blur to keep anonymity.

7. The part of the dataset that is not published will be deleted on termination of the project.

8. Please email 18678532@students.lincoln.ac.uk for any further information or if you have any questions.

Figure 50 The information sheet handed out when obtaining consent for study participation and image use (in-person submissions).

E

Pole Dance Pose Recogniser App & Pole Dance AI Image Consent Form

My name's Oakleigh Weekes and I'm a Poler and Computer Science student. For my final year project, I am creating a Computer Vision app that will recognise pole poses you are unsure of. I need your help with training the machine learning algorithm with example moves. If you would like to submit any or all of these poses: Scissor sit/ superman/ butterfly; please fill out this consent form. All images will only be used for training machine learning algorithms and examining use cases of Artificial Intelligence in pole sport. You can also submit multiple images of each type.

If you have any questions please email me at 18678532@students.lincoln.ac.uk.

The name and photo associated with your Google Account will be recorded when you upload files and submit this form. Your email address is not part of your response.

*Required

Email *

Your answer

I would like to give consent for my image(s) to be used to train machine learning algorithms. These trained algorithms will be used in an app that recognises pole moves and examing use cases of AI in pole sport. All data will be deleted on project end. Email 18678532@students.lincoln.ac.uk to remove your image data at any time. *

○ I consent

Next Clear form

Figure 51 Entry page to fill out consent and submission form (online submissions).

F

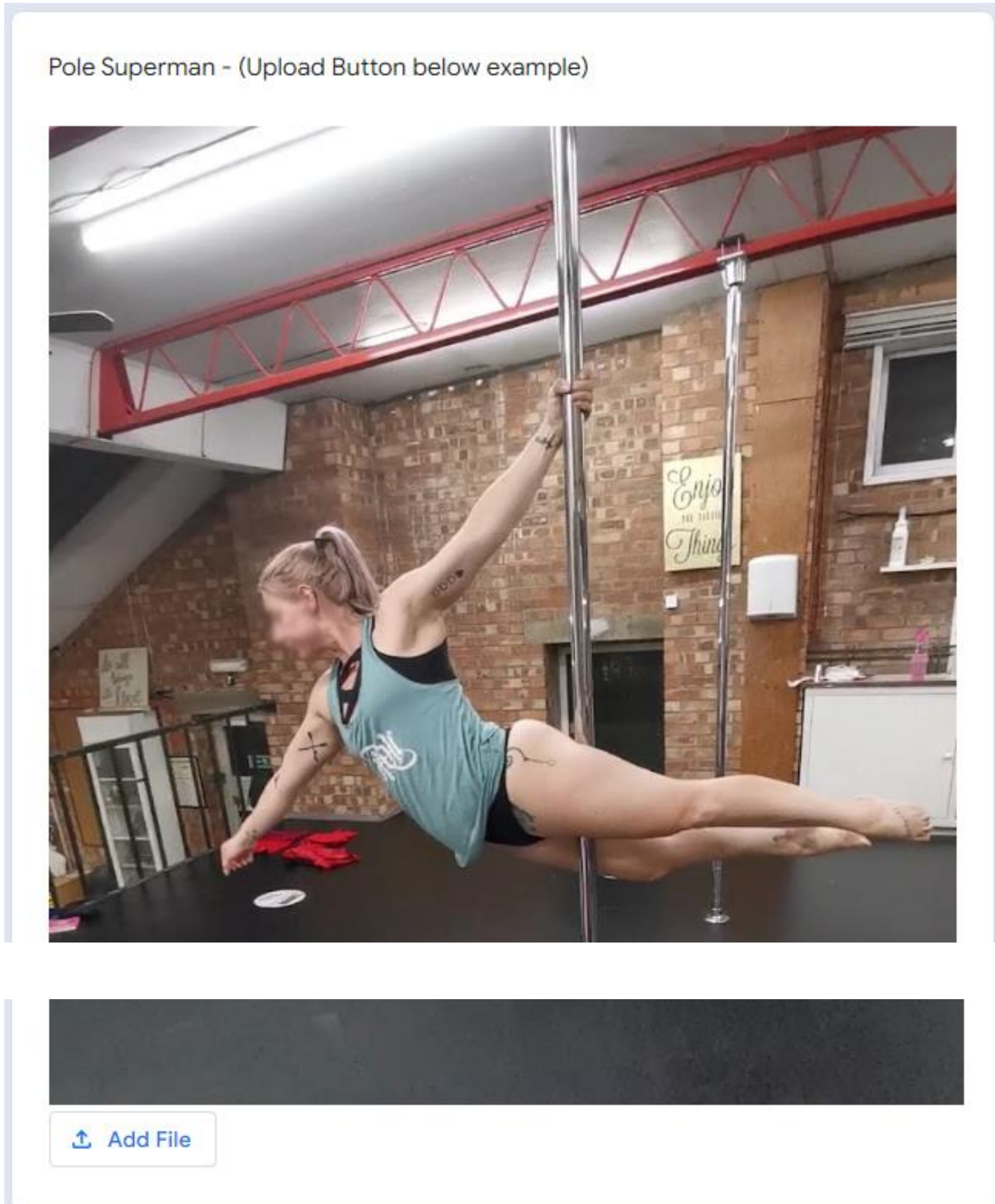


Figure 52 Upload section on the online submission form for 'superman' image.

G

```
// Classification is done in two stages:
//   * First we pick top-K samples by MAX distance. It allows to remove
samples that are almost
//     the same as given pose, but maybe has few joints bent in the other
direction.
//   * Then we pick top-K samples by MEAN distance. After outliers are
removed, we pick samples
//     that are closest by average.

// Keeps max distance on top so we can pop it when top_k size is reached.
PriorityQueue<Pair<PoseSample, Float>> maxDistances = new PriorityQueue<>(
    maxDistanceTopK, (o1, o2) -> -Float.compare(o1.second, o2.second));
// Retrieve top K poseSamples by least distance to remove outliers.
for (PoseSample poseSample : poseSamples) {
  List<PointF3D> sampleEmbedding = poseSample.getEmbedding();

  float originalMax = 0;
  float flippedMax = 0;
  for (int i = 0; i < embedding.size(); i++) {
    originalMax =
        max(
            originalMax,
            maxAbs(multiply(subtract(embedding.get(i),
sampleEmbedding.get(i)), axesWeights)));
    flippedMax =
        max(
            flippedMax,
            maxAbs(
                multiply(
                    subtract(flippedEmbedding.get(i),
sampleEmbedding.get(i)), axesWeights)));
  }
  // Set the max distance as min of original and flipped max distance.
  maxDistances.add(new Pair<>(poseSample, min(originalMax, flippedMax)));
  // We only want to retain top n so pop the highest distance.
  if (maxDistances.size() > maxDistanceTopK) {
    maxDistances.poll();
  }
}

// Keeps higher mean distances on top so we can pop it when top_k size is
reached.
PriorityQueue<Pair<PoseSample, Float>> meanDistances = new PriorityQueue<>(
    meanDistanceTopK, (o1, o2) -> -Float.compare(o1.second, o2.second));
// Retrive top K poseSamples by least mean distance to remove outliers.
for (Pair<PoseSample, Float> sampleDistances : maxDistances) {
  PoseSample poseSample = sampleDistances.first;
  List<PointF3D> sampleEmbedding = poseSample.getEmbedding();

  float originalSum = 0;
  float flippedSum = 0;
  for (int i = 0; i < embedding.size(); i++) {
    originalSum += sumAbs(multiply(
        subtract(embedding.get(i), sampleEmbedding.get(i)), axesWeights));
    flippedSum += sumAbs(
        multiply(subtract(flippedEmbedding.get(i), sampleEmbedding.get(i)),
axesWeights));
  }
  // Set the mean distance as min of original and flipped mean distances.
  float meanDistance = min(originalSum, flippedSum) / (embedding.size() *
2);
```

Figure 53 Code Abstract showing a two-step method of K-NN Classification, as seen in the 'PoseClassifier.java' file.

H

```
public ClassificationResult getSmoothedResult(ClassificationResult
classificationResult) {
  // If we are at window size, remove the last (oldest) result.
  if (window.size() == windowSize) {
    window.pollLast();
  }
  // Insert at the beginning of the window.
  window.addFirst(classificationResult);

  Set<String> allClasses = new HashSet<>();
  for (ClassificationResult result : window) {
    allClasses.addAll(result.getAllClasses());
  }

  ClassificationResult smoothedResult = new ClassificationResult();

  for (String className : allClasses) {
    float factor = 1;
    float topSum = 0;
    float bottomSum = 0;
    for (ClassificationResult result : window) {
      float value = result.getClassConfidence(className);

      topSum += factor * value;
      bottomSum += factor;

      factor = (float) (factor * (1.0 - alpha));
    }
    smoothedResult.putClassConfidence(className, topSum / bottomSum);
  }

  return smoothedResult;
}
```

Figure 54 Smoothing algorithm supplied by MLKit's Vision-Quickstart.

I

Table 14 Full censorship analysis table including training image combinations, splits, and resulting metrics.

Image Classification													
Trained on	**Training**	**Validation**	**Test**	**Pole Test**	**Callback**	**stopping criteria**	**Threshold**	**Standard Test Accuracy**	**Standard Sensitivity**	**Standard Specificity**	**Pole Acc.**	**Pole Sens.**	**Pole Spec.**
Pole; Porn	253; 253	72; 72	37; 37	75	Accuracy	90	0.5	(Against Neutral) 0.514	0.027	1	0.679	0.027	1
"Easy" Neutral; Porn	2100; 2100	600; 600	300; 300	362	Accuracy	90	0.5	0.825	0.656	0.99	0.386	0.656	0.162
"Difficult" Neutral; Porn	2100; 2100	600; 600	300; 300	362	Accuracy	90	0.5	0.913	0.857	0.97	0.396	0.857	0.014
"Easy"/"Difficult" Neutral; Porn	2100(Mix); 2100	600; 600	300; 300	362	Accuracy	90	0.5	0.857	0.97	0.743	0.482	0.97	0.077
"Easy"/"Difficult" Neutral/ Pole; Porn	2100(Mix); 2100	600; 600	300; 300	75	Accuracy	90	0.5	0.628	0.62	0.637	0.696	0.62	1
"Easy"/"Difficult" Neutral/ Pole; Porn	2100(Mix); 2100	600; 600	300; 300	75	Val Accuracy	93	0.5	0.898	0.91	0.887	0.928	0.91	1
"Easy"/"Difficult" Neutral/ Pole; Porn	2100(Mix); 2100	600; 600	300; 300	75	Val Accuracy	93	0.03	N/A	N/A	N/A	0.984	0.99	0.96

"Easy"/"Difficult" Neutral/ Pole; Porn	2450(Mix); 2450	700; 700	350; 350	75	Val Accuracy	93	0.5	0.917	0.983	0.851	0.986	0.98	1
"Easy"/"Difficult" Neutral/ Pole; Porn	2450(Mix); 2450	700; 700	350; 350	75	Val Accuracy	93	0.004	N/A	N/A	N/A	0.79	1	0.88

J

Table 15 Full video classification output table visualising efficiency of created algorithms.

Video Classification (Based on "Best" Model; Trained on Most Easy, Neutral, Pole, Porn images)							
Discrimination Threshold = 0.5	Correct; Safe	Correct; edge	Incorrect				
Video Category	**No. Frames**	**Mean Classification**	**Median Classification**	**Weighted: Beginning**	**Weighted: Centre**	**Weighted: End**	**Weighted: Full**
Pole 01 (Standard Video)	363	0.604	0.651	0.233	0.467	0.373	0.358
every '5th' frame	60	0.623	0.711	0.244	0.478	0.39	0.371
Pole 02 (x2 speed & rotation)	772	0.027	~0	0.015	0.025	0.012	0.017
every '5th' frame	128	0.026	~0	0.015	0.025	0.012	0.017
Pole 03 (Far Perspective)	1082	0.223	0.149	0.109	0.179	0.114	0.134
every '5th' frame	180	0.214	0.143	0.109	0.174	0.106	0.13
Pole 04 (Moving Video)	511	0.056	0.004	0.031	0.05	0.025	0.035
every '5th' frame	85	0.063	0.005	0.037	0.056	0.027	0.04

Pole 05 (UltraViolet Light Video)	1627	0.095	0.014	0.064	0.069	0.03	0.054
every '5th' frame	271	0.093	0.012	0.062	0.068	0.032	0.054
Porn 01 "WsM"	814	0.999	~1	0.5	0.749	0.5	0.583
every '5th' frame	135	0.999	~1	0.503	0.749	0.503	0.585
Porn 02 "WsM"	658	~1	1	0.501	0.75	0.501	0.584
every '5th' frame	109	~1	1	0.505	0.75	0.505	0.586
Porn 03 "MsM"	73	0.806	0.94	0.437	0.605	0.38	0.474
every '5th' frame	12	0.799	0.799	0.49	0.622	0.39	0.5
Porn 04 "MsM"	274	~1	~1	0.502	0.75	0.502	0.584
every '5th' frame	45	~1	~1	0.511	0.75	0.511	0.591
Porn 05 "WsW"	402	0.479	0.429	0.2	0.328	0.281	0.269
every '5th' frame	67	0.49	0.489	0.214	0.333	0.284	0.277
Porn 06 "WsW"	177	0.867	0.998	0.461	0.645	0.412	0.506
every '5th' frame	29	0.877	0.998	0.483	0.658	0.426	0.522
Porn 07 "Cartoon WsM"	414	~1	1	0.501	0.75	0.501	0.584
every '5th' frame	69	~1	~1	0.507	0.75	0.507	0.588
Porn 08 "Cartoon WsM"	39	~1	~1	0.513	0.75	0.514	0.592
every '5th' frame	6	~1	~1	0.62	0.75	0.625	0.667

K

Oakleigh M Weekes (18678532)
To: sandra@ic.unicamp.br
Cc: James Brown

Mon 1/17/2022 2:14 PM

Dear Prof. Dr. Sandra Avila,

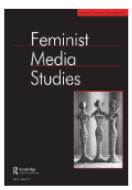
I am a Level 3 BSc Computer Science student at the University of Lincoln, UK, currently undertaking my final year project. My project is based around exploring ways in which Vision and Machine Learning can benefit the Pole Sport Community. So far, I have been able to classify pole moves successfully using pose recognition.

Another problem facing the pole community is the Social Media algorithm. Pole is a sport where showing skin is highly beneficial for being able to grip and perform certain poses and moves. Videos and photos that polers post of their moves are often taken down by the algorithms mistaking them for nudity or pornography (Are ,2021). I want to see if a better classification algorithm can be created by training a CNN on pole images and some from your dataset.

Prior to sending a form and requesting access, I would like to know if your NPDI Pornography Database is still available.

Kind Regards,
Oakleigh Weekes,
18678532.

References:
Are C (2021) The Shadowban Cycle: an autoethnography of pole dancing, nudity and censorship on Instagram. Feminist Media Studies. 1-18. Available from https://www.tandfonline.com/doi/full/10.1080/14680777.2021.1928259



The Shadowban Cycle: an autoethnography of pole dancing, nudity and censorship on Instagram

World risk society and Instagram. This paper applies Beck's (1992, 2006) and Giddens' world risk society theory to Instagram moderation.A sense of security is a crucial part of the social contract, a trade-off between individual liberty and security (Barbara Hudson 2003).Risks are undesired, threatening events (ibid) that become apparent through what Beck calls "techniques of ...

www.tandfonline.com

[accessed 10 November 2021]

Figure 55 Communications with Dr Sandra Avila (I).

Oakleigh M Weekes (18678532)
To: sandra@ic.unicamp.br
Cc: James Brown

Wed 1/19/2022 8:18 AM

NPDI_Pornography_Database...
107 KB

Dear Prof. Dr. Sandra Avila,

Following my last email, I have received approval from my supervisor to carry out my project. Attached is my form for approval.

Kind Regards,
Oakleigh Weekes,
BSc Level 3 Computer Science,
The University of Lincoln, UK.

...

Figure 56 Communications with Dr Sandra Avila (II).

Sandra Avila <sandra@ic.unicamp.br>
To: Oakleigh M Weekes (18678532)
Cc: James Brown

Fri 1/21/2022 3:35 AM

Dear Oakleigh Weekes,

The data can be downloaded at:

[redacted]

We would also kindly ask you to cite the following work on any publications making use of the dataset:
Sandra Avila, Nicolas Thome, Matthieu Cord, Eduardo Valle, Arnaldo de A. Araújo. Pooling in Image Representation: the Visual Codeword Point of View. Computer Vision and Image Understanding (CVIU), volume 117, issue 5, p. 453-465, 2013.

If you have any problems or questions, don't hesitate to contact me.

Sincerely,
Sandra.

...

Prof. Dr. Sandra Avila
www.ic.unicamp.br/~sandra
Institute of Computing
University of Campinas (Unicamp)
Chair, Information Systems Department

Artificial Intelligence Lab. (recod.ai)

Reply | Reply all | Forward

Figure 57 Communications with Dr Sandra Avila (III).