Aiding Diagnosis of Rare Diseases from Photographs using Machine Learning

Mitchell Dawson

Oriel College



University of Oxford

A thesis presented for the degree of

Doctor of Philosophy

Trinity 2019

This thesis is submitted to the Department of Engineering Science, University of Oxford, in fulfilment of the requirements for the degree of Doctor of Philosophy. This thesis is entirely my own work, and except where otherwise stated, describes my own research.

Mitchell Dawson, Oriel College

Copyright © 2019
Mitchell Dawson
All Rights Reserved

Abstract

The objective of this thesis is to further research in the field of computational syndrome diagnosis from ordinary photographs of patient faces. As well as providing insight into the phenotypic scope of specific genetic disorders, the development of such tools may eventually help to shorten the process for patients to receive a clinical diagnosis.

The success of computer vision algorithms in this domain crucially depends on sourcing large labelled data sets of patient syndrome faces. In this thesis, we address this issue by creating a download pipeline to obtain facial images of genetic disorder patients from the published biomedical literature. The development of this pipeline has led to the creation of a novel biomedical figure classifier; and a syndrome annotation tool, which enables rapid diagnostic labelling of retrieved patient images.

Critical to modelling the craniofacial manifestations associated with genetic disorders is the need to separate phenotypic signals caused by the genetic disorder from the typical variation found among inherited facial features. Unfortunately, existing kinship data sets display several biases which undermine the task of facial kinship modelling. To demonstrate this, we build a classifier which can determine whether two facial images are cropped from the same original photo. Despite this classifier having no understanding of kinship, it is able to achieve comparable performance to many previously published methods on several existing kinship data sets.

To counter this, we construct two novel kinship data sets collected from the Geni ancestry website, being careful to avoid several common biases found in other kinship data sets. Finally, we attempt to benchmark the kinship signal in our newly produced data sets using feature aggregation methods. The data sets and techniques discussed in this thesis are currently being used to produce the next generation of diagnostic syndrome face models.

Acknowledgements

Firstly, I would like to thank my supervisors, Christoffer Nellåker and Andrew Zisserman, for the opportunities, lessons and guidance they have given me over the last four years.

I also wish to thank the students and researchers of VGG group. In particular to Sam, Karel, Abhishek, Yujie, Carlos, Sophia, Ernesto, João, Aravindh, and Elliot, for their willingness to pause their own research to share their knowledge and assist in mine. Also, to Mohsan, who first helped get me up to speed with the field of machine learning, and has a been a great ally throughout my DPhil.

In addition, I am grateful the Systems Biology DTC, the EPSRC, and Oxford Metrics for financial supporting my research.

A very big thank you to my Mum and Dad, and the Dawson, Haynes, Jacobs, Parker, Sutton and Luke families for their love, support, encouragement and enthusiasm. Finally, and most importantly, thank you to Nicola, without whom this simply would not have been possible.

Table of Contents

1 Introduction **1**

1.1 Main challenges . . . 3

1.2 Contributions and thesis outline . . . 4

1.3 Publications . . . 6

I Mining syndrome faces from research articles **7**

2 Literature review – syndrome face classification **8**

2.1 Computer aided diagnosis . . . 9

2.2 Machine learning and deep learning for CAD . . . 10

2.3 Discriminative facial features using deep learning . . . 11

2.4 Computational syndrome diagnosis . . . 12

2.5 Obtaining patient images . . . 15

2.6 Extracting and classifying biomedical figures . . . 16

3 Mining facial photographs from genetic disorder articles **18**

3.1 PubMed download pipeline . . . 20

3.2 Results of mining . . . 29

3.3 Summary . . . 30

4 Classifying biomedical figures using deep learning **32**

4.1 Training a classifier using synthetic data . . . 34

4.2 Testing the classifier on real biomedical figures . . . 38

4.3 Case study – rare genetic disorders . . . 42

4.4 Summary . . . 45

5 Annotating syndrome faces **46**

5.1 Annotation tasks and tool design 48

5.2 Results of annotation 54

5.3 Summary 58

II Facial kinship **59**

6 Literature review – kinship verification **60**

6.1 Kinship verification data sets 61

6.2 Bias in kinship data sets 63

6.3 Approaches to kinship verification 64

7 From same photo bias in kinship data sets **66**

7.1 Training a classifier to detect faces from the same photo 68

7.2 Testing the FSP classifier on kinship data sets 73

7.3 Results of testing FSP classifier on kinship data sets 75

7.4 Summary 78

8 The Geni ancestry kinship data set **80**

8.1 Mining Geni ancestry website 82

8.2 Cleaning the raw Geni data 84

8.3 Classifying image usefulness 88

8.4 Managing data set bias 90

8.5 Geni pairs and triples kinship data sets 94

8.6 Summary 99

9 NetVLAD feature aggregation for facial kinship verification **100**

9.1 Pre–training NetVLAD for facial recognition 103

9.2 Training NetVLAD for kinship feature aggregation 105

9.3 Testing on Geni MFC triplets 107
9.4 Summary 109

10 Summary **110**

10.1 Achievements 110
10.2 Future work 112

1 | Introduction

From a clinical perspective, cranial malformations can often provide visual evidence that a patient may be suffering from a rare disorder (see Figure 1.1). Rare disorders individually affect less than one in every 2000 members of the general population (European Parliament & the Council of the European Union, 2000). However, collectively, they are surprisingly common: directly impacting around seven percent of the human population (Rare Disease UK, 2019). Currently, there are between 6000 and 8000 known rare genetic disorders, and it is likely that there are many more in existence that are, as yet, uncatalogued.

Many of these disorders cause symptoms which can greatly reduce a person's quality of life. However, it is believed that the majority of people living with a genetic disorder around the world have not received a clinical, let alone genetic, diagnosis (Hart & Hart, 2009). Such a diagnosis could allow a patient to undergo more targeted, diagnostically informed treatment for their condition.

In the last five years, several groups have attempted to develop computer vision methods for the task of syndrome classification from ordinary photographs of patient faces (Ferry et al., 2014; Gurovich et al., 2018). Although there has been some success, widespread progress has been restricted by the lack of large, high–quality, publicly available syndrome face data sets. In addition to providing the necessary algorithmic training examples, such a data set would allow researchers to assess the generalisability of their models more accurately.

The task of syndrome classification is made particularly difficult by the sheer variation and subtleties with which genetic disorders can manifest as facial phenotypes. Often, it is a challenging task to visually separate craniofacial characteristics which are indicative of a particular disorder from those which are due to healthy variation, for example due to age or sex. Some groups have used average syndrome faces to try and delineate salient phenotypic features (Ferry et

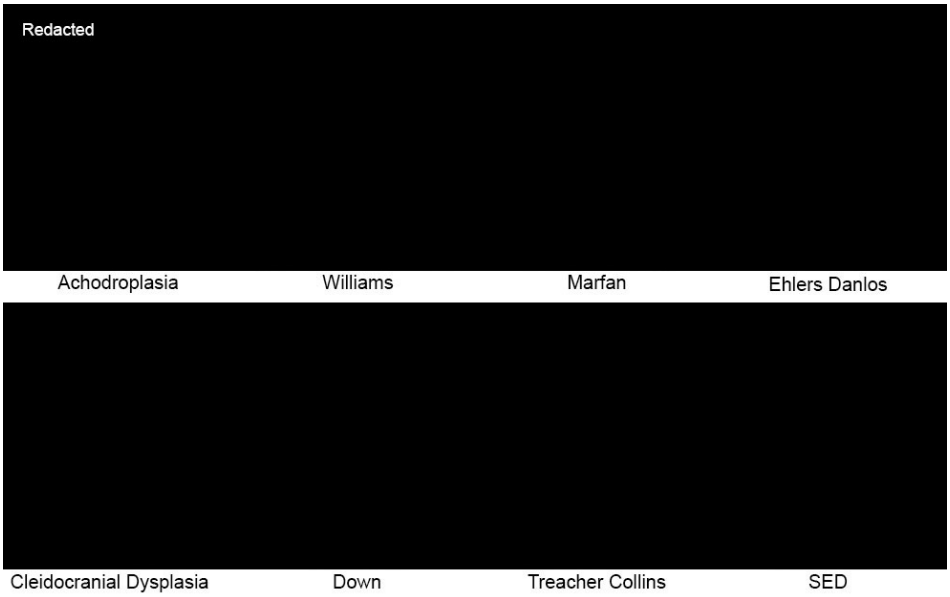


Figure 1.1: Examples of public figures known to have been diagnosed with a genetic disorder. In some cases, craniofacial features can provide visual evidence for the diagnosis of a particular syndrome.

al., 2014; Bengani et al., 2017). However, in a practical diagnostic context, an approach more personalised to individual patients is required. In particular an understanding of the interaction between a syndrome and the patient's inherited facial features could be greatly informative.

In this thesis, we follow two major tracks of investigation, both of which aim to support research into diagnosing rare genetic disorders using only an ordinary photograph of a patient's face. Firstly, to directly support the field of computational syndrome classification, we design and implement a method for obtaining large numbers of rare genetic disorder patient faces labelled by syndrome. Our approach to achieve this goal is to identity and obtain faces of patients from published biomedical research articles.

Secondly, we undertake research in the field of computational facial kinship verification, with the shared goal of understanding how one might model the visual relationship between a child and parent's faces. We assess the suitability of current kinship data sets, and subsequently produce a novel, large–scale data set for the task of kinship facial modelling. Finally, we wish

to make the preliminary steps in modelling facial kinship by using our new data set to produce a kinship classification model.

1.1 Main challenges

Large–scale syndrome face data sets. Modern computer vision methods are able to extract discriminative features from facial images. Convolutional neural networks (CNNs) have been shown to be able to accurately classify human characteristics such age, sex, and ancestral background from a single face image (Rothe et al., 2015; Buolamwini & Gebru, 2018; M. Wang et al., 2018). These results have been enabled by the existence of large labelled data sets of relevant images. Individual clinical groups often collect facial images of patients with a particular syndrome (Gorlin et al., 2001), and at least one private syndrome face data set has been collected by the company Face2Gene (Gurovich et al., 2018). Currently, however, no such publicly available, large–scale data set exists for the task of syndrome classification.

Utilisation of patient case studies. Over the last 50 years, a huge amount of research into the phenotypic manifestations of different genetic disorders has been undertaken; with much knowledge published within scientific journal articles. These case studies provide a rich source of syndrome face patient photographs. However, identifying and gathering patient images from these articles manually presents a laborious and time intensive task. Furthermore, matching a photograph of a patient with the corresponding diagnosis is made difficult by the variety of ways in which patient information is presented. Case studies involving multiple patients, or containing diagnostic information stored in tables, present a barrier to the deployment of automated natural language processing methods.

Modelling facial inheritance. For the opinion of a computer algorithm to be practically incorporated into the formulation of a medical diagnosis, it is crucial that an understanding of the decision evidence used by the algorithm can be plainly attributed (Adebayo et al., 2018;

Jamaludin et al., 2017; Lipton, 2016). Broadly, aspects of the morphology of a healthy human face can be attributed to several characteristics including: age; sex; ancestral background; environmental factors; and the specific facial characteristics inherited from ones parents. As described above, many of these factors have already been modelled with great success using deep learning computer vision methods. By modelling the relationship between the faces of parents and their children, it is hoped that the craniofacial features caused by a patient's syndrome can be more clearly delineated. In the future, this will enable clear, personalised visualisation of the evidence used by a facial syndrome classifier to suggest a diagnosis.

Bias in kinship data sets. As with all learning algorithms, the extent to which deep neural networks are able to model facial kinship depends crucially on the quality of the data sets used in training. Several varieties of kinship verification data sets have been produced over the last decade (Fang et al., 2010; Qin et al., 2015; Robinson et al., 2016). However, many of these data sets are simply too small for models tested on them to confidently claim generalisability. Furthermore, these data sets contain clear biases which make it easy to cheat the task of kinship verification visually. For example, if pairs of images are cropped from the same original family photo,or were taken in photographically identical environments, the task of determining kinship is reduced to simpler tasks such as matching background and lighting conditions. The small size and biases present in kinship data sets has likely led to the exaggeration of the accuracy with which current computer vision algorithms are able to model facial kinship.

1.2 Contributions and thesis outline

Syndrome face download pipeline (Chapter 3). The first contribution made in this thesis is the development of a novel download pipeline for finding and obtaining articles of interest from the PubMed database. The pipeline consists of over 40 modular components, and is capable of identifying and retrieving articles from multiple research publishers. In order to extract syndrome faces, we construct several individual tools to filter articles, figures and face images.

Biomedical figure classification (Chapter 4). We develop a convolutional neural network for the task of identifying biomedical figures which contain particular image types of interest. We deploy our classifier for the task of identifying biomedical figures containing facial photographs. Usually training such a classifier would require access to a large data set of relevant, labelled biomedical images. However, retrieving such images is the very task we wish to build the classifier to perform. To avoid this causation issue, our training method relies solely on the use of synthetically produced figures; and thus can be readily adapted to work with other categories of biomedical images.

Syndrome face annotation tool (Chapter 5). To label the large number of facial images obtained from our newly developed download pipeline, we develop a bespoke syndrome annotation tool. Our website–based tool includes several features designed specifically to aid the process of syndrome face annotation from genetic disorder articles. The annotation tool has already been successfully deployed, and continues to produce a steady stream of annotated facial images.

From same photo bias in kinship data sets (Chapter 7). In the second half of this thesis, we focus on research in the field of kinship verification. We investigate a common bias present in several kinship data sets; namely whether two facial images are cropped from the same original photograph. To benchmark this 'from same photo' signal, we train a novel CNN classifier to determine whether a pair of cropped facial images are extracted from the same base photograph. We show that despite our network having no concept of kinship, it is able to perform with comparable accuracy to many previously published kinship models.

The Geni kinship ancestry data set (Chapter 8). In response to the biases found in existing kinship data sets, we collect a new kinship data set using the ancestry website Geni.com. To achieve this, we mine a large number of ancestry profile pages for family information and facial images. We produce two kinship data sets: a pair–wise data set consisting of parent–child and sibling–sibling relationships; and a tri–subject mother–father–child (MFC) data set. The Geni

MFC triples data set is over 150 times larger than any previously published.

Processing the raw data obtained from Geni into a usable kinship data necessitated the implementation and development of several computer vision methods for the task of face detection, duplicate detection and usefulness classification. In addition, we take several steps to ensure common kinship data set biases are reduced in our data sets.

NetVLAD for tri–subject kinship verification (Chapter 9). Finally, we apply methods of feature aggregation to the task of mother–father–child kinship verification using our newly developed Geni data set. In particular, we are the first to implement a NetVLAD architecture for the task of kinship verification. Our preliminary results suggest that NetVLAD is able to perform with better accuracy when compared to a linear feature aggregation approach. Furthermore, by using the bias–reduced Geni MFC data set, we demonstrate a clearer representation of the difficulty of computational kinship verification.

1.3 Publications

The work presented in this thesis has thus far been developed into two peer–reviewed research papers. The work described in Chapter 4 was presented orally at ACM–BCB 2017 (Boston, USA) (Dawson et al., 2017). The work presented in Chapter 7 was accepted as a poster at ACCV 2018 (Perth, Australia) (Dawson et al., 2018). The remaining chapters are expected to form the underpinnings of several papers in the near future.

Part I

Mining syndrome faces from research articles

2 | Literature review – syndrome face classification

The development of the structure and shape of the human head has a clear and significant genetic component: children look more similar to their parents than to unrelated individuals, and monozygotic twins look more similar than dyzygotic twins (F. Liu et al., 2012). The face is one of the most variable features observable among the human population. Indeed, it has been shown that the correlations between facial traits are lesser than other related characteristics, possibly due in part to a selection pressure for faces to be easily recognisable (Sheehan & Nachman, 2014).

The process of craniofacial development is a highly complex cascade of orchestrated embryonic sub–processes (Twigg & Wilkie, 2015). As such, the specific links between an individual's genotype and their facial appearance are not yet fully understood (F. Liu et al., 2012). However, it is clear that the complexity of craniofacial development makes the process susceptible to genetically and environmentally caused malformations.

Over the last twenty years, there has been much work to discover the causative mutations associated with rare genetic disorders and craniofacial development. Linkage studies and genome wide association studies both attempt to identify relevant portions of the genome that occur more frequently in individuals who share a common phenotype. With the rise of next generation sequencing, it now possible to examine the entire genome of a large group of patients (Bush & Moore, 2012). Whole exome sequencing has already enjoyed significant success, being first used to find the genetic cause of several mendelian disorders, including Miller syndrome, Freeman–Sheldon syndrome and Barttman syndrome (Ng et al., 2010, 2009; Choi et al., 2009).

2.1 Computer aided diagnosis

Using computers to aid in medical diagnosis is by no means a new phenomenon. Mathematicians were discussing models and techniques which could be used to automatically diagnose patients in the late 1960s (Doi, 2007). By 1972, A research group based at the surgical unit of the Leeds General Infirmary had created a computer algorithm which it was claimed could diagnose patients presenting with acute abdominal pain with greater accuracy than their most senior clinician (De Dombal et al., 1972).

Originally, the focus of computer aided diagnosis (CAD) research was to create algorithms which might conceivably automate the diagnostic work of clinicians. Nearly 50 years ago, Croft (1972) predicted that 'General practitioners and medical clinics might someday have desk computer terminals where they can do patient record keeping, medical information retrieval, and specific disease diagnosis. In this last function a time–shared computer with diagnostic models could perform the work of an on–line consulting specialist'. Croft also listed what he saw as the three main challenges faced by the computer aided diagnosis community at the time, stating that 'the challenges are not computational', but instead 'a lack of standard medical definitions; a lack of large, reliable medical databases; and a lack of acceptance of computer aided diagnosis by the medical profession'. Interestingly, these problems are still among the most relevant and pressing challenges faced by the field.

Over the years, the end goal of integrating CAD into a clinical setting has necessitated a shift towards algorithms which aid and augment the ability of clinicians to diagnose, rather than replace them. This complementary approach has facilitated the gradual incorporation of CAD systems into many areas, such as detection of breast cancer through mammograms, the detection of lung cancer through CT scans, and the diagnosis of coronary artery disease (De Dombal et al., 1972; Firmino et al., 2014; Diamond et al., 1983). The most widely used CAD systems are able to synthesise and report non–obvious data and statistics to clinicians to provide tie–breaking information in ambiguous clinical scenarios (Moorman et al., 2016). In fact, it is now

often more conventional to evaluate a new computational diagnostic technique not by direct comparison with clinicians, but instead by assessing the performance increase of a clinician who is able to use the diagnostic tool to aid and improve their judgement (De Dombal et al., 1972).

2.2 Machine learning and deep learning for CAD

The sheer volume and diversity of diagnostic information potentially available to clinical experts has grown considerably over the last 50 years. Clinicians are now able to evaluate patient symptoms alongside information gained through tests from fields such as biochemical analysis and biomedical imaging (Amato et al., 2013). This has necessitated the use of statistical techniques which are able to analyse a diverse and complex range of medical data. This data driven approach has meant that machine learning is particularly suitable for aiding contemporary healthcare, as modern hospitals and clinics have the ability to gather, store and share vast quantities of multi–modal patient data (Kononenko, 2001).

The most common approach used to leverage machine learning in CAD has been to first explicitly define a list of medical tests deemed relevant to a certain disorder. The results of these tests for a particular patient can then be collated into a 'feature vector', which is evaluated alongside other such vectors to classify the patient into a diagnostic category. Shallow machine learning classification methods such as K–nearest neighbours algorithms and support vector machines (SVM) are ubiquitous throughout the CAD literature (Ramírez et al., 2013; Sluimer et al., 2003; Y. L. Huang et al., 2008; D. Y. Liu et al., 2012; Murphy et al., 2009).

Artificial neural networks have also been used to classify patient feature vectors (Amato et al., 2013). For example, Temurtas et al. (2009) trained a three-layer neural network using only eight features (such as including BMI, age, and the number of times patients had previously been pregnant) to diagnose Type 2 Diabetes in a female population with 82% accu-

racy (chance = 50%). Neural networks have also been used to aid in the diagnosis of Epilepsy (Nigam & Graupe, 2004). Patients are often diagnosed with Epilepsy on the basis of up to a weeks worth of scalp electroencephalogram (EEG) readings. Neural networks can be trained to identify epileptic seizures from this time series, and by highlighting these regions of importance, the time and effort required for a trained clinician to reach a diagnostic decision can be significantly reduced.

2.3 Discriminative facial features using deep learning

In the last few years, the field of computer vision has been revolutionised by the development of deep artificial neural networks, which can now be practically trained due to the availability of large image data sets (Russakovsky et al., 2015), the availability of high performance computing, and the development of sophisticated training algorithms (Simonyan & Zisserman, 2014). One of the most powerful techniques which has been developed in this field is the convolutional neural network (CNN) (LeCun et al., 1989). Convolutional neural networks exhibit the property of being able to combine local low–level image data into higher–level complex concepts (Mahendran & Vedaldi, 2016). The use of CNNs has led to a leap forward in the accuracy of object detection, localisation and classification.

Perhaps the clearest evidence that convolutional neural networks are able extract discriminative facial features is seen in the domain of facial recognition. Development in this field has been driven by the release of large facial recognition data sets. In 2014, DeepFace was introduced by researchers at Facebook (Taigman et al., 2014) . DeepFace used a nine–layer CNN trained with four million face images of over 4k people, and was able to achieve 97.4% accuracy on the 'labelled faces in the wild' (LFITW) data set (G. B. Huang et al., 2007). Soon after, Schroff et al. (2015) produced FaceNet, a CNN which was trained, using triplet loss, to map an individual's face into a Euclidean embedding. This was done in such a way that the similarity of faces could be measured by finding the distance between the related feature vectors in this Euclidean space.

FaceNet was able to achieve an even higher classification accuracy of 99.63% on LFITW.

In response to these efforts by private companies, Parkhi et al. (2015) released the VGGFace data set: a collection of over 2.6 million faces, each belonging to one of 2.6k celebrities, which they used to train a deep convolutional neural network to perform facial recognition. This method used far fewer images than FaceNet for training, but was still able to achieve 98.9% accuracy on the LFITW data set. Recently, Cao et al. (2018) released VGGFace2, which contains over 3.3 million images of over 9k public figures. It was shown that an SENet network trained on this data set achieved a top–5 classification accuracy of 95% on the 1845–identity IJB–B data set (Hu et al., 2018; Whitelam et al., 2017). It is clear from these performance statistics, as well as results in other tasks such as facial landmark detection (Chang et al., 2017; Teixeira et al., 2019), that convolutional neural networks are capable of extracting highly detailed representations of a human face.

2.4 Computational syndrome diagnosis

The presence of distinct craniofacial morphologies known to be associated with particular genetic disorders has naturally led research scientists to question whether it is possible to accurately diagnose these disorders using craniofacial shape and appearance alone. This approach will likely not replace genetic diagnosis, but could act as a screening tool for patients who require additional biochemical testing (Kosilek et al., 2015).

One of the first papers in the field was published by Boehringer et al. (2006), who analysed gray–scale facial photographs of 147 patients with one of ten genetic syndromes. By applying techniques of principal component analysis and linear discriminant analysis on the features extracted from participants' faces, they achieved a diagnostic classification accuracy of 76%, and were also able to develop a method of visualising which facial locations are most important for determining classifications.

Within the last ten years, several groups have further established the proof of concept for computational syndrome classification. Collecting images of patients in a controlled setting can be a difficult and laborious task. Therefore, due to the comparative ease with which they can be collected, the most prevalent approach used to date has been to employ ordinary facial photographs of patients. Researchers generally use computer vision software to first detect landmark features of the face (such as the edges of eyes, bottom of nose etc.). This information is then processed to extract specific facial geometry (Kosilek et al., 2015; Rai et al., 2015). Also, texture and appearance filters will often be applied locally to facial landmarks, and the resulting information then collated into a feature vector. When enough of these labelled feature vectors have been collected, they can be used to construct discriminative embeddings. The faces of new patients are then diagnosed according to the location of their feature vector in the trained embedding.

In the last five years, the number of papers published in this field has rapidly accelerated. Q. Zhao et al. (2014) used geometric and local texture features to diagnose Down syndrome in children under the age of three. By integrating the results of three separate shallow classifiers, they were able to achieve a diagnostic accuracy of 96.7% (chance = 50%) on a test set of 130 frontal facial images. Cerrolaza et al. (2016) presented a framework for detecting genetic disorders from 2D facial pictures. They used a support vector machine approach to classify 145 patients, 73 of which had one of 15 genetic syndromes, with an accuracy of 95%. The features used included relative geometric descriptions of facial landmarks, alongside landmark specific image filters. These image filters were optimised to maximise between–class difference and within–class similarity of the feature vectors belonging to patients with different disorders .

For several years, the state–of–the–art for diagnosing genetic disorders through computer vision techniques was formulated by Ferry et al. (2014). Their pipeline processes each image using facial recognition software to identify the locations of 36 specific facial features, such as the edges of eyes, or lips. These locations are then used to construct shape and appearance feature vectors, which undergo principal component analysis to reduce their dimensionality. These feature vectors then determine the location of a face in a feature vector space known as Clinical

Face Phenotype Space (CFPS). It was shown that in this CFPS, vectors corresponding to the faces of people with the same disease tended to cluster in a particular region. Therefore, once a sufficient number of images for each syndrome were encoded into CFPS, one could diagnose a new patient by taking an ordinary photograph of them, transforming the photo into a feature vector, and using a K–nearest neighbours algorithm to find which cluster the new point most likely belongs to.

The method used by Ferry et al. (2014), utilises 1363 images of patients who have been diagnosed with one of eight syndromes, alongside 1515 healthy controls, as the baseline points in CFPS. Their approach was first tested on faces of people diagnosed with one of the eight training syndromes, achieving a classification accuracy of 99.5% using a leave–one–out method. They also tested their embedding using the faces of people with 75 genetic disorders not in the original set of eight syndromes. The images of people with the same disease, again, clustered together with a much greater concentration than by random chance. Interestingly, this feature of the method naturally facilitates the clustering of patients with the same genetic disease, even if the disease is as yet unnamed or undiscovered. Most importantly for the future understanding of the relationship between phenotype and genotype, it was shown that the distance between disorder clusters in CFPS was positively correlated to protein–protein interaction distances for genetically characterised syndromes.

Recently, a group of researchers linked to the commercial Face2Gene organisation published the DeepGestalt network (Gurovich et al., 2018). Their method first uses landmark detection to extract multiple regional crops of a face. Each regional face image is then passed through separately trained CNNs, before classification scores are aggregated. Although not publicly released, the data set used to train the multi–class version of the network was the largest used to date: containing 26k images of patients with one of 216 syndromes. It was shown that on a held–out test set of 500 images, DeepGestalt was able to achieve a top–1 classification accuracy of 60%, and a top–10 classification accuracy of 91%.

2.5 Obtaining patient images

The approaches described in Section 2.4 illustrate several effective non–invasive tools for determining whether a person suffers from a particular genetic disorder. However, due to their machine learning nature, the power of such methods is heavily modulated by the need for large sets of representative training examples. It is very likely that the size of the training data set used by Ferry et al. (2014) and Gurovich et al. (2018) were an important factor in the comparative success of their approaches. Although there is no upper limit on the amount of training data that could be used with such algorithms, it has been suggested that a training set consisting of facial images from between 500 and 1000 subjects would be sufficient to provide practical use in a clinical setting as a screening tool (Kosilek et al., 2015).

For the most common genetic disorders, one approach which could be used to collect images of patients would be to use standard internet search engines such as Google or Bing. However, this approach becomes far less applicable for obtaining images relating to the rarest genetic syndromes. More importantly, it is often very difficult to confirm that the people depicted in these images have a particular disorder, ruling out their use as training examples.

A better source for obtaining photographs of patients is published scientific literature. Research groups often document individual cases, especially involving novel or rare disorders, by publishing research articles which contain patient photographs. This data comes with the added benefit of being peer reviewed and potentially containing phenotypic and genotypic information alongside photographs.

The most established resource for the curation of biomedical literature is the NCBI PubMed database. The size of the PubMed database has increased exponentially over the last 25 years to now include over 30 million citations. In last few years, NCBI has also introduced and developed the Entrez application program interface, to allow researchers to automatically query PubMed programmatically using a variety of computer programming languages (Sayers, 2019).

2.6 Extracting and classifying biomedical figures

Extracting images from articles is now easier than ever before with the development of open source software projects such as xPDF[1], and the related rendering library Poppler[2]. The Poppler library was recently used to develop PDFFigures 2.0: a software solution which is able to extract figures and tables alongside their respective captions from research articles, with a precision and recall of over 90% (Clark & Divvala, 2016).

Several research groups have also shown means to collect, categorise and store biomedical images in a way that is easily searchable (Demner-Fushman et al., 2012; S. Xu et al., 2008). These image databases frequently employ captions or textual information to allow the user to query–by–search–term (Koike & Takagi, 2009). However, these databases often only include a comparatively tiny subset of all the figures contained in biomedical articles.

Mining, processing and analysing the information stored in the text of biomedical articles is now a reasonably well–established research field. In contrast, the classification of biomedical figures still presents a great challenge. Although there has been huge success over the last ten years in classifying photographs in the wider computer vision community, the variability in scientific figure design, combined with the ubiquity of multi–panelled figures has severely hampered progress in the field (Rodriguez-Esteban & Iossifov, 2009). It has been estimated that around two–thirds of the figures found in the PubMed database are composite in nature, requiring dismantling before classification (Lee et al., 2017).

One of the most successful approaches to the problem of figure classification has been to focus on one particular kind of image. For example, Kuhn et al. (2014) used the PubMed database to mine information associated with gel electrophoresis images. Their pipeline used a modular design with specialised methods for 'figure extraction, segmentation, text recognition, gel detection, gel panel detection, named entity recognition and relation extraction'. This algorithm

[1] xPDF: www.xpdfreader.com
[2] Poppler: poppler.freedesktop.org

was then successfully deployed on a set of over 880k figures held in the PubMed database, from which over 86k gel panels were detected alongside associated information.

More recently, Giannakopoulos et al. (2015) attempted to automatically categorise biomedical figures into one of five sub types: charts, diagrams, geometric shapes, medical images, and photographs. The researchers used colour histograms, edge detection, line detection and a histogram of oriented gradients (HOG) to create feature vectors describing each image. These feature vectors were then classified using a deep belief network. When tested on a labelled data set of 1500 figures, their algorithm was able to achieve an average precision and recall across the five categories of around 76%.

3 | Mining facial photographs from genetic disorder articles

Over the last few years, there has been much research to develop computer vision solutions for the task of syndrome classification from ordinary photos of patient faces (Ferry et al., 2014; Gurovich et al., 2018). As discussed in Section 2, many of these efforts have used modern computer vision techniques, often based upon convolutional neural network (CNN) architectures. Although CNNs can be powerful image classification tools, they typically require training sets of many thousands, or even millions, of images depending on the difficulty of the task they are developed to solve. Classifying syndromes from ordinary patient photographs is a visually complex task, where multiple facial characteristics must be considered concurrently, whilst ignoring potentially irrelevant signals such as age, gender, ancestral background and photographic conditions.

Collecting sufficiently large data sets of patient syndrome faces is currently difficult for several reasons. Firstly, many genetic disorders are individually very rare: by definition occurring in less than one in 2000 births (European Parliament & the Council of the European Union, 2000). Therefore, being able to find patients with these disorders to photograph and subsequently share with the research community is naturally less common. Also, even if patients are photographed, there are stringent ethical considerations that may impede the distribution of potentially identifiable medical information, such as a patient face image alongside a syndrome diagnosis.

There are currently two methods that have been employed by researchers for collecting face images of patients with genetic disorders. The first, and most common, is to collect images directly from a clinical setting (Gorlin et al., 2001). A benefit of this approach is that photographs can be taken in a controlled manner, depicting the syndrome clearly. Also the accompanying syndrome diagnosis is likely to be accurate as it has been given by an expert clinician, and

often confirmed using genetic testing. Unfortunately collecting data in this way can be a slow process, relying heavily on the number and variety of patients treated by the clinic, and also the time available to a clinician to input patient data.

A second method which could potentially be used to collect patient faces is to retrieve images directly from internet search engine results. Although this may seem to be a fast and targeted method to obtain many images for particular syndromes, practically it is difficult to obtain corresponding diagnosis labels accurately. Less relevant search results often include photographs of people with different disorders from the search target, or no disorder at all. The lack of a standard diagnosis description attached to these images rules out the chance to obtain a high number of faces for targeted syndromes in this way.

A third option, mining the biomedical literature for facial images of patients with rare genetic disorders, is motivated by the observation that many of the results returned by search engines relating to genetic disorders are figures from academic papers. Clinicians and research groups will often document case studies of patients with rare genetic disorders as published articles. The diagnoses given to patients in these papers are likely to be accurate as they have checked by the expert authors, perhaps confirmed genetically, and undergone peer review. Additionally, in reported case studies of patients with rare or novel disorders, clinicians have already sought consent for publication, which makes further analyses of such published images uncontroversial from an ethical standpoint.

In this chapter, we develop a download pipeline of software to retrieve images of potential syndrome face patients from the biomedical literature. We mine the PubMed database, which contains nearly 30 million citations of life science articles. We use the PubMed interface to search for facial images of 100 genetic disorders and find over 20k patient face images from 4.8k research articles.

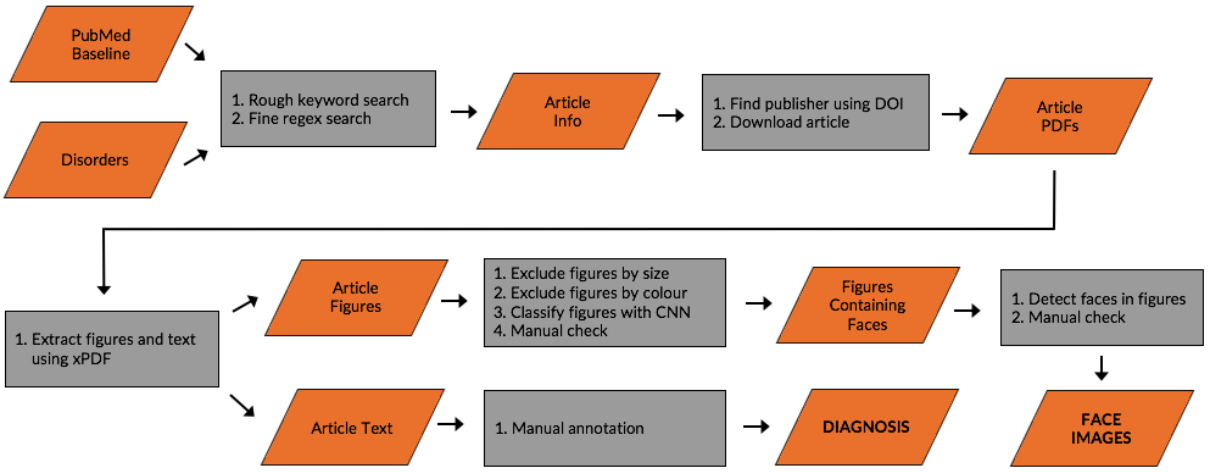


Figure 3.1: High–level overview of the download pipeline we develop to retrieve faces of patients with rare genetic disorders from research articles.

3.1 PubMed download pipeline

In this section, we describe our software pipeline for mining faces from genetic disorder articles. The pipeline we create consists of solutions to more than 40 individual tasks, divided into three main stages: finding potentially relevant articles; obtaining the articles locally; and then processing articles to obtain the facial images within. A high–level overview of the pipeline is shown in Figure 3.1.

3.1.1 Finding potentially relevant articles in PubMed

The first phase of the download pipeline involves obtaining the PubMed baseline, a snapshot of the PubMed database at the end of a given year. We then decide on a list of syndromes for which we wish to retrieve relevant articles. Finally, through an initial rough search and subsequent fine–grained search, we are able to identify articles within the baseline which may be relevant to the target syndromes.

Table 3.1: Size of the PubMed yearly baseline (2019). These files contain meta data for all of the academic citations held in the PubMed database at the end of a given year.

Data size	XML files	Lines	Citations
51 GB	972	5.5 Billion	30 Million

The PubMed baseline[1] consists of 51 Gigabytes of XML files containing information on the 30 million citations held in the PubMed database (see Table 3.1). Each XML file is divided into separate sections for each citation, with each section containing information such as the title of the work, year of publishing, author names, abstract etc.

We choose a list of 100 disorder names to target – the full list is shown in Table 3.2. These disorders were chosen for several reasons, including their relevance to previous works (Ferry et al., 2014). Others were chosen to increase the size of existing data sets of patient photos provided to us by other research groups. Many of the remainder of syndromes chosen are among the most common rare genetic disorders, increasing the likelihood of patient photos existing. Finally, we chose several disorders which, although rare, are known to cause a distinct facial phenotype, and so could potentially amenable to classification by computer vision methods.

We find articles which may relate to a particular genetic disorder using two searches. This is made necessary by the size of the PubMed database and the tendency for disorders often to be named using surnames. We first deploy a fast, keyword–match based search across the database and then a finer, slower regular expression based search. For the rough search, we use aliases belonging to each disorder, gathered using various resources including Orphanet, OMIM, NORD, and GARD[2]. In total, we use 445 aliases belonging to the 100 disorders, examples of those found are shown in Table 3.3. Upon searching the PubMed baseline we found over 4 million lines that contained a disorder and alias name.

A fine search is then performed on the lines found in the rough search. Some disorder names are specific enough that all occurrences of them in the PubMed baseline are found in articles

[1] www.nlm.nih.gov/databases/download/Pubmed_medline.html
[2] orpha.net, omim.org, rarediseases.org, rarediseases.info.nih.gov

relating to the corresponding disorder e.g. 'Rubinstein–Taybi'. However, many of the results returned by a keyword search often contain matches from articles that clearly do not relate to the target disorder. Examples of these false positives include matches with common surnames

Table 3.2: Names of genetic disorders we target for retrieval from the PubMed baseline.

17q21	Dubowitz	Myotonic dystrophy
22q11	Dyggve–Melchior–Clausen	Neurofibromatosis
3C	EEC	Noonan
Aarskog–Scott	Ehlers–Danlos	OAV dysplasia
Achondroplasia	Ellis–Van Creveld	Oculodentodigital
Acrofacial Dysostosis	Fetal Alcohol	Opitz–Kaveggia
Alagille	FG	Orofaciodigital
Albright	Floating–Harbor	Otopalatodigital
Angelman	Fragile X	PACS1
Apert	Freeman–Sheldon	Pfeiffer
Beckwith–Wiedemann	Frontometaphyseal	Phelan–Mcdermid
Bloom	Goldenhar	Pitt–Hopkins
Branchio–Oculo–Facial	Gorlin	Prader–Willi
Burn–Mckeown	Greig	Progeria
Cantu	Hajdu–Cheney	Proteus
Cardiofaciocutaneous	Hallermann–Streiff	Richieri–Costa–Pereira
Cartilage Hair Hypoplasia	Incontinentia pigmenti	Rieger
CHARGE	Johanson–Blizzard	Robinow
Cherubism	Kabuki	Rothmund–Thomson
Cleidocranial dysostosis	Kleefstra	Rubinstein–Taybi
Cockayne	Klippel–Feil	Russell–Silver
Coffin–Lowry	Klippel–Trenaunay	Saethre–Chotzen
Cohen	Koolen–de Vries	Schwartz–Jampel
Cornelia de Lange	Langer–Giedion	Seckel
Costello	Larsen	Sotos
Craniofrontonasal Dysplasia	Lenz–Majewski	SED
Craniometaphyseal Dysplasia	Mabry	Stickler
Cri du Chat	Mandibulofacial Dysostosis	Treacher–Collins
Crouzon	Marfan	Trichorhinophalangeal
Cutis Laxa	Marshall–Smith	Turner
Diastrophic dwarfism	Melnick–Needles	Van Den Ende–Gupta
Diastrophic dysplasia	Miller	Waardenburg
Donnai–Barrow	Moebius	Weaver
Down	Muenke	Williams

Table 3.3: Examples of disorder names and their corresponding aliases. When searching for relevant articles, we must consider that many disorders have multiple aliases by which they are referred to.

Disorder common name	**Example disorder alias**
Cutis Laxa	Chalazoderma
3C syndrome	Schinzel–Ritscher syndrome
FG syndrome	Opitz–Kaveggia syndrome
Cohen syndrome	Pepper syndrome
Charge syndrome	Hall–Hittner syndrome
Donnai–Barrow syndrome	Holmes–Schepens syndrome
Hallermann–Streiff syndrome	François syndrome
Richieri–Costa–Pereira syndrome	Pierre Robin sequence

such as ‘Cohen’, ‘Turner’ or ‘Miller’, as well as commonly used words and acronyms such as ‘Down’, ‘Charge’ or ‘FG’. In many cases, filtering these results can be achieved by confirming a word such as ‘syndrome’ or ‘disorder’ appears on the same line as the keyword. However, more complicated regular expressions are required to match disorders which can be spelt with special characters such as ‘Möbius syndrome’, or that can be spelt with several forms of hyphenation. The fine search narrows the 4 million lines found in the rough search down to 300k lines, which more likely belong to genetic disorder articles.

We then portion out the section of XML belonging to the article surrounding each fine search result line. This is done by searching forward and backward in the baseline for XML tags indicating the start and end of the article respectively. By extracting the XML belonging to a single article, further processing of the large PubMed files is avoided. From this process we identify 140k articles potentially relating to the 100 disorders. We show in Figure 3.2 the distribution of articles found by disorder. Some disorders names such as ‘Down’ and ‘Neurofibromatosis’ are very common keywords within the PubMed database, and so filtering out articles by textual methods alone is difficult. Many of these articles will be removed later in processing at a later stage as the figures found in the article will likely contain no faces.

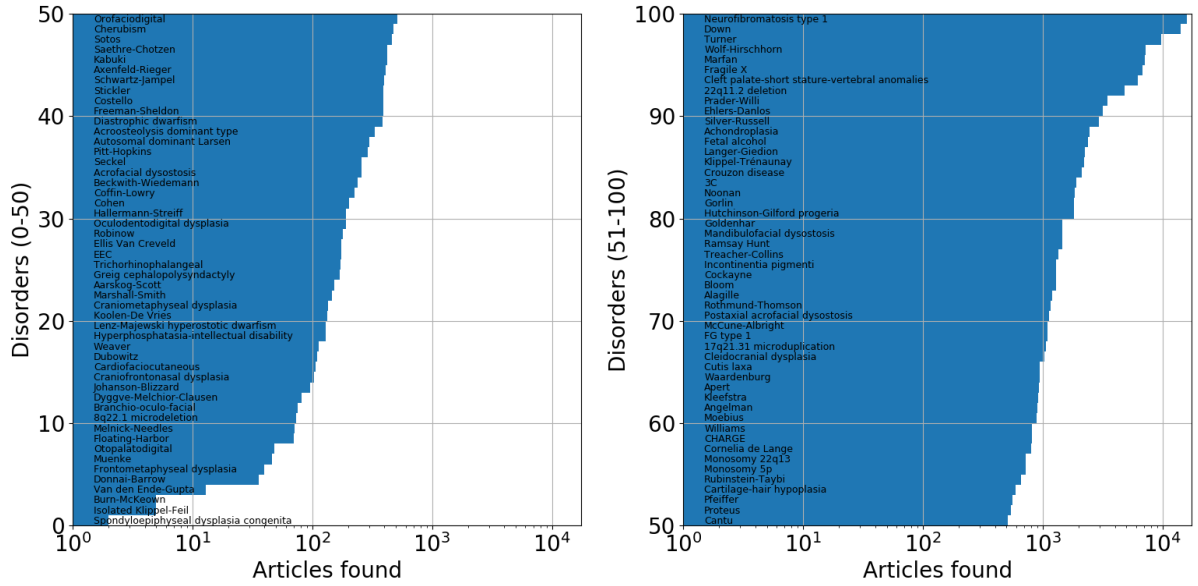


Figure 3.2: Distribution of the number of articles found in the PubMed baseline by disorder. This figure is best viewed on screen.

3.1.2 Obtaining research articles

In this section, we describe the steps required to retrieve the target articles identified previously. This process involves first extracting a Digital Object Identifier (DOI) from the meta data of the article. A DOI is a unique alphanumeric string that can be assigned to electronic resources, and acts as a persistent identifier, so that a resource can be found even if its hosted location changes. For each article, we use its respective DOI to determine the article's publisher. We then design bespoke web mining solutions for a number of publishers to retrieve the PDFs of target articles.

For each article we extract four pieces of information from the meta data stored in the PubMed baseline: The article title, the year the article was published, the PubMed ID assigned to the article, and the DOI of the article. We record the publisher of each article by programmatically resolving the articles DOI[3], and parsing the resulting response to find the publisher name. In practice, we have found that only 64k (46%) of the articles we targeted have a registered DOI in

[3] http://dx.doi.org/<doi>

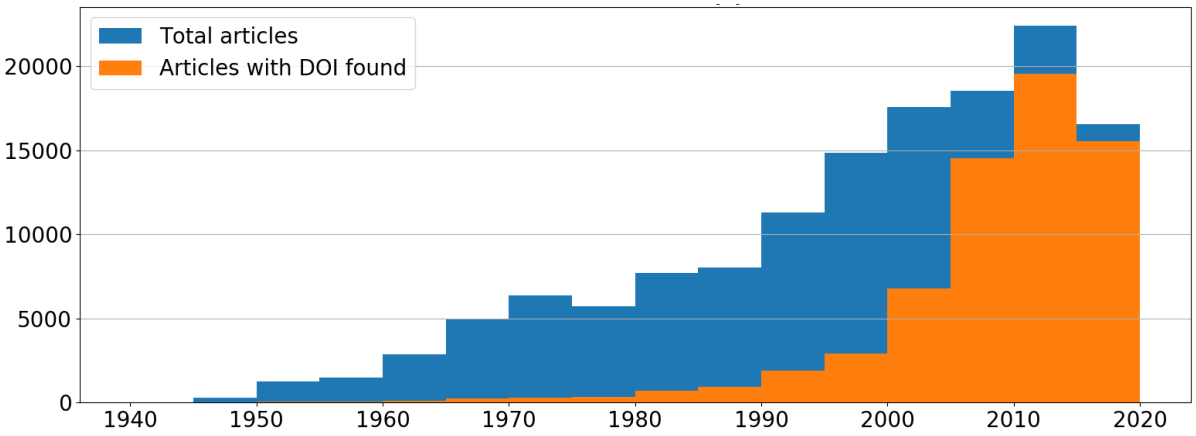


Figure 3.3: Total target articles published in a given year (blue), compared to the number of articles with a valid digital object identifier (orange). We are currently unable to programmatically retrieve articles without a valid DOI number. Note the percentage of target articles with a valid DOI is low before 2005.

the PubMed baseline. This is most commonly because an article is too old to have been assigned one, although some highly cited, older research articles have been assigned a DOI retroactively. In Figure 3.3, we show the number of articles we have found by the year in which they were published. We also show that the percentage of articles with a valid DOI drops noticeably for articles published before 2005.

Of the 64k articles we target, the majority are hosted by a small number of publishers. As shown in Figure 3.4, the top four most prolific publishers: Elsevier, Wiley, Springer/Nature and OVID collectively host around 40k (63%) of the articles found to have a DOI number. Each publishing website stores research articles according to their own distinct website layout. Furthermore, articles from the same publisher can be stored in a variety of ways internally as sites are updated partially over time. Developing software to navigate these websites can be a complex and time–intensive task. Thus far we have chosen to create tools to automatically download articles from Elsevier, Wiley, Springer/Nature and Ovid, as each subsequent publisher yields diminishing returns.

Broadly, we obtain research articles by first resolving the DOI number of the article to find its

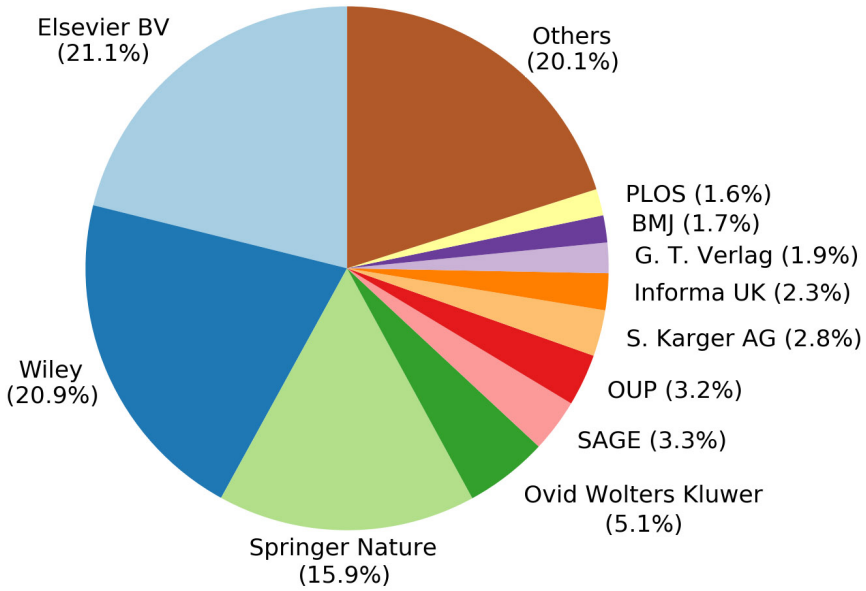


Figure 3.4: Percentage of target articles digitally hosted by publishers. We develop software to programmatically retrieve article data from the four most prolific publishers: Elsevier, Wiley, Springer and Ovid. This accounts for over 63% of all targeted disorder articles.

rough location on a publisher's website, then recursively finding and following links in the webpage HTML. In this way we find a route to the article PDF matching that taken by a typical web browser user. Downloading articles is performed programmatically using Python's Requests package[4], with care taken to respect the rate limit restrictions imposed by each publisher. Three of the four publishers allow users to automatically download full article PDFs. However, Elsevier uses an API which only permits the download of article text and figures. This is ultimately compatible with our objectives and pipeline, as we would wish to obtain this data at a later stage anyway.

[4] http://2.python-requests.org

Table 3.4: Effects of removing images from our pipeline using size and colour statistics on an annotated subset of downloaded data. We are able reduce the number of images we extract from articles by over 70% from further processing, whilst keeping images that do contain at least one face.

Processing step	Images with faces	Images without faces
Images found	223	16,286
Remove by size	223	10,258
Remove by colour	223	4,771

3.1.3 Processing research articles and figures

Due to some loss we are able to download 37k research articles downloaded as described in Subsection 3.1.2 to obtain the face images held within. This is performed in three steps. The first is to extract the images from articles. We then disclude images which clearly do not contain faces. The remaining figures are then passed through a CNN classifier to determine whether they contain faces. Finally, we use a face detector on the positively classified figures and extract the face images individually.

We extract images from articles using the xPDF software package[5]. In total, we extract 392k images from the 37k articles we have downloaded. As we wish to only find figures containing faces, we can discount many images from further processing using simple image size and colour statistics. Firstly, we exclude images that are too small, thin or narrow to contain a usable face image. Secondly, we exclude images that consist of too few distinct colours; or too many black or white pixels. The thresholds chosen for this classification step were found by annotating a group of over 16k figures by whether they contained a face. A threshold was selected which did not cause any images containing face images to be erroneously excluded from further processing. As shown in Table 3.4, We find that we can remove 47% of images at this stage just by using size statistics, and can remove a further 53% of images using colour statistics, without excluding any figures which contain faces.

[5] www.xpdfreader.com/



Figure 3.5: Examples of patient images obtained from research articles found using the download pipeline. There is a large variation in the syndromes and facial phenotypes we have retrieved.

The remaining figures are then passed through the figure classifier CNN described in Chapter 4. Each figure is assigned a classification score by the CNN indicating the classifier's confidence that the figure contains a face. This score is then used to rank figures, where they can be manually checked quickly. Of the 99k figures we run through the figure classifier, 7.5k were found to contain one or more images of a face. These figures are then passed through a face detection network which proposes bounding boxes for individual faces within a figure. These bounding boxes are then adjusted as required using the VGG image annotation tool (Dutta & Zisserman, 2019). Finally, faces are extracted from figures into individual images programmatically.

Table 3.5: Overview of data collection statistics at each stage of the download pipeline.

Stage	Articles	Figures	Faces
PubMed baseline	30M	-	-
Find article info	140k	-	-
Has valid DOI	64k	-	-
Hosted by top 4 publishers	40k	-	-
Article images retrieved	37k	392k	-
Irrelevant images removed	24k	99k	-
Find and extract faces	4.8k	7.5k	20k

3.2 Results of mining

As a result of mining, we identified over 140k research articles with potential relevance to one of 100 genetic disorders. Of these, we were able to obtain 37k. We extracted 392k images from these articles, of which 7.5k, belonging to 4.8k articles, were found to contain one or more faces. From these figures we obtained 20k face images. Examples of faces found using the download pipeline are shown in Figure 3.5.

As described in Section 3.1, the pipeline is able to distil potentially relevant data in several stages: preserving potential syndrome face images whilst discarding articles and figures which are irrelevant. An overview of the amount of data at each stage of the pipeline is shown in Table 3.5. We can see that the largest percentage filtering of data occurs very early in the pipeline, where we identify less than half a percent of the PubMed baseline for further processing.

We are only able to obtain images and text from 26% of articles that we target as potentially being relevant to our 100 target disorders. This is partly due to only implementing automated article download for the four largest publishers, but also because articles with valid DOIs are much less common for articles published before 2005, as shown in Figure 3.3.

Although we extract 392k images from 37k articles, we are immediately able to discount 75% of these images as not containing a face using simple image statistics. Of the remaining 24k figures, 7.5k were found to contain one or more faces by our figure classifier.

3.3 Summary

In this chapter we have built a software download pipeline for the extraction of facial images from published articles. We have shown how articles can be identified using keyword and regular expression searches on the PubMed yearly baseline; how PDFs of these articles can be obtained locally, and their figures processed to find face images. In total we have found over 20k face images from 4.8k research articles using the pipeline. In Chapter 5, we describe how these images can be paired with a corresponding patient diagnosis using the relevant article text, thereby making the images usable for training supervised syndrome classification models.

This method has several benefits compared with those previously used, including the ability to target specific disorders, and the accompanying high–quality diagnostic information describing each patient found in the article text. This method also has the additional benefit of retrieving some photographs of patients whose phenotype is uncommon for the disorder they have been diagnosed with. Patients with atypical facial phenotypes are often published to aid the medical communities understanding of the scope of a particular syndrome's manifestation.

Data is progressed through the pipeline largely through the execution of automated scripts, which each perform singular data processing functions and require little or zero user input. Running the pipeline in its current format requires manual user intervention in only a handful of stages. For example, the user must choose the rough search terms with which to search articles; and then manually build the regular expressions used to filter these rough searches at the fine search stage, as described in Section 3.1.1. Furthermore, manual checking is expected for both the figure classification and face detection stages of pipeline that are outlined in Section 3.1.3. The labour involved in these tasks is significantly reduced as figures can be viewed in ranked order using classification scores, and face detections can be easily adjusted using the VGG image annotation tool (Dutta & Zisserman, 2019).

Errors produced by the pipeline occur as missed oppurtunities to retrieve relevant syndrome

faces, and as the retrieval of faces without the target rare genetic disorder. The first type of error occurs mainly at the article–search stages of the pipeline, as it is difficult to benchmark whether a given set of keywords will retrieve all articles relating to a particular disorder. We expect that very few face images are lost during image and article processing, as figure classifications and face detections are manually checked and corrected. The second type of error manifests most commonly as the retrieval of patients with rare genetic disorders related to, but distinct from, the target syndrome. This causes little issue in this particular circumstance, as obtaining syndrome face photographs is generally valuable to our wider research goals. As will be discussed further in Section 5.2, around 6% of individuals with facial photographs extracted from the retrieved articles do not have a rare genetic disorder. Both of these cases are identified and handled using the manual annotation tool described in Chapter 5.

It is clear that the general process outlined in this chapter is readily adaptable for the retrieval of other categories of biomedical images from research articles held in the PubMed database. Such categories include: clinical photographs; X–ray images; ultrasound images; cell histology slides; CT scans etc. As computer vision methods are applied ever more to widely to biomedical image data, this pipeline provides a method of obtaining relevant image data sets, more quickly and cheaply than organising clinical data collection. To support this end, we have made the code which underpins the pipeline publicly available[6].

[6] https://www.github.com/Nellaker-group/pubmed_pipeline

4 | Classifying biomedical figures using deep learning

As described in Section 2, research in the sector of biomedical data mining has largely focused on the retrieval and extraction of information from the text of research articles or patient records (Gonzalez et al., 2015). However, figures and images contained in these articles can also constitute a valuable resource for biomedical scientists. These figures often provide a representation of a concept more clearly and entirely than a written description could. Although, to be utilised in the development and testing of medical imaging algorithms, images must often be explicitly labelled.

Retrieving relevant images from the great number of published biomedical figures is a task well suited to a convolutional neural network (CNN). This type of classifier would provide a fast and accurate method of determining amongst a collection of biomedical figures, which ones are most likely to contain a particular object of interest e.g. cell cultures, protein structures, X–ray images.

However, training a deep CNN to robustly classify complex images typically requires the use of huge sets of labelled training data (Krizhevsky et al., 2012). This creates a causality dilemma. To train a CNN, one must first possess a large data set of biomedical images labelled by whether they belong to the target image class or not. However, if it were already possible to accurately and rapidly label images in this way, there would be no need to build a classifier originally. Furthermore, collecting a sufficient number of relevant figures from the biomedical literature to effectively train a CNN is currently impractical.

In this chapter, we show that this problem can be resolved by training a classifier using *only* synthetic images, designed to mimic common biomedical figures. Our approach is inspired by

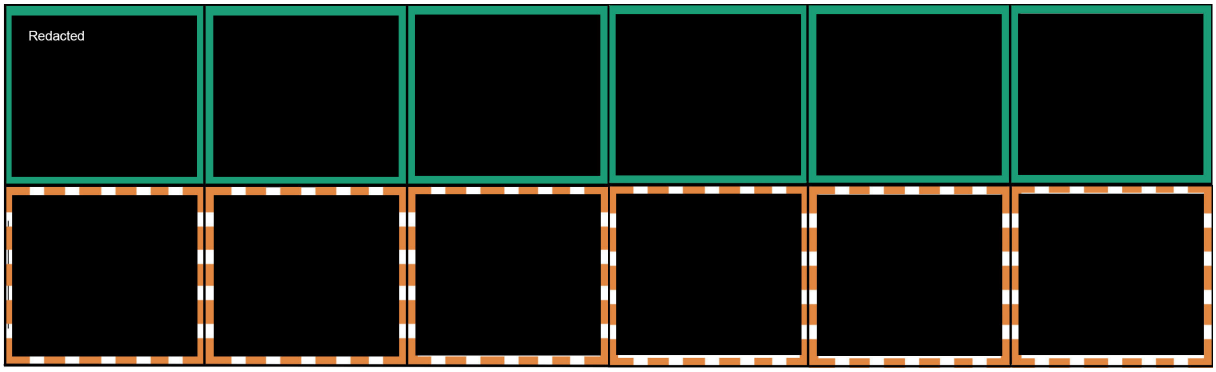


Figure 4.1: Examples of figures, containing faces (green solid borders) and not containing faces (orange dashed borders), taken from research articles held in the PubMed database. Many of the figures are composite in nature: they combine images of faces alongside other body parts, X–ray imaging, molecular structures, graphs and charts etc.

research in the fields of text localisation (Gupta et al., 2016; Jaderberg et al., 2014) and object detection in natural images (Rozantsev et al., 2015), which have shown that synthetic images may be used for training purposes in lieu of domain specific data.

We demonstrate this approach by training a classifier to label whether figures contain images of human faces[1]. We show that although the classifier is trained on synthetic images, it is capable of adapting to the domain of real images taken from biomedical articles.

In testing, the classifier is able to determine whether an image contains a face with an average precision of 94.8% on a data set of over 31k images taken from a subset of the real biomedical articles obtained in Chapter 3. We also present a case study using a subset of our retrieved images to benchmark the effectiveness of the classifier in helping to mine images of patients with rare genetic disorders.

The approach developed in this chapter is particularly suitable for cases where a specific category of image, that may exhibit some intra–class variation, needs to be retrieved, and sufficiently many examples from the image category can be obtained either through internet search engines

[1]We have found that standard face–detection algorithms are unsuitable for the task of retrieving patient faces from biomedical figures, as they tend to miss too many patients with severe craniofacial malformations, as well as producing many false–positive results when applied to common biomedical figures, such as cellular images, X–rays, CT scans etc.

or other means. We have demonstrated the approach for the case of faces, but the method could be readily adapted to retrieve other categories of biomedical images, for example: electrophoresis gels, hands, mice, protein structures, cell cultures, human tissues, X–ray images, etc.

4.1 Training a classifier using synthetic data

This section describes our method for training a CNN classifier to be able to label biomedical figures by whether they contain an image of a face. Our approach is to train the classifier using only synthetic figures, fabricated from images taken from an internet search engine, which resemble figures found in the biomedical literature.

The classifier is trained in two stages, using a bootstrap approach. In the first stage, an *initial* classifier is trained using synthetic images, and its performance assessed on a synthetic evaluation set. The misclassified images from this evaluation set are then analysed to identify visual themes which reoccur amongst the failure cases of the initial classifier. A second, larger, synthetic data set is then generated that specifically includes more examples of images exhibiting these themes. In the second stage, this larger set is used to retrain the classifier. The success of this bootstrapping approach is then measured by evaluating the initial and final classifier on a synthetic test set.

4.1.1 Generating synthetic data

Figures found in biomedical articles can often be divided into a small number of themes (photographs, tables, X–ray imaging, graphs etc.). Furthermore, to adhere to journals' editorial guidelines, researchers will frequently concatenate individual images, as shown in Figure 4.1, to form compound grid–like figures. Therefore, to ensure our classifier is able to adapt to the domain of real biomedical images, we train it on synthetic grid–like images composed of sub–images which are representative of the categories often found in biomedical literature.

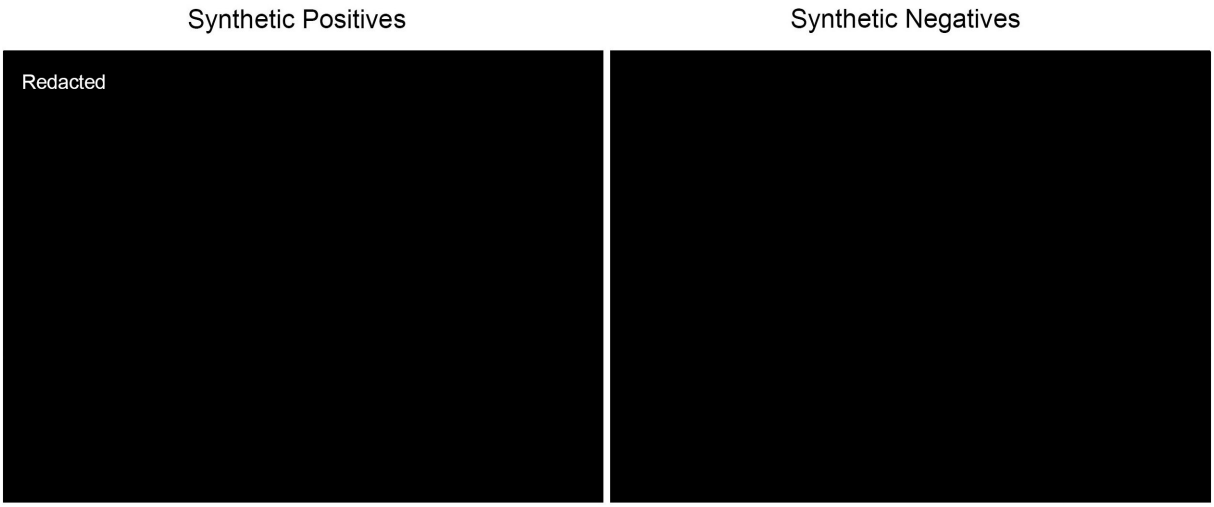


Figure 4.2: Examples of the synthetic grid figures produced using images downloaded from the Bing search engine. These images were used in training the initial classifier. Examples of positive and negative class training images are shown on the left and right respectively.

To generate this synthetic data, we first compiled a list of search queries which were used to download images from the Bing search engine[2]. To collect facial photographs, 37 search terms such as 'face', 'adult face', and 'child face' were used. To collect non–facial images, representative of figures found in biomedical papers, we used 56 terms such as 'line–graph', 'protein structure', and 'brain scan'. Images were downloaded and categorised by whether they contained a face or not. This produced a data set of 'positive' and 'negative' labelled images, which were further separated into training, validation and evaluation partitions. The images were then pasted onto white backgrounds in 18 different tiled configurations, chosen to mimic the most common sub–panel arrangements found in a sample of figures taken from real biomedical research articles. Examples are shown in Figure 4.2.

In total, 9.6k images were downloaded, of which 1.8k contained at least one face. These images were used to produce the initial data set of synthetic grid images, as well as the initial synthetic evaluation set of 19.5k images. The statistics of the training and validation split of the initial data set are given in Table 4.1. We ensure the synthetic data sets are balanced, with roughly half

[2] www.bing.com/images

Table 4.1: Summary of the number of images in the initial and final synthetic data sets, and the synthetic and real test sets.

Data set	**Train**		**Val**		**Test**	
	Pos	Neg	Pos	Neg	Pos	Neg
Initial synthetic	28865	28865	9597	9597	24884	24884
Final synthetic	74165	74165	24368	24368		
Real	-	-	-	-	725	30731

of synthetic figures containing at least one image of a face. This helps to prevent the classifier from learning a bias towards the larger class of images.

4.1.2 Training and evaluating the initial classifier

The initial classifier was trained using the synthetic data set generated as described in Section 4.1.1. The CNN implemented in this chapter is based upon the VGG–M architecture (Chatfield et al., 2014) and is constructed using MatConvNet, a Matlab–based framework for implementing CNNs (Vedaldi & Lenc, 2015). The classifier used was pre–trained on the ILSVRC 2012 data set to classify images into one of 1,000 object categories, as part of the ImageNet challenge (Russakovsky et al., 2015). Pre–training involves using a CNN initialised with a set of weights which have been found to be useful in another image classification task (Sharif Razavian et al., 2014). These weights can then be fine–tuned to tackle the new task, typically requiring much less training data than if the parameters were initialised randomly. Additionally, we made a simple architectural change to the CNN, reducing the final fully–connected layer of the network from 1000 nodes (corresponding to the object categories in the ImageNet challenge) to 2 nodes (corresponding to 'does not contain face' or 'does contain face'). The pre–trained VGG–M was fine–tuned on the synthetic training data using stochastic gradient descent to minimise a cross entropy loss objective function.

We evaluated the initial classifier on an initial evaluation set of synthetic images to direct the training of an improved classifier. For each image, I, the CNN outputs a score, $S(I)$, where

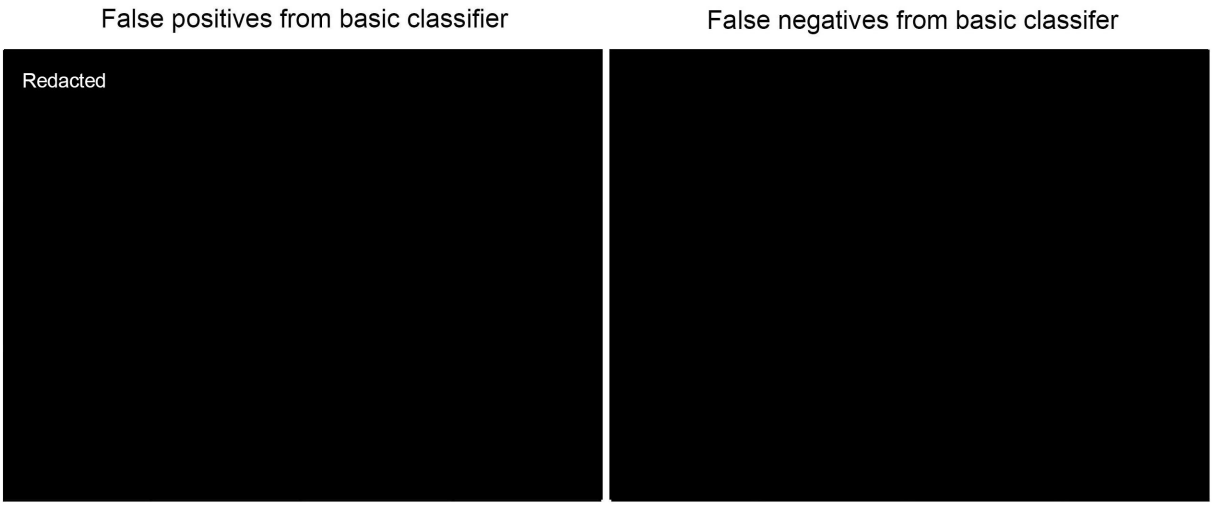


Figure 4.3: Examples of synthetic grid images which were misclassified by the initial classifier. Common themes amongst the false negative images include: black and white photographs; faces in profile; small images of faces; and people with irregular poses or facial expressions. Common themes amongst the false positives include: X–ray images; protein and crystal structures; and images of bacteria and cells. This image is best viewed on screen.

$0 < S(I) < 1$, corresponding to the classifier's confidence that an image contains a face. The performance of the initial classifier was measured by ranking every image in the evaluation set by its respective score. Observing images in this way allowed the identification of themes which were present amongst those images that were classified incorrectly by the initial classifier. Examples of the highest ranked false positives and lowest ranked false negatives are shown in Figure 4.3.

4.1.3 Creating the final synthetic data set and synthetic test set

Identifying common themes amongst the images which were missed, or misclassified, by the initial classifier, guided the creation of a second data set (examples of themes are given in the caption of Figure 4.3). This final synthetic data set, used to retrain the CNN classifier, is over 2.5 times larger than the initial.

We created the final synthetic data set by expanding the set of search terms from the original set

of 93 used to download images from Bing. We added 46 new search phrases relating to faces (e.g. 'greyscale faces', 'side profile of face'), and 26 associated to other biomedical entities (e.g. 'X–ray imaging', 'cells under microscope'). This resulted in an image pool of 37,801 images, of which 23,268 contained at least one face. The final synthetic training data set and synthetic test set (see Table 4.1) were then generated using these images, through the method described in Section 4.1.1.

4.1.4 Training and evaluating the final classifier

We use the final synthetic data set to retrain the CNN, employing the same regime as described for the initial classifier in Section 4.1.2. The performance of the initial and final classifiers is compared on the synthetic test set using receiver operating characteristic (ROC) and precision–recall (PR) curves. It is depicted in Figure 4.4 that, although both classifiers perform well on this test set, the final classifier achieves considerably better results than the initial. The initial and final classifiers achieve an average precision of 96.58% and 98.48%, and an equal error rate of 9.72% and 5.73% respectively. This indicates a marked reduction in the classification error.

In these experiments we have fine–tuned all layers of the network. To measure the effectiveness of this approach, we repeated the above experiments, fine–tuning only the last layer of the pre–trained network. We found the difference in results to be negligible for the initial classifier, and notably worse for the final classifier, indicating that training all layers of the network is a more effective approach, given a suitably large training data set.

4.2 Testing the classifier on real biomedical figures

In this section, we test the final classifier on a subset of real figures we have collected from biomedical articles in Chapter 3. Up to this point, the CNN classifiers have been tested on synthetically generated figures, and it has been shown that the final classifier is able to outperform

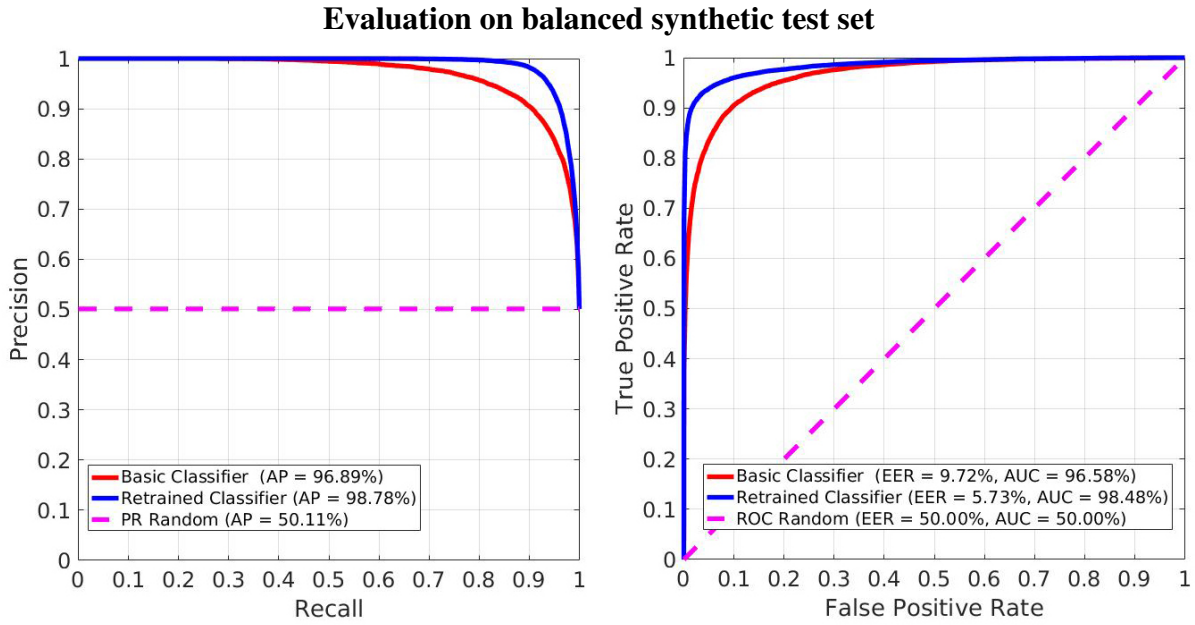


Figure 4.4: PR and ROC curves displaying the performance of the initial and final classifiers on a balanced test set of 49.8k synthetic grid images. The final classifier outperforms the initial classifier on the test statistics of average precision, equal error rate, and area under ROC curve.

the initial classifier on these images. However, the true test of the classifier is whether it is able to accurately determine whether figures taken from *real* biomedical articles contain an image of a face.

The data set we have annotated for testing the classifier consists of 31k figures from research papers held in the PubMed database. These images were gathered from 3.8k distinct articles, each of which relates to one of 72 biomedical search terms. The figures were extracted using open source software provided by xPDF[3], and display a wide variation in lighting, pose, occlusion, facial morphology and size. We further ensure that the images used to generate our synthetic data are disjoint from these real figures. Examples of the images extracted from biomedical articles are shown in Figure 4.1.

Aside from being comprised of real images, this data set also differs from the synthetic training set by being heavily unbalanced. Although balanced data sets were used in training to avoid

[3]xPDF: www.xpdfreader.com/

Table 4.2: Results of testing the initial and final classifiers on the synthetic and real test sets. These classifiers were trained on the initial and final synthetic data sets respectively. The average precision (AP) and area under receiver operating characteristic curve (ROC AUC) are reported. We see that the final classifier is able to achieve much better performance both when applied to synthetic and, in particular, real biomedical figures.

Data set	Classifier AP (%)		Classifier ROC AUC (%)	
	Initial	Final	Initial	Final
Synthetic test	96.89	98.78	96.58	98.48
Real	59.26	94.80	91.10	99.66

introducing biases into the classifier, only 725 (2.3%) of the figures in the real test set contain one or more photographs of a face. This is representative of the actual ratio of face images obtained by mining, and so provides a realistic test of the classifier's performance on unbalanced data.

4.2.1 Results on real biomedical figures

It can be seen in Table 4.2, that the final classifier achieves a high average precision of 94.8% on the real unbalanced test set. This level of performance is sufficient to ensure the classifier would be of pragmatic use in determining amongst a set of biomedical figures, which contain faces.

As described in Section 4.1.2, each image, I, is assigned a score, $S(I)$, by the classifier, representing the classifier's confidence that the figure contains a face. In practice, once the images in a data set have been classified, they can be ranked in descending order according to the value of $S(I)$. The highest ranked images often include a very high proportion of correctly classified images which contain faces.

The CNN can also be used as a simple binary classifier to determine, amongst a batch of images, which contain a face. The PR curve shown in Figure 4.5 can be used to determine a decision threshold for binary classification, based on the precision and recall requirements of a user. A decision threshold, T, is a fixed constant, such that images with $S(I) \geq T$ are classified as

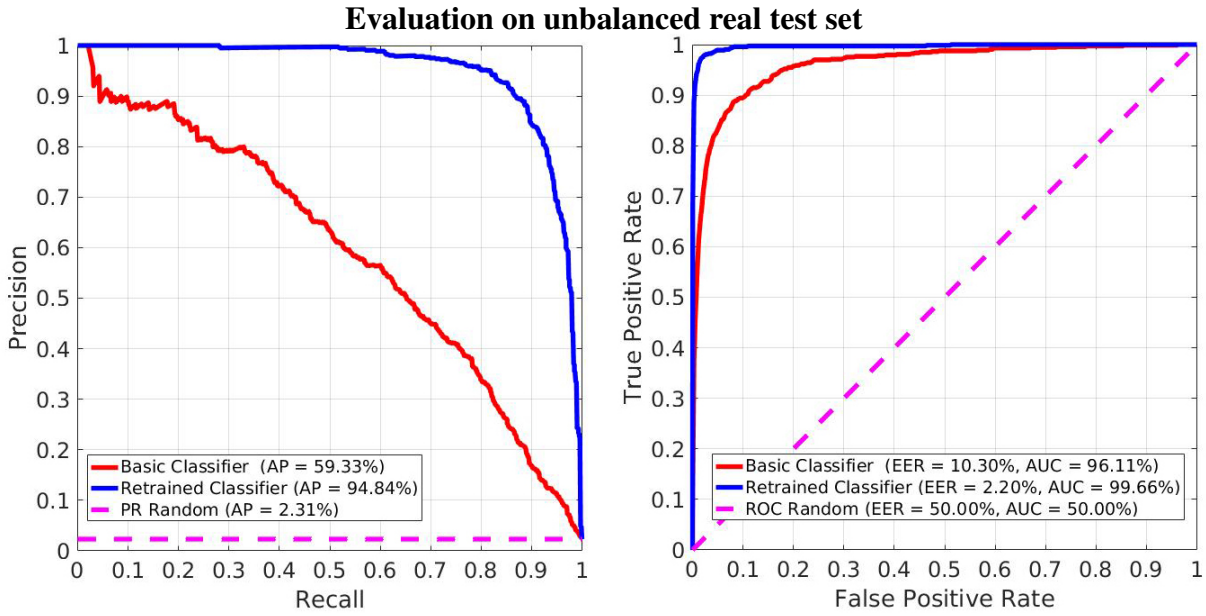


Figure 4.5: PR and ROC curves displaying the performance of the initial and final classifiers on 31456 real images (725 containing one or more faces) extracted from biomedical research articles. The final classifier appreciably outperforms the initial classifier on the test statistics of average precision, equal error rate, and area under ROC curve.

containing faces. If the decision threshold is set at $T \approx 0.63$, the classifier achieves a precision and recall of 69% and 95% respectively, meaning that 95% of all images containing a face are retrieved, and 69% of retrieved images are true positives. Alternatively, if a higher decision threshold of $T \approx 0.91$ is chosen, the precision and recall of the classifier are both approximately equal to 89%.

Figure 4.5 also highlights the effectiveness and necessity of the bootstrapping approach used to create the final classifier. In contrast to the performance of the final classifier, the initial classifier achieves an average precision of only 59.3% when tested on real figures. One reason for this is that the real data set is heavily unbalanced towards negative examples, so the rate at which false positive errors are committed by the classifiers is amplified. Retraining the CNN with a much larger data set, and including more image themes which were originally missed, or misclassified, appears to have helped make the classifier more robust against false positive errors.

4.3 Case study – rare genetic disorders

We test the efficacy of the classifier by retrieving images of patients with rare genetic disorders from published biomedical articles. In this section, we benchmark the capability of the classifier to aid the collection of images of patients with one of ten rare genetic disorders; chosen for their relevance to work by Ferry et al. (2014). The articles are collected from several journals known to publish research articles in the field of rare disorders. A full list of the disorders is given in Table 4.3.

4.3.1 Obtaining faces of patients with rare genetic disorders

We began by obtaining a list of candidate articles using the PubMed E–utilities API [4]. This resulted in a list of 1.1k articles, each potentially containing images of patients, for which the corresponding PDF document was obtained.

Next, we extracted the images from each of the candidate articles, and prepared them for analysis by the CNN classifier. Aside from the legitimate numbered figures found within papers, the image extraction software often retrieved small regions of white space, which had been used as padding by authors formatting their work. Other small images commonly found included copyright stamps, and the logos of journals and institutions. These small images comprised a non–trivial proportion of the images initially retrieved, particularly from older papers. Given that numbered figures used in research papers need to be large enough to convey meaningful information, images with a height or width of under 50 pixels were immediately discarded from further analysis. This resulted in a data set of 3.1k images, each relating to one of ten search terms.

[4]Entrez E- utilities: www.ncbi.nlm.nih.gov/books/NBK25497/

Table 4.3: Summary of classifier results on figures taken from a subset of articles relating to ten rare genetic disorders. We report the number of images found for each disorder; how many images actually contained faces; how many images the classifier predicted contained faces; and how many face images the classifier missed. The classifier greatly reduces the search space for face images with almost no missed positives.

Search term	**Images**	**GT faces**	**Classified +ves**	**Missed +ves**
Angelman	615	43	54	1
Apert	68	5	8	0
Cutis Laxa	210	26	40	1
Fragile X	854	20	24	0
Kabuki	61	13	14	0
Noonan	450	32	46	0
Prader–Willi	275	27	34	0
Rubinstein–Taybi	87	16	16	0
Treacher Collins	223	8	10	0
Williams–Beuren	281	13	20	0
Total	**3124**	**203**	**266**	**2**

4.3.2 Results on case study images

As shown in Table 4.3, of the $3,124$ images found, 203 (6.5%) actually contained a face. The final classifier predicted that 266 of the $3,124$ images contained a face, missing only two positive examples. In this experiment, the decision threshold of the final classifier was set using the equal error rate found during the real image testing described in Section 4.2. Under this regime, the classifier achieves an overall precision of 76% $(201/266)$, and an overall recall of 99% $(201/203)$. Most importantly, the classifier was able to reduce the search space of potentially relevant figures by over 91% from $3,124$ to 266. Examples of some of the face images correctly retrieved by the classifier are shown in Figure 4.6.

A small minority of the figures retrieved by the classifier, whilst containing faces, did not contain faces of patients with the particular disorder of interest. In several examples, images of healthy relatives of individuals with a disorder were retrieved. These images were included in case reports to contrast the facial morphology displayed by the patient. In other examples,

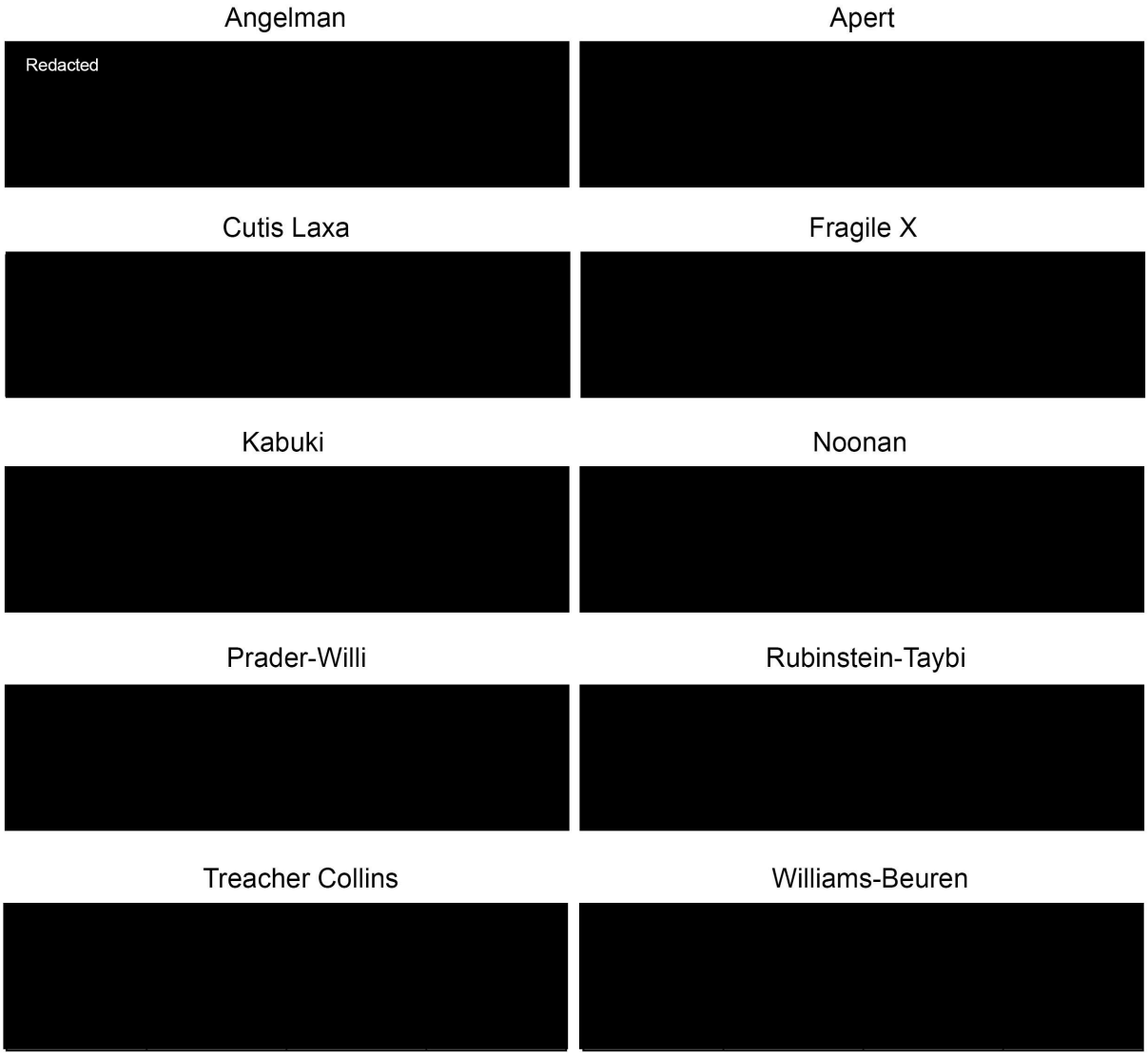


Figure 4.6: Examples of face images of patients from genetic disorder articles found using the biomedical figure classifier we have developed.

articles discussed multiple syndromes, or explicitly dismissed the target disorder as a potential diagnosis. Despite these instances, the performance of the classifier in this case study provides proof of concept that this approach can be used to aid the mining of images of patients with rare genetic disorders. Furthermore, such a pipeline could be easily scaled to accrue more images by targeting articles from a wider array of journals, relating to a greater variety of rare disorders.

4.4 Summary

In this chapter we have demonstrated that a deep convolutional neural network can be trained using only synthetic images to classify whether figures extracted from biomedical research articles contain faces. Our implementation consists of a two–stage bootstrapping approach. First, an initial classifier is trained using synthetic images designed to mimic real biomedical figures. Then, by identifying themes in the images missed by the initial classifier, a larger data set of synthetic images, which includes more of these hard examples, is generated and used to retrain the classifier. This strategy led to considerable improvement in classification precision and recall when tested on synthetic images, and real images taken from biomedical articles. It was also shown that this approach is able to assist in the mining and labelling of biomedical figures to identify potentially relevant patient photographs for rare disease phenotyping research.

5 | Annotating syndrome faces

In Chapter 3, we described a download pipeline for obtaining facial images from research articles relating to genetic disorders. At the time of writing, this pipeline has extracted over 20k face photographs from 4.8k biomedical articles. However, to be utilised for the task of syndrome classification using supervised computer vision techniques, each face image must be annotated with the disorder of the patient shown.

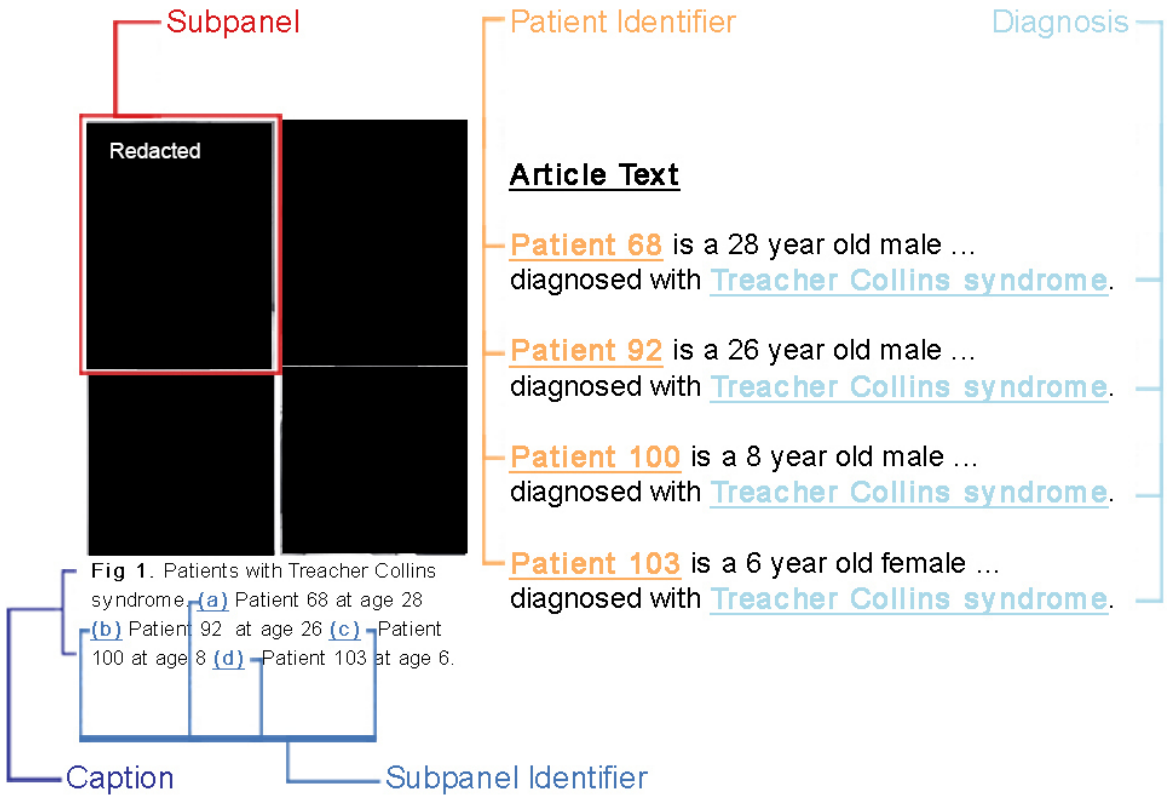
This diagnostic information can be consistently found in the text of the article from which the face was originally extracted. Within a case study article, each patient is referred to using a de–identified handle such as *Patient 3*, *the proband*, *P-IV* etc. This identifier is then used to introduce the phenotype, genotype and diagnosis of the patient throughout the article text. Each facial image of a patient is then linked to the corresponding patient identifier through a figure caption.

However, to retrieve this information precisely requires the solution of several entity matching problems, as shown in Figure 5.1. Broadly, to match a patient identifier with the relevant face images would require: understanding how patients are labelled in the article text; finding the captions in the article which refer to the patient; matching the caption to the corresponding figure; and finally using the caption and figure to determine the specific subpanels in which the patient is shown.

Automating this identifier–image matching is a complex task due to the great variability in how different articles are structured and presented. Furthermore, employing multiple NLP and computer vision methods to solve this problem would likely be hindered as errors in one method would be prone to propagate throughout the overall solution.

Due to these difficulties, we instead choose to develop a tool to facilitate the human annotation of syndrome faces from research articles. We use a website for this task for many benefits

Figure 5.1: Illustration of the several matching problems involved in labelling face images from research articles. Each face image must be matched with the subpanel identifier within a caption to a patient identifier, and the patient identifier to a diagnosis. Solving these problems accurately using automated methods presents a great challenge.



including: the quality of the structured data produced; the availability of website development tools, and the option for annotators to work remotely. Moreover, this method also provides data which can be used to train automated methods to extract relevant structured information from research articles in the future.

In the rest of this chapter, we describe the components and layout characteristics of the annotation tool which enable annotators to label syndrome faces quickly. We describe solutions to several challenges such as collating patient information from multiple annotation sources, and monitoring annotator progress. Finally, we describe the output of the annotation tool (and ultimately download pipeline), a collection of labelled syndrome faces which far exceeds the size and diversity of several previously published data sets.

5.1 Annotation tasks and tool design

In this section we describe the design of an annotation tool for the rapid ground truthing of syndrome face images from research articles. We split each annotation into three distinct tasks of retrieving caption, patient and disorder information. Alongside the direct annotation of each face image, we also provide tools to annotate several other important article features which will be used in future efforts to automate the process of information extraction from article text.

5.1.1 Three annotation subtasks: caption, patient, disorder

The first annotation subtask is to identify the caption of the figure from which the face image was originally extracted (see Figure 5.2). Identifying figure captions manually is necessary as even state–of–the–art PDF parsers cannot perform this task with complete accuracy (Clark & Divvala, 2016), particularly for older PDFs which often have complex internal file structures.

The second subtask is to identify information relating to the patient (see Figure 5.2). Annotators are asked to note how the patient shown in the displayed photograph is referred to in the article text, as well as report information such as: the sex of the patient; whether the patient has other family members described in the article; whether the patients eyes are blocked in the image shown; and whether the patient has information recorded in a table. The information we annotate at this stage is important for several reasons. Determining the sex of patients may in the future facilitate the training of syndrome classifiers which are able to explicitly incorporate this information in their decision making. Recording whether other family members are shown in an article will, in the future, allow us to build a data set of patient faces and unaffected family members, potentially leading to an understanding of how syndrome manifestations are affected by ancestral background. We annotate whether a patient's eyes are blocked, as this occlusion is likely to affect the performance of computer vision methods for syndrome face classification, and so affords us the option of excluding faces with blocked out eyes from training sets. This is

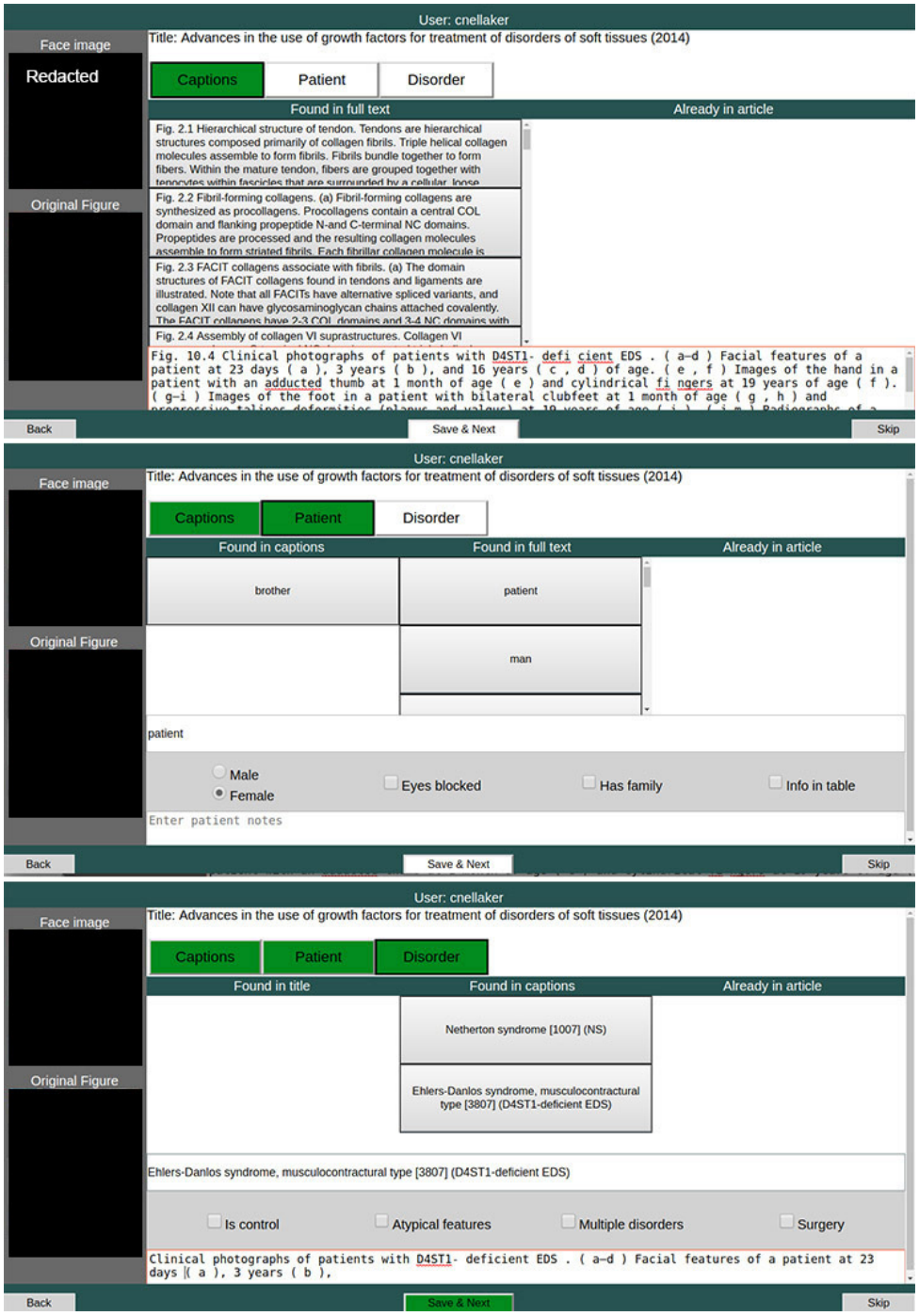


Figure 5.2: The annotation tool we have designed to aid the rapid labelling of patient face images from rare genetic disorder articles. The layout consists of separate tabs for annotating (a) caption, (b) patient and (c) disorder information. Each tab changes to green when the section has enough information. When all three tabs are complete, the save button becomes active.

precautionary as, if an image of a patient's face has been published even with the eyes blocked out, it is likely that the important phenotypic information is still present in the rest of the face.

The final annotation subtask is to record the syndrome diagnosis of the person shown in the face image, and to identify the piece of text from the article which confirms that diagnosis (see Figure 5.2). Non syndromic individuals are marked by annotators as controls, so that they may be excluded from further processing. Additionally, we annotate whether an individual shown has atypical features for the disorder they have been diagnosed with; if a patient has multiple suspected disorders; and whether the patient has had facial surgery. This information affords us the option of excluding particular types of facial images from use in training syndrome classification models through supervised learning. Note that we choose not to annotate any genetic mutation information, despite the information often being present in disorder case studies. Manually retrieving this information was found to noticeably slow down the process of human annotation, as its complexity required annotators to have reasonably advanced genetic expertise.

5.1.2 Annotation tool characteristics

Database backend. The annotation tool is chiefly a graphical user interface designed to allow the accurate labelling of syndrome faces as quickly as possible. The information required and collected by annotators is both stored and distributed using a relational database backend. A description of this database structure is shown in Figure 5.3. The database holds all preprocessed information relating to a syndrome face and the article from which it was obtained. Using this database structure affords us several benefits including: being able to move articles, figures and face images in and out of circulation; limiting the workload of an individual annotator; and forcing multiple annotations of the same facial image. This final property is necessary as a small proportion of syndrome faces are particularly difficult to annotate. This is typically because information is missing from an article's text or is presented in a non–standard manner. In these cases, re–annotation by multiple users can allow us to reach a consensus.

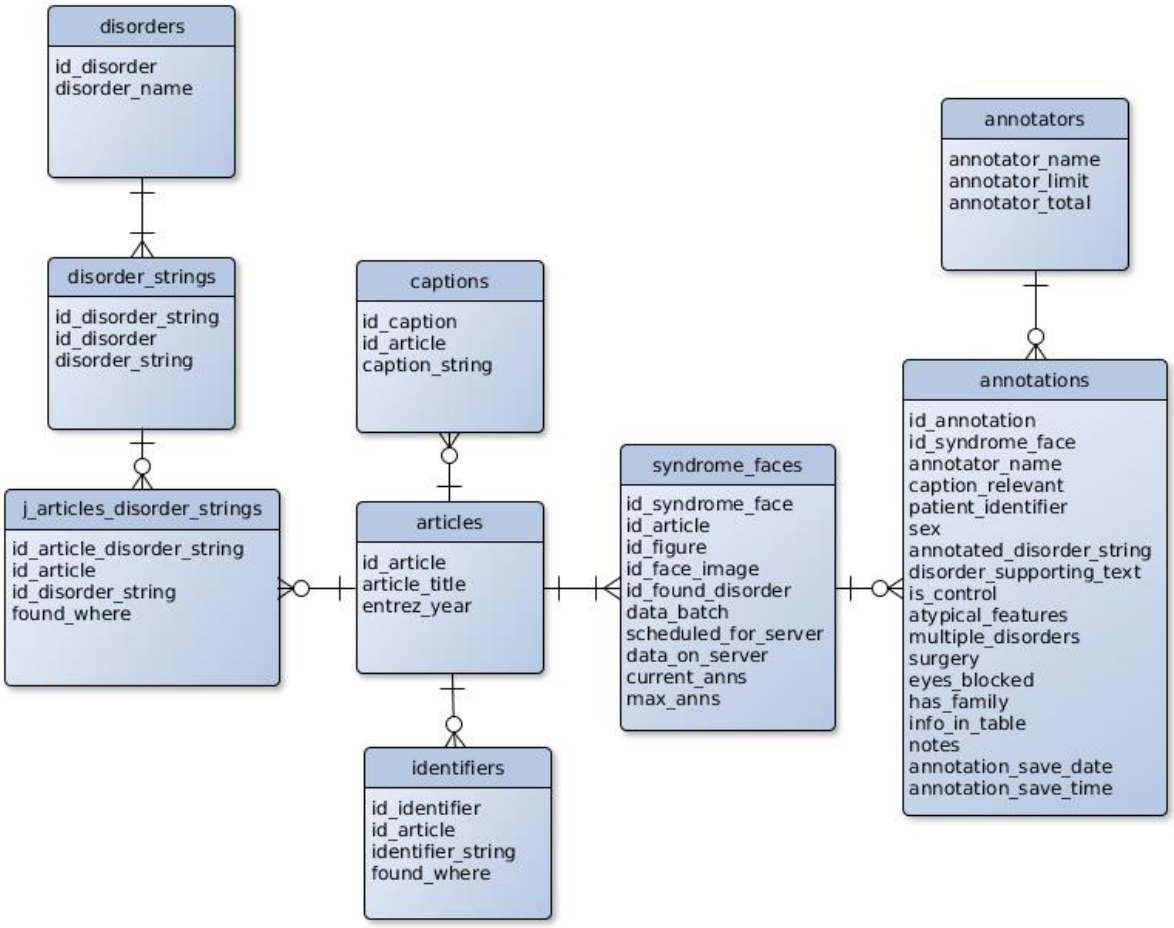


Figure 5.3: Structure of the database used to store annotation information as part of the website's backend.

Suggested annotations. To accelerate each annotation task, we perform several preprocessing steps on the text of each syndrome face article. These steps are used to identify potential annotation information, which can then be suggested to users, thereby speeding up annotation progress. We use a regular expression based search to find figure captions and potential patient identifiers in the full text of the article. A similar method is used to identify any disorder names or aliases in the title or captions of the article. Focusing on these specific areas of the article text is necessary as searching for the names and aliases of thousands of genetic disorders for many articles takes a prohibitively long time.

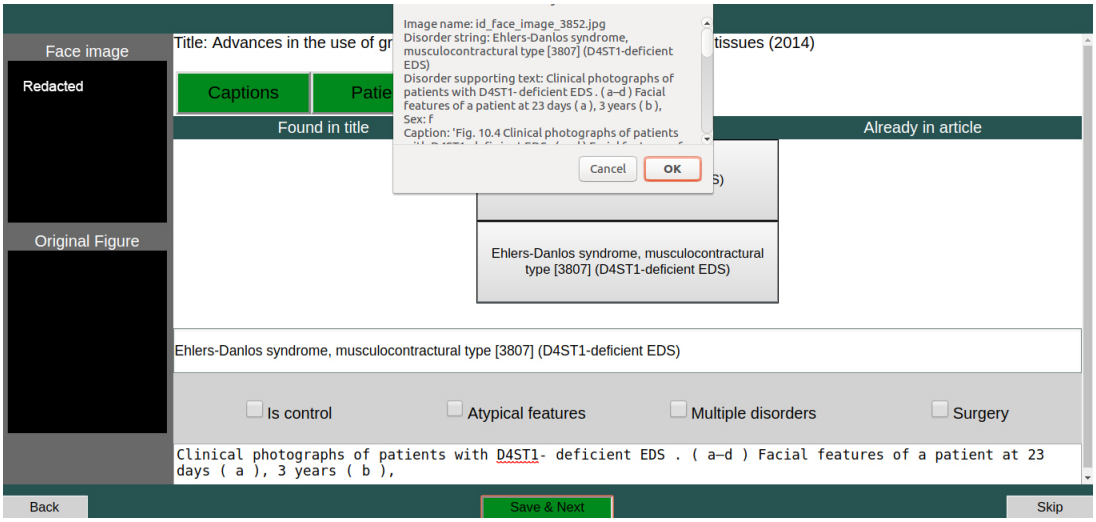


Figure 5.4: Example of a completed annotation and the save confirmation popup, used by annotators to double check their input before it is recorded in the database backend.

User experience. During the annotation task, each user is presented with a consecutive stream of single face images, alongside the corresponding original figure. Images are shown in order of the article from which they came from, and articles are shown in order of the potential syndrome they refer to. This helps annotators quickly build expertise in annotating both a particular article and syndrome. On presentation of a new face image, the relevant article is automatically opened in the user's web browser. The annotator is then asked to find the relevant caption, patient, and disorder information the text of the loaded article. If information is difficult to find, users may skip particular face images to maintain progress. When each individual subtask has been completed to a minimum standard, the corresponding tab header changes colour to green as an indicator. When enough information has been provided for all three subtasks, the save button will change colour, inviting the user to save the annotation to the database as shown in Figure 5.4. Information that has been recorded for a particular article can then be used as a suggested annotation for future users. This helps to expedite the process of annotating many patients with the same disorder in one article, or of annotating many images of the same patient.

User support. Each annotator is initially provided with a user guide to the website which they can be used to refer back to for specific tasks. They are then asked to annotate 20 images as part of a gold standard set. This acts as a form of quality control, and allows us to give individual feedback before further annotation. We have found that annotators are able to find the relevant task information with very high accuracy before being trained. After feedback we would expect this performance only to increase.

Additionally, we continually limit the number of faces given to each annotator, and regularly provide feedback on annotation quality. We are particularly vigilant of annotators spending too long on a single face image as we have found this to have a strong negative impact on overall annotation progress. We further support the annotation process using an online feedback document, where users can report problematic articles or syndrome faces; report bugs and issues; or request components for future versions of the annotation tool.

5.1.3 Collating patient annotations

When training syndrome classification models, there is a need to identify images belonging to a single patient. This helps to prevent informational leakage between training and testing sets, as well as discouraging models from learning a bias toward a small number of over represented patients. However, using multiple annotators to label patient data can lead to issues with labelling consistency. In particular, there can be disagreement between the identifiers used to refer to individual patients. This most commonly occurs in cases where a single patient is referred to using multiple handles in the article text: for example, authors may use ‘patient 1’, ‘P1’ and ‘the patient’ all to refer to the same underlying person.

We can reconcile this problem by making two assumptions on the nature of patient annotations. Firstly, we assume that if two annotators have chosen two distinct patient identifiers for a single facial image, then both identifiers refer to the same underlying patient. Secondly, if two face images are annotated with the same patient identifier, we assume that the same patient appears in both images.



Figure 5.5: Examples of 15 face images which were labelled by syndrome using the annotation tool. So far, patients with over 240 different disorders have been found using the annotation tool.

Using these assumptions, collating annotations of the same underlying patient can be formulated as a problem of graph theory. Each node of the graph represents an annotation of a single face image by one annotator. Edges then connect nodes (annotations) which have the same underlying image or are labelled with the same patient identifier. Annotations referring to a single patient can then be found approximately as the connected components of this graph, requiring only a small manual effort to merge together sub–graphs of annotations which do not share an image or identifier, but ultimately refer to the same intrinsic patient.

5.2 Results of annotation

As described in Chapter 3, we have obtained 20k facial images of potential syndrome faces from biomedical research articles using the PubMed database. Over a 14 week period, we have

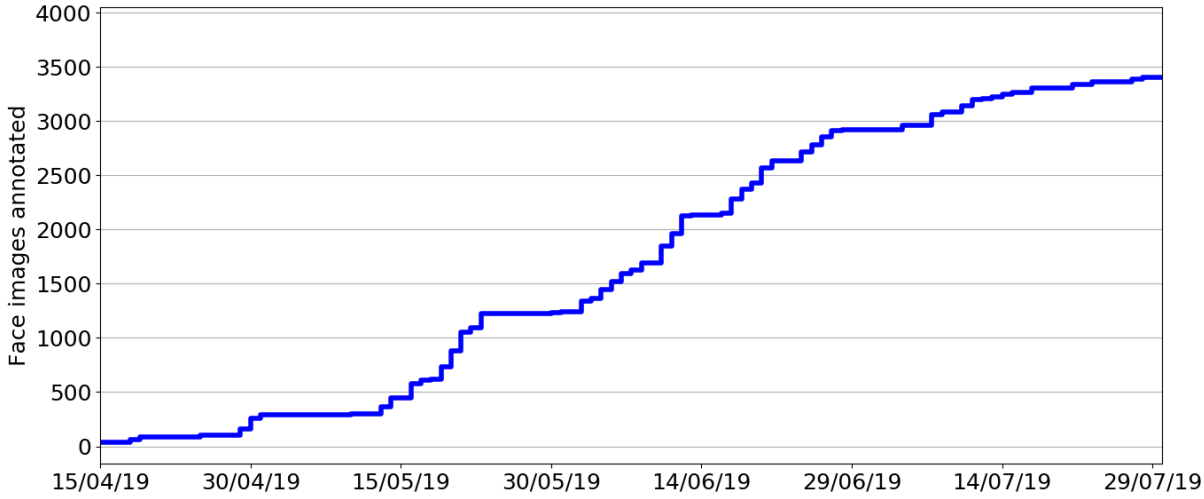


Figure 5.6: Total number of face images labelled by syndrome using our annotation tool over a 14 week period. We have achieved a mean average rate of 250 syndrome face images annotated per week.

given five users access to our annotation tool. These users were non–technical data entry staff, who were not required to have any specific background knowledge of the field. In this time, they have produced 4461 total annotations, labelling 3713 individual facial images, extracted from 1400 figures, belonging to 883 articles. From these, 1754 individual patients were found, who are diagnosed with one of 242 genetic disorders. Examples of faces annotated by disorder using the tool are shown in Figure 5.5.

After an initial familiarisation stage, the annotators were collectively able to record around 250 annotations per week as shown in Figure 5.6. This allows us to estimate that, if progress proceeds at the current rate, annotating the full 20k facial images produced by the download pipeline may take over a year at the current rate, and potentially lead to data for 9k total patients.

In Figure 5.7, we show the distribution of the number of patients found for each of the 242 disorders annotated so far. Many syndromes are so rare that patient images are only found in the literature once or twice. For example, we see that 107 disorders have only a single patient found for them. Conversely, 44 disorders have 10 or more respective patients found so far. This is typically the minimum class size need to train syndrome classification models. Finally,

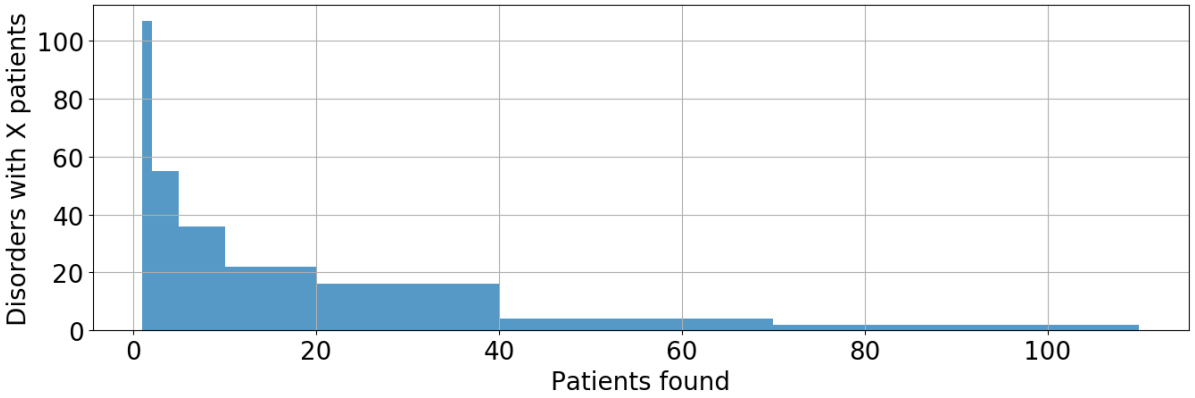


Figure 5.7: Distribution of number of patients annotated using our tool by disorder. So far, ten or more patients have been found for 44 distinct disorders.

106 (6%) of people found in articles have been identified as having no genetic disorder to date. Often these images belong to unaffected family members, or authors who have included their portrait at the end of their work. A comparison of the data produced so far with other published syndrome face data sets is shown in Table 5.1.

5.2.1 Monitoring annotation

Each annotation saved to the website database is labelled with a time stamp, allowing us to monitor the respective progress of each individual annotator. In Figure 5.6, we show the cumulative number of faces annotated by each of five annotators over a 14 week period. In Figure 5.8, we see that two annotators are responsible for the large proportion of annotations. Furthermore,

Table 5.1: Size comparison of our current kinship data set versus other previously published syndrome face data sets. Note these numbers are expected to grow significantly as annotation progresses.

Data set	Patients	Face images	Disorders
Ferry et al. (2014)	-	2,878	8
Gurovich et al. (2018)	-	>10,000	>2,500
Our pipeline[1]	1,754	3,713	242

[1]Results correct as of August 2019

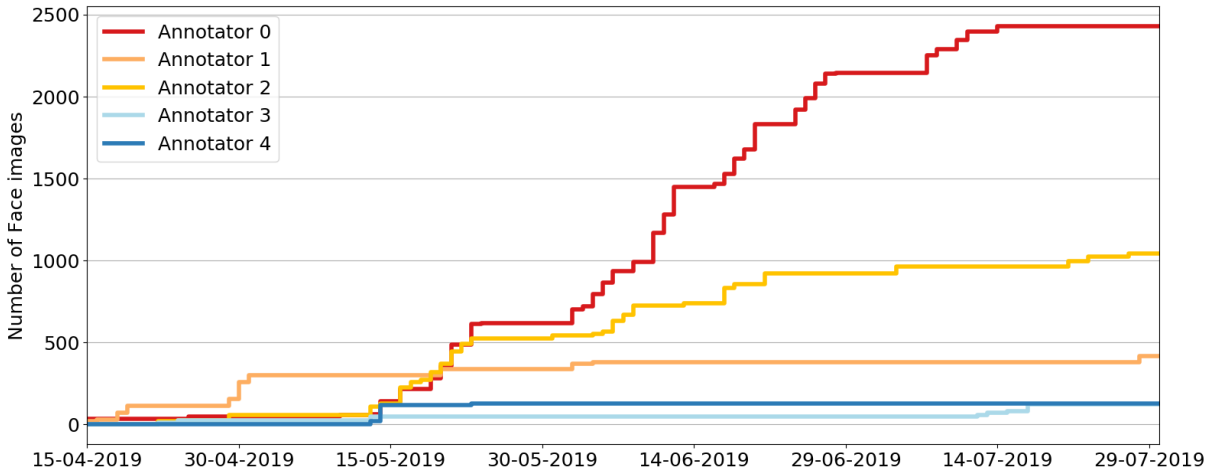


Figure 5.8: Times series showing the cumulative number of annotations produced by each of the five annotators over a 14 week period. 82% of annotations were completed by only two of the five annotators.

some annotators have worked very quickly in a short period of time and then reduced their output, whereas others have annotated faces at a slower, but more sustained pace over the entire period. Currently, the two most prolific annotators have contributed over 82% of the total annotations.

We have also developed a graphical dashboard to monitor individual annotator performance more precisely. An example dashboard for one annotator is shown in Figure 5.9. These plots allow us to view how many images were annotated on a given day, and to visualise the distribution of time taken to perform each annotation. For example, we see that for this user, 50% of faces are able to be annotated in under 45 seconds, and that 90% of faces can be annotated in under 4 minutes. Monitoring these statistics can be used to tell if changes to the tool are leading to a significant speed–up in annotations.

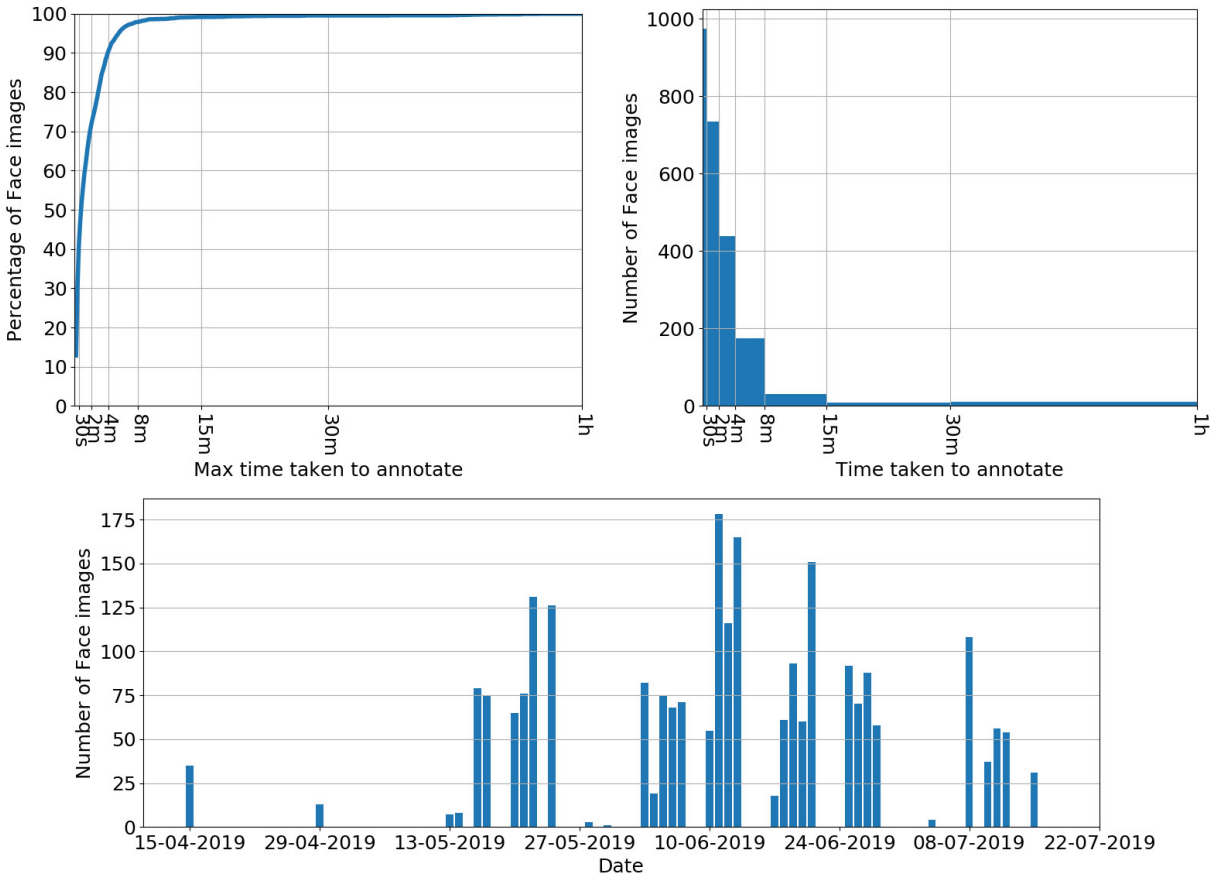


Figure 5.9: Example annotator dashboard plots. These graphs are used to monitor a single annotators progress. In the top left plot, we can see this individual is able to annotate over 90% of face images they have encountered within four minutes.

5.3 Summary

In this chapter, we have developed an annotation tool to facilitate the rapid annotation of genetic syndrome face images from the biomedical literature. The website we have created guides users through three annotation subtasks to extract information on the figure captions, patients and diagnoses found within the article text. These tasks are accelerated by preprocessing the article text to extract and display potential annotation suggestions.

The website has been used by five annotators over a 14 week period to produce one of the

largest syndrome face data sets collected to date, containing over 1.7k patients with over 240 genetic disorders. Annotation will continue to run over the coming months, and we predict to extract from 10k patients from the current batch of 20k face images obtained using the download pipeline described in Chapter 3. The patient data produced by the annotation tool is, and will continue, to be used to train the next generation of syndrome classification algorithms.

Part II

Facial kinship

6 | Literature review – kinship verification

It is well known that children look more similar to their parents than to unrelated adults of the same sex (Alvergne et al., 2007). Perceptually, human's ability to recognise kinship from two faces is closely related to our ability to detect facial similarity. DeBruine et al. (2009) showed that the similarity scores assigned to pairs of faces by human annotators were strong predictors of the probability that a different group of observers would identify pairs as siblings. In our early lives, facial similarity between fathers and their children plays an important role in indicating paternity. The perception of this similarity has been shown to correlate with the quality of the father–child relationship, as well as paternal investment in a polygynous community (Burch & Gallup Jr., 2000; Alvergne et al., 2009).

Computationally, kinship verification is the task of determining whether two or more people are closely related, using only photographs of their respective faces (Fang et al., 2010). Potential uses of kinship verification include: organisation of digital family photo albums (such as those found on social media accounts); aiding ancestral and genealogical research; detecting lost children using CCTV; identifying potential cases of child trafficking; and in distinguishing family traits from environmental or disease related changes in a clinical research context. Aside from these applications, there also potentially exists several unethical uses of kinship verification, such as nonconsensual paternity testing. The facial kinship research community must be aware of these potential misuses, and manage the release of algorithms and data sets accordingly, in the future.

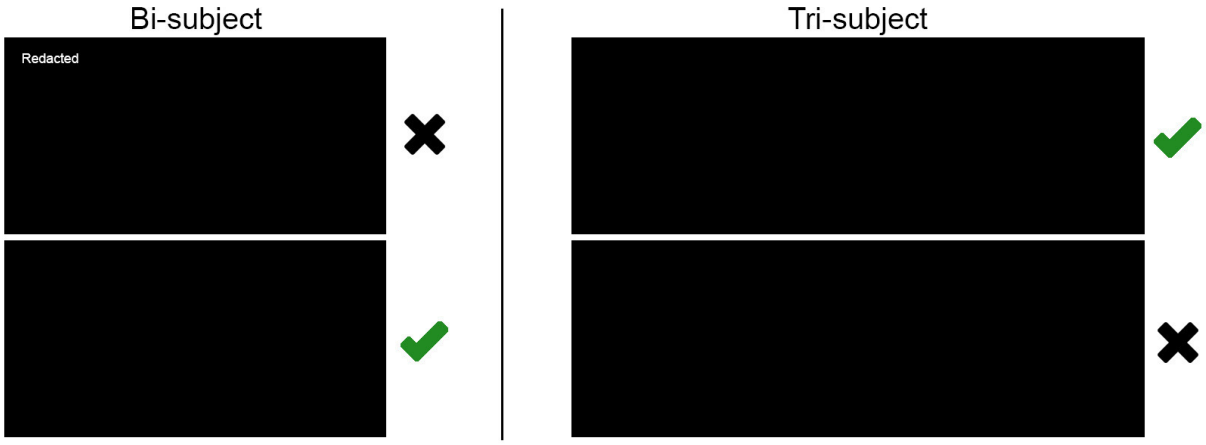


Figure 6.1: Visual description of the tasks of bi–subject and tri–subject kinship verification. For bi–subject verification one must determine whether two individuals are directly related. For tri–subject verification one must decide whether three images represent a valid mother–father–child triple.

6.1 Kinship verification data sets

Over the last nine years, the field of kinship verification has been driven by the collection and public distribution of several data sets. The tasks presented by these data sets have served to branch the field into distinct challenges (as shown in Figure 6.1. The canonical task of the field is bi–subject or one–to–one kinship verification, where the goal is to determine whether two individuals are related using only images of their faces. This task was extended to one–to–many family classification, where one must determine which family a person is related to from a gallery (Robinson et al., 2016; Fang et al., 2013). Intermediate to these two tasks is tri–subject kinship verification, which presents triples of a mother, father and potential child; and asks whether the third person is the biological child of the two parents (Qin et al., 2015).

The first kinship data set was published by Fang et al. (2010), and consisted of just 150 image pairs of the parents and children of public figures and celebrities. Soon after UBKinFace–V1, and the subsequently expanded UBKinFace–V2, were released by S. Xia et al. (2011, 2012). Both data sets included images of parents as both young and old adults, with UBKinFace–

V2 containing 200 parent–child pairs. S. Xia et al. (2012) also released the FamilyFace data set which contained 214 images of groups of related and unrelated individuals collected from Facebook and FlickR. However, the first explicit family classification task was set by Fang et al. (2013), with the release of the Family 101 data set in 2013. This was the largest kinship data set published at the time with nearly 15k images of 600 public figures from over 200 nuclear families.

Classifying kinship using video analysis was first attempted by Dibeklioglu et al. (2013), who modelled facial expression dynamics to predict kinship by adapting the UvA–NEMO smile data set (Dibeklioglu et al., 2012). The first purpose–built kinship video data set was Kinship Face Videos in the Wild (KFVW) which contains 418 pairs of face videos of 152 individuals, and was produced by Yan & Lu (2017). Recently, Kohli et al. (2019) released the KIVI video data set which contains a single video for each of 503 individuals.

For many years, KinFaceW–I and KinFaceW–II were the two most widely used kinship verification data sets (Lu et al., 2014). The KinFaceW–I data set contains pairs of images from the four main bi–subject kinship verification categories: mother–daughter (127 pairs), mother–son (116 pairs), father–daughter (134 pairs), father–son (156 pairs). The major differences between the two data sets are that KinFaceW–II is larger, and contains a greater number of photos cropped from the same original image. These data sets formed the basis of the first kinship verification challenge, FG 2015, which received submissions from five institutions (Lu et al., 2015).

TSKinFace was introduced as an alternative to the growing number of bi–subject data sets (Qin et al., 2015). For many of the use situations described for kinship verification, such as organising family photo albums or locating missing children, it is likely that pictures of both parents will be available. TSKinFace is the largest data set of kinship image triples published to date. The data set contains 787 images of 2589 individuals belonging to 1015 tri–subject kinship groups (father–mother–child). The images from TSKinFace are collected from family photographs. The data set is split by the gender of the child with 513 positive father–mother–son (FM–S) triplets and 502 positive father–mother–daughter (FM–D) triplets.

The largest kinship data set released publicly is the families in the wild (FIW) data set (Robinson et al., 2016). It contains 1000 families, 10676 people and 30725 face images. For the task of bi–subject kinship verification, the data set consists of 418k pairs of images, heavily weighted towards over 37k pairs for each of the four major parent–child relationships, and over 75k for sibling–sibling relationships. FIW now forms the standard data set used for modern kinship challenges (Robinson et al., 2017).

6.2 Bias in kinship data sets

As the field of kinship verification has grown, researchers have identified characteristics of public data sets which undermine the generalisability of trained models. The need for larger, more reliable data sets was identified by Wu, Boutellaa, Feng, & Hadid (2016). They suggested that models trained on kinship data sets may be learning clues such as image background or average colour, and further recommended that future methods be tested on several kinship data sets to ensure generalisability. This problem was also raised by López et al. (2016), who recommended that the two KinFaceW data sets should no longer be used in kinship research. Their work showed that comparable results could be achieved on the data sets using a method with no understanding of kinship: classifying pairs of images based upon the distance between the chrominance averages of images in the Lab colour space. Further to this work, Wu, Boutellaa, López, et al. (2016) showed that results on kinship data sets can be improved by using colour images rather than grey–scale images. Although one might expect that skin colour could help to identify kinship relations, it also gives more information about the background and colour distribution of the image, a confounding variable which can be used to improve results on data sets containing images cropped from the same photo.

Other groups have attempted to avoid this issue by using images where each family member is cropped from different original photographs, such as the Family 101 and UBKinFace data sets (Fang et al., 2013; S. Xia et al., 2011). However, such images are expensive to find and so

photographs of famous people and their children, who have many public images available, are frequently included, introducing a new bias into the data sets.

Other errors enter kinship data sets as automated production becomes more common, and manual checking becomes less practical. For example, two unrelated groups of actors from the TV shows *My Family* and *Modern Family* were originally included in the positive TSKinFace data set. Similarly, the Families in the Wild data set contains several incorrectly recorded and/or impossible family relationships. Checks such as ensuring familial relationships are symmetric, or that the biological parents of a child must be of opposite sexes have been overlooked.

6.3 Approaches to kinship verification

The size of the kinship verification research community has grown significantly over the past decade. As a greater variety of kinship data sets have been proposed, many groups have developed or applied existing methods to the task. Broadly, advances among kinship verification models can be roughly divided into two categories: those which utilise new image features, and those which deploy new learning algorithms.

Initially, many shallow computer vision features were used for the task of kinship verification including: Histogram of Gradients, Weber Normalisation, Spatial pyramids, Gabor–based gradient orientation pyramids, Local binary patterns, and DAISY image descriptors (Fang et al., 2010; Kohli et al., 2012; Zhou et al., 2011, 2012; Lu et al., 2014; G. Guo & Wang, 2012).

Often these features were compared using cosine similarity or trained classification methods such as SVMs (Fang et al., 2010; Zhou et al., 2011, 2012). However, many groups have attempted to use metric learning approaches to build kinship embeddings. Lu et al. (2014) developed 'neighbourhood repulsed metric learning' to create an embedding whereby unrelated people are pushed further apart than related people. Furthermore, Qin et al. (2015) used a metric learning approach to model tri–subject kinship verification to benchmark the TSKinFace data

set.

Often, researchers will use several types of image feature, fusing the collective information using multi–view algorithms (Hu et al., 2014; Yan et al., 2014). This year, Northeastern SMILE Lab hosted a bi–subject kinship verification Kaggle challenge using the FIW data set[1]. The competition had over 500 participants, and was evaluated on a private test set. The winner achieved a verification accuracy of 92.3%, by extracting features of pairs of faces from 37 different networks and combining the scores linearly to produce a final output prediction. Many of the top competition submissions used ensembles of multiple facial feature extraction networks.

In Chapter 2, we described how deep learning revolutionised the task of facial verification. Similarly, deep learning features are now commonly used kinship verification algorithms (K. Zhang et al., 2015; Xiong et al., 2015; Nandy & Mondal, 2019). The range of techniques used has followed the trends of deep learning architectures. Several research groups have now attempted to develop approaches based upon auto–encoder architectures (Dehghan et al., 2014; Kohli et al., 2019). Furthermore, generative adversarial networks (GANs) have also been applied in the field. For example, S. Wang et al. (2018) used a GAN to synthesise younger facial images of parents to reduce the age gap distribution, before comparing images of young adult parents and their respective children.

[1] www.kaggle.com/c/recognizing-faces-in-the-wild

7 | From same photo bias in kinship data sets

Over the last few years, several kinship verification data sets have been developed for research groups to benchmark against. These data sets of images and videos have been produced by many different research groups, and each has its own unique characteristics including variations in the number of images, collection methods, and types of relation included.

In building these data sets for kinship verification, often different individuals' images have been cropped from larger family photos. While this is a fast way to find people in the same family, it inadvertently builds in a clue to the task – that people from the same photo are related. This has been overlooked as a potential issue, as it is far easier to identify whether two face images are from the same original photo than it is to determine if they are kin.

There are several cues that can be used to determine if two face images are cropped from the same original photograph[1], and by inference are more likely to be kin. For example, common lighting and shadows, background, blur, resolution, tone, contrast, clothing, camera specific noise and exact regional overlap between crops. Another confounding signal, which is present when parent and child are cropped from the same photo, is that the age difference between the people shown in each crop will be roughly the same as the average age which parents give birth to children. Illustrations of such cues are shown in Figure 7.1.

Deep neural networks are notorious for finding subtle data shortcuts to exploit in order to 'cheat' and thus not learn to solve the task in the desired manner; an example is the misuse of chromatic aberration by Doersch et al. (2015) to solve the relative–position task. Thus, deep learning

[1]Note that two individuals do not need to be cropped from the same photograph for their respective facial images to display near identical photographic conditions. A similar effect would be produced if one were to take two separate facial photographs of two subjects in quick succession within a single photographic session. We use the phrase 'from same photo' as a shorthand for this wider range of conditions, broadly representing a situation where two facial photographs appear to have been taken under near identical photographic conditions.

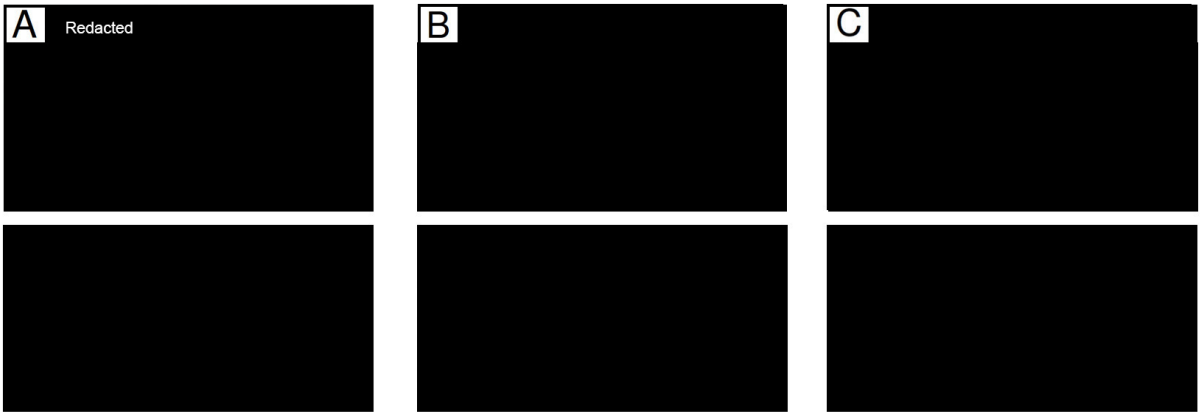


Figure 7.1: Representative examples of how cropping face images from the same original photo can make the kinship verification a trivial task. Confounding, non–kinship information includes camera specific noise, background similarity (A,C), the historical era in which the photo was taken (B), image lighting and tone (A,B,C), and the relative age difference between parents and children (A,B,C).

models are liable to pick up on these FSP cues to gain a boost in performance on kinship tasks.

Many models have already reported on these data sets infected by this bias. Therefore, we would like to benchmark the extent to which these results could have used the 'from same photo' signal. This will provide a clear understanding of how much of the accuracy reported is based upon kinship signals, and how much could have been 'from same photo' signal.

We achieve this by creating a new data set of 25k images, containing 145k faces, and 914k pairs of faces taken from non–familial group photos. We use this data set to train a CNN classifier to determine if two facial images are cropped from the same original photo or not. Crucially, this classifier has no understanding of kinship relations.

We present results on the KinFaceW–I (Lu et al., 2014), KinFaceW–II (Lu et al., 2014), Cornell KinFace (Fang et al., 2010), Families in the Wild (FIW) (Robinson et al., 2016), and TSKinFace (Qin et al., 2015) data sets. We show that the 'from same photo' signal exists for all of these five data sets. For some, the signal is fairly weak, whereas for others it explains a large proportion of the state–of–the–art results. In many cases we achieve comparable results with

other submissions using only the ‘from same photo’ signal.

7.1 Training a classifier to detect faces from the same photo

In this section, we describe our approach to train a CNN classifier to be able to determine whether two facial images are cropped from the same original photograph or not. For positive training data we use pairs of face images cropped from non–family group photographs (see Figure 7.2). This was done to ensure the classifier could not learn kinship signals between faces. We evaluate the ability of our classifier to classify faces from the same photo on a held–out test set.

7.1.1 Generating FSP data set

To generate our FSP data set, we have collected face images that were cropped from the same original photo, but crucially where the people shown in the face images do not share a kinship relation. We began by creating a list of 125 search terms that describe groups of people, who are unlikely to share a kinship relation. These include terms such as ‘school students’, ‘business meeting’, ‘sport team photos’. We then use the Bing Azure API (Microsoft Corporation, 2018) to find URLs of images related to these search terms. We ensure that the images returned are



Figure 7.2: Examples of photos found using non–kin group search terms and the Bing Azure API (Microsoft Corporation, 2018). Pairs of faces found in these images are used in creating the From Same Photo (FSP) data set.

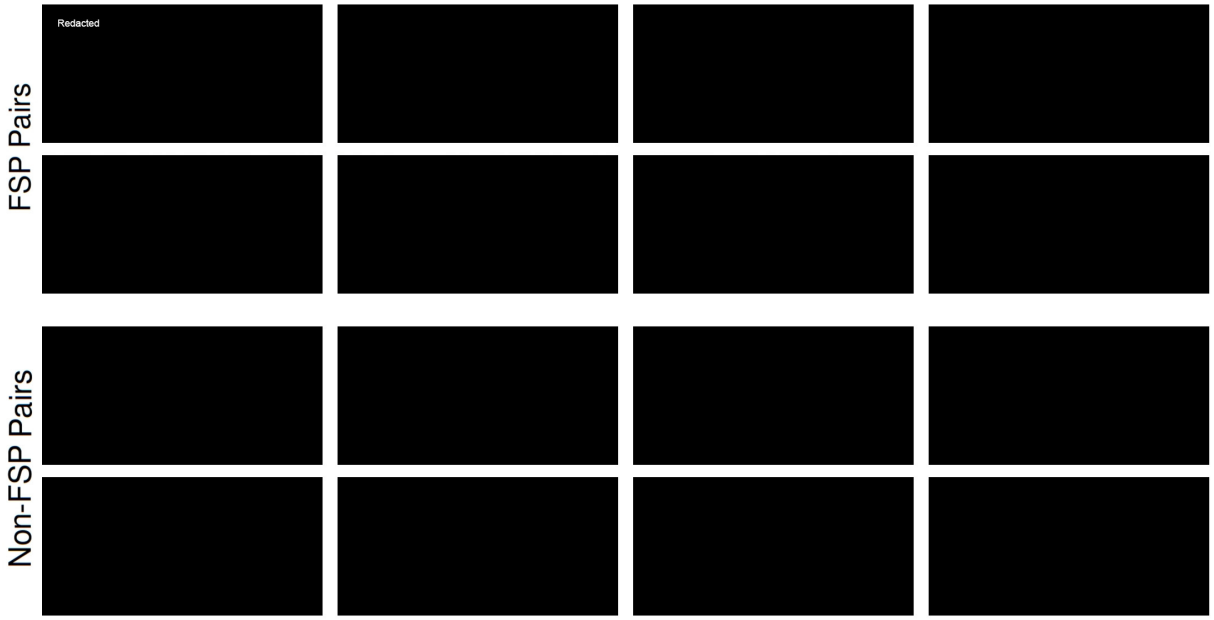


Figure 7.3: Examples of pairs of faces used to train the From Same Photo (FSP) classifier. FSP pairs are cropped from the same non–kin group photo. Images in Non–FSP pairs are cropped from separate non–kin group photos.

large enough to potentially contain high resolution faces, and that each URL is returned only once in an attempt to avoid duplicate images. These searches result in a list of 82k URLs: an average of 654 per search term. We download each image and, with some loss due to moved content and unusable file formats, obtain 76k images.

The Dlib face detector (King, 2009) is used to find faces within these images. For our purposes, a usable group photo is defined to be an image which contains at least two faces, where the height and width of both face detection regions is not less than 50 pixels. This results in a maximum usable positive data set of 24.8k group photos, which contains 146k face images and 915k possible pairs of face images. We partition this data set into training (70%), validation (10%) and testing (20%), as shown in Table 7.1.

Although the vast majority of pairs of faces in the positive FSP data set are correctly labelled, it was found that there are some cases of negative pairs making it through the processing pipeline. For example, some of the group images collected were composite collages of separate photos,

and so the faces in the collages were not originally taken from the same photo. Another example which could lead to falsely labelled positive pairs comes from photographs of people standing next to portrait paintings or photographs which contain a face. These faces are also detected by Dlib, and so lead to pairs of faces being falsely labelled as positives. Furthermore, we cannot exclude that some images will have true kin related people present in them. Overall, these examples make up a very tiny proportion of the data set, and so it is not expected that they skew our results significantly.

Training the FSP classifier also requires creating a corresponding negative set of pairs of faces that are cropped from different original photographs. This is achieved by taking each image in the positive data set and randomly swapping one of the faces for another in the same training/validation/testing subset. This ensures that pairs of faces in the negative data set are not from the same photo, and that face photos do not leak across train, validation and testing splits. Furthermore, matching the total number of positive pairs and negative pairs in this way also helps to ensure that the FSP classifier does not learn a bias towards predicting a particular class. Examples of pairs of images used to train the FSP classifier are shown in Figure 7.3.

7.1.2 The FSP classifier

In this section we describe the CNN classifier used to detect whether two facial photographs are cropped from the same original image. For training data we use the balanced FSP data set described above in Section 7.1.1. We evaluate the performance of the FSP classifier on a test set of images from the FSP data set, and analyse the results using the standard receiver operating characteristic (ROC) and precision–recall (PR) curves.

Architecture. The CNN architecture can be split into four distinct parts as shown in Figure 7.4. The first part consists of two parallel copies of a pre–trained VGG–M–Face, up to the output from the fully connected FC6 layer. This architecture was first used by Chatfield et al. (2014) and has been trained here for facial recognition on the VGGFace data set. The VGG–M–Face

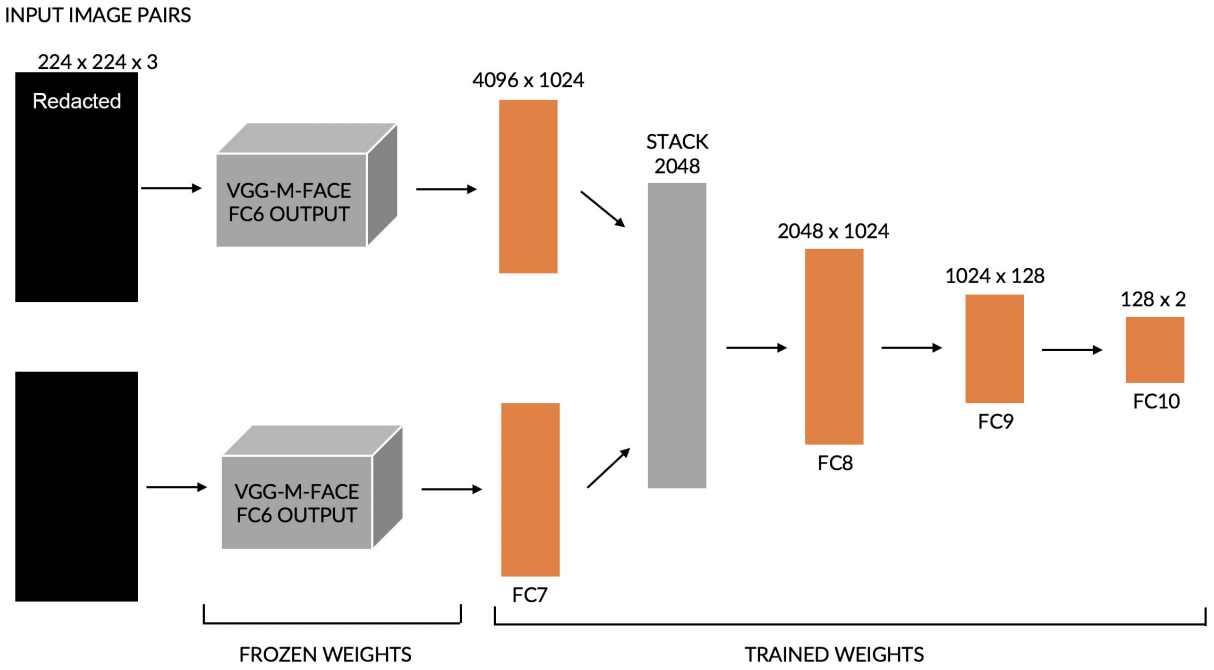


Figure 7.4: The FSP network architecture.

networks generate a 4096-dimensional feature vector for each face image of the input pair. Both feature vectors are then fed to a trainable fully connected layer to reduce their dimensionality. These reduced feature vectors are then stacked into a single vector and fed through three more fully connected layers and ReLU activation functions to produce a two–dimensional vector giving the probability that the pair belongs to the class 'from same photo' or 'not from same photo'. The network is implemented in the PyTorch framework, and the code (and data set) will be made publicly available.

Training. The FSP network was trained using stochastic gradient descent to minimise a softmax cross entropy loss function. During training we freeze the weights of the VGG–M–Face part of the model, but train all other layers. The initial learning rate was set to 0.1, and decreased by a factor of 0.5 every five epochs, until the classifier's performance on the validation set began to saturate. In training we use dropout with $p = 0.1$ between layers FC8 and FC9 to prevent overfitting. During training we augment our input images by resizing them up to 256×256 pixels and taking a random 224×224 pixel crop from the resulting image.

Table 7.1: Summary of the number of pairs of face images in the data sets used to train, validate and test the FSP classifier. The area under receiver operating characteristic curve (ROC AUC) is reported for the FSP classifier on the test set of images

Data set	**Train**	**Validation**	**Test**	**ROC AUC (%)**
FSP	1,321,440	184,358	323,236	98.1

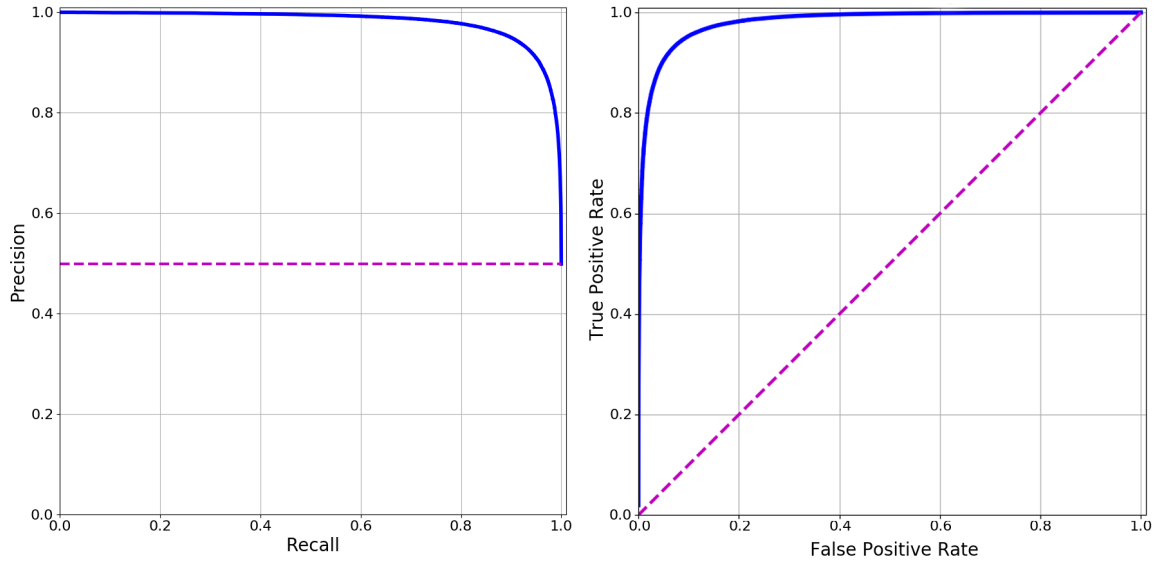


Figure 7.5: Precision–recall and ROC curves for testing the FSP classifier on the FSP test set. The area under the ROC curve is 98.09%, and the classifier performs with an equal error rate of 6.97% and an average precision of 98.07%.

Testing. We evaluate the FSP network on a test set taken from the FSP data set that was not used in training or validation. The testing set consists of 323k pairs of face images, half of which are cropped from the same original photo. At test time we test a pair of images ten times using different permutations of region crops and horizontal flips. The result of these ten tests is then averaged to give the classifier score for a pair of images.

As can be seen in Figure 7.5, the FSP classifier is able to tell whether two images are from the same photograph with very high accuracy. Testing resulted in an area under ROC curve of 98.09%, with an equal error rate of 6.97%.

7.2 Testing the FSP classifier on kinship data sets

In this section, we examine five popular kinship data sets. We describe the testing procedure used for each data set to benchmark the degree to which FSP is a biasing factor for their respective one–to–one and triplet kinship verification task challenges.

KinFaceW–I and KinFaceW–II (Lu et al., 2014) Typically, kinship models are tested on these data sets are using five–fold cross validation, where both the positive and negative pairs are specified for each of the five folds. Here we collect all the positive and negative pairs across all five folds to create a balanced test set. The KinFaceW–I test set contains 1066 pairs of face images, and the KinFaceW–II test set contains 2000 pairs of face images, with 250 for each of the four major relations.

Cornell KinFace (Fang et al., 2010) This is the smallest data set among the five tested, consisting of only 144 positive pairs. To define the negative portion of the test set we randomly substitute a parent or child from the positive pair with a random parent or child from the remaining positive set. To avoid bias from choosing a particularly easy negative set we average over five different versions of the negative set, randomised uniquely each time.

Families in the Wild (FIW) (Robinson et al., 2016) The positive and negatives test pairs of faces are already determined with this data set, as it was used in the Recognising Families in the Wild (RFIW2018) kinship verification challenge. Here we test on the 100k pairs of images in the challenge evaluation set.

TSKinFace (Qin et al., 2015) In this data set, related triplets of father–mother–son and father–mother–daughter are specified. For the negative test set, we randomly combine images of parents with an image of different child of the same gender as their own. To reduce the probability of reporting results on a particularly easy negative set, we repeated our experiments with a different set of permutations for the children and parents in the negative set each time.

At test time we split each triplet into two pairs, father–child and mother–child. Each pair is then

scored by the FSP classifier. We take the maximum of these two scores as the test score for a triplet. This corresponds to asking whether at least one of the parents is cropped from the same original photo as the child.

7.2.1 Implementation details

We apply the FSP classifier to the kinship data sets naively, that is to say without further training for the intended kinship verification task. To achieve this, we extract each image at three crop sizes. This is required as the Dlib face detector tends to propose tight square regions around the centre of a face, often cutting out chin, tops of heads and ears. However, the face images contained within two of the data sets we test on are more loosely cropped. We found that the greatest accuracy was achieved on KinFaceW–I, KinFaceW–II and FIW using the FSP classifier trained with images cropped to the standard Dlib size. Whereas on Cornell KinFace and TSKinFace, the best results were obtained by expanding the width and height of the images by 15%.

We report the accuracy our FSP classifier is able to obtain on each kinship test set. To determine the accuracy, we set the threshold of our FSP classifier using five–fold cross validation on the test sets. We determine the threshold which produces the maximum achievable accuracy for the

Table 7.2: Results of using the FSP classifier to predict kinship on KinFaceW–I, KinFaceW–II, Cornell KinFace, FIW and TSKinFace data sets. Accuracy percentages are shown for the pair subsets of mother–daughter (MD), mother–son (MS), father–daughter (FD), and father–son (FS), and the triplet subsets of father–mother–daughter (FMD) and father–mother–son (FMS), as well as across the entire test set.

Data set	MD	MS	FD	FS	FMD	FMS	All
KinFaceW–I	86.0	78.3	74.1	74.6			76.8
KinFaceW–II	94.8	90.3	84.5	92.3			90.2
FIW	60.3	59.3	59.0	57.5			58.6
TSKinFace					88.6	89.4	88.6
Cornell KinFace							76.7

FSP classifier on 80% of the test set, then calculate the accuracy on the remaining 20% of the test data for five splits. The mean accuracy across the splits is then reported.

7.3 Results of testing FSP classifier on kinship data sets

We show the results for the FSP classifier as a kinship classifier on the five data sets in Table 7.2. For each of the bi–subject kinship verification tasks we report high accuracies: KinFaceW–I 76.8% , KinFaceW–II 90.2%, FIW 58.6%, Cornell KinFace 76.7%. Examples of from same photo bias in these data sets are shown in Figure 7.6.

There is variance between the tasks depending on the gender of the parent and child in the image. In kinship verification tasks this could be interpreted as perhaps biases in facial similarities between genders, however this is not something we should expect from the FSP classifier. It is more likely to be due to biases in likelihood for various gender pairs of family members to show up together in photographs, or even just stochastic variance from data set sampling.

In Table 7.3, we show summary statistics across a large number of published kinship verification models on the various data sets, and show that the FSP classifier performs kinship classification as well or better than most of them. We also achieve high accuracies for the tri–subject verification data set beating many published models and near the state–of–the–art published results (Table 7.3, TSKinFace 88.6%). In this instance there is no significant gender bias in accuracy of verification of kinship for the gender of the child.

Finally, we can see that the FSP classifier consistently outperforms human classification of kinship. We can expect that the deep learning models will have inadvertently learnt the FSP signal as a means to solve the kinship verification task, but can only speculate if human classifiers uses the same type of information to simplify the task. Examples of positive kin pairs and triplets found by the FSP classifier are shown in Figure 7.6. A selection of pairs the classifier falsely predicted as being from the same photo are displayed in Figure 7.7.

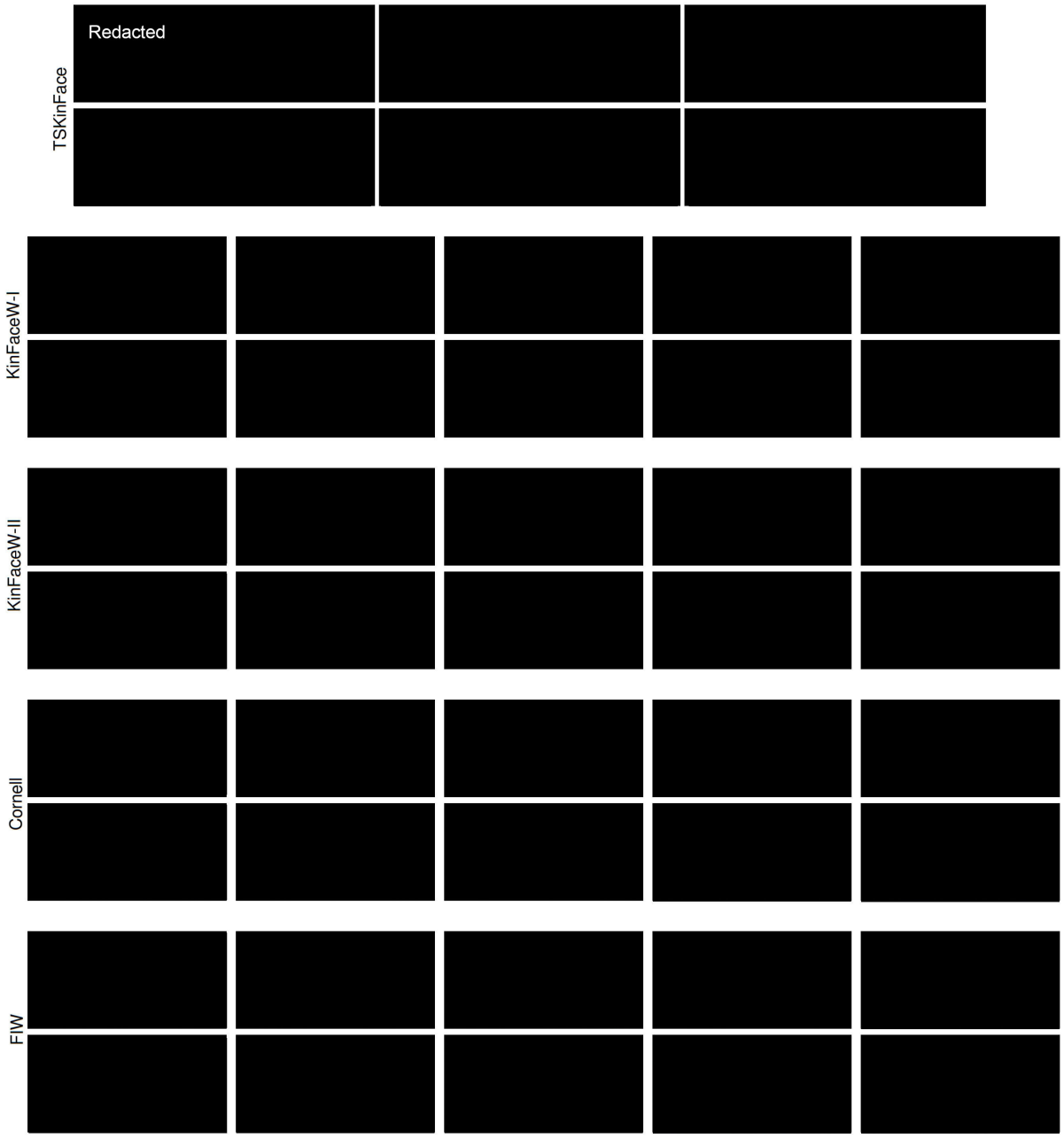


Figure 7.6: Examples found in kinship data sets which we predict are taken from the same photo. Each row corresponds to examples from one of the five analysed kinship data sets. Note the similarities in background, facial pose, lighting, and overall image tone between images cropped from the same original photo.

Table 7.3: Results of using the FSP classifier to predict kinship, and previously published kinship classifiers on all five kinship data sets. KinFaceW is abbreviated as KFW. Note the high accuracies achieved by using the FSP classifier as a kinship classifier on each data set, in comparison with the average and state–of–the–art maximum accuracies over results reported in previous publications. Also note the available human annotation benchmarks which are all exceeded by the FSP classifier model. Some results have been omitted due to page length constraints. Human results are thanks to Y. G. Zhao et al. (2018); Fang et al. (2010); Qin et al. (2015)

Paper	KFW–I	KFW–II	Cornell KF	FIW	TSKinFace
FSP classifier (ours)	76.8	90.2	76.7	58.6	88.6
Mean accuracy	78.3	82.0	79.2	67.2	85.7
Median accuracy	78.8	82.8	79.0	68.8	87.2
Max accuracy	96.9	97.1	94.4	74.9	93.4
Human	71.0	74.0	67.2		79.5
Aliradi et al. (2018)	80.6	88.6			
Moujahid & Dornaika (2018)	88.2	88.2			
Lopez et al. (2018)	68.4	66.5			
Yan (2019)	77.6	78.5			
Robinson et al. (2018)	82.4	86.6			
Kohli et al. (2019)	96.9	97.1	94.4		
Zhou et al. (2019)	82.8	85.7	81.4		
Mahpod & Keller (2018)	79.8	87.0	76.6		
Y. G. Zhao et al. (2018)	81.5	82.5	81.7		84.5
S. Wang et al. (2018)				69.5	
C. Xia et al. (2018)					90.7
Yang & Wu (2017)	88.6	90.3			93.4
X. Chen et al. (2017)	83.3	84.3			
Lu et al. (2017)	83.5	84.3			
Patel et al. (2017)	78.7	80.6			
Kohli et al. (2017)	96.1	96.2	89.5		
Laiadi et al. (2017)			83.2	54.8	
Duan & Zhang (2017)				66.6	
Dahan et al. (2017)				65.0	
Li et al. (2017)				74.9	
S. Wang et al. (2017)				68.8	
Zhou et al. (2016)	78.8	75.7			
M. Xu & Shang (2016)	77.9	77.1			
Q. Liu et al. (2016)	77.9	81.4	75.9		
J. Zhang et al. (2016)					89.7
Robinson et al. (2016)				71.0	
Qin et al. (2015)					85.4
Lu et al. (2014)	69.9	76.5	66.5		72.1
Fang et al. (2010)			70.7		



Figure 7.7: Examples of non–kinship pairs that the classifier incorrectly predicted as being from the same photo. Pairs shown, from left to right, are from the KinFaceW–I, Families in the Wild and Cornell KinFace data sets respectively.

7.4 Summary

In this chapter we have applied a 'from same photograph' (FSP) classifier as a naive classifier of kinship to five data sets regularly used for kinship verification tasks, and thereby have estimated the degree to which the FSP signal contaminates these data sets. The FSP classifier performs amongst the best published kinship classifier models, despite not being trained for this primary task. It is likely that deep neural network models built with the intention to verify kinship, are instead primarily using this much easier to detect FSP signal. We have also obtained a new data set for training classifiers to detect facial crops from the same photograph to begin to address this problem.

Furthermore, it is important to note that there are many other spurious signals that one should expect in kinship verification data sets. For instance, deep learning kinship classifiers would also be expected to learn biases in distributions of age, gender and particularly ancestral backgrounds. Due to the way the FSP classifier has been trained it is blind to many of these other confounding non–kinship signals contained within existing kinship data sets.

We recommend that an FSP classifier should be an important part of kinship data set production pipelines: either to ensure the FSP signal is removed entirely, or is balanced between positive and negative kin pairs. It should be considered that inappropriate construction of negative sets can introduce biases by overly simplifying the task, such as generating father–father–child tri–subjects, mismatched ancestral backgrounds, and implausible age distributions (such as a child

older than the parents). Kinship verification is a challenging task made more difficult by inherent biases in how data occurs in the wild. Sharing a photograph does not make us relatives, but we are likely to share a photograph if we are.

8 | The Geni ancestry kinship data set

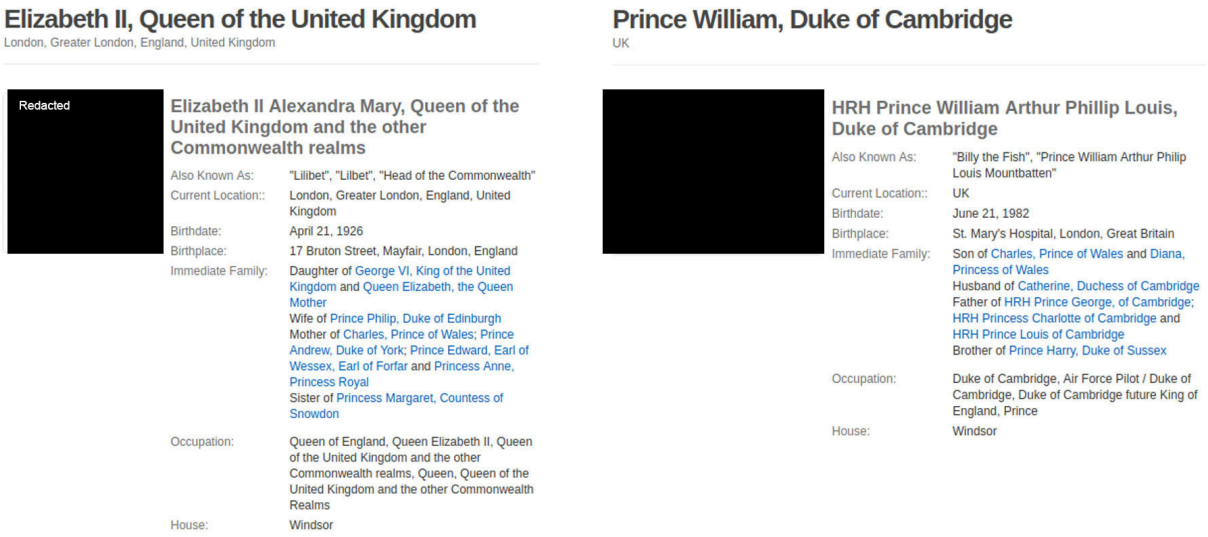


Figure 8.1: Two examples of profile pages from the ancestry website Geni. Profiles include images, links to immediate family, birth date, and other genealogically relevant data.

As discussed in Section 6, the successful application of modern computer vision techniques to the task of visual kinship verification critically depends on the production of large kinship data sets. Furthermore, these data sets must be carefully constructed to minimise common task–specific biases. To address this issue, we create two data sets for the task of one–to–one kinship verification and mother–father–child (MFC) tri–subject kinship verification. We build these data sets by sourcing family data and images collected from an ancestry website.

Geni is an open access ancestry website containing hundreds of millions of individual profiles. Each person on the Geni website has an associated profile page where they can store related images as well as genealogical information such as gender, birth and death dates, and familial relationships in the form of links to other profiles. Examples of a Geni profile pages are shown in Figure 8.1.

A major goal of Geni is to connect not just individual family groups, but to allow the merging of completed family information into the 'World Family Tree'[1]. This is a family tree containing over 136 million people. Once a person has established a local family tree and established a link to a node in the main tree they can then connect their tree to the 'Big tree'.

Creating a kinship data set using images from Geni requires overcoming several technical challenges necessitated by the sheer scale and complexity of data which must be obtained. Firstly, a robust automated method of collecting profile, family, and image data from Geni is needed. Once this data is obtained, images must undergo several stages of processing and cleaning before being suitable for use in a kinship data set. This includes face detection to find single face images, and duplicate image detection to remove multiple copies of the same facial images. Finally, images which are unsuitable for a visual kinship verification data set, such as photographs of statues and portrait paintings, must be identified and removed.

In Chapter 7, we described how cropping images from the same original photo could lead to unintended visual signals arising in a kinship data set. When making the Geni kinship verification data sets, we wish to avoid or minimise the impact of biases which make the task of kinship verification trivially easy. For example, the over representation of large families, and the use of photograph era estimation to eliminate clearly impossible kinship relations.

We are aware of the ethical implications of obtaining and potentially releasing family ancestry photos to the wider research community. Laws pertaining to copyright, data protection, GDPR, and universal personality rights will all be considered as we decide how to share this data. At the current time, the basis of study has been put through an ethics review and approved by the University of Oxford's tropical ethics review committee (ref: 511-17).

In the rest of this chapter, we describe our method to create a kinship data set using profile information mined from the Geni ancestry website. We build a bi–subject and tri–subject kinship

[1] https://www.geni.com/worldfamilytree

verification data set, taking steps to avoid common biases. Finally, we compare our data set to those published previously, and benchmark the from same photo bias signal within.

8.1 Mining Geni ancestry website

In this section, we describe an automated method to obtain raw family data from the Geni ancestry website. Our overall approach to obtain familial data is to sample from profiles on Geni by searching for profiles with particular names. We then extract familial information and images from the nuclear families (mother, father, direct children) associated with each profile into a local database. Families with sufficiently many images are identified and their respective images retrieved programmatically.

We begin by collecting a list of 998k full names taken from the MS–Celeb–1M data set (Y. Guo et al., 2016). These names broadly correspond to public figures, artists and other celebrities from around the world. Each name is individually searched using the Geni API to find profiles belonging to individuals with similar names. On average, seven profiles were found per name searched. For each profile found, we use the Geni API to check whether there is at least one image linked. If this is the case, we then find profiles of immediate family, and again check for linked images. In this way we find groups of profiles belonging to a single nuclear family, where each profile has at least one linked image.

Profile, family, and image data is collected locally in a relational database. A schematic of the database structure is shown in Figure 8.2. The database is used to store information on: the structure of nuclear families (unions); the people within those unions and their respective relationships; the images found linked to each profile; and individual image meta data. To speed up data collection, we initially only mine the minimum necessary information needed to identify a profile, images and familial links. Further information, such as gender or birth date, is collected at a later stage of processing as required. Images related to each profile are

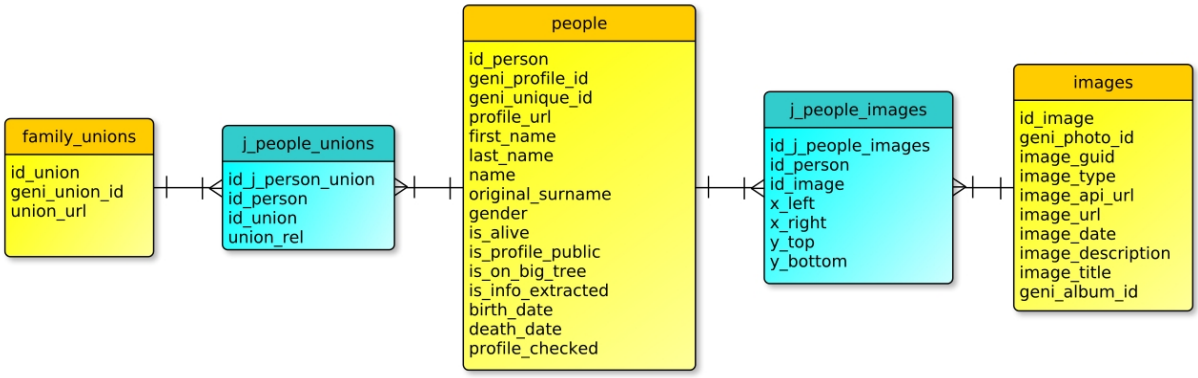


Figure 8.2: Structure of the database used to store information retrieved from the Geni website. The database allows a person to be linked to the nuclear families to which they belong, and the images stored on their Geni profile.

downloaded programmatically from a publicly accessible web server using the URLs provided by the Geni API.

There are two main barriers which impede the rapid mining of data from Geni. The first is that one is prevented from accessing any meaningful data from user profiles which are set to private. This occurs in around 42% of profiles, as some users do not wish for other members of the site to have access to their information. The second issue is that Geni's API is rate limited to 5 requests every 10 seconds. As well as restricting the speed of data collection, rate limiting also increases the likelihood of an automated solution being halted. This often occurs due to errors and exceptions which inevitably arise when querying a website for an extended period of time over an imperfect connection.

8.1.1 Results of mining

Using the list of 998k names described above, our mining tool was able to process 6.95 million individual profiles in over 75 days of computer processing time. A visualisation of the cumulative progress of downloading data from Geni is shown in Figure 8.3. In total, we found 4.04 million (58%) non–private profiles, among which 1.27 million profiles had at least one image

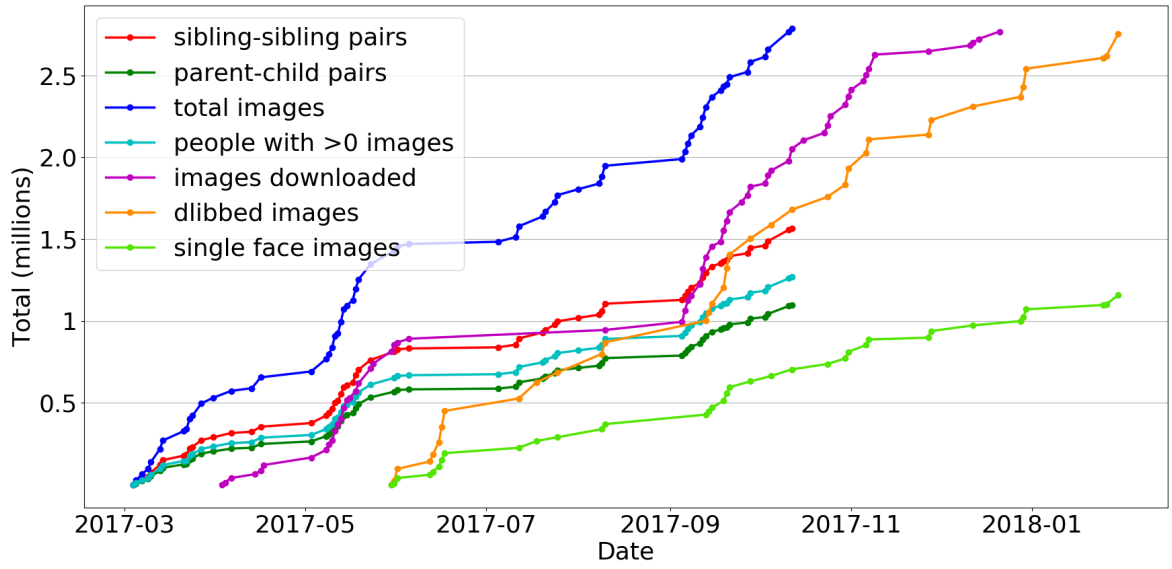


Figure 8.3: Cumulative progress of collecting Geni data. Our software searched nearly 7 million profiles over a 9 month period.

linked to them. Processing these profiles produced 314k nuclear families that contained at least two related individuals, who each have at least one image linked to their respective profiles. In total these families contained 873k children and 485k parents. At this stage, these individuals could be combined into 1.6 million potential sibling–sibling pairs and 1.1 million potential parent–child pairs.

In total, 2.76 million images (175 GB) relating to target profiles were downloaded. The process took over 27 days of computer processing time, with an average rate of 1.2 images downloaded per second. Less than 1% of images were unable to be found due to URL changes or file corruption.

8.2 Cleaning the raw Geni data

In this section, we address the first two steps required to process the raw data mined from Geni into a kinship data set. We use a face detection model to identify images containing a single

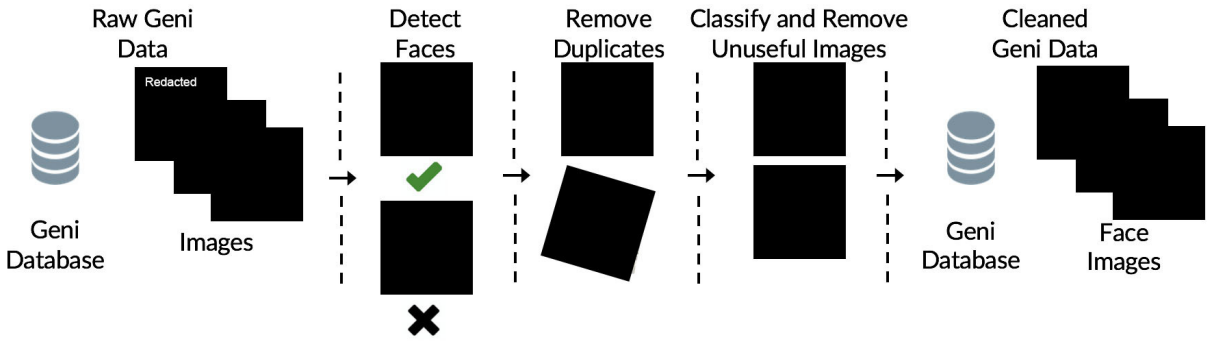


Figure 8.4: High–level overview of the processes we undertake to clean the raw Geni data, before it can be developed into a kinship data set.

face only. We then remove duplicate faces images by using a perceptual hashing algorithm. In the following section, we describe a CNN classifier used to further filter images unsuitable for use in a kinship verification data set. A schematic of the data cleaning pipeline is shown in Figure 8.4.

8.2.1 Detecting face images

Images stored on the Geni website typically have the purpose of aiding the preservation of family history. As such, there is a wide variety of different subject matters displayed in these photographs. On a single profile, one will typically find photos containing a single face, as well as one or more multi–person family photos. In addition, profiles often contain images of gravestones; schools; family homes; birth and death certificates etc.

We make the assumption that if an image contains a single face, then the face belongs to the person for whom the profile was created. In this way we can immediately exclude images with zero or many faces, and also individuals and families for whom we do not have single face images, from further processing. Note that this approach has the additional benefits of reducing the prevalence of ‘from same photo’ bias in our final data sets.

With the goal of removing non–single face images from further processing, we analyse the 2.76

Table 8.1: Number of faces detected in each of the images we have retrieved from Geni profiles. We make the assumption that single face images depict the profile owner, and therefore only use these images for the Geni kinship data sets.

Faces detected	Images found	Percentage
0	1.02M	36.9%
1	1.15M	41.6%
2	285K	10.3%
3	110K	4.0%
4	65K	2.4%
5	40K	1.4%
6–9	74K	2.7%
≥ 10	16K	0.6%
Total	2.76M	100.0%

million images obtained from Geni using an off–the–shelf face detection package (King, 2009). In total, 3.3 million faces were detected, with 1.15 million images found to contain only a single face. A summary of the results of applying face detection to Geni images is shown in Table 8.1.

8.2.2 Removing duplicate images

The second task required to clean the raw Geni data is to remove duplicate facial images. Geni profiles contain sufficiently many duplicate pairs of images that this issue must be addressed to prevent unduly skewing any evaluation statistics calculated on a contaminated kinship data set. Most commonly duplicate images of the same face are found in different resolutions.

To solve this challenge, we use a shallow perceptual hashing algorithm, *dHash*[2]. In brief, the algorithm: resizes a digital image to a standard pixel width and height; converts the resulting image to greyscale; and outputs a binary string. Each place value of the output binary string corresponds to whether a particular pixel in the standardised, greyscale image is brighter or darker than its neighbour. The similarity of two images can then be compared by using the Hamming distance between the output binary strings of each respective image.

[2] www.hackerfactor.com/blog/index.php?/archives/529-Kind-of-Like-That.html

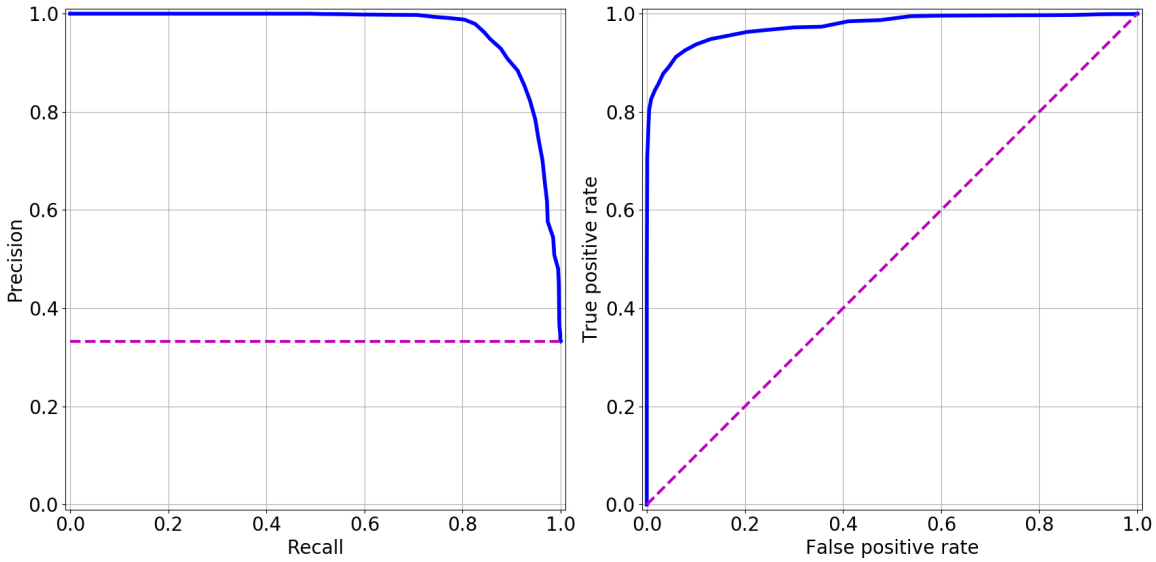


Figure 8.5: Precision–recall and ROC curves for using Hamming distance applied to dHash encodings for detecting duplicate pairs of images among a subset of the Geni data. The method achieves an average precision of 96.4% and a ROC AUC score of 97.5%, with an equal error rate of 7.8%.

We evaluate the performance of the dHash algorithm using a set of 42k pairs of images taken from the available Geni data. Manual annotation found 14k duplicate pairs of images within this group. Each image pair in the test set was passed through the dHash algorithm, and scored using the Hamming distance metric. In Figure 8.5, we plot the values of precision and recall for different threshold values using the Hamming distance as a classification score. For duplicate detection, we found that the dHash method had an average precision of 96.4%, which is sufficiently high for our purposes. An operating point for the classifier was chosen based upon the equal error rate of the classifier, giving a precision of 92.6% and a recall of 85.4%.

Confident with the performance of our duplicate detection method, we apply the dHash algorithm across the entire Geni data set, testing pairs of single face images belonging to the same profile. We found that from among the 1 million face images included in testing, the dHash classifier identified 242k (23%) as duplicates of another image using our set operating threshold. We collate single face duplicates, taking only the largest resolution version from each set of single face images for further processing.



Figure 8.6: Examples of unuseful images found during processing of the Geni data. These images would not be suitable for use in a facial kinship data set. Examples include artistic impressions of faces, pieces of text and maps.

8.3 Classifying image usefulness

To further clean the images mined from Geni, false positive errors produced by our face detection algorithm must be removed. Furthermore, face images which are generally unsuitable for use in a kinship verification data set must be excluded. Typical examples include artistic impressions of a face such as paintings, drawing or sculptures as shown in Figure 8.6. Images containing many small curved lines such as writing, scripts and maps are also particularly common among false positive errors among produced by the face detection algorithm. Face images must also be excluded if no meaningful kinship signal could possibly be gleaned from them. Typically these images are blurry; display an extreme facial pose; are partially occluded; or have been taken from a long distance away.

Filtering images according to the above criteria is a complex visual task. We define ‘useful’ images in this context as any face image which would be suitable for use in a kinship data set; designating images fulfilling the above criteria as ‘unuseful’. In this section we describe our

	Training (70%)	**Validation (10%)**	**Testing (20%)**	**Total**
Useful (50%)	5561	793	1594	7948
Not useful (50%)	5561	793	1594	7948
Total	11122	1586	3188	15896

Table 8.2: Size of data set used to train, validate and test the usefulness classifier. This classifier is used to remove images which were retrieved from Geni and found to contain a face, but would not be suitable for use in a kinship data set. Examples of such images ae shown in Figure 8.6.

method to retrieve useful single faces by training a CNN classifier, using manually annotated Geni images.

8.3.1 Training a classifier to detect usefulness

To train the usefulness classifier, we take a subset of images from the Geni data set and manually label them as either suitable for use in a facial kinship data set or not. To help prevent the model from learning a bias towards a particular class, these images are subsampled to produce a balanced data set. The images are split into training (70%), validation (10%) and testing (20%) sets. The number of images used in each subset is given in Table 8.2.

The architecture of the usefulness classifier is based upon a VGG–M network which has been pre–trained on ImageNet (Chatfield et al., 2014). The final layer of the network is replaced by a fully connected layer with two outputs: corresponding to scores for 'useful' and 'not useful' classes. The network is trained on our usefulness data set in an end–to–end fashion using stochastic gradient descent and a cross–entropy loss function.

8.3.2 Testing and deploying the usefulness classifier

We test the usefulness classifier on a held–out test set of 3188 manually ground truthed images taken from the Geni data set. We see that the classifier performs with an average precision of 90.2% and an equal error rate of 16.4%. In Figure 8.7, we show the precision–recall (PR) and

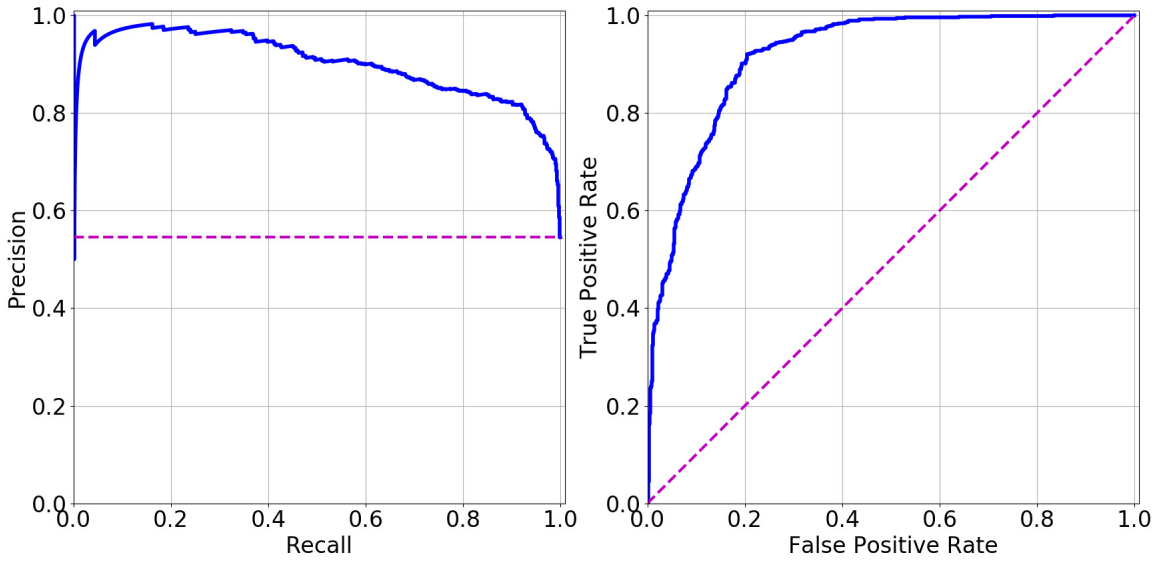


Figure 8.7: PR and ROC curves from testing the usefulness classifier on a set of 3200 images from an annotated subset of the Geni data. The classifier performs with an average precision of 90.2% and an equal error rate of 16.4%.

receiver operating characteristic (ROC) curves of the classifier evaluated on the usefulness test set.

For evaluation on the full Geni data set, we choose an operating threshold for the classifier of 0.478, giving the classifier an operating precision and recall of 84% respectively. This level is performance provides sufficient accuracy for the task to removing unsuitable images from the Geni data set. We apply the trained usefulness classifier to all images collected in the previous steps. In total 850k (74%) single face images were classified as being of sufficient quality by the usefulness classifier. Examples of images found to be useful in the full Geni data set are shown in Figure 8.8.

8.4 Managing data set bias

In creating the Geni kinship data sets we wish to avoid incorporating several common biases which undermine the task of kinship verification for computer vision models. In Chapter 7, we



Figure 8.8: Examples of images found to be useful using the classifier. Note that despite the variation in image era, these are all clear facial images, free from extreme poses, occlusion or blur.

described how using images cropped from the same family photo in kinship data sets makes the task much easier. Therefore, in the Geni data set, we attempt to avoid this bias by using single–face profile images only. In this section, we discuss four other examples of biases which make the task of kinship verification easier, and the steps that can be taken to minimise their impact.

8.4.1 Identity recognition

A key bias used by human annotators for the task of kinship verification is to recognise the people shown in a photograph and remember that two or three people are related. For example, many of the people shown in Figure 7.1 are sufficiently famous, that one can recognise individual people from their photograph, and use learned knowledge exterior to the task to confirm that the two individuals are related (Lopez et al., 2018). When training machine learning models for the task of kinship verification, this bias can be counteracted by ensuring members of a fam-

ily appear only in one of the training, validation or testing subsets. Furthermore, within these subsets, members from a family should be included in both the positive and negative examples. This helps to prevent models from simply remembering the faces of individuals, and learning that whenever one or more specific individuals appear within a pair or triple, that it must belong to the positive or negative class.

8.4.2 Large family thresholding

A second bias we wish to avoid is an imbalance of the number of kinship pairs or triples produced by different families. Naturally some nuclear families will contain more children than others. Furthermore, some families, such as royalty, are photographed appreciably more than average. The number of kinship pairs or triples of images that can be produced from these families grows combinatorially, and if left unchecked, may come to represent a substantial proportion of the overall data set.

To prevent this family imbalance within the Geni kinship data set, we catalogue the number of image pairs or triples produced by each respective relationship pair or MFC triple. The percentage of relationship pairs or triples which have less than a given number of associated image pairs or triples. We find that just under 70% of relationship pairs produce only a single image pair, and 90% of relationship pairs produce three or fewer image pairs. We choose to set the threshold at a maximum of six images per relationship pair or triple. This allows us to keep over 95% of the image pairs and 92% of triples produced in our data set, whilst avoiding large family saturation.

8.4.3 Photographic era and sex normalisation

Other biases present in kinship verification data sets often arise from the manner in which the negative set of image pairs or triples is produced. Unrelated groups of individuals are easily

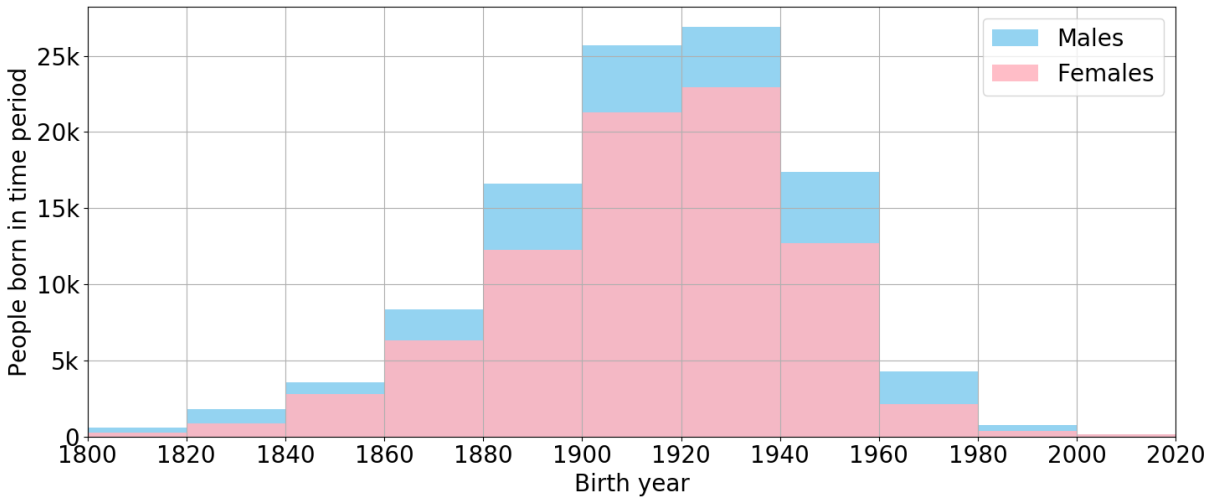


Figure 8.9: Distribution of birth year by sex for individuals used in the Geni pairs data set.

produced in kinship data sets by choosing two or three members from separate nuclear families. However, naively creating the negative set in this way produces many examples of pairs or triples of people who clearly do not represent the tested kinship relationship.

As an example, consider the situation where one trains a specific model for kinship verification of mother–daughter relationships. Using any images of males in the negative training set is likely to lead to this signal being learnt by a classifier, as determining sex is visually simpler than determining kinship.

A further bias that can be introduced by naive creation of the negative kinship set is caused by the use of historical ancestry data sets, where images are taken from across a large span of time. For example, in the case of classifying mother–father–child triples, if the two images of the parent have been taken with a modern colour camera, whereas the image of the child is an older, lower quality, greyscale photograph. A combination of such images will never be seen in the positive set, and so gives a clear indication to the classifier that the triple is from the negative class.

Our approach to prevent these biases from contaminating the Geni data sets is to match the distribution of photographic era and sex in our negative sets. To achieve this, we first query

the birth year and sex of each relevant individual from Geni. Only 20% of people were found to not have a birth year recorded, and 0.02% did not have a sex recorded. The distribution of people in the Geni data set by age and birth year is shown in Figure 8.9. When creating the negative data sets we sample from the same gender as the individual being exchanged – female for female, male for male. Furthermore, we also sample from among people who were born within ten years of the person being replaced.

8.5 Geni pairs and triples kinship data sets

We use the images processed to create two kinship verification data sets. The first is a bi–subject (pairs) data set for the task of one–to–one kinship verification. The second data set is a tri–subject (triples) data set, where the goal is to determine whether a particular facial image represents the child of a given mother and father.

To make the tri–subject data set, we first find all relationship triples of mother, father and child. The Geni MFC triples data set consists of over 116k individual people from over 25k nuclear families. It contains roughly 55k positive triplets of people and 110k positive image triples. To create the negative data sets, for each image in the positive set, we assign a different child to the triple, respecting the bias considerations outlined in Section 8.4. We keep the order of mother, father and child the same within triples, to allow the network to specialise the features coming from each position within a family. Examples of triples from the data set are shown in Figure 8.10

To produce the bi–subject data set, we take all usable pairs of 206k parent–child and 204k sibling–sibling relationships processed thus far. The Geni pairs data set consists of over 270k individual people. It contains 411k positive pairs of people and 269k positive image pairs. To create the negative pairs set, we exchange individuals randomly using the bias considerations described in Section 8.4. Examples of pairs from the data set are shown in Figure 8.11.

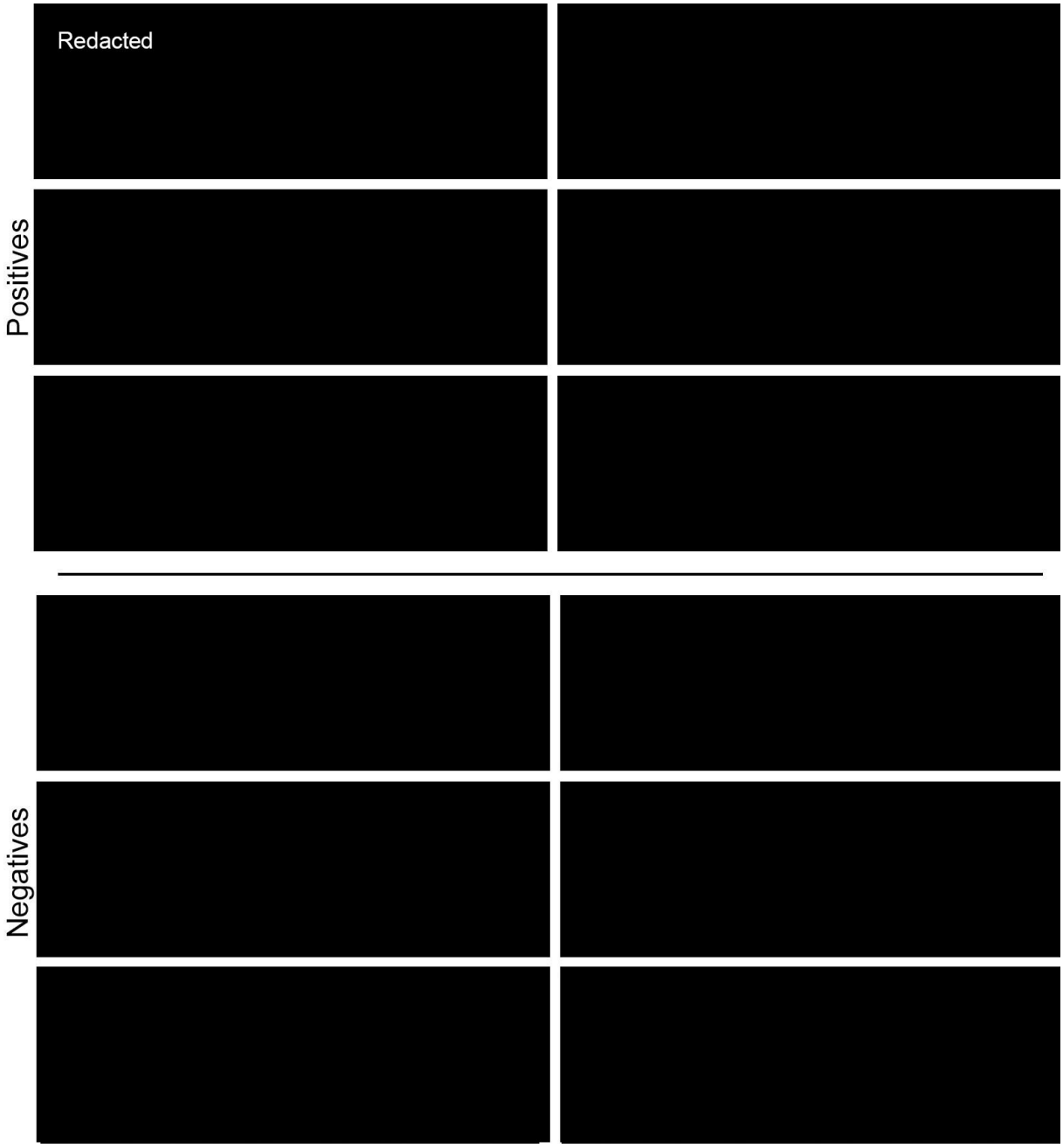


Figure 8.10: Examples of positive triples from the Geni tri–subject mother–father–child data set. It is clear that classifying kinship from these images is a visually difficult task.

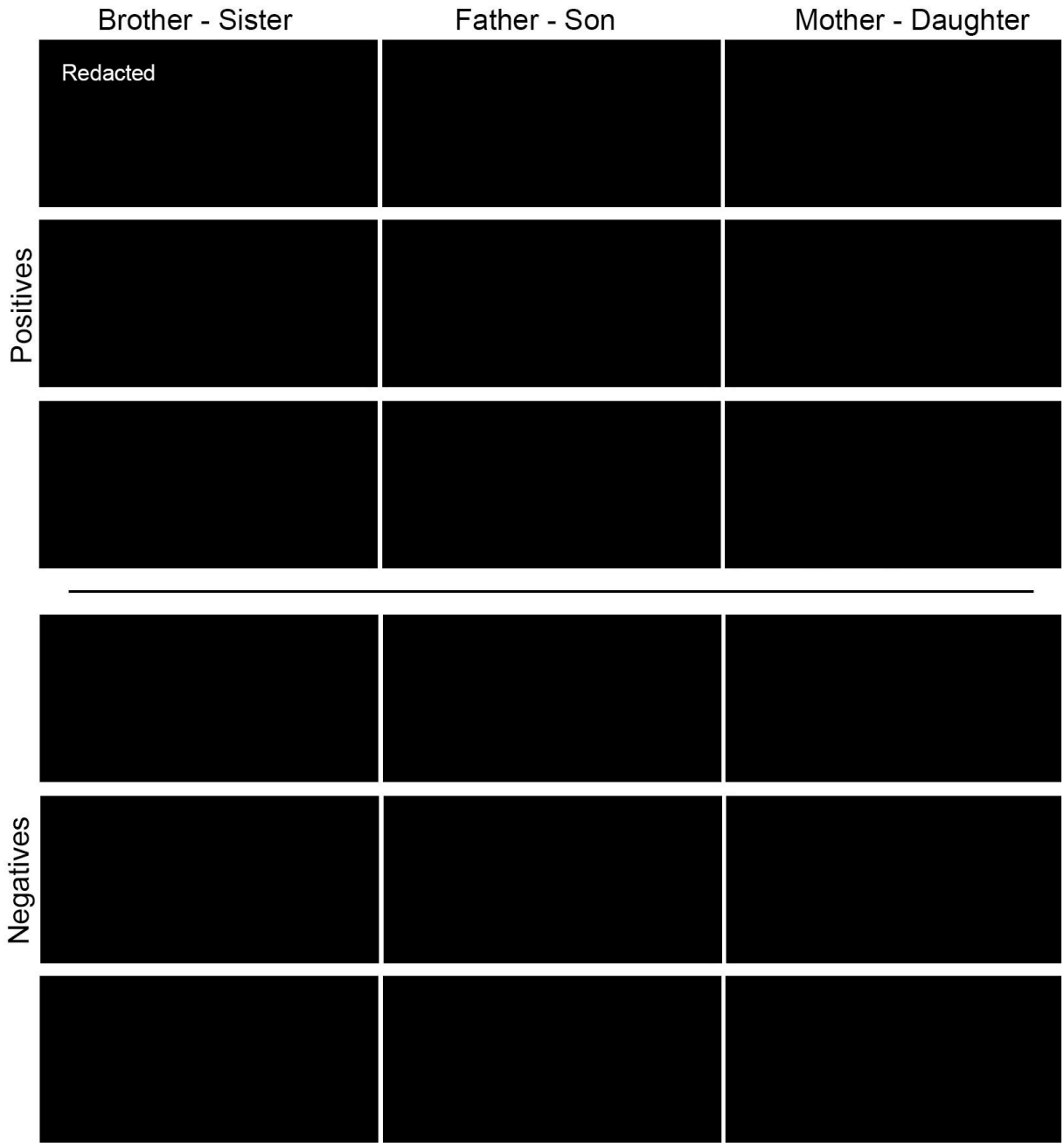


Figure 8.11: Examples of positive pairs from the Geni bi–subject kinship data set. Parents are shown on the left and their children on the right for each pair. Again, confirming facial kinship on this data set is difficult as simplifying biases have been removed.

Table 8.3: Size comparison of the Geni pairs and mother–father–child triples data sets to other previously published kinship data sets. Note that the Geni triples data set is over 150 times larger than the next largest MFC kinship data set.

Data set	**Pairs**		**Triples**
	Parent–Child	Siblings	Mother–Father–Child
Cornell KinFace	150	-	-
KinFaceW–I	533	-	-
KinFaceW–II	1000	-	-
TSKinFace	-	-	1015
FIW	169k	247k	-
Geni (ours)	295k	269k	153k

8.5.1 Comparison to other kinship data sets

We compare the size of the Geni bi–subject and tri–subject data sets to those previously published in Table 8.3. As shown, the Geni tri–subject data set contains over 150 times as many image triples as TSKinFace, the largest previously published tri–subject data set. Furthermore, the Geni bi–subject data set contains 74% more parent–child image pairs, and 9% more sibling–sibling image pairs as the largest previously published bi–subject kinship data set, Families In The Wild.

8.5.2 From same photo signal in Geni data sets

We assess the extent to which the 'from same photo' signal bias, discussed in Chapter 7, is present in both our created data sets. Although we attempt to minimise the presence of this bias, by using only single face images from profiles, Geni users may crop individual images from a larger group photo, and link the individual photos to separate profiles. Currently, rather than choosing to explicitly remove these examples from our data set, we use the FSP classifier, outlined in Chapter 7, in to benchmark the extent to which the FSP signal is present.

We run the FSP classifier on the Geni pairs and Geni triplets data set naively as a kinship classifier. For triples, we use the same testing protocol as described for TSKinFace in Chapter 7.

Table 8.4: Comparison of accuracy achieved on previously published kinship data sets using the FSP classifier developed in Chapter 7. As can be seen the Geni data set has a very low FSP signal.

Data set	Cornell KF	KinFaceW–I	KinFaceW–II	FIW	TSKinFace	Geni (ours)
Pairs	76.7	76.8	90.2	58.6	-	55.3
Triples	-	-	-	-	88.6	58.8

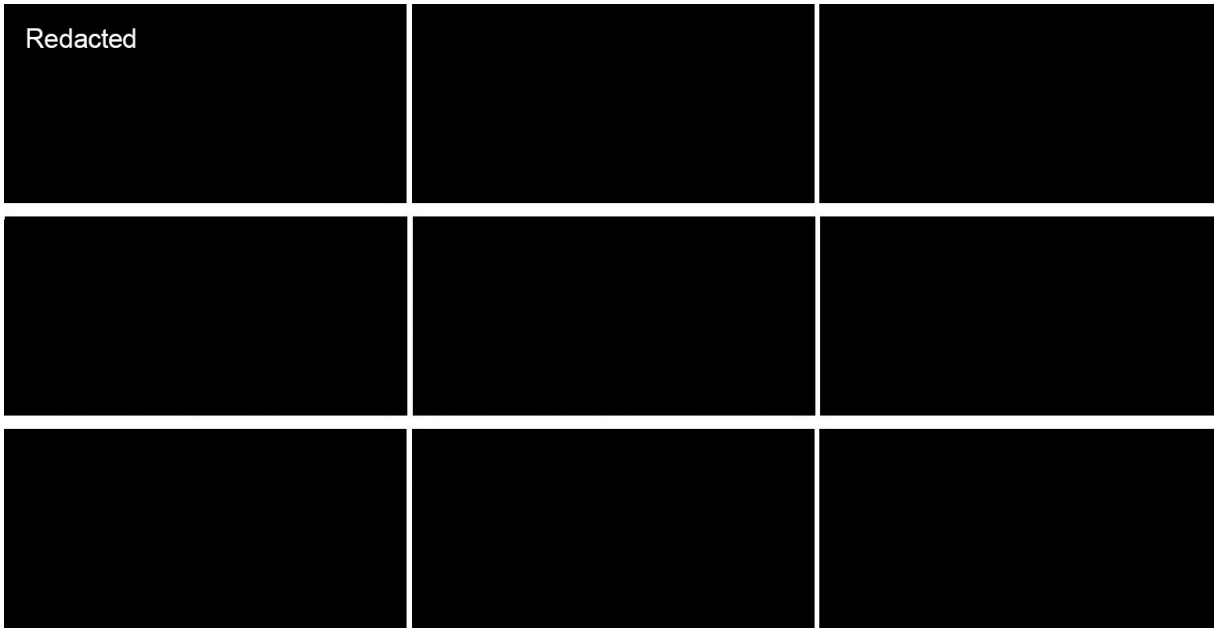


Figure 8.12: Examples of pairs of kin faces cropped from the same photo that were found in the FSP–included Geni pairs data set. We remove FSP pairs and triples from our final data sets.

We find that the FSP classifier achieves an average precision of 55.3% on the Geni pairs data set, and 58.8% on the Geni triples data set. Examples of pairs of images from the Geni data set classified as being from the same photo are shown in Figure 8.12. These results are comparable to previously largest published kinship data set, Families in the Wild. In Table 8.4, we compare the strength of the FSP signal in the Geni data set with the FSP signal present in other major kinship data sets.

8.6 Summary

In this chapter, we have described a method for obtaining and processing data from the Geni ancestry website to produce the largest bi–subject and tri–subject kinship verification data set published to date. Our method searches the Geni website for profiles with names similar to those of public figures, mining these profiles for image data and familial relations. We have also described how to process the raw Geni data to remove duplicates and images non–single face images. We also outline our method of building a CNN classifier to remove images unsuitable for use in a kinship data set. Finally, we build our pairs and triples data sets carefully to avoid common kinship data set biases, such as unrealistic negative examples.

9 | NetVLAD feature aggregation for facial kinship verification

As described in Chapter 6, kinship verification tasks come in three main forms. One–to–one verification, tri–subject (mother–father–child) verification and family classification. In this chapter we consider the second of these. Previous computer vision models for tri–subject verification have attempted to extract and combine individual's facial features before classifying them.

The second half of such an approach crucially relies upon methods capable of aggregating discriminative features in a meaningful way. For example, a child may share more facial characteristics with one parent than another, and so simple averaging will likely be ineffective. Furthermore, such a method must ideally also combine features efficiently: particularly in the case of family classification, where nuclear families can contain many children.

One of the canonical forms of feature aggregation used in a machine learning context is the bag–of–words model used for language encoding. The method represents passages of text using the frequency with which words from a set vocabulary appear. The bag–of–words model was adapted for use in computer vision by Sivic & Zisserman (2003); in which images are treated analogously to documents, with local image patch features replacing words. A vocabulary of visual words can be compiled by applying clustering algorithms to the feature patches found within a data set of images. Whole images can then be compared using the frequency of their respective image patch features.

As image data sets have grown larger, and more complex image features developed, memory constraints have become a limiting factor in search and retrieval systems. To address this issue, Jegou et al. (2011) developed a new image representation termed the 'vector of locally aggregated descriptors' or VLAD. This method of feature aggregation was inspired by the bag–of–

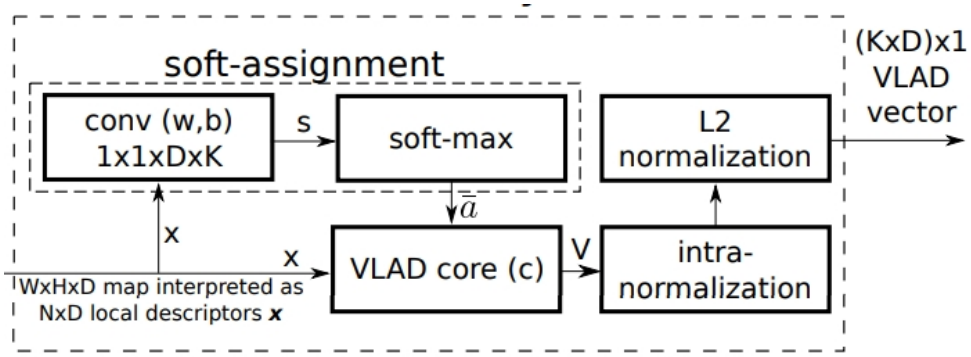


Figure 9.1: The NetVLAD feature aggregation module. Our experiments use 8 cluster centres (K), and an image descriptor size dimension (D) of 128. Figure reproduced from Arandjelović et al. (2017)

visual features approach, and similarly uses clustering to identify an image feature vocabulary. Rather than simply counting the frequency with which each feature appears, VLAD accumulates the signed distances between each feature found in an image, and its respective nearest image feature patch in the learned vocabulary. This allows for much more expressive features to be learned: VLAD outperforms bag of words for same size image representation (Jegou et al., 2011). Crucially, the length of VLAD representations can be reduced using principal component analysis (PCA) to a few hundred dimensions without impact accuracy noticeably. Subsequent improvements were made to VLAD by Arandjelović & Zisserman (2013), including intra–class normalisation. This normalisation addressed the problem of repeating patterns in an image causing a few large components in the VLAD vector to dominate, thereby affecting similarity calculations.

NetVLAD was introduced by Arandjelović et al. (2017), as a deep learning adaption of VLAD. NetVLAD uses standard CNN components and can be used to aggregate features from standard CNN architectures to produce a fixed size representation for images. NetVLAD weights feature channels for each image by performing cluster assignment using a softmax classifier. This means that the network can be trained end–to–end using deep learning methods. An overview of the NetVLAD module, reproduced from Arandjelović et al. (2017), is shown in Figure 9.1.

NetVLAD was originally deployed, and continues to be used for, the task of place recognition (Arandjelović et al., 2017; Yu et al., 2019). However, the ability of NetVLAD to aggregate visual components into a fixed length representation rapidly, while retaining individual features, makes it a very widely applicable method. Other implementations of NetVLAD have been used to pool features for large–scale video classification (Miech et al., 2017; S. Chen et al., 2017; Tang et al., 2018); as well as being used for crowd counting tasks, where features belonging to individual people must be accumulated (Shi et al., 2018).

In this chapter, we take the first steps towards applying a generalisable deep feature aggregation method for facial kinship. We adapt a NetVLAD architecture, which has been previously trained for multi–image face recognition, to the task of mother–father–child (MFC) kinship verification.

Significantly, NetVLAD has two characteristics which make it particularly suitable for this task. Firstly, the output feature vector produced by NetVLAD aggregation does not change size, regardless of how many input images are used. Secondly, NetVLAD aggregates features from different inputs in the same way. This means that classifying a NetVLAD representation of a mother–father–child triple asks whether three images form a valid mother–father–child triple, rather than simply determining whether a person is the child of two parents.

In the rest of this chapter, we describe our two–stage method for implementing the NetVLAD architecture for the task of tri–subject kinship verification. We first train the NetVLAD module for the task of facial recognition, and show that the module is able to preserve facial identity information. We then train the NetVLAD aggregator using the Geni MFC triplets data set described in Chapter 8. Finally, we discuss experiments of different training regimes, and show that NetVLAD is able to achieve superior performance when compared with a linear aggregation network architecture. We must stress that our results are preliminary and will be further developed over the coming months.

Table 9.1: Size of the adapted VGGFace2–triples data set used to pre–train the NetVLAD module for tri–image facial verification.

Data set	Identities	Training	Validation	Testing	Total image triples
VGGFace2 (triples)	8631	2.81M	147k	172k	3.13M

9.1 Pre–training NetVLAD for facial recognition

In this section we describe our approach for training a tri–image aggregation network using NetVLAD. Our overall approach is to append a NetVLAD module to a pretrained face verification network, then train the module using image triples adapted from the VGGFace2 data set (Cao et al., 2018). The performance we achieve confirms that NetVLAD is able to aggregate discriminative facial features from image triples. Moreover, we show that using multiple faces and NetVLAD for the task of face verification appears to improve classification accuracy compared with using only single faces.

The original VGGFace2 data set (see Figure 9.2) contains over 3.3 million single face images of over 9000 public figures. We adapt the VGGFace2 data set by creating the VGGFace2–triples data set. This consists of triples of facial images belonging to the same identity. We avoid information leakage in this new data set by splitting each individual image into separate sets for training (90%), validation (5%) and testing (5%) subsets. We then find unique combinations of image triples for each individual within these subsets. In total we make 3.13 million triples for 8631 identities, making the data set of a comparable size to the original VGGFace2. A description of the VGGFace2 triples data set is given in Table 9.1.

We choose to base our classifier upon the VGGFace2 facial recognition networks originally produced by Cao et al. (2018). In particular, we use the versions built using a squeeze and excitation version of ResNet–50 introduced by Hu et al. (2018). This base network has been



Figure 9.2: Examples of facial images for three identities from the VGGFace2 data set. Images from this data set are used to pre–train the NetVLAD module for the task of tri–image face recognition. VGGFace2 data set contains over 3.3 million face images for over 9k individuals.

re–implemented from the original in the PyTorch development library[1] and includes an extra fully–connected layer to reduce the size of the output feature vector down from the original 2048 to 128. Inspired by Zhong et al. (2018), we append an 8–node NetVLAD aggregation module and four more fully connected, batch normalisation and normalisation layers as shown in Figure 9.3.

Our training protocol is also heavily directed by the approach taken by Zhong et al. (2018). We initialise the NetVLAD module and subsequent fully–connected layer weights using K–means clustering and principal component analysis, applied to 10% of the original VGGFace2 data set. This approach appears to speed up convergence. In training, we freeze the weights in the base SENet and train the remaining network using a one–vs–hardest–20 logistic loss function. We use a two–tier learning rate scheme, with a learning rate of 0.1 for the weights and biases in the NetVLAD module's soft assignment convolutional layer, as well as for the final fully connected

[1] github.com/ox-vgg/vgg_face2

classification layer. However, we use a much lower learning rate for the weights determining the NetVLAD centroids and fully connected compression layer directly after NetVLAD, as we expect these weights to already be well suited to the task, given their initialisation procedure. Furthermore, we use several types of data augmentation as suggested by Zhong et al. (2018), including: horizontal flip; colour jitter; image rotation up to ten degrees; and resizing and random cropping. The performance of our network converged rapidly: within one epoch of training on the full VGGFace2 triples data set.

9.1.1 Results of pre–training

We test our NetVLAD face recognition network using a held–out test set of 172k triples of images from the VGGFace2 data set described above. We find the network achieves an average precision of 99.98%, with an equal error rate of 0.15%. This level of accuracy is much higher than was originally reported by the base SENet classifier. When tested on the original, single–face VGGFace2 data set, the base SENet was found to have an average precision of 99.76% with an equal error rate of 0.76%.

Given that the number of training examples is similar in both the VGGFace2 and our VGGFace2 triples data set, we suggest that the task of classification is made easier by the aggregation of information from multiple face images. These results would indicate that the NetVLAD module is able aggregate facial feature descriptors in a discriminative manner. We have repeated these experiments using the 256–dimensional version of SENet as a base network, and found no noticeable change in performance.

9.2 Training NetVLAD for kinship feature aggregation

We use the learned weights from the tri–image facial verification task as a base for training our NetVLAD architecture for tri–subject kinship verification. We adapt the final layer of the

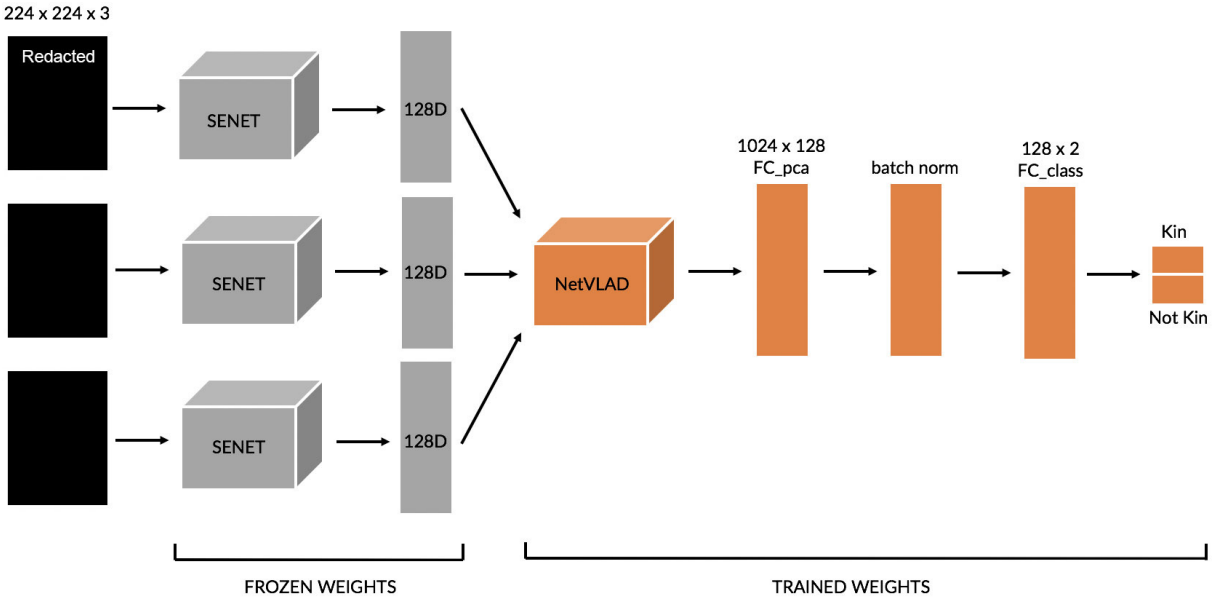


Figure 9.3: Architecture of the NetVLAD feature aggregation network we use for tri–subject kinship verification.

architecture replacing the 128×8631 dimensional classification layer, previously used for face recognition, with a 128×2 fully connected layer; where each output corresponds to scores for 'kin' or 'not–kin'.

To test the extent to which 'from same photo' affects the performance of a network trained on the task, we train our network both using the full Geni MFC data set, and a slightly smaller version; where we have removed all positive triples which were classified correctly using an FSP classifier. To balance the number of positive and negative triples we then randomly sample an equivalent number of triples from the negative data set. A description of the data sets used in training is given in Table 9.2.

Table 9.2: Description of the size of the Geni MFC data sets used to train our NetVLAD classifier. We use two versions of the data set, both with and without the FSP signal removed.

Data set	**Training**	**Validation**	**Testing**	**Total MFC triples**
Geni MFC (FSP not removed)	123k	17k	36k	176k
Geni MFC (FSP removed	107k	15k	31k	153k

To train our networks, we use a similar protocol to that described in Section 9.1: freezing the base network, and training the NetVLAD module with a much greater learning rate for both the NetVLAD assignment layer and final classification layer. However, we now use a binary cross entropy loss function during training, which lasts for 50 epochs with an exponentially decreasing learning rate.

For comparison, we also train a simpler linear aggregation CNN architecture on the Geni MFC data set. This network consists of three frozen copies of the off–the–shelf base SENet–128 trained on the original VGGFace2 described above, on to which we append a further fully connected layer of size 384×2. During training of the linear aggregation network, we pass each image from the MFC triple through the base networks and stack the three resulting 128–dimensional features vectors. This stacked feature vector is then passed through the final linear layer to produce a kinship prediction score. This network is trained using the same data set, augmentation methods and learning rate scheduling as the NetVLAD architecture.

9.3 Testing on Geni MFC triplets

We test the trained NetVLAD architecture on a held–out test set of 36k image triples from the Geni MFC data set. As shown in Figure 9.4, neither our NetVLAD network nor the linear aggregation network is able to show good performance on the Geni MFC data set. Our best performing network achieves an average precision of only 53.3%, and an equal error rate of over 47% on the test set. These results are very low compared with those reported on other published kinship verification sets and are only slightly better than random chance.

In Table 9.3, we see a comparison between the results of training and testing both architectures on versions of the Geni MFC data set where FSP bias has been explicitly removed or left in. We see that for both architectures, removing from FSP bias from the training and testing sets

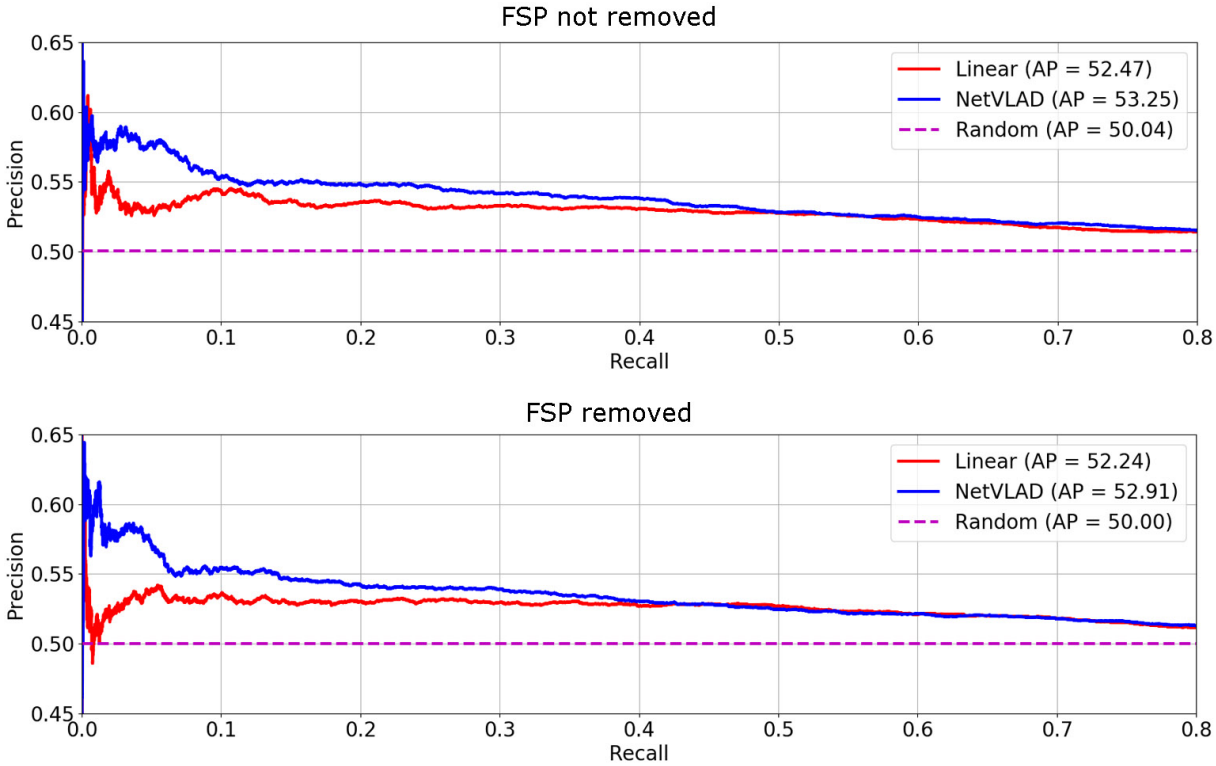


Figure 9.4: PR and ROC curves of the NetVLAD and linear aggregation networks applied to two versions of the Geni MFC data set. Results are shown for a data set with FSP signal explicitly removed (top), and with the signal left unremoved (bottom). We see that both networks struggle to classify kinship. The NetVLAD architecture does perform slightly better than a linear aggregation method. Removing FSP signal from our data sets appears to make the task slightly harder.

causes only a small decrease in accuracy. This is likely due to the easiest triples being removed from the data set, thereby making the overall task more difficult.

In both experiments where we have included and removed the FSP bias, the NetVLAD architecture is able to achieve a higher average precision than a simple linear aggregation model. However, for all experiments it is difficult to draw conclusions on performance, as results are very close to random guessing. In the future we will conduct further experiments to ascertain whether these results represent a failure of the aggregation method or, the underlying difficulty of the kinship verification task, when data set biases are limited.

Table 9.3: Results of testing the NetVLAD and linear aggregation networks on the tri–subject Geni MFC kinship verification data set. Removing FSP signal from the data sets appears to make the task more difficult.

Aggregation method	**FSP signal**	**Accuracy (%)**	**AP (%)**	**EER (%)**
Linear	Removed	52.3	52.2	47.5
Linear	Not removed	52.6	52.5	47.3
NetVLAD	Removed	52.5	52.9	47.7
NetVLAD	Not removed	52.7	53.3	47.2

9.4 Summary

In this chapter we have implemented a NetVLAD feature aggregation network for the task of tri–subject (mother–father–child) kinship verification. We originally trained the NetVLAD module for the task of tri–image face recognition; and used transfer learning to re–train the feature aggregation network using the Geni MFC data set described in Chapter 8. For comparison we trained a linear feature aggregation network in an equivalent manner.

Our results show that both our current approaches struggle to classify kinship from triples in the Geni data set. However, the NetVLAD architecture does achieve slightly better performance. We find that removing the FSP signal from our data sets makes the task only slightly more difficult.

This chapter represents the first attempt to classify kinship using the Geni MFC data set. The data set was designed specifically to be free from biases which make the task of kinship verification easier. The performance of our evaluated methods is much lower than results achieved previously on other kinship data sets. We expect that this may be due to our implementation methodology, as well as the comparative lack of undermining biases, as well as the greater diversity of images, found within the Geni data set.

10 | Summary

In this thesis we have developed tools and data sets in the field of computational syndrome classification from ordinary patient photographs. Our approach has been to obtain a new data set for the task, as well as conduct research in the field of kinship modelling.

In this chapter we summarise the main contributions of this thesis. We discuss the impact both for the goal of computational syndrome diagnosis, as well as for the field of kinship verification. Finally, we discuss work which can be completed in the future to extend the results that have been presented.

10.1 Achievements

Research article download pipeline. In chapter 1, we developed a download pipeline for the identification and retrieval of biomedical research images of interest from the PubMed database. We deployed this pipeline for the task of retrieving facial images of patients with rare genetic disorders and analysed over 40k articles. We were able to mine articles from the four main publishers of identified articles of interest. The syndrome face images collected from this pipeline are currently being used to train the next generation of syndrome classification models.

Biomedical figure classifier. To support our retrieval pipeline, we developed a CNN classifier capable of determining whether a biomedical figure contains an image type of interest. We demonstrated that it is possible to train such a classifier using only synthetically produced images designed to look like images from the target distribution. We showed that although the classifier is trained using synthetic images, it is able to adapt to real biomedical figures with very high precision and recall. Crucially, this approach is readily adaptable to retrieve images from other categories of interest.

Syndrome annotation tool. Labelling patient faces retrieved from biomedical articles allows them to be used to train supervised syndrome classification algorithms. In Chapter 5 we developed a platform which successfully facilitates the rapid syndrome annotation of patient face images from genetic disorder articles. The tool has been successfully used by five annotators over the course of 14 weeks, to annotate over 3.5k facial images of patients with over 242 disorders.

Benchmarking bias in current kinship data sets. In Chapter 7, we recognise one of the key biases in current kinship verification data sets: that positive kin pairs are often cropped from the same original family photo. We demonstrate that a CNN classifier, trained using groups of unrelated individuals, and with no knowledge of kinship, is able to perform among many of the state–of–the–art methods on several kinship data sets. This result should provide a warning to the kinship community that their models may not be learning the signals they expect. We suggest the FSP classifier should be an integral part of the building of future kinship data sets.

Development of two novel kinship data sets. In Chapter 8, we address two of the major issues common among kinship data sets; namely their small size, and the presence of underlying biases which are easier to learn than the kinship signal. We built an automated processing pipeline to extract two of the largest kinship data sets produced to date from the ancestry website Geni. We have made attempts to remove as much of underlying bias as possible using large family thresholding, from same photo bias, and sex and age normalisation.

Exploration of feature aggregation for kinship verification. Finally, we presented an application of the NetVLAD feature aggregation network to the task of kinship verification. As we cannot confirm that our model would not simply learn the biases inherent in other kinship data sets, we test on two versions of the Geni MFC triples. We undertook preliminary experiments on our new data set, and showed that NetVLAD is able to understand kinship at least as well as a network using linear aggregation; and also that removing the FSP signal from a kinship data set makes the task noticeably harder.

10.2 Future work

In this section, we discuss several directions in which the work presented in this thesis could potentially be extended in the future. Going forward, our objective is to utilise the syndrome–labelled patient faces obtained using the PubMed download pipeline, in addition to those collected from other sources, to train the next generation of syndrome recognition algorithms. The work completed in the second half of this thesis will support this objective, as effective modelling of facial kinship will enable clearer delineation and attribution of the craniofacial characteristics caused by an underlying disorder. It is hoped that this work will lead to syndrome classification algorithms which are invariant, or blind, to the kinship signal at the feature representation level (Alvi et al., 2018; Glastonbury et al., 2018).

10.2.1 Mining syndrome faces

Deployment of the download pipeline. A clear direction for improvements to be made to the research article download pipeline would be to reduce the amount of data which is currently out of scope for the mining pipeline. For example, a non–trivial proportion of articles are held on publisher's websites that we have not yet developed automated mining tools for. Furthermore, we were unable to develop a method for targeting articles with no assigned DOI number. Addressing these two issues would greatly increase the number of patient faces we are able to retrieve. Going forward, we expect the download pipeline to be run periodically: thereby capturing patient faces from the most recently published genetic disorder articles.

Scaling and automating annotation. Currently, we use human annotation to solve the problem of locating and matching face images, patient identifiers, and syndrome diagnoses in the text of published case studies. There are several ways in which we could increase the rate at which faces can be annotated. Importantly, we are aware that syndrome face annotation, particularly over an extended period of time, is an emotionally difficult task. This motivates the need to

employ more syndrome annotators, either through internal means, or potentially by opening the task up externally (for example, to Mechanical Turk annotators). In the future, we hope to develop automated methods to retrieve patient information. This goal will naturally become more viable as the current annotators provide us with supervision labels for the next generation of algorithms. Particular sub–problems which may be readily solvable using computer vision and NLP methods include: linking patient identifier and diagnosis; and linking figure subpanels with their respective caption identifier.

Retrieving genotypic and phenotypic information. Currently, we do not ask annotators to record a genetic description of a patient's mutation even though this information exists in many of the case studies we retrieve. We have found that asking annotators for this information significantly slows down the overall task, particularly for an untrained individual. In addition, we have found that case studies often include phenotypic descriptions of patients. Extracting this information sensibly may be possible, as the human phenotype ontology (HPO) project provides a comprehensive naming system of human phenotype characteristics (Köhler et al., 2013). Furthermore, methods do exist to automatically extract and normalise phenotypic descriptions into HPO terms (Groza et al., 2015). Obtaining this information could help to provide a greater understanding of the relationship between a specific genetic mutation and the subsequent phenotypic manifestations associated with a disorder.

10.2.2 Facial kinship

Expanding the Geni data set. In this thesis, we have collected information from nearly 7 million Geni profiles. This represents only 5% of the Geni's 'big tree' which contains over 135 million profiles. Our efforts were restricted both by time constraints, and the explicit rate limiting of the Geni API. Obtaining a larger data set which includes information from the remaining Geni profiles may lead to the development of more generalisable kinship models.

Incorporating multiple face images. In creating the Geni data set, we chose to ignore images which contained multiple faces: making the assumption that single face images depict the profile owner. In the future, we hope to develop a facial verification solution which could be used to identify all facial images of the profile owner. Any 'from same photo' signal that is subsequently introduced into the data set from using images cropped from family photos can then be removed by our FSP classifier. Of the 2.76 million images we tested in creating the Geni data sets, over 21% were found to contain two or more face images. Processing and incorporating these valuable images will further expand the size and variation of the Geni data sets.

Further cleaning of the Geni data sets. The sheer scale of the Geni data set has meant that it has not been possible within the scope of this work to check every single kinship pair or triple. It is clear from our results in Chapter 8 that our methods of detecting faces, removing duplicates and classifying usefulness work well, but not perfectly. More complex state–of–the–art algorithms could be deployed for each of these tasks. Furthermore, visualisation of classification results achieved on these data sets will help us to identify remaining inappropriate images, pairs and triples. We accept that the Geni data sets may never contain perfectly accurate kinship labels due to issues with paternity, unregistered adoptions and misrecordings of complex genealogies.

Removing ancestral background bias. We have been careful to ensure that signals such as age and sex cannot be used in the Geni data sets to make the task of kinship verification easier. However, we did not account for the signal of ancestral background. The Geni ancestry website contains profiles of individuals from around the world. Therefore, easy negative examples of pairs or triples can be produced when an image of a child or parent is swapped out for someone from a different ancestral background. Computer vision methods are capable of classifying ancestral background with a high accuracy (M. Wang et al., 2018). Thus, we expect to be able to control for these signals in a similar manner to photographic age.

Further kinship experiments. Finally, we would like to benchmark the performance of humans and other computer vision methods on our newly developed kinship data sets to gain a better understanding of our preliminary results. Over the coming months, we will continue to experiment using the Geni kinship data sets to further understand the effects of variations in architecture, training protocol, data set restriction etc. We will also make the kinship verification community aware of our new data sets soon by publishing a paper outlining them.

Bibliography

Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., & Kim, B. (2018). Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems* (pp. 9505–9515).

Aliradi, R., Belkhir, A., Ouamane, A., & Elmaghraby, A. S. (2018). DIEDA: discriminative information based on exponential discriminant analysis combined with local features representation for face and kinship verification. *Multimedia Tools and Applications*, 1–18.

Alvergne, A., Faurie, C., & Raymond, M. (2007). Differential facial resemblance of young children to their parents: who do children look like more? *Evolution and Human behavior*, *28*(2), 135–144.

Alvergne, A., Faurie, C., & Raymond, M. (2009). Father–offspring resemblance predicts paternal investment in humans. *Animal Behaviour*, *78*(1), 61–69.

Alvi, M., Zisserman, A., & Nellåker, C. (2018). Turning a blind eye: Explicit removal of biases and variation from deep neural network embeddings. In *Proceedings of the european conference on computer vision (eccv)* (pp. 0–0).

Amato, F., López, A., Peña-Méndez, E. M., Vaňhara, P., Hampl, A., & Havel, J. (2013). Artificial neural networks in medical diagnosis. *J Appl Biomed*, *11*, 47–58.

Arandjelović, R., Gronat, P., Torii, A., Pajdla, T., & Sivic, J. (2017). NetVLAD: CNN architecture for weakly supervised place recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*(6), 1437–1451.

Arandjelović, R., & Zisserman, A. (2013). All about VLAD. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition* (pp. 1578–1585).

Bengani, H., Handley, M., Alvi, M., Ibitoye, R., Lees, M., Lynch, S. A., ... Malmgren, H. (2017). Clinical and molecular consequences of disease-associated de novo mutations in SATB2. *Genetics in Medicine*, *19*(8), 900.

Boehringer, S., Vollmar, T., Tasse, C., Wurtz, R. P., Gillessen-Kaesbach, G., Horsthemke, B., & Wieczorek, D. (2006). Syndrome identification based on 2D analysis software. *European Journal of Human Genetics*, *14*(10), 1082.

Buolamwini, J., & Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency* (pp. 77–91).

Burch, R. L., & Gallup Jr., G. G. (2000). Perceptions of paternal resemblance predict family violence. *Evolution and Human Behavior*, *21*(6), 429–435.

Bush, W. S., & Moore, J. H. (2012). Genome-wide association studies. *PLoS computational biology*, *8*(12), e1002822.

Cao, Q., Shen, L., Xie, W., Parkhi, O. M., & Zisserman, A. (2018). VGGFace2: a dataset for recognising faces across pose and age. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)* (pp. 67–74).

Cerrolaza, J. J., Porras, A. R., Mansoor, A., Zhao, Q., Summar, M., & Linguraru, M. G. (2016). Identification of dysmorphic syndromes using landmark-specific local texture descriptors. In *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)* (pp. 1080–1083).

Chang, F. J., Tuan Tran, A., Hassner, T., Masi, I., Nevatia, R., & Medioni, G. (2017). Faceposenet: Making a case for landmark-free face alignment. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 1599–1608).

Chatfield, K., Simonyan, K., Vedaldi, A., & Zisserman, A. (2014). Return of the devil in the details: Delving deep into convolutional nets. *arXiv preprint arXiv:1405.3531*.

Chen, S., Wang, X., Tang, Y., Chen, X., Wu, Z., & Jiang, Y.-G. (2017). Aggregating frame-level features for large–scale video classification. *arXiv preprint arXiv:1707.00803*.

Chen, X., An, L., Yang, S., & Wu, W. (2017). Kinship verification in multi–linear coherent spaces. *Multimedia Tools and Applications*, *76*(3), 4105–4122.

Choi, M., Scholl, U. I., Ji, W., Liu, T., Tikhonova, I. R., Zumbo, P., ... Sanjad, S. (2009). Genetic diagnosis by whole exome capture and massively parallel DNA sequencing. *Proceedings of the National Academy of Sciences*, *106*(45), 19096–19101.

Clark, C., & Divvala, S. (2016). PDFFigures 2.0: Mining figures from research papers. In *2016 IEEE/ACM Joint Conference on Digital Libraries (JCDL)* (pp. 143–152).

Croft, D. J. (1972). Is computerized diagnosis possible? *Computers and Biomedical Research*, *5*(4), 351–367.

Dahan, E., Keller, Y., & Mahpod, S. (2017). Kin–verification model on FIW dataset using multi–set learning and local features. In *Proceedings of the 2017 Workshop on Recognizing Families In the Wild* (pp. 31–35).

Dawson, M., Zisserman, A., & Nellåker, C. (2017). Mining faces from biomedical literature using deep learning. In *Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics* (pp. 562–567).

Dawson, M., Zisserman, A., & Nellåker, C. (2018). From same photo: cheating on visual kinship challenges. In *Asian conference on computer vision* (pp. 654–668).

DeBruine, L. M., Smith, F. G., Jones, B. C., Roberts, S. C., Petrie, M., & Spector, T. D. (2009). Kin recognition signals in adult faces. *Vision research*, *49*(1), 38–43.

De Dombal, F., Leaper, D., Staniland, J. R., McCann, A., & Horrocks, J. C. (1972). Computer–aided diagnosis of acute abdominal pain. *British Medical Journal*, *2*(5804), 9–13.

Dehghan, A., Ortiz, E. G., Villegas, R., & Shah, M. (2014). Who do I look like? Determining parent–offspring resemblance via gated autoencoders. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1757–1764).

Demner-Fushman, D., Antani, S., Simpson, M., & Thoma, G. R. (2012). Design and development of a multimodal biomedical information retrieval system. *Journal of Computing Science and Engineering*, *6*(2), 168–177.

Diamond, G. A., Staniloff, H. M., Forrester, J. S., Pollock, B. H., & Swan, H. (1983). Computer-assisted diagnosis in the noninvasive evaluation of patients with suspected coronary artery disease. *Journal of the American College of Cardiology*, *1*(2.1), 444–455.

Dibeklioglu, H., Salah, A. A., & Gevers, T. (2012). Are you really smiling at me? spontaneous versus posed enjoyment smiles. In *European conference on computer vision* (pp. 525–538).

Dibeklioglu, H., Salah, A. A., & Gevers, T. (2013). Like father, like son: Facial expression dynamics for kinship verification. In *2013 IEEE International Conference on Computer Vision* (pp. 1497–1504).

Doersch, C., Gupta, A., & Efros, A. A. (2015). Unsupervised visual representation learning by context prediction. In *International Conference on Computer Vision (ICCV)*.

Doi, K. (2007). Computer–aided diagnosis in medical imaging: historical review, current status and future potential. *Computerized medical imaging and graphics*, *31*(4-5), 198–211.

Duan, Q., & Zhang, L. (2017). AdvNet: adversarial contrastive residual net for 1 million kinship recognition. In *Proceedings of the 2017 Workshop on Recognizing Families In the Wild* (pp. 21–29).

Dutta, A., & Zisserman, A. (2019). The VGG image annotator (VIA). *arXiv preprint arXiv:1904.10699*.

European Parliament & the Council of the European Union. (2000). Regulation (EC) No 141/2000 of the European Parliament and of the Council of 16 December 1999 on orphan medicinal products. *Official J Eur Communities*, *43*, L18.

Fang, R., Gallagher, A. C., Chen, T., & Loui, A. (2013). Kinship classification by modeling facial feature heredity. In *Image Processing (ICIP), 2013 20th IEEE International Conference on* (pp. 2983–2987).

Fang, R., Tang, K. D., Snavely, N., & Chen, T. (2010). Towards computational models of kinship verification. In *Image Processing (ICIP), 2010 17th IEEE International Conference on* (pp. 1577–1580).

Ferry, Q., Steinberg, J., Webber, C., FitzPatrick, D. R., Ponting, C. P., Zisserman, A., & Nellåker, C. (2014). Diagnostically relevant facial gestalt information from ordinary photos. *Elife*, *3*, e02020.

Firmino, M., Morais, A. H., Mendoça, R. M., Dantas, M. R., Hekis, H. R., & Valentim, R. (2014). Computer–aided detection system for lung cancer in computed tomography scans: review and future prospects. *Biomedical engineering online*, *13*(1), 41.

Giannakopoulos, T., Foufoulas, I., Stamatogiannakis, E., Dimitropoulos, H., Manola, N., & Ioannidis, Y. (2015). Visual-based classification of figures from scientific literature. In *Proceedings of the 24th international conference on world wide web* (pp. 1059–1060).

Glastonbury, C. A., Ferlaino, M., Nellåker, C., & Lindgren, C. M. (2018). Adjusting for confounding in unsupervised latent representations of images. *arXiv preprint arXiv:1811.06498*.

Gonzalez, G. H., Tahsin, T., Goodale, B. C., Greene, A. C., & Greene, C. S. (2015). Recent advances and emerging applications in text and data mining for biomedical discovery. *Briefings in bioinformatics*, *17*(1), 33–42.

Gorlin, R. J., Cohen Jr, M. M., & Hennekam, R. C. (2001). *Syndromes of the head and neck*. Oxford University Press.

Groza, T., Kohler, S., Doelken, S., Collier, N., Oellrich, A., Smedley, D., ... Robinson, P. (2015). Automatic concept recognition using the human phenotype ontology reference and test suite corpora. *Database*, *2015*, 1–13.

Guo, G., & Wang, X. (2012). Kinship measurement on salient facial features. *IEEE Transactions on Instrumentation and Measurement*, *61*(8), 2322–2325.

Guo, Y., Zhang, L., Hu, Y., He, X., & Gao, J. (2016). MS-celeb-1M: A dataset and benchmark for large–scale face recognition. In *European conference on computer vision* (pp. 87–102).

Gupta, A., Vedaldi, A., & Zisserman, A. (2016). Synthetic data for text localisation in natural images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2315–2324).

Gurovich, Y., Hanani, Y., Bar, O., Fleischer, N., Gelbman, D., Basel-Salmon, L., ... Bird, L. M. (2018). DeepGestalt – identifying rare genetic syndromes using deep learning. *arXiv preprint arXiv:1801.07637*.

Hart, T., & Hart, P. (2009). Genetic studies of craniofacial anomalies: clinical implications and applications. *Orthodontics & craniofacial research*, *12*(3), 212–220.

Hu, J., Lu, J., Yuan, J., & Tan, Y.-P. (2014). Large margin multi–metric learning for face and kinship verification in the wild. In *Asian conference on computer vision* (pp. 252–267).

Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7132–7141).

Huang, G. B., Ramesh, M., Berg, T., & Learned-Miller, E. (2007). *Labeled faces in the wild: A database for studying face recognition in unconstrained environments* (Tech. Rep.). University of Massachusetts; Amherst, MA, USA. Report 07-49.

Huang, Y. L., Chen, D. R., Jiang, Y. R., Kuo, S. J., Wu, H. K., & Moon, W. K. (2008). Computer–aided diagnosis using morphological features for classifying breast lesions on ultrasound. *Ultrasound in Obstetrics and Gynecology*, *32*(4), 565–572.

Jaderberg, M., Simonyan, K., Vedaldi, A., & Zisserman, A. (2014). Synthetic data and artificial neural networks for natural scene text recognition. *arXiv preprint arXiv:1406.2227*.

Jamaludin, A., Kadir, T., & Zisserman, A. (2017). SpineNet: Automated classification and evidence visualization in spinal MRIs. *Medical image analysis*, *41*, 63–73.

Jegou, H., Perronnin, F., Douze, M., Sánchez, J., Perez, P., & Schmid, C. (2011). Aggregating local image descriptors into compact codes. *IEEE transactions on pattern analysis and machine intelligence*, *34*(9), 1704–1716.

King, D. E. (2009). Dlib–ml: A machine learning toolkit. *Journal of Machine Learning Research*, *10*, 1755-1758.

Köhler, S., Doelken, S. C., Mungall, C. J., Bauer, S., Firth, H. V., Bailleul-Forestier, I., ... Campbell, J. (2013). The human phenotype ontology project: linking molecular biology and disease through phenotype data. *Nucleic acids research*, *42*(D1), D966–D974.

Kohli, N., Singh, R., & Vatsa, M. (2012). Self–similarity representation of weber faces for kinship classification. In *2012 IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems (BTAS)* (pp. 245–250).

Kohli, N., Vatsa, M., Singh, R., Noore, A., & Majumdar, A. (2017). Hierarchical representation learning for kinship verification. *IEEE Transactions on Image Processing*, *26*(1), 289–302.

Kohli, N., Yadav, D., Vatsa, M., Singh, R., & Noore, A. (2019). Supervised mixed norm autoencoder for kinship verification in unconstrained videos. *IEEE Transactions on Image Processing*, *28*, 1329–1341.

Koike, A., & Takagi, T. (2009). Classifying biomedical figures using combination of bag of keypoints and bag of words. In *2009 International Conference on Complex, Intelligent and Software Intensive Systems* (pp. 848–853).

Kononenko, I. (2001). Machine learning for medical diagnosis: history, state of the art and perspective. *Artificial Intelligence in medicine*, *23*(1), 89–109.

Kosilek, R., Frohner, R., Würtz, R., Berr, C., Schopohl, J., Reincke, M., & Schneider, H. (2015). Diagnostic use of facial image analysis software in endocrine and genetic disorders: review, current results and future perspectives. *European journal of endocrinology*, *173*(4), M39–M44.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097–1105).

Kuhn, T., Nagy, M. L., Luong, T., & Krauthammer, M. (2014). Mining images in biomedical publications: Detection and analysis of gel diagrams. *Journal of biomedical semantics*, *5*(1), 10.

Laiadi, O., Ouamane, A., Benakcha, A., & Taleb-Ahmed, A. (2017). RFIW 2017: LPQ–SIEDA for large scale kinship verification. In *Proceedings of the 2017 workshop on recognizing families in the wild* (pp. 37–39).

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, *1*(4), 541–551.

Lee, P. S., West, J. D., & Howe, B. (2017). Viziometrics: Analyzing visual information in the scientific literature. *IEEE Transactions on Big Data*, *4*(1), 117–129.

Li, Y., Zeng, J., Zhang, J., Dai, A., Kan, M., Shan, S., & Chen, X. (2017). KinNet: Fine–to–coarse deep metric learning for kinship verification. In *Proceedings of the 2017 workshop on recognizing families in the wild* (pp. 13–20).

Lipton, Z. C. (2016). The mythos of model interpretability. *arXiv preprint arXiv:1606.03490*.

Liu, D. Y., Chen, H. L., Yang, B., Lu, X. E., Li, L. N., & Liu, J. (2012). Design of an enhanced fuzzy k-nearest neighbor classifier based computer aided diagnostic system for thyroid disease. *Journal of medical systems*, *36*(5), 3243–3254.

Liu, F., Van Der Lijn, F., Schurmann, C., Zhu, G., Chakravarty, M. M., Hysi, P. G., ... Ikram, M. A. (2012). A genome-wide association study identifies five loci influencing facial morphology in europeans. *PLoS genetics*, *8*(9), e1002932.

Liu, Q., Puthenputhussery, A., & Liu, C. (2016). A novel inheritable color space with application to kinship verification. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on* (pp. 1–9).

López, M. B., Boutellaa, E., & Hadid, A. (2016). Comments on the 'kinship face in the wild' data sets. *IEEE transactions on pattern analysis and machine intelligence*, *38*(11), 2342–2344.

Lopez, M. B., Hadid, A., Boutellaa, E., Goncalves, J., Kostakos, V., & Hosio, S. (2018). Kinship verification from facial images and videos: human versus machine. *Machine Vision and Applications*, *29*(5), 873–890.

Lu, J., Hu, J., Liong, V. E., Zhou, X., Bottino, A., Islam, I. U., ... Chen, S. (2015). The FG 2015 kinship verification in the wild evaluation. In *2015 11th IEEE international conference and workshops on automatic face and gesture recognition (FG)* (Vol. 1, pp. 1–7).

Lu, J., Hu, J., & Tan, Y. P. (2017). Discriminative deep metric learning for face and kinship verification. *IEEE Transactions on Image Processing*, *26*(9), 4269–4282.

Lu, J., Zhou, X., Tan, Y. P., Shang, Y., & Zhou, J. (2014). Neighborhood repulsed metric learning for kinship verification. *IEEE transactions on pattern analysis and machine intelligence*, *36*(2), 331–345.

Mahendran, A., & Vedaldi, A. (2016). Visualizing deep convolutional neural networks using natural pre-images. *International Journal of Computer Vision*, *120*(3), 233–255.

Mahpod, S., & Keller, Y. (2018). Kinship verification using multiview hybrid distance learning. *Computer Vision and Image Understanding*, *167*(C), 28–36.

Microsoft Corporation. (2018). *Bing Image Search API.* (azure.microsoft.com/en-us/services/cognitive-services/bing-image-search-api/)

Miech, A., Laptev, I., & Sivic, J. (2017). Learnable pooling with context gating for video classification. *arXiv preprint arXiv:1706.06905*.

Moorman, J. R., Lake, D. E., & Moss, T. J. (2016). Computers in white coats: How to devise useful clinical decision support software. *Critical care medicine*, *44*(7), 1449–1450.

Moujahid, A., & Dornaika, F. (2018). A pyramid multi–level face descriptor: application to kinship verification. *Multimedia Tools and Applications*, 1–20.

Murphy, K., van Ginneken, B., Schilham, A. M., De Hoop, B., Gietema, H., & Prokop, M. (2009). A large–scale evaluation of automatic pulmonary nodule detection in chest ct using local image features and k-nearest-neighbour classification. *Medical image analysis*, *13*(5), 757–770.

Nandy, A., & Mondal, S. S. (2019). Kinship verification using deep siamese convolutional neural network. In *2019 14th IEEE international conference on automatic face & gesture recognition (FG 2019)* (pp. 1–5).

Ng, S. B., Buckingham, K. J., Lee, C., Bigham, A. W., Tabor, H. K., Dent, K. M., ... Nickerson, D. A. (2010). Exome sequencing identifies the cause of a mendelian disorder. *Nature genetics*, *42*(1), 30.

Ng, S. B., Turner, E. H., Robertson, P. D., Flygare, S. D., Bigham, A. W., Lee, C., ... Eichler, E. E. (2009). Targeted capture and massively parallel sequencing of 12 human exomes. *Nature*, *461*(7261), 272.

Nigam, V. P., & Graupe, D. (2004). A neural–network–based detection of epilepsy. *Neurological research*, *26*(1), 55–60.

Parkhi, O. M., Vedaldi, A., & Zisserman, A. (2015). Deep face recognition. In *British Machine Vision Conference (BMVC)* (Vol. 1, p. 6).

Patel, B., Maheshwari, R., & Raman, B. (2017). Evaluation of periocular features for kinship verification in the wild. *Computer Vision and Image Understanding*, *160*, 24–35.

Qin, X., Tan, X., & Chen, S. (2015). Tri–subject kinship verification: Understanding the core of a family. *IEEE Transactions on Multimedia*, *17*(10), 1855–1867.

Rai, M. C. E., Werghi, N., Al Muhairi, H., & Alsafar, H. (2015). Using facial images for the diagnosis of genetic syndromes: a survey. In *2015 international conference on communications, signal processing, and their applications (ICCSPA 15)* (pp. 1–6).

Ramírez, J., Górriz, J., Salas-Gonzalez, D., Romero, A., López, M., Álvarez, I., & Gómez-Río, M. (2013). Computer–aided diagnosis of alzheimer's type dementia combining support vector machines and discriminant set of features. *Information Sciences*, *237*, 59–72.

Rare Disease UK. (2019). *What is a rare disease?* Retrieved 2019-09-29, from www.raredisease.org.uk/what-is-a-rare-disease/

Robinson, J. P., Shao, M., Wu, Y., & Fu, Y. (2016). Families in the wild (FIW): Large–scale kinship image database and benchmarks. In *Proceedings of the 2016 ACM on multimedia conference* (pp. 242–246).

Robinson, J. P., Shao, M., Wu, Y., Liu, H., Gillis, T., & Fu, Y. (2018). Visual kinship recognition of families in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Robinson, J. P., Shao, M., Zhao, H., Wu, Y., Gillis, T., & Fu, Y. (2017). Recognizing families in the wild (RFIW): Data challenge workshop in conjunction with ACM–MM 2017. In *Proceedings of the 2017 workshop on recognizing families in the wild* (pp. 5–12).

Rodriguez-Esteban, R., & Iossifov, I. (2009). Figure mining for biomedical research. *Bioinformatics*, *25*(16), 2082–2084.

Rothe, R., Timofte, R., & Van Gool, L. (2015). Dex: Deep expectation of apparent age from a single image. In *Proceedings of the IEEE international conference on computer vision workshops* (pp. 10–15).

Rozantsev, A., Lepetit, V., & Fua, P. (2015). On rendering synthetic images for training an object detector. *Computer Vision and Image Understanding*, *137*, 24–37.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... Bernstein, M. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, *115*(3), 211–252.

Sayers, E. (2019). *Entrez programming utilities help.* Retrieved 2019-09-29, from www.ncbi.nlm.nih.gov/books/NBK25500/

Schroff, F., Kalenichenko, D., & Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 815–823).

Sharif Razavian, A., Azizpour, H., Sullivan, J., & Carlsson, S. (2014). CNN features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (pp. 806–813).

Sheehan, M. J., & Nachman, M. W. (2014). Morphological and population genomic evidence that human faces have evolved to signal individual identity. *Nature communications*, *5*, 4800.

Shi, Z., Zhang, L., Sun, Y., & Ye, Y. (2018). Multiscale multitask deep NetVLAD for crowd counting. *IEEE Transactions on Industrial Informatics*, *14*(11), 4953–4962.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large–scale image recognition. *arXiv preprint arXiv:1409.1556*.

Sivic, J., & Zisserman, A. (2003). Video google: A text retrieval approach to object matching in videos. In *International Conference on Computer Vision (ICCV)* (Vol. 2, pp. 1470–1477).

Sluimer, I. C., van Waes, P. F., Viergever, M. A., & Van Ginneken, B. (2003). Computer–aided diagnosis in high resolution CT of the lungs. *Medical physics*, *30*(12), 3081–3090.

Taigman, Y., Yang, M., Ranzato, M., & Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1701–1708).

Tang, Y., Zhang, X., Ma, L., Wang, J., Chen, S., & Jiang, Y. G. (2018). Non–local NetVLAD encoding for video classification. In *Proceedings of the European Conference on Computer Vision (ECCV).*

Teixeira, B., Tamersoy, B., Singh, V., & Kapoor, A. (2019). Adaloss: Adaptive loss function for landmark localization. *arXiv preprint arXiv:1908.01070*.

Temurtas, H., Yumusak, N., & Temurtas, F. (2009). A comparative study on diabetes disease diagnosis using neural networks. *Expert Systems with applications*, *36*(4), 8610–8615.

Twigg, S. R., & Wilkie, A. O. (2015). New insights into craniofacial malformations. *Human molecular genetics*, *24*(R1), R50–R59.

Vedaldi, A., & Lenc, K. (2015). Matconvnet: Convolutional neural networks for matlab. In *Proceedings of the 23rd ACM international conference on multimedia* (pp. 689–692).

Wang, M., Deng, W., Hu, J., Peng, J., Tao, X., & Huang, Y. (2018). Racial faces in the wild: Reducing racial bias by deep unsupervised domain adaptation. *arXiv preprint arXiv:1812.00194*.

Wang, S., Ding, Z., & Fu, Y. (2018). Cross-generation kinship verification with sparse discriminative metric. *IEEE transactions on pattern analysis and machine intelligence*.

Wang, S., Robinson, J. P., & Fu, Y. (2017). Kinship verification on families in the wild with marginalized denoising metric learning. In *Automatic face & gesture recognition (FG 2017), 2017 12th IEEE international conference on* (pp. 216–221).

Whitelam, C., Taborsky, E., Blanton, A., Maze, B., Adams, J., Miller, T., ... Allen, K. (2017). Iarpa Janus benchmark-B face dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (pp. 90–98).

Wu, X., Boutellaa, E., Feng, X., & Hadid, A. (2016). Kinship verification from faces: Methods, databases and challenges. In *2016 IEEE international conference on signal processing, communications and computing (ICSPCC)* (pp. 1–6).

Wu, X., Boutellaa, E., López, M. B., Feng, X., & Hadid, A. (2016). On the usefulness of color for kinship verification from face images. In *Information forensics and security (WIFS), 2016 IEEE international workshop on* (pp. 1–6).

Xia, C., Xia, S., Zhou, Y., Zhang, L., & Shao, M. (2018). Graph based family relationship recognition from a single image. In *Pacific rim international conference on artificial intelligence* (pp. 310–320).

Xia, S., Shao, M., & Fu, Y. (2011). Kinship verification through transfer learning. In *Proceedings of the 22nd international joint conference on artificial intelligence* (Vol. 3, pp. 2539–2544). AAAI Press.

Xia, S., Shao, M., Luo, J., & Fu, Y. (2012). Understanding kin relationships in a photo. *IEEE Transactions on Multimedia*, *14*(4), 1046–1056.

Xiong, C., Liu, L., Zhao, X., Yan, S., & Kim, T. K. (2015). Convolutional fusion network for face verification in the wild. *IEEE Transactions on Circuits and Systems for Video Technology*, *26*(3), 517–528.

Xu, M., & Shang, Y. (2016). Kinship verification using facial images by robust similarity learning. *Mathematical Problems in Engineering*, *2016*.

Xu, S., McCusker, J., & Krauthammer, M. (2008). Yale image finder (YIF): a new search engine for retrieving biomedical images. *Bioinformatics*, *24*(17), 1968–1970.

Yan, H. (2019). Learning discriminative compact binary face descriptor for kinship verification. *Pattern Recognition Letters*, *117*, 146–152.

Yan, H., & Lu, J. (2017). Video–based facial kinship verification. In *Facial kinship verification* (pp. 63–80). Springer.

Yan, H., Lu, J., Deng, W., & Zhou, X. (2014). Discriminative multimetric learning for kinship verification. *IEEE Transactions on Information forensics and security*, *9*(7), 1169–1178.

Yang, Y., & Wu, Q. (2017). A novel kinship verification method based on deep transfer learning and feature nonlinear mapping. *DEStech Transactions on Computer Science and Engineering*, *AIEA*.

Yu, J., Zhu, C., Zhang, J., Huang, Q., & Tao, D. (2019). Spatial Pyramid–Enhanced NetVLAD With Weighted Triplet Loss for Place Recognition. *IEEE transactions on neural networks and learning systems*.

Zhang, J., Xia, S., Pan, H., & Qin, A. (2016). A genetics–motivated unsupervised model for tri–subject kinship verification. In *Image processing (ICIP), 2016 IEEE international conference on* (pp. 2916–2920).

Zhang, K., Huang, Y., Song, C., Wu, H., & Wang, L. (2015, September). Kinship verification with deep convolutional neural networks. In *Proceedings of the British Machine Vision Conference (BMVC)* (p. 148.1-148.12).

Zhao, Q., Werghi, N., Okada, K., Rosenbaum, K., Summar, M., & Linguraru, M. G. (2014). Ensemble learning for the detection of facial dysmorphology. In *2014 36th annual international conference of the IEEE engineering in medicine and biology society* (pp. 754–757).

Zhao, Y. G., Song, Z., Zheng, F., & Shao, L. (2018). Learning a multiple kernel similarity metric for kinship verification. *Information Sciences*, *430*, 247–260.

Zhong, Y., Arandjelović, R., & Zisserman, A. (2018). GhostVLAD for set–based face recognition. In *Asian conference on computer vision* (pp. 35–50).

Zhou, X., Hu, J., Lu, J., Shang, Y., & Guan, Y. (2011). Kinship verification from facial images under uncontrolled conditions. In *Proceedings of the 19th ACM international conference on multimedia* (pp. 953–956).

Zhou, X., Jin, K., Xu, M., & Guo, G. (2019). Learning deep compact similarity metric for kinship verification from face images. *Information Fusion*, *48*, 84–94.

Zhou, X., Lu, J., Hu, J., & Shang, Y. (2012). Gabor–based gradient orientation pyramid for kinship verification under uncontrolled environments. In *Proceedings of the 20th ACM international conference on multimedia* (pp. 725–728).

Zhou, X., Shang, Y., Yan, H., & Guo, G. (2016). Ensemble similarity learning for kinship verification from facial images in the wild. *Information Fusion*, *32*, 40–48.