Detecting Facial Expressions in Professional Tennis Matches

Stephanie Kobakian, Mitchell O'Hara-Wild, Dianne Cook, Stephanie Kovalchik

13 February 2017

Contents

1 Introduction **1**
1.1 Project Aim . . . 2
1.2 Sample and sampling approach . . . 2
1.3 Software Selection . . . 3

2 Methodology **3**
2.1 Manual Annotation . . . 5
2.2 Software Interaction . . . 5
2.3 Data Processing . . . 6
2.4 Analysis . . . 6

3 Results **9**
3.1 Results Overview . . . 13

4 Future Work **20**

References **21**

1 Introduction

Many tennis professionals believe that tennis is a game heavily affected by the mental states of the players. The opportunity for researching this "inner game" presents itself with the hope of improving the playing and coaching of tennis players by improving their "mental game". By statistically analyzing the faces and expressions of players during a match there is a hope that insight may be gained into the effects of the mental state on the outcome of a match. Facial expressions during competition provide the most direct insight into a player's thoughts. The aim of this project is to begin to develop methods to collect accurate information about the facial expressions of elite tennis athletes during match play.

In this report, we investigate the performance of several popular facial recognition software's through their Application Programming Interfaces (APIs), and evaluate their performance when applied to the broadcasted videos of elite tennis matches. Using the broadcasted videos gives an objective insight to emotions as the player progresses through a match. While it is impossible to know the thoughts and feelings of a player during a match, professionals may be able to infer this information through results produced by a recognition software. As opposed to the approach of previous studies that have used Player's recollections after a game to determine their emotions.

Making use of the recognition software's currently available presents a challenge as high performance sports are not the intended uses of such software's. Their capabilities are often limited to their intended security and surveillance uses. Barr (2014) addresses the 'lack of robustness of current tools in unstructured environments' that this paper faces and applies to a sports environment. This report aims to analyse the application of these software's to a broadcast to find a suitable software and API to use to analyse a previously recorded tennis broadcast file.

1.1 Project Aim

The aims of the present study were to determine the feasibility of using currently available facial recognition algorithms for extracting facial information from players during broadcasts of professional matches by comparing the performance of several popular facial recognition APIs. This limited selection was based on accessible APIs that we believed would produce appropriate and useful facial recognition. The performance of the evaluated software was compared against manual classification obtained notation tool developed by the authors. In addition to looking at the overall performance, we also evaluated image factors that influences the performance of each service.

1.2 Sample and sampling approach

The goal of the sample was to be representative of the video files that will be used for future facial recognition analysis.

- 6406 Australian Open images (2.8GB)
- 800x450px size frames from 105 match broadcast videos
- Video frames taken every 3 seconds over a 5 minute segment

The sample consisted of a set of 6404 still images. To produce these images, a still shot of the frame was taken at every three seconds, for the length of each 5 minute segment. The stills were provided by Tennis Australia for use in this research, these segments were taken from 105 video files, which were the broadcast of the tennis Matches shown on the Seven Network during the Australian Open 2016. The sample included an equal amount of singles tennis matches played between females and males. The rounds of the competition vary as to not limit the pool of players to only those who progressed, though there was a higher chance of advancing players reappearing.

The sample included images that contained the faces of many people, this included players, staff on the court and fans in the crowd. These faces were included in the manual annotations as they were likely to be found by the software selected. We felt including these additional faces would not only increase the sample by which to judge the software's capabilities but also allow provision of information on how to differentiate between players and other people for further research. Therefore the sample was not filtered at this initial stage.

There are many matches played during the Australian Open, and they are played on the range of courts available at Melbourne Olympic Park. Therefore the sample was selected to be representative of the seven courts that have the Hawk Eye technology enabled.

1.3 Software Selection

The choice of the initial software considered for this research were informed by a report that reviewed 'commercial off-the-shelf (COTS) solutions and related patents for face recognition in video surveillance applications.'

The process of software selection to determine which we would compare was based on several criteria. Firstly, we based our choices on the results of the report as it considered processing speed and feature selection techniques, as well as the ability to perform both still-to-video and video-to-video recognition.

From the software's analysed we considered availability for use within the time frame of the report. This led us to choose Animetrics FaceR. The report outlines that for Animetrics, 'one requirement is that image/face proportion should be at least 1:8 and that at least 64 pixels between eyes are required'. We realize this could present challenges given our data set. It will also allow for an extension from detecting to recognizing people in the data set.

After considering several other off-the-shelf products, we did not choose any other software's from those analysed as they were not as readily available as other products on the market.

This led to SkyBiometry, an API that also allows for both detection and recognition. The cloud-based software as a service, is a 'spin-off of Neurotechnology', a software considered by the report.

We then chose to consider companies who are expanding their API ranges. This resulted in the choice of Microsoft API, provided by Microsoft Cognitive Services. This detects faces and return a square area where the face was located, and predicts facial features. It also allows the possibility of video stream detection.

The final software we chose to analyse was Google Vision API. Due to Google's expansion in many web based solutions we searched for a facial recognition software.

We were able to try the online demos to see whether these software were viable, we used the following Image displayed below in Figure 1:

2 Methodology

An annotation tool was constructed to create a base for comparative analysis, we refer to this as Manual Classifications. These manual Classifications involved describing the features of the Scenes.

The aim was to collect specific information on each different face within the scene. To determine which of the sometimes many faces in the scene it would be reasonable for software to detect a standard was created for reasonable detection.

The faces of players were recorded if it showed their face at a minimum of 20 by 20 pixels. The back of the head was not detected as a face by any software, these faces were classified manually but reclassified as other. Crowd shots provided difficultly in determining which faces were reasonable to classify. These faces were not the intended targets of the recognition however these faces contributed to our understanding of the software. The same face size standard applied to crowd members, but focus was placed on the most prominent faces. For each of these faces, we collected information on the following attributes:

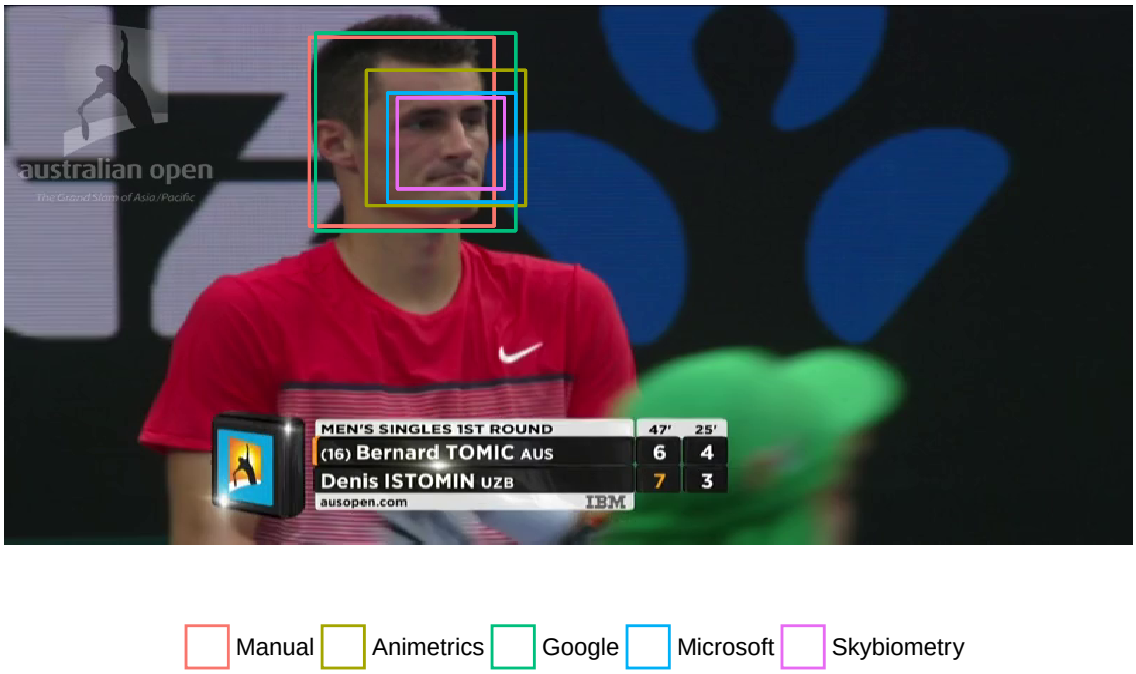

Figure 1: This image of Bernard Tomic was chosen as a trial image to be presented to each of the software before they were included in the research. It was expected that the software would be able to find this face, despite the player facing away from the camera.

Table 1: This table lists the possible image descriptions that are associated with the attributes of each image. The most appropriate descrition from each list was selected. There are more options for more complex attributes, there were five expected situations the image may depict.

Attribute	Choices
Graphic	Live Image, 2D Graphic
Background	Crowd, Court, Logo Wall, Not Applicable
Person	Yes, No
Shot Angle	Level With Players, Birds Eye, Upward Angle
Siuation	Court in Play, Court Player Close-Up, Court Close-Up Not Player, Crowd, Off Court Close Up of

Table 2: This table lists the possible face descriptions that were appropriate for each individual face. The most appropriate descrition for each face was built by selecting one option from each list describing a particular attribute of the face that was selected. Most of these categories selections were made obvious in the image, Obscured and Head Angle were more difficult to choose for some faces.

Attribute	Choices
Detectable Person	Player, Other Staff Member (on court), Fan, Not Applicable
Obscured Face	Yes, No
Lighting	Direct Sunglight, Shaded, Partially Shaded
Head Angle	Front On, Back of Head, Profile, Other
Glasses	Yes, No
Visor or Hat	Yes, No

2.1 Manual Annotation

To record the details of attributes for each face and scene a Shiny (2016) App was created. We called this Application our ManualClassificationProgram[1]. This helped to provide information for all attributes quickly and consistently.

If there was a face in the image the annotator was able to highlight a section of the image to create a square 'Face Box'. This changed the display and presented a set of Attributes with radio buttons, this allowed information to be recorded for the face in the specific 'Face Box'. This recorded the x and y coordinates of the corner points of a box drawn by the mouse, and when the save button was hit it saved all the radio button selections and the 'Face Box' coordinates to a CSV file[2].

When a face was not selected, the radio buttons showed the Scene attributes and the radio buttons with the possible selections the annotated was able to choose from. When in this display, selecting the save button would then save the Scene selections to a specific CSV file[3].

If there were issues, the CSV files were able to be edited, this was reserved for extreme circumstances. As a lot of care was taken to ensure the first selections were correctly submitted and applied to the correct Faces and Scenes.

All the annotations for this sample were completed by one author. This was chosen to provide consistency across the sample of faces annotated manually. However the initial choices of what would be reasonably detected were made by several of the authors.

2.2 Software Interaction

The software choices allowed for POST requests to be sent via the internet. To access the APIs through R we enlisted the httr package, using functions from this package a script was written for

[1]https://github.com/mvparrot/face-recognition/blob/master/ManualClassificationProgram.R
[2]https://github.com/mvparrot/face-recognition/blob/master/ManualClassifiedFaces.csv
[3]https://github.com/mvparrot/face-recognition/blob/master/ManualClassifiedScenes.csv

Google, Animetrics[4], Microsoft[5] and Skybiometry[6]. These scripts contained loops that would move through the images, individually posting a request for each image to be analysed. These scripts included retrieving the information provided and converting it into a usable format for our analysis. One interesting anomaly was found when using the Skybiometry software as it limited the amount of requests per minute. We accounted for this by stalling the posts for the amount of waiting time the software notified, and checking until the time lapsed and the script could continue looping.

2.3 Data Processing

The data needed for our analysis was spread across six files. For each software we had the information on the location of the Facial Bounding Boxes, as well as the time taken for the software to find the information. Some of the software also provided a more detailed level of information.

The collation of the results from the Manual Recognition Program created two CSVs, ManualClassifiedFaces[7] and ManualClassifiedScenes[8].

A single data set was created to combine all necessary information in the previously mentioned files for our analysis. The information in the data set[9], was carefully considered. It considers the identify of each face, and all relative face attributes, as well as the image file the face was found in, from this information each face was able to be uniquely identified. Also included was information on the software that found it, and the time it took the software to identify the face. It also has a record of how many faces had been identified in the image by counting each additional recognized face. To do so, we gathered the name of the file the face was found in and the software Type the potential Face Boxes was determined by. The automatically determined time values were also included. The minimum and maximum x and y values were drawn from different values in each software's CSV files. This required some processing to align the differing values to be comparable.

To find whether the software were recognizing the same faces a function was created. As the location and size of the boxes around the faces were recorded, these values were used to see if a particular identified face box matched a manually identified face, or a region found by another software. This function uses the information of each face and compares the intersecting regions of the polygons created by the x,y coordinates of Manual Faces and other software's faces, to determine if the same face was recognized. We determined the ratio of intersecting area to total area must be greater than 0.1 to be considered the same face. This allowed us to compare the identification areas, as well as contrast the identified faces of each software. This contributed another variable, boxID, to the data set[10].

2.4 Analysis

Using the data set[11] of the combined API and manual results, we were able to compare the performance of the software. Firstly, we considered how many individual faces the software were

[4]https://github.com/mvparrot/face-recognition/blob/master/SoftwareRequestScripts/animetrics.R
[5]https://github.com/mvparrot/face-recognition/blob/master/SoftwareRequestScripts/microsoftAPI.R
[6]https://github.com/mvparrot/face-recognition/blob/master/SoftwareRequestScripts/autoSkybiometry.R
[7]https://github.com/mvparrot/face-recognition/blob/master/ManualClassifiedFaces.csv
[8]https://github.com/mvparrot/face-recognition/blob/master/ManualClassifiedScenes.csv
[9]https://github.com/mvparrot/face-recognition/blob/master/ALLmetIMG.csv
[10]https://github.com/mvparrot/face-recognition/blob/master/ALLmetIMG.csv
[11]ALLmetaIMGnamed

able to detect in Figure 3.

However, individual faces are not beneficial if they do not correspond to the faces manually annotated. Figure 4 was created by defining groups depending on the API that recognized each particular face. The UpSetR (2017) package helps visualise set intersections. Where in this circumstance potential Faces Boxes may overlap on the same face, each bar shows the number of individual faces that have potential Face Boxes resulting from each of the APIs highlighted below the bar.

2.4.1 Modelling Face Detection Probabiity

This shows how influential certain attributes are in determining whether a face will be detected, a hit, or not, a miss.

A step wise method was used to determine the best regression model for predicting a hit or miss. The regression model that provided the best AIC, included the scene attributes: Shot Angle, Background, the interaction between these two; whether the scene was a graphic, and the situation on the court when the image was taken. It also included the lighting on the particular face, and whether the specific face was accessorised with Glasses and a Visor or Hat.

Regression analysis showed that the situation variable had significant differences between intercept level of "Court Close-Up Not Player" and the categories of "Court in Play" and "Court Player Close Up". This would mean that the probability of all the APIs finding the face that had been found manually was significantly less if the situation depicted was "Court in Play" and "Court Player Close Up" rather than a "Court Close-Up Not Player".

The bar charts of the situations show that the situation is important in influencing the detection of a face. When the face is accessorised with them, all APIs have a significantly increased chance of finding the face in comparison to the base rate category of the "Court in Play". While the image being a Graphic had a large impact it was not significant, this is likely because of the small number of images with this attribute. It also shows that the shot angle is only significant for one or two APIs at each level.

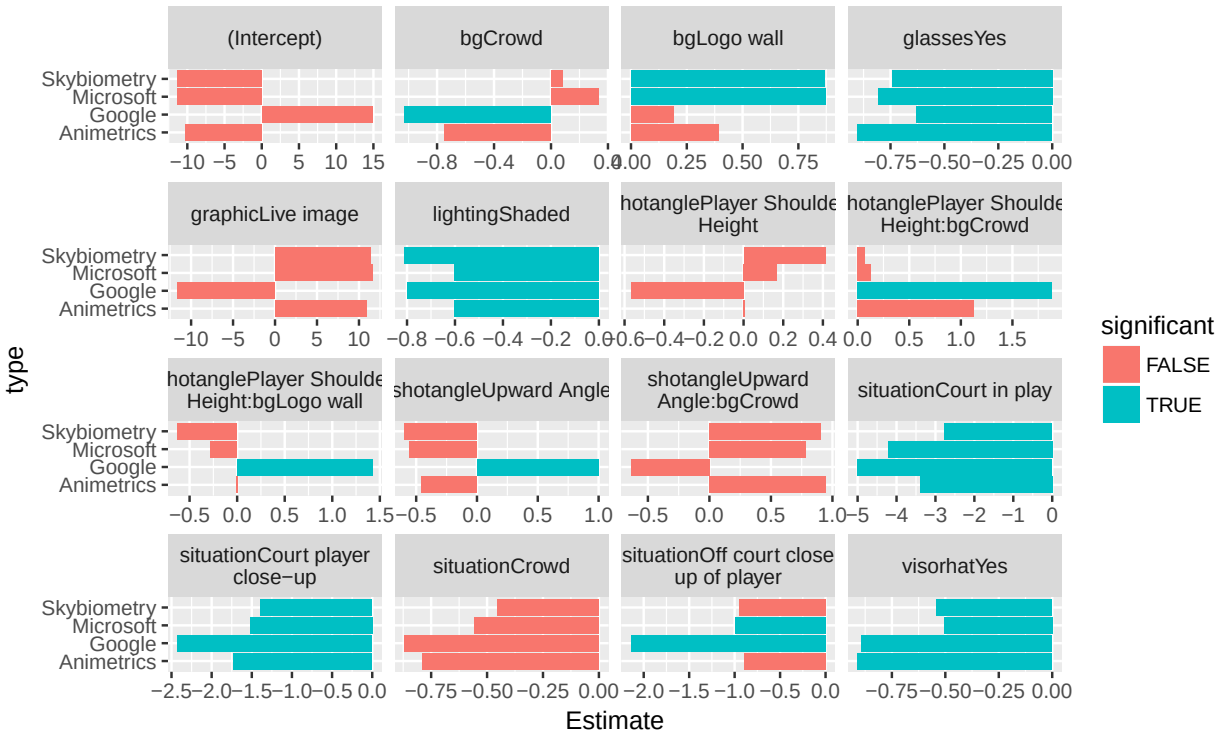

Figure 2: The figure depicts the change in the probability of a face being detected by the APIs given a certain attribute level associated with the face. Wearing either glasses, or visor or a hat significantly decreased the likelihood of a manually annotated face being detected by the APIs.

3 Results

Each API returned areas that indicated the potential location of a face. These areas were defined by a Bounding Box, created using the four points returned by the APIs to make a box around each face. The Google Vision API detected a large amount of potential faces, much more than Microsoft, Skybiometry or Animetrics. This can be seen below in Figure 3.

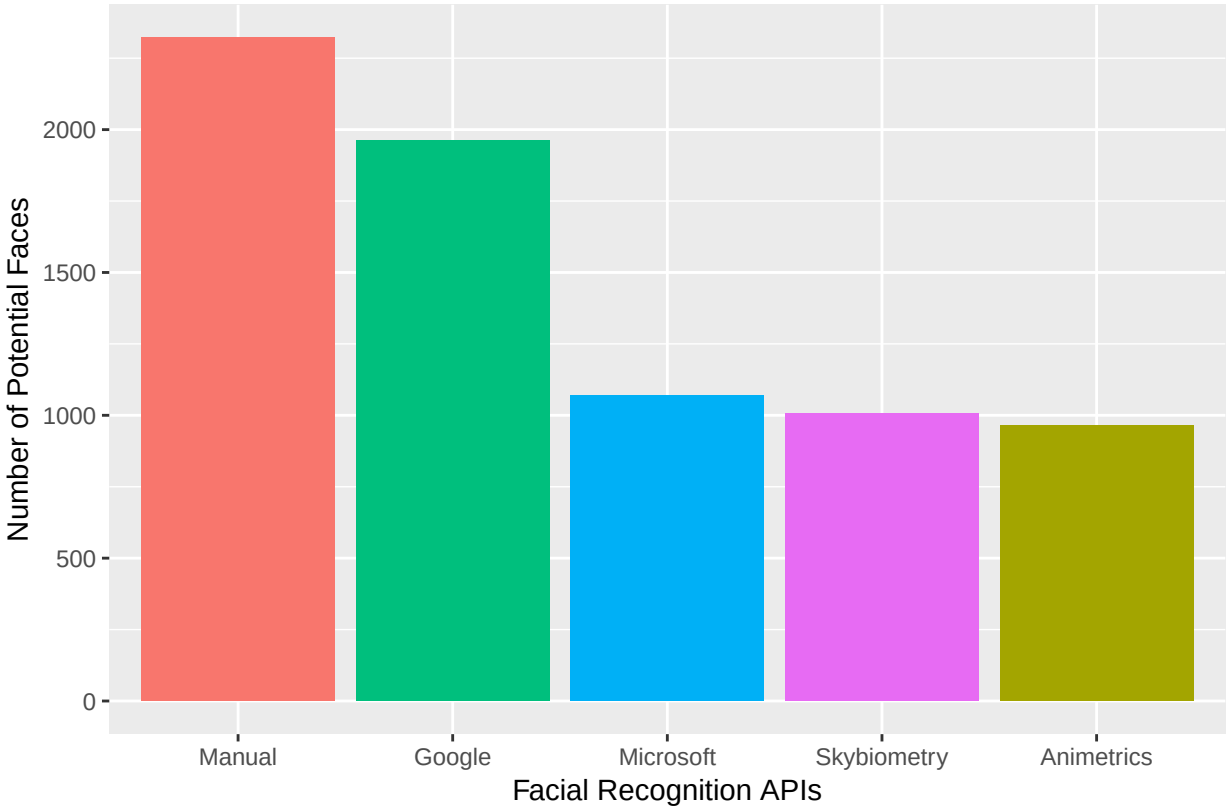

Figure 3: Potential Face Boxes Per API: The bar chart shows the number of Potential Face Boxes produced by each API, comparing the height of the bars indicates that Google's Facial Recognition API recognized almost 1000 more potential faces than the next best API, Microsoft.

To evaluate the performance in terms of the overall accuracy of each algorithm we considered the amount of faces they classified that matched faces that were selected manually. Using a sample that contains all the manually annotated faces and all the faces recognized by the four APIs.

To consider how many Type I errors occurred, where a face was detected incorrectly, we look at the potential faces that do not match manually annotated faces.

Matches Manual	API Type Animetrics	Google	Microsoft	Skybiometry	Total
FALSE					
N	528	638	505	512	2183
Row(%)	24.1869%	29.2258%	23.1333%	23.4540%	43.6600%
Column(%)	54.7718%	32.5344%	47.1963%	50.9453%	
Total(%)	10.56%	12.76%	10.10%	10.24%	
TRUE					
N	436	1323	565	493	2817
Row(%)	15.4775%	46.9649%	20.0568%	17.5009%	56.3400%
Column(%)	45.2282%	67.4656%	52.8037%	49.0547%	
Total(%)	8.72%	26.46%	11.30%	9.86%	
Total	964	1961	1070	1005	5000
	19.28%	39.22%	21.4%	20.1%	

All the potential facees that Google found which do not match manually annotated faces were correctly identifying faces.

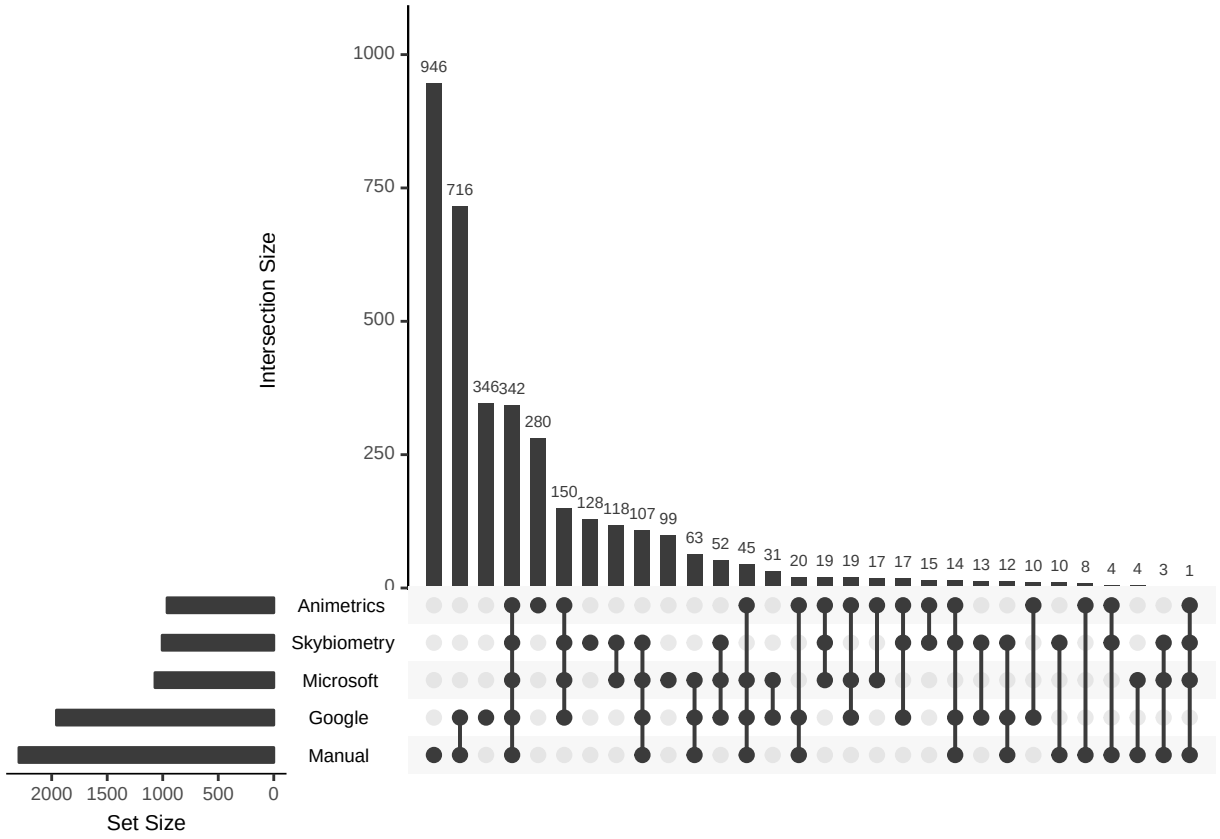

Figure 4: potential facees Per API Combination. The Bar Chart shows the potential facees that were recognised by multiple API or found manually. The largest group, with 809 faces, is potential facees only found by Manual annotations. The following group were the 716 faces recognized both Manually and by the Google API.

This shows that there are many faces that were identified by some APIs and not others. These combinations may give some indication as to the circumstances when some APIs perform better than others.

The image below is an example of how some faces were only found by certain APIs, and how some APIs produced unusual false discoveries.

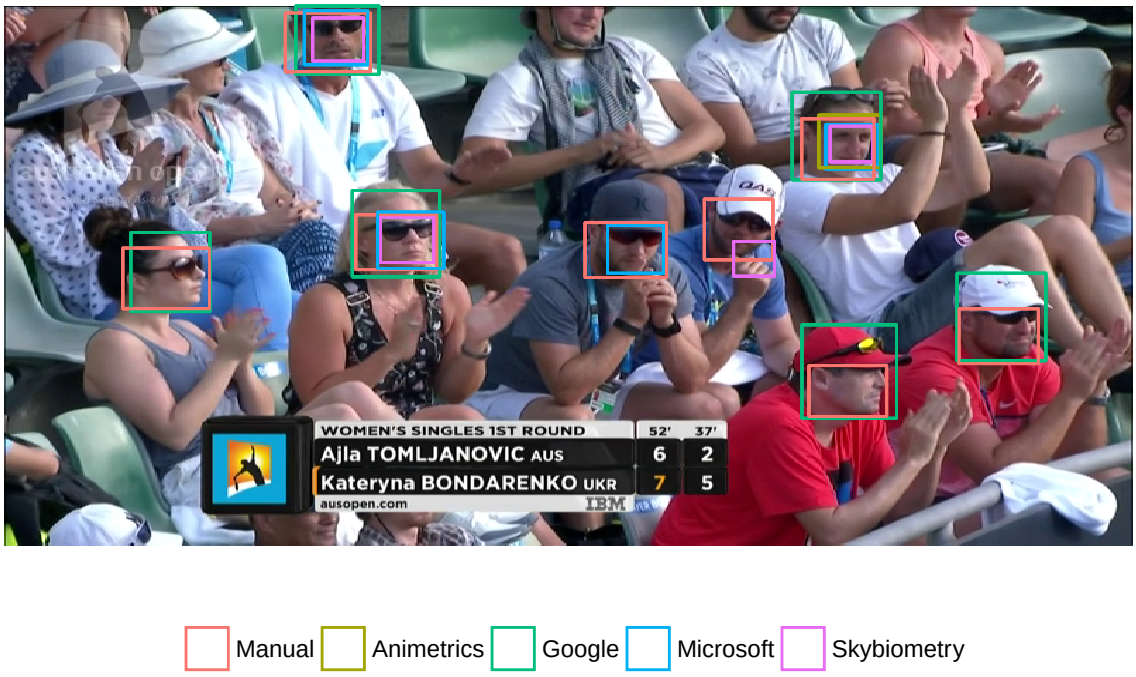

Figure 5: There is an unusual classification where just right of the center, the smaller box actually captures a fist, not a face.

We then considered the characteristics of the images that the API found Potential Faces in.

Table 4: This table outlines the combinations of image attibutes that are most common in the image set. The combination of the Logo Wall background and the Player Shoulder Height angle are shared by three of the five image combinations.

situation	bg	shotangle	detect	count
Court player close-up	Logo wall	Player Shoulder Height	Player	734
Court in play	Logo wall	Player Shoulder Height	Player	241
Crowd	Crowd	Upward Angle	Fan	149
Court player close-up	Court	Birds Eye	Player	133
Court player close-up	Logo wall	Player Shoulder Height	Other staff on court	117

We then considered that there would be an uneven amount of faces with certain image attributes. This is considered in the mosaics below in Figure 7.

Front on	Other	Profile
111	549	290

3.1 Results Overview

Figure 3, the bar chart of the potential faces shows Google produces the most detections. This would be a strong incentive to use Google for facial recognition in sports application. It was considered that Google's API may have been finding more unwanted faces than the other APIs. Table **??** details the amounts of faces that were found by each API that matched or did not match a face found manually. There are slightly more faces found by APIs that do match the faces found manually, 56.34%, than faces that do not match, 43.66%. This was unexpected as it was presumed that majority of the faces annotated manually would be found.

The largest group of images were those annotated by manual annotations and found by the Google API, these faces consisted of 26.46% of the total sample. This set of images are also seen in Figure 4, which shows the amount of faces that were found by each combination of APIs. The group of faces that were found by manually and by the Google API are represented across the various bars. Their ownership of the faces is denoted by the black bubbles below the bars.

Visual inspection showed that the 638 potential faces noted in Table **??** that were found by Google but not found manually were actually faces, however they were not all of players, and some of these crowd memebers were beyond what would be expected of an emotion recognition application for tennis players. These were deemed unlikely to be detected and neglected during manual annotation.

Table **??** shows Animetrics had the least amount of potential faces that matched Manually annotated faces. Also, visual inspection of the 528 potential faces that did not match manually annotated faces showed the Animetrics results contained many potential faces that were not faces.

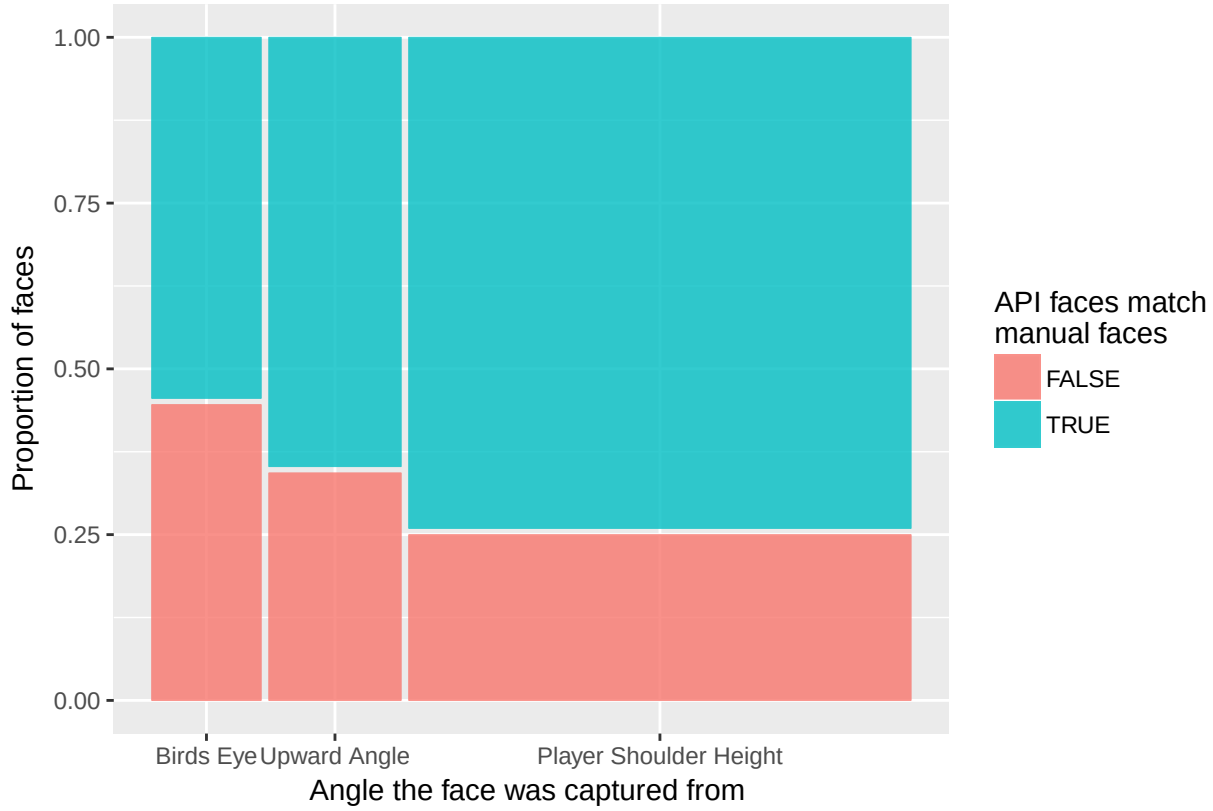

Figure 6: The colours in the mosaic show the amount of faces that matched those found manually given the angle the faces were captured from. There is a greater amount of API faces that matched the faces found manually than those that did not match. However given the face was captured from a birds eye angle there were less faces that matched those found manually. The most common faces in our set were those captured at Player Shoulder Height and found manually and by an APIs.

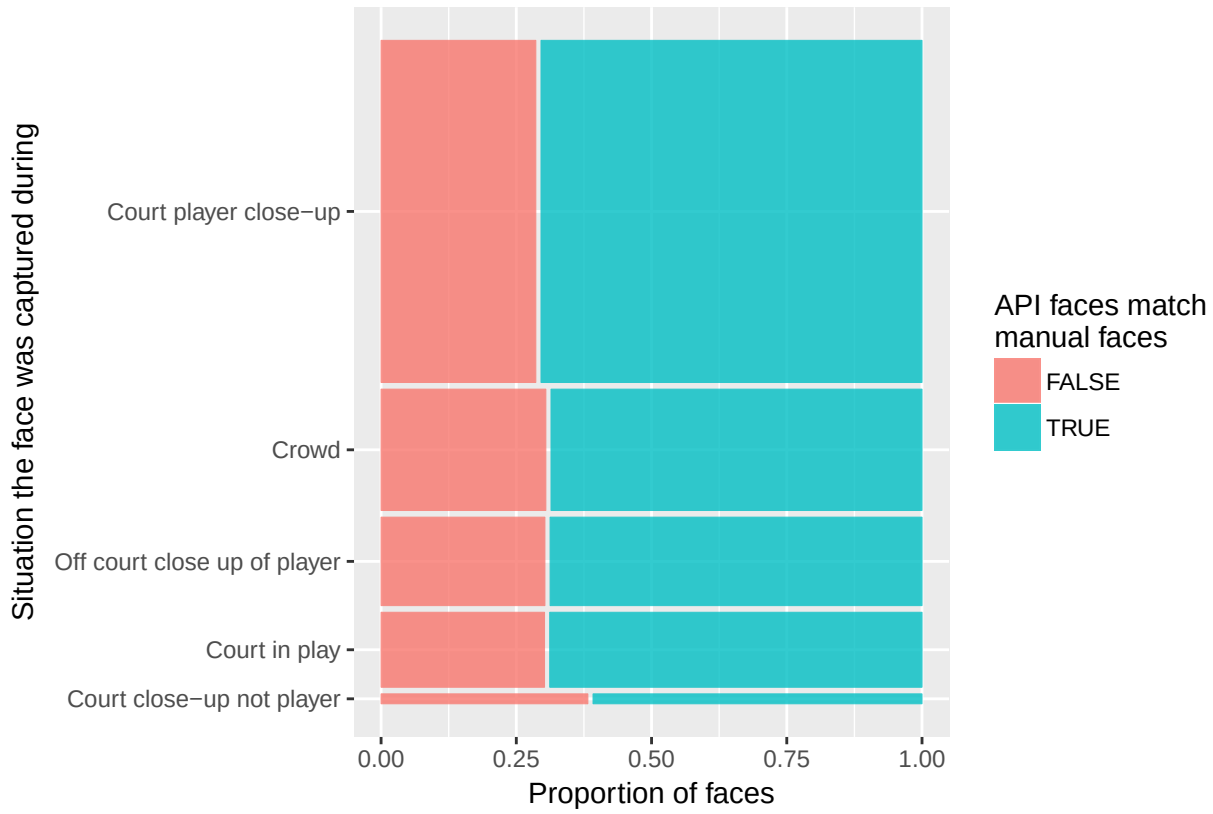

Figure 7: This mosaic shows the number of Faces captured during each possible situation. It contrasts how many of the images captured in situations either did or did not match the faces annotated manually using the colour pink for potential faces that did not match, and blue for those that did. It can be seen that the largest portion of the faces were captured in a close up situation while the players were on court.

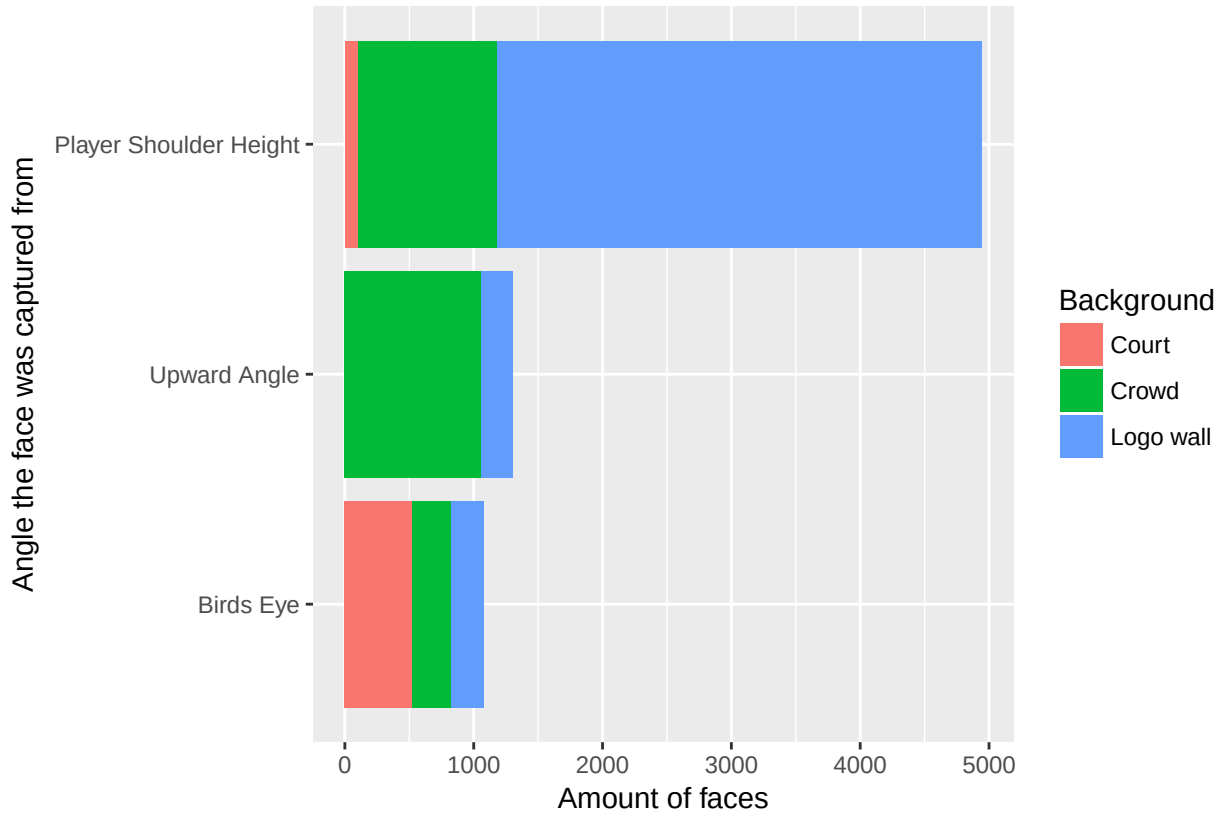

Figure 8: This stacked bar chart allows consideration of the interaction between two image attributes. We are able to see that the background being the Logo Wall and the angle of Player Shoulder Height is common to the highest amount of faces. There are also no images that were taken at an upward angle with the background of the court.

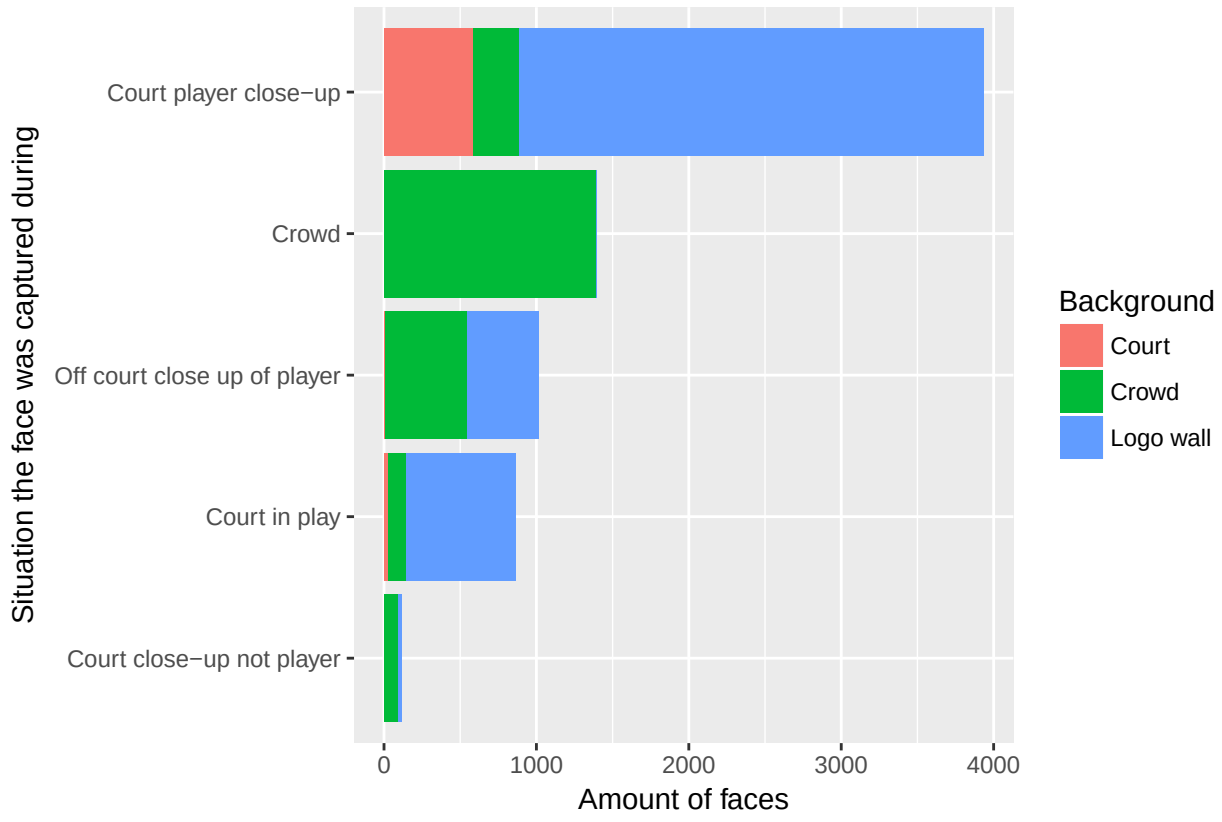

Figure 9: This stacked bar chart allows consideration of the interaction between the background of the image captured and the situation that could possibly be occurring. As seen previously, the logo wall is the most common background, but it is never the background to an image of the crowd. The court is the background of an image only when players are captured, this occurs during close ups and while the court is in play.

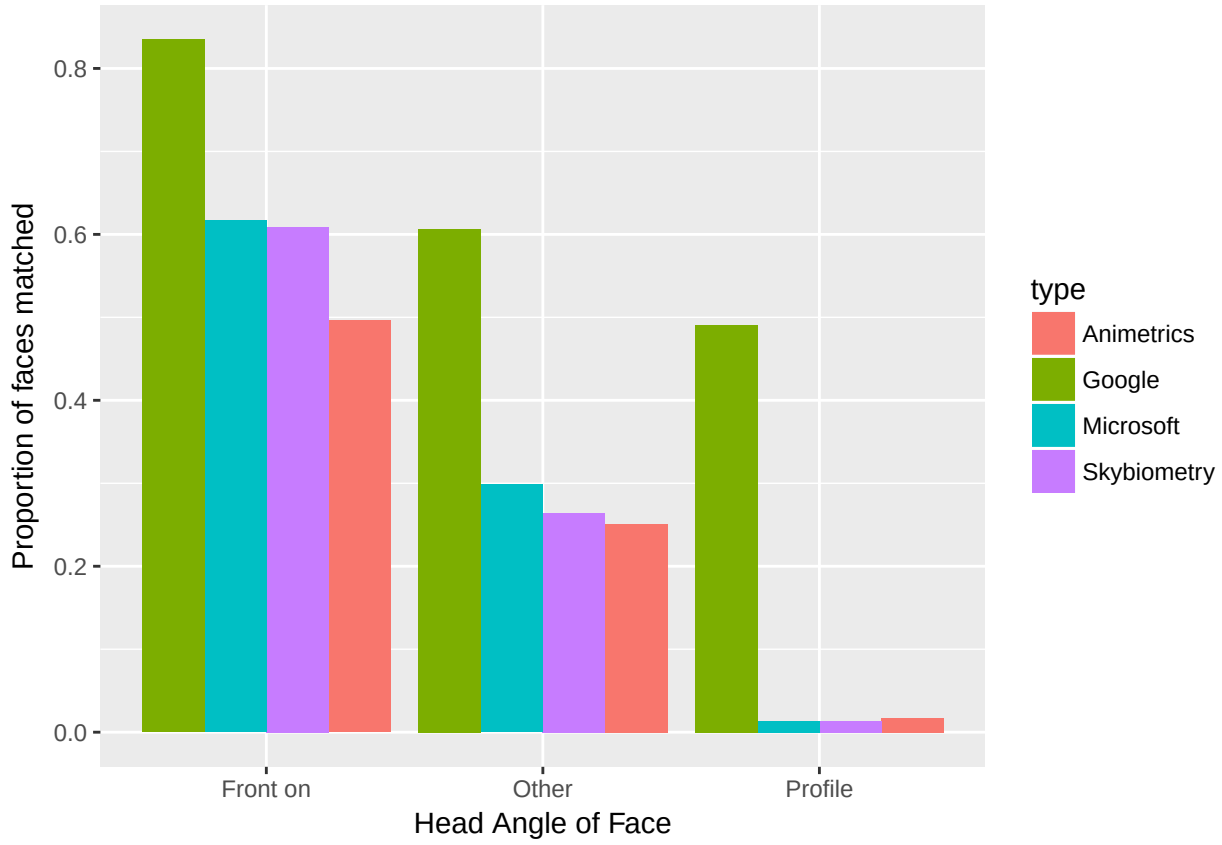

Figure 10: Google performs much better in comparison to the other API when the head-angle is Profile. This is outperforming unusually well, however this could also be due to the poor performance of the API in this circumstance.

The images were all considered manually. The scene information was recorded and the combinations were shown to find how many potential faces were found with the combination of scene attributes. This showed that crowd members faces were often recognized, this is both helpful and unhelpful as it shows a strong ability of Google's algorithm to recognize faces, even when these faces are not the goal of the research.

Image 1 shows the images preferable for future research. Where the faces will be recognized and allow both the identity and emotion of the players to be recognized.

Google gave the optimum results in this image as it found the face of the player, but did not locate the face of the staff member behind him on the court.

While Google's recognition's mostly matched the Manually annotated set of faces, there were some that did not. These were all actually faces and were missed during manual annotations.

Table 6 shows that 190 of the faces that were not found manually occurred in the scene of a Court player close-up, with a background of a logo wall, where the shot was taken at player shoulder height.

This shows it was performing extremely well and not resulting in unexplained potential faces as some of the other software were. This is a strong indicator that applying Google's software for further research would result in the recognition of desired faces.

These results have contributed to the choice of Google given the optimum scene as described above. Implementing a filtering process, either using current or alternative footage[12] would allow Google to provide Tennis Australia with the most applicable results.

We moved to considering the characteristics of the faces. This helped to distinguish where Google performed well in comparison to the other software options.

Table 7 showed the combinations of attributes that were found for each face[Given that information was not recorded where Google provided facial recognition for faces not Manually annotated these could not be considered.]. The use of accessories, Glasses and Visors or Hats, was considered as the Australian Open takes place on both indoor and outdoor courts. To apply this research all courts that elite Tennis players compete on had to be included. It was assumed that outdoor courts would led to the use of these accessories and these accessories may contribute to the performance of a recognition software. It may be implied by the table that Glasses prohibits recognition as all but one of the combinations have 'No' for the Glasses variable. However, we are cautious of validating this as Table 8 tells that there are many more faces recognized, by both the Google recognition's and manual annotations, that do not have Glasses. This disproportionate sample of faces with Glasses means that we considered it proportionally rather than as a total.

Graph 2 demonstrates that the presence of Glasses on faces annotated manually did affect recognition by Google's algorithm, while it outperformed the other software in both instances, faces were identified more often if the person did not wear glasses.

Moving to looking at the characteristics that were considered manually shows that the use of glasses by players coincided with less faces being annotated.

The box-plot in graph encourages our comparisons to not consider the size of potential face boxes as a measure of how the software performs on small faces.

[12]See future research for further information on these options

3.1.0.1 Challenges

It is understandable that there would be many more faces to recognize in these shots than in shots where there is only a player, and therefore many more faces recognized. This provides many faces to sort through to find emotions of a player.

We faced the challenge of accessing usable images of players, and specifically their faces. - Availability of software - Using the software - Time constraints

Method, automated the process to reduce data cleaning and help group characteristics

Pricing

Employ the Google Vision API, which would allow the use of still images, or video (TEST VIDEO) files, reducing the need for stills. This product - cost in relation Ease of access - API calling

4 Future Work

The Long Term goal of this research is to better understand how the emotion's felt by a player during a match affect player performance. Ultimately we would aim to create a program that automated the collection of player emotion data from throughout a match. This information would be presented in a timeline that allowed match performance, in the form of points won, to be aligned with the emotions felt at certain times throughout.

Considering the images used during our study were stills derived from Broadcast video files, it would be useful to extend further research to deal with the video files directly. The Google Vision API used in this research which produced the best recognition in images does not yet have the potential to detect faces and emotions in a video.

It should also be considered that these are software focused on providing recognition in certain controlled scenarios. If the study was controlled to focus on certain camera angles that align with the facial angles these security programs are intended to recognize faces in.

Given that Google found many faces that did not match manually annotated face, we considered that we should check for manual errors. There is the possibility that we could create another app that shows the Facial Bounding Boxes identified by each program, this would allow the annotator to confirm manually whether or not these are faces.

Given that certain Scene attribute combinations produced more facial recognition than other combinations we should consider limiting the sample of images sent to Google Vision API. This would not only reduce cost but also provide a greater level of detail of the emotions felt by a player during a match. To provide a greater level of information at all points in a match it would be beneficial to derive images from a single camera feed. This feed should match the Scene attributes that provided the most Google faces.

To undertake sentiment analysis, we would take the boxes of faces found in this set of images. Allowing each face a border of pixels, we would crop the images and produce an individual face image that would form the data set for emotion recognition. We also feel that incorporating audio information from the microphones worn by players may assist in sentiment analysis. By including this information we would be able to define differences between certain emotions that may not be able to be found by facial features only.

References

Barr, Bowyer, J. 2014. "The Effectiveness of Face Detection Algorithms in Unconstrained Crowd Scenes." *IEEE Winter Conference on Applications of Computer Vision (WACV)*. doi:10.1109/WACV.2014.6835992.

Chang, Winston, Joe Cheng, JJ Allaire, Yihui Xie, and Jonathan McPherson. 2016. *Shiny: Web Application Framework for R*. https://CRAN.R-project.org/package=shiny.

Gehlenborg, Nils. 2017. *UpSetR: A More Scalable Alternative to Venn and Euler Diagrams for Visualizing Intersecting Sets*. https://CRAN.R-project.org/package=UpSetR.