HIDING IN PLAIN SIGHT? THE IMPACT OF FACE RECOGNITION SERVICES ON PRIVACY

by

JAMES RICHARD ORTEGA

Presented to the Faculty of the Graduate School of
The University of Texas at Arlington in Partial Fulfillment
of the Requirements
for the Degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

UNIVERSITY OF TEXAS AT ARLINGTON

December 2019

ABSTRACT

HIDING IN PLAIN SIGHT? THE IMPACT OF FACE RECOGNITION SERVICES ON PRIVACY

James Richard Ortega, MS

University of Texas at Arlington, 2019

Supervisor: Shirin Nilizadeh

Committee Members: David Levine and Won Hwa Kim

The public at large is increasingly concerned with privacy online. While the focus is on the data privately collected by platforms, there are also privacy concerns in the realm of public data. Seemingly innocuous information shared in public, on online platforms, can be pieced together to detrimentally affect one's privacy in unexpected ways. On YouTube there exists a rich public dataset for adversaries to analyze for the purposes of breaching privacy; particularly due to the intersection of location and facial data. The goal of this work is to characterize the privacy risks that exists on YouTube, and explore the viability of large-scale analysis of video data through easily accessible means for the purposes of identifying users by face in different videos. This work's threat model presumes an adversarial actor is interested in identifying faces across several videos. In phase one focus was on the efficient collection of visual data. In phase two the compiled dataset was characterized, and it's associated facial images were analyzed with Microsoft Azure Face API. In phase three the data obtained in the prior steps was analyzed with a focus on the implications to personal privacy. In conclusion, this work finds that there may be privacy concerns for bystanders who are unaware they are being recorded in public, and for video publishers who are relying upon security through obscurity. Specifically, it was possible to identify the same persons across several videos and YouTube channels in the San Francisco area within a two-week time span.

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Professor Shirin Nilizadeh, for the opportunity, and guidance in doing this work. Additionally, I would like to thank my committee members, Professors David Levine, and Won Hwa Kim, for their time, and feedback. Lastly but importantly, I would like to thank my friends and family for their warmth and support throughout my time accomplishing this work.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

API	**A**pplication **P**rogramming **I**nterface
CPU	**C**entral **P**rocessing **U**nit
GPU	**G**raphical **P**rocessing **U**nit
HD	**H**igh **D**efinition
ID	**I**dentity **D**ocument
IP	**I**nternet **P**rotocol
LFW	**L**abelled **F**aces in the **W**ild
LPG	**L**arge **P**erson **G**roup
PGP	**P**erson **G**roup **P**erson
PPTT	**P**rice **P**er **T**housand **T**ransactions
SD	**S**tandard **D**efinition

CONSTANT VALUES & FORMULAE

Pertaining to the Examined Location:

San Francisco Coordinates (SFC): $\text{SFC}= (37.757023, -122.434513)$ (1)

San Francisco Radius (SFR): $\text{SFR}= 8km$ (2)

Pertaining to Frames &

Extraction of Face Images:

f (Frame): $f = a\ video\ frame$

f' (Frame with a face): $f' = \{f \mid f\ contains\ a\ face\ according\ to\ local\ detection\}$

f'' (Face image after AFA detection): $f'' = f' \mid f'\ contains\ a\ face\ according\ to\ AFA\}$

V_i (Frames belonging to video "i"): $V_i = \{f'' \mid f''\ belongs\ to\ video\ i\}$

f''_s (Face image saved to a PGP): $f''_s = \{f'' \mid f''\ is\ successfully\ added\ to\ PGP\}$

Pertaining to the AFA Cost of Use:

n_{gm} (Number of calls to

AFA "group" method): $n_{gm} = \sum_{i=0}^{max(i)} \lceil |V_i|/1000 \rceil$

n_p (Number of PGP's): $n_p = |\{group \mid group\ is\ returned\ by\ group\ call\}|$

$n_{p'}$ (Number of PGP's

that contain a saved face): $n_{p'} = |\{group \mid group\ contains\ at\ least\ one\ f''_s\}|$

$Cost$ (Cost of Analysis w/ AFA, in Dollars): $$Cost = \frac{|f'| + n_{gm} + n_p + f''_s + f''_s/1000 + n_{p'}/10}{1000} * PPTT \quad (3)$$

Pertaining to Graph Representations:

f'''^i_{rf} (Representative Face): $f''_{rf} = largest(\{f''_s \mid f''_s\ within\ PGP\ i\})$

Representative Face Directed Graph (RFDG): $\text{RFDG}= \{Vertices = \{All\ f'''^i_{rf}\ and\ PGPs\},$

$Edges = \{f'''^i_{rf}, PGPs, Relation\ Confidence\ Score\}\}$ (4)

PGP Relationship Undirected Graph (PRUG) $\text{PRUG}= \{Vertices = \{All\ PGPs\},$

$Edges = \{PGPs, PGPs, Relation\ Confidence\ Score\}\}$ (5)

Dedicated to my loving grandmother...

CHAPTER 1

Introduction

1.1 Motivation

In a world where the amount of information being stored online is ever increasing, the public is concerned about their private information. With concerns at an all time high, the focus now is on data that is privately collected by platforms; however there are also privacy issues with public data. Despite the public nature of a subset of online data, many assume that, due to the bulk of the data online, it is reasonable to expect their publicly exposed information is virtually private with the exception of the few people within in their social circles. One platform where this appears to be the case is YouTube. With the YouTube platform, there are many public videos where people voluntarily expose their face, location, and other information they may not want certain people to be aware of, or track. Additionally, to make matters worse, there have been reports of negative social behaviors on the YouTube platform when users become aware of another user's location; one such behavior has earned its own term and is known as "swatting" [15]. This project's goal is to characterize the privacy risks that exist on YouTube while exploring the viability of large-scale analysis of video data,through local and cloud computing means. This is for the specific purposes of identifying users and bystanders, by face, within in different videos. Ultimately to the ends of, assessing the impact of face recognition services on privacy via the use of a threat model 2.1.2, where an adversary is attempting to violate the privacy of a defenseless YouTube user-base.

1.2 Background

1.2.1 On YouTube

YouTube currently has over 2 billion users, and hundreds of hours of video data are being uploaded to the YouTube servers each day [24]. The structure of YouTube revolves around what are known as YouTube "channels", which can also be thought of as accounts (YouTube accounts are associated with Google) with video publishing ability. Anyone can start a YouTube channel, and typically users who upload videos, especially casual users, will only ever have one YouTube channel. YouTube users can visit and view most YouTube videos without an account (excepting those flagged for mature content), but do require an account to like/dislike or comment on videos. Additionally, both YouTube channels themselves, and the videos they host are often associated with metadata such as descriptions and links to other social media. Furthermore, videos are often tagged with geolocation information, possibly automatically, depending on a channel owner's google account settings. As shown in figure 4.1, these locations are visible as "phrases" to YouTube users, however, the videos are actually tagged with specific geolocation coordinates which makes searching by specific region possible using the YouTube API [60]. While not something that is accessed by the typical YouTube user, the YouTube API is freely available. Users of YouTube may not appreciate vulnerability presented by sharing location information when uploading videos to YouTube due to the misleading search interface on the YouTube webpage. As presented in figure 4.1, the YouTube search interface does not support searching by location coordinates/radius, but only by phrase. This may mislead the user into believing YouTube is not storing or making accessible to anyone, any specific location information. Also, related to the specificity of the location information stored on the YouTube servers, the YouTube phone app allows users

to record videos directly in the app, possibly then drawing upon the location information reported by the phone to be stored on YouTube servers.

1.2.2 On Video Structure

In order to conduct face recognition, fundamentally, images must be used. Videos can be thought of as image frames displayed in a sequence over time. These images are known as frames, and most videos, including those handled in this work, use a structure revolving around "key-frames" (also known as i-frames) [9], which encode an image independently without reliance on other frames.Secondary frame types are used that are reliant on key-frames (for file size compression) to display an image at the appropriate time during video playback [9]. This is outlined here because it is the key-frames which will be of focus in this work.

1.2.3 On Cloud Services & Microsoft Azure

Microsoft Azure is a cloud computing platform with many different capabilities. As a part of their "cognitive services" suite, they provide what is known as the "Azure Face API" (AFA) [34]. The AFA provides a variety of cloud computing services pertaining to facial recognition. Namely they provide functionality which includes the detection of faces, grouping faces, identifying faces, comparing faces, storing faces in "bins" referred to at various levels of the AFA as "persons" or in what are called "face lists". Additionally, while it is a black-box and the methodology is unknown, the AFA does provide a functionality that allows a user to train face recognition such that the system attempts to be able to discern the identities of the aforementioned "persons". While the AFA provides much functionality, on several levels

these functions are limited in a manner that will necessitate some of the tactics employed in the course of this work, as mentioned in section 2.2.3.

1.2.4 On Face Recognition

While in this work, the bulk of facial recognition utilized a cloud service provider, and is thusly a black-box system, there are fundamental factors which broadly influences all face recognition technique performance. Specifically, factors such as target face image quality (typically defined as pixel between eyes) and target image focus [5]. Lastly, related to facial recognition is the question of how to prevent it, and while this is not explored in this work it is discussed in section 6 which pertains to future work.

1.3 Related Works

1.3.1 On Social Media Privacy as it Relates to Facial Data

Privacy concerns within the Information Security realm, as it relates to big data social media platforms, continue to be a topic of much discussion [33]. Highly relevant to this work, Smith et al. explore the problem of "geo-tagged" information in public social media, and it is found that such geo-tagged information is common [51]. Additionally, in one paper, the author Loebel ponders "Is privacy dead?" as a result of the common link between geo-tagged information facial recognition systems [30]. Many social media platforms will capture geolocation data and associate it with image or video content, perhaps without a user's realization. The privacy issues on social media platforms with respect to facial recognition remain active, despite legal proposals to counteract them [32]. The author Reidenberg, explores the problem surrounding the "reasonable expectation of privacy" legal standard is explored, and is argued to be insufficient to protect privacy [47]. This applies widely to social

media content, particularly content which capture private information in the context of a public space. Acquisti, Gross, and Stutzman, used face images (not videos) within various social media accounts to associate these accounts across different social media platforms [3]; which could result in widespread privacy breaking data collection on individuals. This topic is very active in the legal realm, and as a specific example, Facebook's known use of facial recognition on it's platform is of concern [49]. Core to privacy concerns such as these, are issues such as the "right to anonymity" or the "right to not be tracked", as discussed in [42]; whether these rights exist, and under what circumstances, are open legal questions. Closely related to these issues of anonymity and tracking, are concerns about the privacy of bystanders in visual data, which has been highlighted as being of particular concern [45]. While there has been work with respect to privacy and social media, specifically involving location information or facial images, there is less work with respect to videos on these platforms. One such work , finds mixed opinions when questioning subjects on their comfort level with respect to videos of them (exercising in public spaces) being shared on social media [25]. Aside from geolocation information, as mentioned earlier, there is also an intersection of other metadata, especially when looking across several social media platforms; which has been documented and purported to possibly rise to the level of a national security concern [1]. On a positive note, if one assumes a fair and accurate investigation which respects civil liberties, social media data, along with facial recognition, has been the topic of discussion where it pertains to improving the work of law enforcement [48].

1.3.2 On Large-Scale Video Analysis

These works are relevant due to the focus on the video medium, and they present a salient counter-point to the notion that there may be privacy in a large-scale video dataset due to

the sheer scale of the information, making analysis untenable. Huang et al. present a method for large-scale facial recognition on a set of videos [20]. Furthermore, a textbook has been written that addresses many video analysis problems [6], most related to this work however, is it's discussion of a distributed network of embedded cameras to assist in facial recognition. Closely related to this also, is the work presented in [10], which aims to address the problem of scaling face targets in a constantly growing dataset, however, it pertains to image sets. Additionally, several papers have drawn upon the power of distribute and clustered computing in order to significantly reduce the time required to process video data [52, 59, 53, 58, 63]. A useful concept that is important to the topic of large-scale video analysis, is the ability to fingerprint frames within a video, and describe them as a set of "features" (as commonly called in the machine learning field). A commonly known and referred to computer-vision algorithm to do this is known as the "Scale-invariant feature transform" [31]. Importantly, this algorithm has been adapted to function within GPUs for real-time analysis [16], which significantly improves its ability to be used in large-scale video analysis in a time efficient manner. Poms et al. present a method for large-scale video analysis is presented which involves breaking a video down into scenes [46]. This differs from the work done here which broadly examines whole videos via key-frames, and could be relevant to reducing the scale of the input data used in future work. Similar to the problems faced here, are those faced by researchers who attempted to create a social network of characters that appeared on television shows using the video data [64]; however in that paper, the authors also made use of auxiliary data other than faces to make these networks, which is a marked deviation from the focus of this work. While this work focuses on public data and solely facial data, there has been a publication describing an engine for processing of video content on a large-scale that makes use of additional metadata [17], even including audio, which of course would

provide even more avenues for privacy penetration. The work in this thesis makes use of the facial recognition service provided by Microsoft Azure, however, there have also been work which make use of the cloud computing services of Microsoft, Google and Amazon, in order to rapidly analyze large-scale video data from a distributed network of cameras [26], or of a large-scale video dataset in general [55]; to note however, these papers were not focused on privacy. Related to this, YouTube can be thought of as a distributed network of cameras, and YouTube has been used to create a video classification dataset [2], albeit a dataset not directly related to facial recognition. In conclusion here, these sources collectively weaken the argument that privacy may be protected due to the size of the data in question, and provide evidence that the work conducted in this thesis can further scaled.

1.3.3 On Public Video Data and Other Classification Problems

There have been publications which involve at their core, the problem of classification, and the use of facial data. However, neither of those presented here involve identification of people across a video dataset. In [54], the response to viral advertisements is explored by analyzing facial expressions in response videos. Additionally, [29], the authors focus on inferring demographic data of marginalized users by examining their faces. This is also related to this work due to it's use of Microsoft Azure Face API. The work presented in this thesis aims to add to this body of work by exploring the problem of identification within the realm of public face data in videos.

1.3.4 On YouTube with Respect to Facial Data

The works to be mentioned here involve using facial data from YouTube for purposes other than privacy analysis, which separates them from the work in this paper. However, for completion, to ensure the reader does not believe there has been no work done that uses YouTube for source data, they are mentioned here. Shen et al. present a facial landmark tracking dataset created using YouTube [50]. Biel, Teijeiro-Mosquera, and Gatica-Perez predicted the emotions expressed by different facial expressions using visual queues within YouTube videos [7]. This could have implications for privacy since it would allow for automated monitoring of an individuals emotions while filming content such as video blogs, which are common on YouTube. Wolf, Hassner, and Maoz present a facial identity dataset based upon celebrities mentioned in another well known face identity benchmark dataset [57, 19]. This differs from this work in that it does not focus on privacy implications of a public video dataset, or privacy implications of a local tool capable of differentiating between two people. It follows then that these authors did not address matters such as time/effort needed to create the dataset, or that the process to create the dataset should ideally allow for programmatically extending the dataset with minimal manual input.

1.4 Recap

Public data does not necessarily mean all owners of the data approve of its viewing, or use, by any party. Related to this, many may assume that, if there is a large amount of data, then whatever risks that may be embedded in that data are mitigated by obscurity. A real dataset where these issues are at play is the dataset of videos present on the YouTube platform. This project's goal is to elucidate the privacy risks that exist on the YouTube platform while

exploring the viability of large-scale analysis of video data, through local and cloud computing means. To further emphasize the risks even in a "best case scenario", where an adversary does not have access to proprietary high-cost tools, or expertise in the computer-vision or machine learning field; this work is conducted using affordable face recognition services, and easily obtainable open-source tools. While there exists related work pertaining to this objective, no work has truly treaded this same path. In the proceeding chapters, the system employed, it's details, and the observations from this system will be documented. Then finally, conclusions and a discussion on potential future works, will be presented.

CHAPTER 2

System Design

2.1 Big Picture

2.1.1 System Goal

The objective of the outlined system, as presented in figure 2.1, is to investigate the potential privacy risks in a dataset where face, time, and location information intersect. For this purpose the data available on YouTube was used, and videos pertaining to a specific time and location where considered. Additionally, in order to highlight potential privacy concerns, the outlined system made use of widely available open-source tools, and the paid computer services of Microsoft Azure. Furthermore, this system is outlined with another outcome in mind, namely the ability to generate larger facial datasets that include identity information along with other metadata associated with privacy, and which represent real world scenarios encountered in videographic facial content. This ability is related to the the question of privacy in that the ability to do so, if easy to do, would imply a greater privacy concern than previously understood.

2.1.2 Threat Model

Prior to the conception of a system, one must first consider the environment in which it will operate, and to that end the following privacy threat model was considered. The scenario of the threat model is such that an adversary has collected a video dataset whereby he is aware of the location of filming for these videos. Additionally, this adversary intends to broadly identify individuals in this dataset, thusly impacting the privacy of those individuals via an untargetted attack, as shown at a high level in figure 2.2A. Furthermore, a consideration in

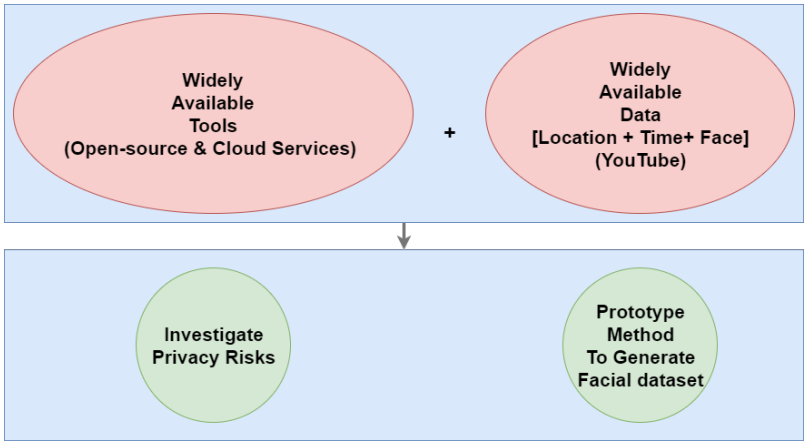

FIGURE 2.1: The system goals ultimately consist of investigating the potential privacy risks on a platform such as YouTube, as well as laying the ground work for producing large face-identity datasets in the future.

S

the threat model includes that the adversary may not have any specific knowledge in machine learning or face recognition, but does have access to a dataset, and can afford cloud-based face recognition tools. In the case of an environment such as YouTube, such an adversary might place a specific focus on individuals who appear in videos owned by different YouTube channel publishers, as this represents an increased opportunity that an individual was filmed without their knowledge. Should the adversary then use the facial information associated with a single individual to further find more instances of that individual appearance online, this would represent a targetted attack as shown at a high level in figure 2.2B. If such an adversary were able to succeed, it would then represent a privacy risk in the context of one of the previously mentioned goals in section 2.1.1.

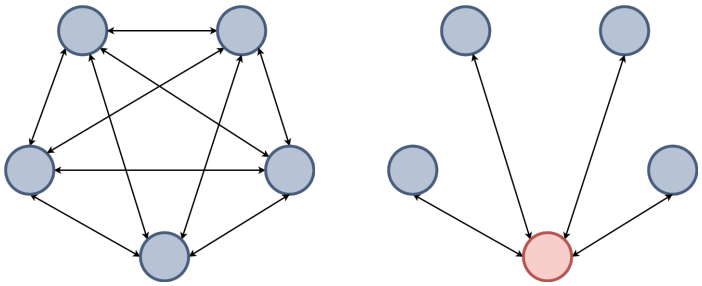

FIGURE 2.2:
(A) The threat model can be thought of as a graph, where nodes are faces, and their relationship is represented by edges; the adversary uses these relations to elucidate the different identities present; this can be thought of as an untargetted attack on privacy.
(B) The threat model can be extended to represent an adversary who is fixated on one set of faces, or one identity; this can be thought of as a targetted attack on privacy.

2.2 Data Pipeline

2.2.1 Overview

The data pipeline can be thought of in two halves, as presented in figure 2.3, one consisting of collecting face image data, and the other consisting of processing the face data to meet aforementioned objectives in the context of the threat model outlined in section 2.1.2. To the end of collecting face image data, first YouTube was queried for videos pertaining to a specific time range, and geolocation coordinate range (latitude/longitude/radius). Subsequently, the videos resulting from the query must be obtained, and processed into images— of which only those containing faces are of interest for the purposes of this work. For the processing of this resulting image data, Microsoft Azure was the primary tool, however a framework, as explained below, that creatively used the tools afforded to us by Azure, was required in order to meet the objective outlined in section 2.1.1. Additionally, it is important to remember that

while processing this data, it was done so with the mindset of the aforementioned adversary in section 2.1.2.

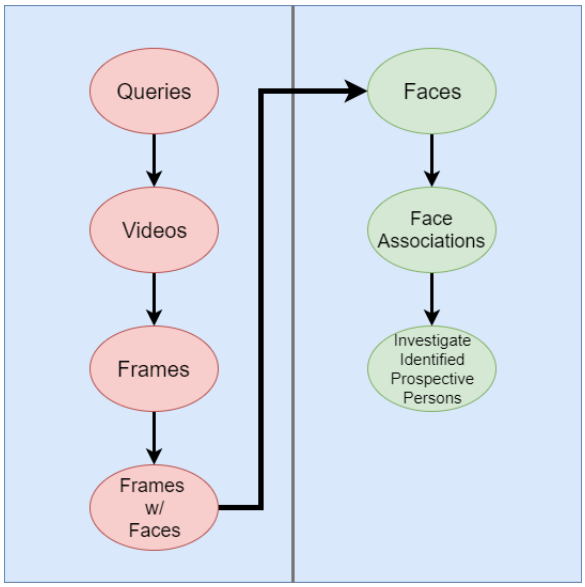

FIGURE 2.3: The Data pipeline as executed to achieve the project goals.

2.2.2 Data Collection

Data collection consisted of 3 clear phases, as outlined in figure 2.4, one consisted of searching for YouTube videos, the second consisted of downloading the videos which appeared in the aforementioned search results, and the third involved converting the videos into an image set, and detecting which of these images contained any faces.

Querying YouTube

YouTube was queried using the YouTube API (v3) [60], a freely available interface provided by YouTube & Google. The query was for all videos in San Francisco area [Constant Values 1 & 2] for a time period of 14 days. The 14 day period was between October 21st 2019,

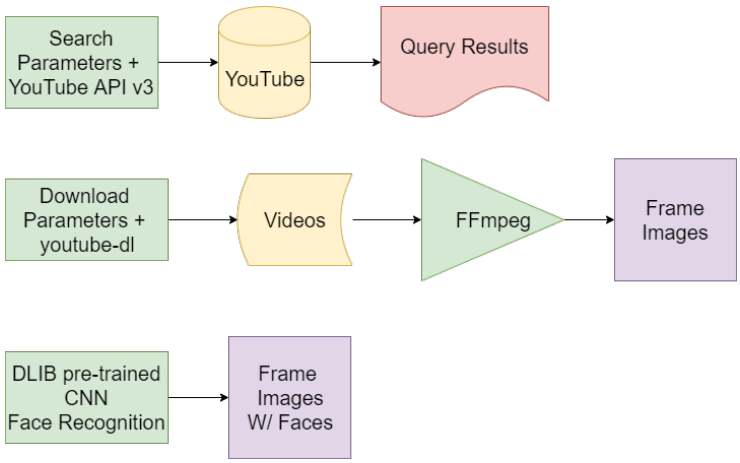

FIGURE 2.4: Data collection presented in three phases.

and November 4th 2019, using local time. No additional search parameters were used. San Francisco was selected due to it's geographic surroundings, which are shown in figure 2.5, and this has the San Francisco Area somewhat isolated from other cities in the San Francisco Bay Area. Additionally, the 14 day window was selected from prior experience with 1 day data collection, suggesting 14 days would be ample data for this work. Additionally, due to the population density of San Francisco [14], and the time frame selected straddling the Halloween holiday, it was thought there would be many videos uploaded in that area during this time. These properties would give a hypothetical adversary, as outlined before in section 2.1.2, the best chance of possibly detecting the same person in an area of interest, across different videos and channels on the YouTube platform. Lastly, metadata for the videos that the queries returned, such as their descriptions, comment counts, like/dislike counts, topics categories, were all made available by the YouTube API and were logged.

FIGURE 2.5: The San Francisco Area Examined, Visualized with a "Google Map Developers" Tool [12], Map data ©2019 Google.

Video Downloading

The best available video with no more than a 2160p resolution or 60 frames-per-second frame rate was downloaded. If the video was not available for any reason, a second attempt was made shortly after to account for possible YouTube server-side errors. No limit was placed on the time length or file size of the videos downloaded. The downloading of the videos was conducted using an open-source tool known as "youtube-dl" [18]; YouTube API does not allow for the downloading of videos.

Video Processing and Filtering

The videos were processed into an image set using tools provided by the open-source ffmpeg project [4], namely the "ffprobe" program for identifying the key-frames within a video, and then the "ffmpeg" program for extracting those key-frame images.

Given these images, it was then determined which images contained at least one face using a tool provided by the Dlib library [28]. Namely, the Dlib library includes a facial detection neural network capable of 99.38% accuracy [27] when tested using the Labelled Faces in the Wild (LFW) dataset [19], which is a known and often used benchmark in the field. These images derived from the frames which did contain faces were used henceforth in the data pipeline. This filtering was performed to reduce the cost of subsequent data processing using Microsoft Azure, as any adversary in the real world would aim to balance cost and performance. This is in keeping with the alluded aim in section 2.1.1, which is to assess the privacy risks in a dataset like YouTube, and if the cost of breaching privacy is too high, this would represent a lower risk of privacy loss.

2.2.3 Data Processing

The data processing can be thought of in two halves, first is the small scale processing, where operations are done on a image or video level (the images pertaining to that video) level. Second, is larger scale processing, where operations are done with the intent of bringing the data together to form cohesive results that an adversary, as described in section 2.1.2, would find interesting. The AFA processing capabilities were used in both of these halves. The data processing pipeline had to be structured in this manner due to one key limitation of the AFA, specifically, the AFA "group" method [36] only accepts an input of 1000 faces; if this method accepted hundreds of thousands of faces, it could be used directly to identify all separate people within a large collection of faces. The way this AFA method works, and how it was employed instead, is elaborated on in the immediately subsequent sections.

Face Detection

The frame images known to contain faces were provided to the AFA's "detect" method [35], which generated bounding boxes for each face it detected in the images, if any, along with a temporary (24hr expiration) face id. These bounding boxes were used to then extract face images, and save these as images separate from their parent frame images.

Small Scale Face Associations

Face images were submitted in batches to the AFA's group method, such that each batch did not span faces from more than one video of origin, and each batch contained faces in sequential order as they appeared in their video of origin. Azure in turn provided a list of groupings for these faces, such that each grouping represented what was suspected to be a different person. For any faces that could not be grouped, Azure returned these as a "Messy Group", and it was assumed that each one of these faces could be a different person. This is termed "Small Scale Face Associations" because at this point in the data pipeline,the faces are arranged by identity for each batch only, and not for the entire dataset from the original YouTube query.

Expandable Scale Face Associations

In order to handle the person-associated faces, and following the structure of the AFA, a "Large Person Group" (LPG) [38] was created which represented a collection of all perspective persons contained in this work's video dataset. Within this LPG, a "Large Person Group Person" (henceforth referred to as a Person Group Person, or PGP) [39] was created, which represented what was suspected to be collections whereby all faces in that collection represented one person, and that each PGP did not necessarily represent a different person compared to

another PGP. The PGPs were created such that they held images of the faces returned by each call to the AFA's group method. The faces were assigned to their respective PGP, and if accepted by Azure, would be given a permanent face ID string per assignment. This permanent face ID was not required in the downstream in the pipeline, however it is useful to have in order to delete that face assignment if so desired. Each face that was returned within a AFA's group method "Messy Group" was assigned to it's own PGP, consistent with the assumption that each of those faces could represent a different real person.

For each PGP, the largest face, with frame order (from the video of origin) as a tie-breaker, was used to "represent" that PGP. This representative face was submitted to the AFA's "identify" method [37], which in turn generated a list of up to 100 PGPs with an associated confidence score representing the likelihood that a face represents the same person as that PGP. As per formula 4, as defined below (definition 1), and as outlined in figure 2.6, each representative face and PGP is considered a node, and each of the confidence scores is considered the weight of a directed edge between a PGP's representative face, and another PGP; this was for the purposes of continued analysis as elaborated on below.

Definition 1 *A bipartite graph, G_{RFDG} = (RF,PGPs,RCS) consists of two disjoint sets of vertices, RF, and PGPs, and a set of directed edges $RCS \subseteq \{rcs = (rf_{pgp}, pgp, rcs_w) : rf \in RF,\ pgp \in PGPs\}$ that represents links between Representative Faces of Person Group Persons, and other Person Group Persons themselves, with a weight of rcs_w to quantify the strength of the link. Where RFDG stands for "Represenative Face Directed Graph".*

Finding Potential Persons of Interest

Related to the prior node/edge structure, an undirected graph where each node is a PGP was constructed as defined in formula 5, as defined below (definition 2), and as presented in 2.6.

In order to construct this undirected graph which would relate PGPs, the aforementioned directed graph was harnessed. From the directed graph, and as shown in figure 2.6, any edges representing the intersection of the confidence score probabilities that involved the two different PGPs were kept, and that intersection would become the new undirected edge's weight. Since there was one representative face per PGP, there were at most 2 confidence scores linking two PGPs. If there is only one directional edge between two nodes, then there is no undirected edge connecting these two PGPs in the constructed undirected graph.

Definition 2 *A graph,* G_{PRUG} *= (PGPs,ICS) consists of a set of vertices, PGPs, and a set of undirected edges* $ICS \subseteq \{ics = (u, v, ics_w) : u, v \in PGPs\}$ *that represents links between Person Group Persons, with a weight of* ics_w *to quantify the strength of the link. Where PRUG stands for "Person Relationship undirected Graph".*

Lastly, to the aim of reducing nodes with representative faces of poor identifying quality, edge pruning was performed as depicted in figure 2.6. If all the directed edges coming from a node had a weight over 0.5, or if no single edge had a score at or over 0.75, these edges were not considered when constructing the undirected graph. In the first case, such a face represents a face of low identification quality because it appears to be related to far too many faces to be plausible. In the second case, this represents a face which is not in any way certain to be related to any single person in this work's dataset. Essentially in both cases the associated edges are treated as noise to be removed. As seen in figure 2.6, the undirected graph structure was used along with a clustering method [8] to obtain clusters which represent what was suspected to be unique persons as an end result. However, following edge pruning, the clustering method was not of particular import since each identity was already clustered as a result of the pruning methodology. The aforementioned pruning cutoffs were determined through experience in manually verifying the resulting clusters. Given the threat model in

section 2.1.2, the clusters of import were those which were associated with more than one single YouTube channel.

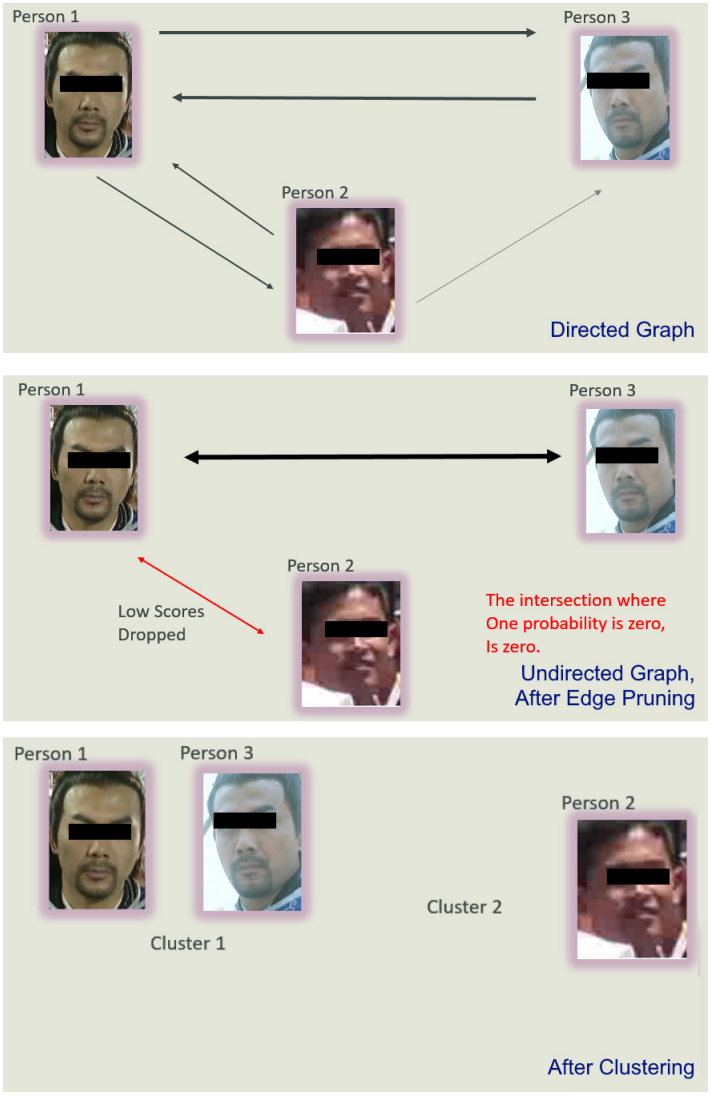

FIGURE 2.6: Structure and handling of the facial identity graph; at the end the clusters represent suspected unique persons.

CHAPTER 3

Additional Experiment Implementation Details

3.1 Local Equipment

The computer used in this work was equipped with a i7-7820X CPU [21], which has 16 cores. Additionally it utilized two Nvidia 2080-ti GPUs [43].

3.2 Data Collection

3.2.1 Querying YouTube

In order to work within the constraints of the YouTube API, namely the limits to the number of API calls per day [61], queries were done for the San Francisco area (Constant values 1 & 2) on a per day basis. That is to say, in the case of this work, 14 separate queries were done, however all were completed within the course of the day. This was done to avoid a scenario where only part of the results could be obtained, which would leave the question of which videos remained. The initial query returns basic information for each video [62], and additional queries were done for each video to get more detailed information as well.

3.2.2 Video Downloading

Videos resulting from the prior queries were downloaded in full, and organized by both their YouTube video ID, as well as their originating date of publishing in local time. The majority of these videos were in MP4 [23] video containers, and the minority used WEBM [56], which version was downloaded was dictated by YouTube as sometimes only one version was available; whichever format held the video of the highest video resolution, and frame rate, was used.

3.2.3 Video Processing & Filtering

Key-frames were first found for the entire video dataset using ffprobe, and as this was a CPU intensive process, multiprocessing was used to maximize concurrent processing. While this proceeded, ffmpeg was utilized using the nvdec [44] video decoder, and as this was a GPU intensive process, two GPUs were used for time efficiency; output from ffmpeg were saved as JPEG files [22] for the associated key-frames. While the prior steps occurred concurrently since one was CPU intensive, and the other GPU intensive, the next step waited for their completion. Face detection to detect which key-frames contained any faces was done using GPU processing, using both GPUs for time efficiency, with batch processing done as well, and with batch sizes determined dynamically depending on the frame image sizes to avoid using too much or too little GPU memory.

3.3 Data Processing

3.3.1 Face Detection

Face detection proceeded using Microsoft Azure, and the faces found were saved as separate files with unique ID strings; multiprocessing was used to ensure processing was done as close to the Azure transaction / second limit (Standard limit as specified by [40]) as possible.

3.3.2 Small Scale Face Associations

Face grouping proceeded using Microsoft Azure, with multiprocessing being used to ensure processing was done as close to the Azure transaction / second limit as possible.

3.3.3 Expandable Scale Face Associations

Creating PGPs and assigning faces to these persons proceeded with multiprocessing being used to ensure processing was done as close to the Azure transaction / second limit as possible. Furthermore, prior to being able to utilize the Azure "identify" method, the LPG required training via the "Train" method; this initiates black-box training for the LPG to be capable of differentiating between PGPs. Additionally, a quirk of the identify method is that it allows for submitting input in batches of 10, and this was done to reduce cost.

3.3.4 Finding Potential Persons of Interest

With regard to the edges generated in the graph schema using the Azure "identify" method, all edges were saved prior to conducting the mentioned edge pruning. Following clustering, cluster summary images were generated using the mentioned Representative faces to concisely view the accuracy of the clustering by manual inspection later.

CHAPTER 4

Observations

4.1 Data Collection

Some observations for the data collection process as a whole include both it's time and cost. For this two week dataset it took approximately one day to collect the data, and it cost $0 (not including equipment cost, or electricity).

4.1.1 Querying YouTube

The structure of the metadata available for each YouTube video as provided by the YouTube API included everything a YouTube user might see on the webpage for a YouTube video. However the metadata includes the specific coordinates associated with the video file, whereas, as shown in figure 4.1, on the YouTube site when a user views a video, it only shows the location by name or phrase. Additionally, the YouTube site's search interface, as presented in figure 4.2, only allows search by location phrase, but not coordinates; creating a false sense of location privacy. The result of the queries across the 14 days totalled to 2145 videos.

FIGURE 4.1: What a user sees when a YouTube video has location data, note it does not inform the user that geo-coordinates are saved, instead a phrase is used such as "Castro District" in this example.

FIGURE 4.2: What a user sees when searching for a YouTube video, note that for searching by location, it does not allow the user to search by geo-coordinates/radius, but only by name/phrase.

4.1.2 Video Downloading

While not explicit measured, video downloading proceeded without any apparent speed throttling, and the entire 288 GB video set downloaded within the course of a day. The

result of the downloading was 2099 videos, as some of the videos were (a) deleted by owner , (b) taken down by YouTube or (c) unavailable due to server-side issues. Additionally, as presented in figure 4.3, most of the videos were at or exceeded 720p (HD) resolution., Furthermore, as shown in 4.4, most videos were the only video uploaded by their respective YouTube channel that was within the YouTube query. Additionally, figure 4.5 shows that most of the videos in the Dataset were of very short duration, with most well under an hour in duration. Lastly, figure 4.6 the videos in the dataset were of low viewer engagement, with most view counts under 100, and even fewer likes/dislikes or comments.

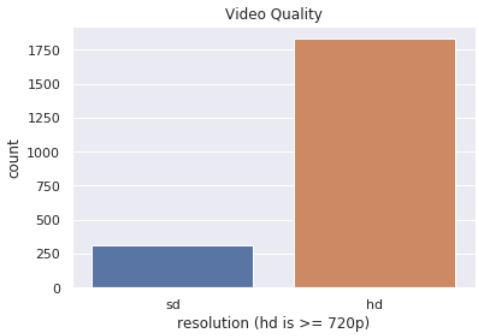

FIGURE 4.3: Video Quality of the obtained YouTube dataset.

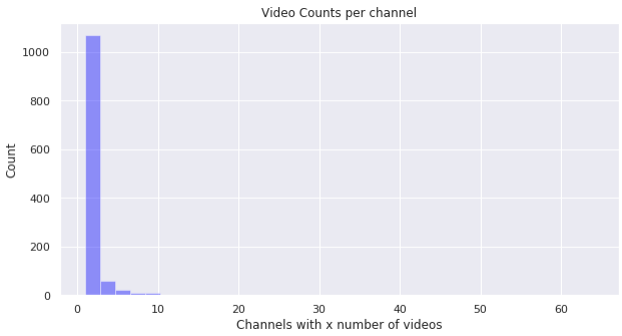

FIGURE 4.4: Number of videos uploaded by YouTube channels in the dataset; note that some channels had large numbers of videos, but they were infrequent given the total 1187 channels in the dataset.

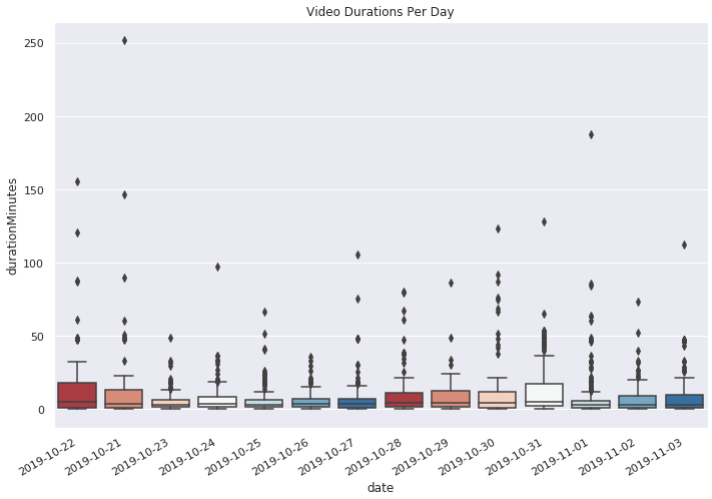

FIGURE 4.5: Video durations per day.

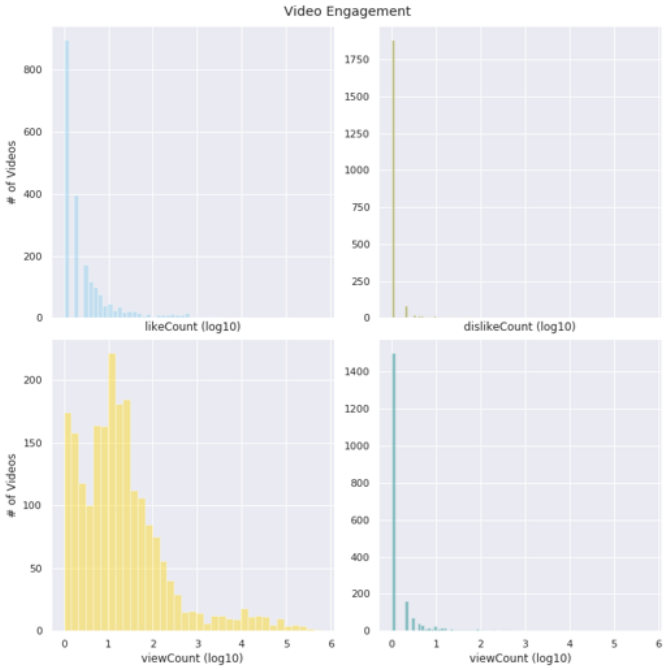

FIGURE 4.6: Video viewer activity in the dataset, log10(x) corrected, with +1 to prevent log of zero.

4.1.3 Video Processing & Filtering

Approximately 316k key-frames ($\boldsymbol{f}$ were extracted from the 2099 video dataset, with approximately 116k of these frames containing at least one face ($\boldsymbol{f'}$).

4.2 Data Processing

Some observations for the data processing process as a whole include both it's time and cost. For this two week dataset it took approximately half a day for this processing, and it cost approximately $320. While the Azure platform allows directly viewing the transactions as they occur within the Azure developer portal, one may want to estimate costs in the future, and to that end a cost function is presented in this work (Formula 3, and as seen here). Within this function, one can break down the cost to it's components, namely the cost due to the detect method, group method, creating PGPs, adding face images to these PGPs, training the LPG, and using the identity method; respectively each of these costs essentially make up the numerator of the cost function, with the denominator being 1000 due to AFA's basing of its costs around the price per thousand transactions (**PPTT**). The PPTT can change based upon the number of transactions used that billing cycle [40], and the prior approximate is based off the PPTT of $1, which is the most expensive, and starting price for the first million transactions used that billing cycle.

The explicit equation to estimate the cost in dollars is as follows (defined as formula 3):

$$Cost = \frac{|f'| + n_{gm} + n_p + f''_s + f''_s/1000 + n_{p'}/10}{1000} * PPTT,$$

where :

$|f'| \propto$ "Detect" Method Cost,

$n_{gm} \propto$ "Group" Method Cost,

$n_p \propto$ "Create PGP" Method Cost,

$f''_s \propto$ "Add face" Method Cost,

$f''_s/1000 \propto$ "Train" Method Cost,

$n_{p'}/10 \propto$ "Identify" Method Cost,

*With variables as defined

in bold in this section

4.2.1 Face Detection

Approximately 245k faces were detected by the AFA's detect method ($\boldsymbol{f''}$).

4.2.2 Small Scale Face Associations

Approximately 20k groups were formed using the AFA's group method, which directly relates to the number of PGPs created ($\boldsymbol{n_p}$), and the number of calls to the group method was approximately 2k ($\boldsymbol{n_{gm}}$).

4.2.3 Expandable Scale Face Associations

Not all groups were successfully translated into "persons" within the AFA framework, this is because faces detected earlier were not always accepted by the "add face" method for persons. Therefore, approximately 17k persons ($\boldsymbol{n_{p'}}$) with a total of 181k faces ($\boldsymbol{f''_s}$) were created, out of the original 20k groups (245k faces). It is believed that this is due to the AFA "person" framework requiring a slightly higher floor quality level for the face images it will accept.

4.2.4 Finding Potential Persons of Interest

Summary of Cluster Findings

Following the outlined edge pruning and clustering procedure as described in section 2.2.3, 98 clusters were found that spanned videos from two or more YouTube channels; these represent prospective persons of interest for an adversary as per the threat model explained in section 2.1.2. The manually verified findings from these 98 clusters is summarized below [Table 4.1]. The clusters were binned into different categories. These clusters are visualized below, detailed information contained within these figures include the maximum and minimum directed edge weights (probabilities), and the undirected edge weights (under "Edge Info" in these figures). The nodes/edges are alluded to are as shown in Figure 2.6. As shown in figures 4.7–4.16, each "node" in the aforementioned graph has a representative face, and pertains to a "person" (here called PGP, or Person Group Person) within the Azure framework— all faces seen in the proceeding cluster figures where the representative face of their PGP.

Cluster Types	**Subtypes**	**Special Cases**	**Amount**	**Totals**
Good Clusters				51
	Celebrities in various contexts		14	
		One included a celebrity with makeup on/off		
	Non-Celebrity Real People		11	
	Photos of Faces in the Real World		10	
	Human-like Statues or Realistic Faces in Video Games		6	
	Non-Celebrities in Duplicate Videos		4	
	Amateur Public Performers		4	
	Underage Users		2	
Bad Clusters				35
	Due to a Mixture of Factors: Quality, Angle, Obstruction		13	
	Children / Toddlers		9	
	Low Quality		6	
	Extreme Side Angles		6	
	Glasses / Hat Obstruction		1	
Other Clusters				12
	Cartoons / Art		11	
	Unable to Verify Cluster		1	
All Clusters				98

TABLE 4.1: Summary of identity clustering results.

Good Clusters

For those that properly contained a single identity (Good clusters), celebrity, non-celebrity, photos in the real world, non-celebrities in duplicated YouTube videos, human-like statues or video game faces, amateur public performers (not shown), and underage users (not shown) were observed in figures 4.7, 4.9, 4.10, 4.11, & 4.12, respectively. With respect to photos in the real world, this was observed because several people were visiting Alcatraz Island, and recorded the posters of the mug shots of several of the island's prior prisoners. A sign of robustness was observed in figure 4.8, with a successful clustering of a celebrity despite a significant difference in face makeup. The amateur public performers were non-celebrities, and were performing at different events in the San Francisco area, with some of the YouTube videos having identifying information in their description.

FIGURE 4.7: Good Cluster - Celebrity: John Travolta appearing in various source material (Hollywood movies).
(YouTube identifying information pixelated)

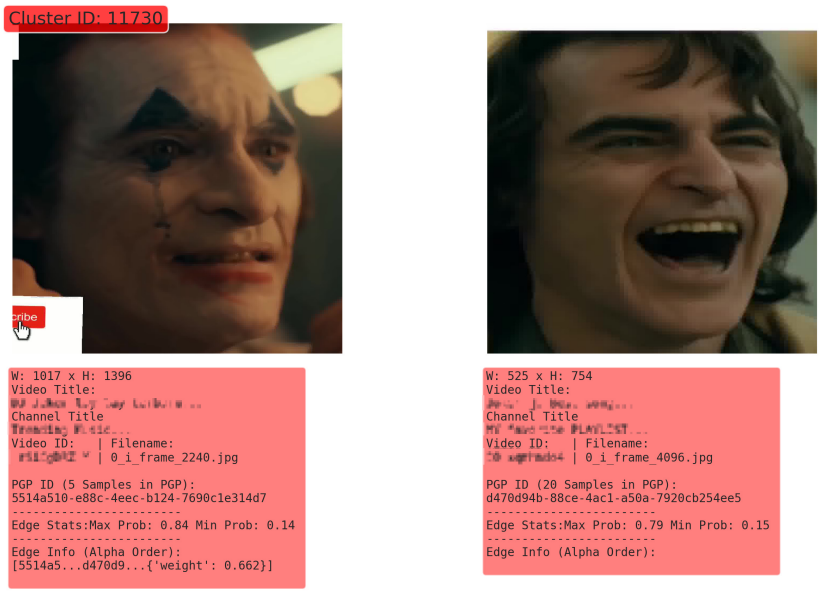

FIGURE 4.8: Good Cluster - Celebrity Special Case: Joaquin Phoenix, with and without makeup.
(YouTube identifying information pixelated)

FIGURE 4.9: Good Cluster - Non-Celebrity: Young Adult Male.
(YouTube identifying information pixelated and blocked out)

FIGURE 4.10: Good Cluster - Photo in Real World: Alcatraz Prison Poster Photos.
(YouTube identifying information pixelated)

FIGURE 4.11: Good Cluster - Non-Celebrity in Duplicate Video: Supporting actor in a movie trailer that was reposed in multiple YouTube channels.
(YouTube identifying information pixelated)

Bad Clusters

For those that did not properly contain a single identity (Bad clusters), there were various reasons for this, which was apparent during manual inspection. These reasons included either, low quality, young child faces (not shown), extreme side incidence angle to face, and

FIGURE 4.12: Good Cluster - Video Game Face: A realistic looking face that commonly appeared in videos which contained playthroughs of video games. (YouTube identifying information pixelated)

obstruction of the face; or at times a combination of the aforementioned reasons (not shown), as exemplified in figures 4.13, 4.15, 4.14, respectively.

Other Clusters

Other cluster observations include one cluster that simply could not be verified by eye or by inspecting the source video (not shown), and clusters involving highly unrealistic faces of a cartoon nature (sometimes properly matched), with a cartoon example shown in figure 4.16.

FIGURE 4.13: Bad Cluster - Low Quality: Where quality is determined by image resolution and clarity. In this case the hat does not appear to be the primary cause of the bad clustering, but rather the images are both nearly 50x50 pixels, which is poor for facial recognition.
(YouTube identifying information pixelated)

FIGURE 4.14: Bad Cluster - Obstruction of the Face: Where obstruction can include things like glasses or hats.
(YouTube identifying information pixelated)

FIGURE 4.15: Bad Cluster - Extreme Side Angle of the Face: Cannot properly see the face from the front in one or more representative images, due to face angle.
(YouTube identifying information pixelated)

FIGURE 4.16: Other Cluster - Cartoon: Unrealistic face images.
(YouTube identifying information pixelated)

Further Investigation of an Identified Individual

The non-celebrity real person shown figure 4.9 was observed at their university and at a political rally in figure 4.17. Additionally, figure 4.17, shows the persons name along with that of other people in the videos in the description. Furthermore, using the posted name, it was trivial to find that persons other social media accounts, and date of birth, with screen-captures showing this presented in figure 4.18; this was observed while acting in accordance with the thread model mentioned in section 2.1.2.

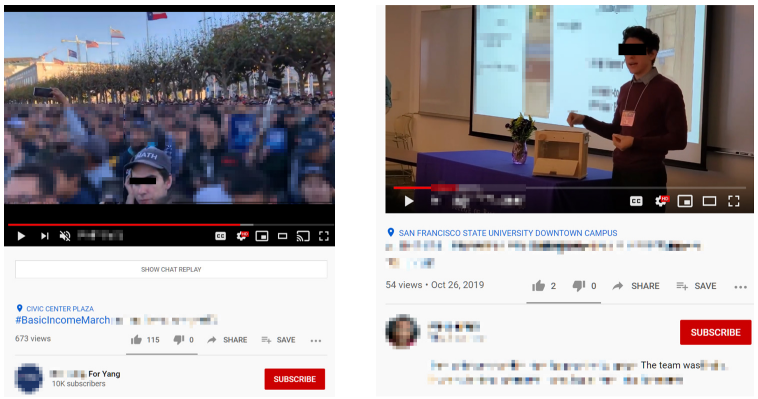

FIGURE 4.17: Good Cluster - Example Privacy Exposure: To the left one can see the video is of a political nature, to the right one can see the video is of an academic presentation. Note that in the video to the right, the description states "the team was", and here the persons name is printed.
(YouTube identifying information pixelated)

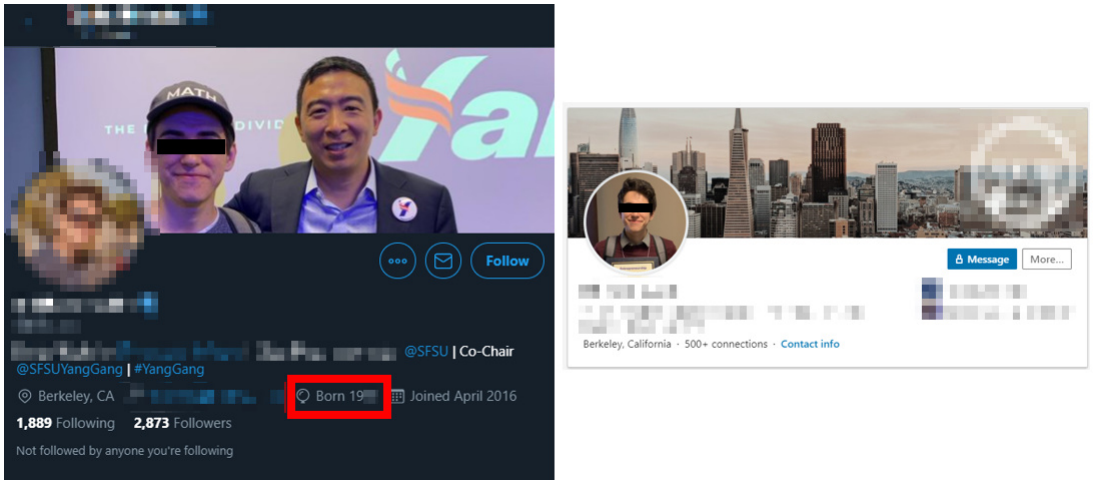

FIGURE 4.18: Good Cluster - Example Privacy Exposure & Other Social Media: Here the social media of the person identified in figure 4.9 can be seen. Organizational affiliations and some personal information, such as date of birth, are present. In this case the user is open about their political affiliations and is posing with a U.S. presidential candidate for 2020, however, this may not always be the case.
(YouTube identifying information pixelated)

CHAPTER 5

Conclusions

In conclusion, there exists a vulnerability to personal privacy where both facial information and location information intersect. In one specific case in the dataset generated by this work, as shown in figures 4.9,4.17, & 4.18, a persons name and date of birth was exposed, as well as their political affiliation. The size of the dataset does not provide additional privacy with regard to the defined threat model presented in section 2.1.2, and the idea that one could hide in plain sight in this scenario is questionable at best. This vulnerability is bolstered by the use of an online face recognition service which does not require extreme funding, or specific expertise in the machine learning field. Furthermore, the improvements to be discussed in the future work section 6 will only serve to improve the results, and thus improve the future ability to penetrate privacy in the dataset. Additionally, it is important to remember that in this work a broad threat model was used, as shown in figure 2.2A; but in the face of a specifically targeted attack, as shown in figure 2.2B, the adversary would have a much simpler problem. This is especially true if the adversary has additional query specifications to reduce the dataset. For example, instead of just location and time, an adversary in some hypothetical scenario may be interested in certain video topics, or key words in the video description. This would reduce the dataset, and thus improve the performance, and reduce the cost for the adversary. Important to remember as well, is the fact that this work pertains to the YouTube dataset, but it can be extrapolated to a scenario such as if an adversary has access to a network of hacked IP-cameras, or camera-containing devices, of which estimates of location can be made based off the IP addresses, [13]. The the risk to privacy in this scenario has a potential to be far greater as there may be more expectation of privacy on the part of the monitored people.

CHAPTER 6

Future Work and Applications

Future work based on limitations found while executing this project, could include several avenues of research. One such limitation is that it cannot be easily determined when two faces are not associated when they should be, and one such remedy would be to use a tool like Amazon Mechanical Turk [11] to examine faces grouped by broad features such as gender. Additionally, improved video frame extraction techniques based upon scene detection could improve efficiency of the data pipeline. Furthermore, the filtering of faces which commonly cause poor clustering results can be considered. To facilitate this, it may be possible to achieve better results by instead filtering out entire videos, or parts of videos, by estimating the quality of the videos; such quality estimations are discussed in another work [41]. However, filtering must be done with caution as to not over-filter possibly useful information.

Other considerations for future work include the training of a local neural network solution; which could be done using data generated by the pipeline outlined in this work. It was a specific goal of this work to lay the ground work for a system capable of creating a large face-identity dataset for largely this reason. Additionally, while a threat model was used in this work , as explained in section 2.1.2, there was no exploration of defenses against an adversary. As alluded to briefly in the introduction section 1.2.4, there has been work to conceal the identity of a person within visual content; typically this involves radical obfuscation of the face region. The author believes there is potential for a privacy solution that conceals identity of a face without the destruction of the quality and utility of the underlying video content, as would be the case with extreme face obfuscation techniques.

Other possible applications that could draw influence from this work include a tool to detect copyright infringement by using when specific faces appear to fingerprint content. Additionally, one could envision organizations, such as public relations firms, being interested in a tool that allows them to easily find appearances of clients or other persons of interest within a set of videos.

References

[1] Abdulhamid, Shafii M et al.
"Privacy and national security issues in social networks: the challenges".
In: *arXiv preprint arXiv:1402.3301* (2014).

[2] Abu-El-Haija, Sami et al. "Youtube-8m: A large-scale video classification benchmark".
In: *arXiv preprint arXiv:1609.08675* (2016).

[3] Acquisti, Alessandro, Gross, Ralph, and Stutzman, Frederic D.
"Face recognition and privacy in the age of augmented reality".
In: *Journal of Privacy and Confidentiality* 6.2 (2014), p. 1.

[4] Bellard, Fabrice. *FFmpeg project*. https://ffmpeg.org/. 2000.

[5] Beveridge, J Ross et al.
"Factors that influence algorithm performance in the face recognition grand challenge".
In: *Computer Vision and Image Understanding* 113.6 (2009), pp. 750–762.

[6] Bhanu, Bir et al. *Distributed video sensor networks*.
Springer Science & Business Media, 2011.

[7] Biel, Joan-Isaac, Teijeiro-Mosquera, Lucía, and Gatica-Perez, Daniel.
"Facetube: predicting personality from facial expressions of emotion in online conversational video".
In: *Proceedings of the 14th ACM international conference on Multimodal interaction*.
ACM. 2012, pp. 53–56.

[8] Blondel, Vincent D et al. "Fast unfolding of communities in large networks".
In: *Journal of statistical mechanics: theory and experiment* 2008.10 (2008), P10008.

[9] Brown, Blain. *The Filmmaker's Guide to Digital Imaging: for Cinematographers, Digital Imaging Technicians, and Camera Assistants*. Routledge, 2014.

[10] Choi, Jae Young et al. "Collaborative face recognition for improved face annotation in personal photo collections shared on online social networks".
In: *IEEE Transactions on Multimedia* 13.1 (2010), pp. 14–28.

[11] Crowston, Kevin. "Amazon mechanical turk: A research tool for organizations and information systems scholars".
In: *Shaping the Future of ICT Research. Methods and Approaches*. Springer, 2012, pp. 210–221.

[12] Developers, Google Map. *Draw Circle Tool*.
https://www.mapdevelopers.com/draw-circle-tool.php. 2019.

[13] Dong, Ziqian et al.
"Network measurement based modeling and optimization for IP geolocation".
In: *Computer Networks* 56.1 (2012), pp. 85–98.

[14] Ewing, Reid H and Hamidi, Shima. *Measuring sprawl 2014*.
Smart Growth America, 2014.

[15] Fagan, Kaylee. *Everything you need to know about 'swatting,' the dangerous so-called 'prank' of calling a SWAT team on someone*. June 2018.
URL: https://www.businessinsider.com/what-does-swatting-mean-2015-3.

[16] Fassold, Hannes and Rosner, Jakub. "A real-time GPU implementation of the SIFT algorithm for large-scale video analysis tasks".
In: *Real-Time Image and Video Processing 2015*. Vol. 9400.
International Society for Optics and Photonics. 2015, p. 940007.

[17] Gibbon, David and Liu, Zhu. "Large scale content analysis engine". In: *Proceedings of the First ACM workshop on Large-scale multimedia retrieval and mining*. ACM. 2009, pp. 97–104.

[18] Gonzalez, Ricardo Garcia. *youtube-dl project*.
https://github.com/ytdl-org/youtube-dl/. 2006.

[19] Huang, Gary B. et al. *Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments*. Tech. rep. 07-49.
University of Massachusetts, Amherst, 2007.

[20] Huang, Zhiwu et al. "Face recognition on large-scale video in the wild with hybrid Euclidean-and-Riemannian metric learning".
In: *Pattern Recognition* 48.10 (2015), pp. 3113–3124.

[21] Intel. *i7-7820X Central Processor Unit*.
https://ark.intel.com/content/www/us/en/ark/products/123767/intel-core-i7-7820x-x-series-processor-11m-cache-up-to-4-30-ghz.html. 2017.

[22] ISO/IEC. *JPEG Standard*. https://www.iso.org/standard/18902.html. 1992.

[23] ISO/IEC. *MP4 Standard*. https://www.iso.org/standard/38539.html. 2004.

[24] Jhonsa, Eric. *How Much Could Google's YouTube Be Worth? Try More Than $100 Billion - Stock Market - Business News, Market Data, Stock Analysis*. May 2018.
URL: https://www.thestreet.com/investing/youtube-might-be-worth-over-100-billion-14586599.

[25] Karlsen, Joakim, Stigberg, Susanne Koch, and Herstad, Jo.
"Probing privacy in practice: privacy regulation and instant sharing of video in social media when running". In: *Proceedings of the 2016 International Conference on Advances in Computer-Human Interactions*. 2016, pp. 29–36.

[26] Kaseb, Ahmed S et al. "A system for large-scale analysis of distributed cameras".
In: *2014 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE. 2014, pp. 340–344.

[27] King, Davis E. *Dlib: High Quality Face Recognition with Deep Metric Learning.* https://blog.dlib.net/2017/02/high-quality-face-recognition-with-deep.html. 2017.

[28] King, Davis E. "Dlib-ml: A Machine Learning Toolkit".
In: *Journal of Machine Learning Research* 10 (2009), pp. 1755–1758.

[29] Kostakos, Panos et al. "Inferring Demographic data of Marginalized Users in Twitter with Computer Vision APIs".
In: *2018 European Intelligence and Security Informatics Conference (EISIC)*. 2018, pp. 81–84.

[30] Loebel, Jens-Martin. "Is privacy dead?–An inquiry into GPS-based geolocation and facial recognition systems".
In: *IFIP International Conference on Human Choice and Computers*. Springer. 2012, pp. 338–348.

[31] Lowe, David G et al. "Object recognition from local scale-invariant features." In: *iccv*. Vol. 99. 2. 1999, pp. 1150–1157.

[32] McClurg, Andrew J.
"In the face of danger: Facial recognition and the limits of privacy law".
In: *Harvard Law Review* 120.7 (2007), pp. 1870–1891.

[33] Mennecke, Brian et al.
"Privacy in the age of big data: The challenges and opportunities for privacy research".
In: (2014).

[34] Microsoft-Azure. *Microsoft Azure Face API.* https://docs.microsoft.com/en-us/azure/cognitive-services/face/overview. 2019.

[35] Microsoft-Azure. *Microsoft Azure Face API - Detect.* https://westus.dev.cognitive.microsoft.com/docs/services/563879b61984550e40cbbe8d/operations/563879b61984550f30395236. 2019.

[36] Microsoft-Azure. *Microsoft Azure Face API - Group.* https://westus.dev.cognitive.microsoft.com/docs/services/563879b61984550e40cbbe8d/operations/563879b61984550f30395238. 2019.

[37] Microsoft-Azure. *Microsoft Azure Face API - Identify.* https://westus.dev.cognitive.microsoft.com/docs/services/563879b61984550e40cbbe8d/operations/563879b61984550f30395239. 2019.

[38] Microsoft-Azure. *Microsoft Azure Face API - Large Person Group.* https://westus.dev.cognitive.microsoft.com/docs/services/563879b61984550e40cbbe8d/operations/599acdee6ac60f11b48b5a9d. 2019.

[39] Microsoft-Azure. *Microsoft Azure Face API - Large Person Group Person.* https://westus.dev.cognitive.microsoft.com/docs/services/563879b61984550e40cbbe8d/operations/599adf2a3a7b9412a4d53f42. 2019.

[40] Microsoft-Azure. *Microsoft Azure Face API Standard Price.* https://azure.microsoft.com/en-us/pricing/details/cognitive-services/face-api/. 2019.

[41] Moorthy, Anush K et al. "Subjective analysis of video quality on mobile devices". In: *Sixth International Workshop on Video Processing and Quality Metrics for Consumer Electronics (VPQM), Scottsdale, Arizona*. Citeseer. 2012.

[42] Naker, Sharon and Greenbaum, Dov. "Now you see me: Now you still do: Facial recognition technology and the growing lack of privacy". In: *BUJ Sci. & Tech. L.* 23 (2017), p. 88.

[43] Nvidia. *2080 ti Graphics Processor Unit.* https://www.nvidia.com/en-eu/geforce/graphics-cards/rtx-2080-ti/. 2018.

[44] Nvidia. *NVDEC.* https://developer.nvidia.com/nvidia-video-codec-sdk. 2017.

[45] Perez, Alfredo J, Zeadally, Sherali, and Griffith, Scott. "Bystanders' privacy". In: *IT Professional* 19.3 (2017), pp. 61–65.

[46] Poms, Alex et al. "Scanner: Efficient video analysis at scale". In: *ACM Transactions on Graphics (TOG)* 37.4 (2018), p. 138.

[47] Reidenberg, Joel R. "Privacy in public". In: *U. Miami L. Rev.* 69 (2014), p. 141.

[48] Ricanek Jr, Karl and Boehnen, Chris. "Facial analytics: from big data to law enforcement". In: *Computer* 45.9 (2012), pp. 95–97.

[49] Shaw, Jonathan. "FACEbook Confidential: The Privacy Implications of Facebook's Surreptitious and Exploitative Utilization of Facial Recognition Technology". In: *Temp. J. Sci. Tech. & Envtl. L.* 31 (2012), p. 149.

[50] Shen, J. et al. "The First Facial Landmark Tracking in-the-Wild Challenge: Benchmark and Results". In: *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*. Dec. 2015, pp. 1003–1011. DOI: 10.1109/ICCVW.2015.132.

[51] Smith, Matthew et al. "Big data privacy issues in public social media". In: *2012 6th IEEE International Conference on Digital Ecosystems and Technologies (DEST)*. IEEE. 2012, pp. 1–6.

[52] Song, Chen et al. "Scalable distributed visual computing for line-rate video streams". In: *Proceedings of the 9th ACM Multimedia Systems Conference*. ACM. 2018, pp. 186–194.

[53] Tan, Hanlin and Chen, Lidong.
"An approach for fast and parallel video processing on Apache Hadoop clusters".
In: *2014 IEEE International Conference on Multimedia and Expo (ICME)*.
IEEE. 2014, pp. 1–6.

[54] Vandal, Thomas, McDuff, Daniel, and El Kaliouby, Rana.
"Event detection: Ultra large-scale clustering of facial expressions".
In: *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*. Vol. 1. IEEE. 2015, pp. 1–8.

[55] Wang, Yongzhe et al. "A cloud-based large-scale distributed video analysis system".
In: *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2016, pp. 1499–1503.

[56] WEBM-Project. *WEBM Project Site*. https://www.webmproject.org/. 2010.

[57] Wolf, Lior, Hassner, Tal, and Maoz, Itay.
Face recognition in unconstrained videos with matched background similarity.
IEEE, 2011.

[58] Yan, Yuzhong and Huang, Lei. "Large-scale image processing research cloud".
In: *Cloud Computing* (2014), pp. 88–93.

[59] Yang, Shuai and Wu, Bin. "Large scale video data analysis based on spark".
In: *2015 International Conference on Cloud Computing and Big Data (CCBD)*.
IEEE. 2015, pp. 209–212.

[60] YouTube. *Youtube API (v3)*. https://developers.google.com/youtube/v3. 2015.

[61] YouTube. *Youtube API (v3) Quota*.
https://developers.google.com/youtube/v3/getting-started#quota. 2015.

[62] YouTube. *Youtube API (v3) Video Details*.
https://developers.google.com/youtube/v3/docs/videos. 2015.

[63] Yu, Kai et al. "A Large-scale Distributed Video Parsing and Evaluation Platform".
In: *Chinese Conference on Intelligent Visual Surveillance*. Springer. 2016, pp. 37–43.

[64] Zhou, Lili, Lv, Jinna, and Wu, Bin. "Social network construction of the role relation in unstructured data based on multi-view".
In: *2017 IEEE Second International Conference on Data Science in Cyberspace (DSC)*. IEEE. 2017, pp. 382–388.