1-6-2017

Intersection of Longest Paths in Graph Theory and Predicting Performance in Facial Recognition

Amy Yates

TITLE: INTERSECTION OF LONGEST PATHS IN GRAPH THEORY AND PREDICTING PERFORMANCE IN FACIAL RECOGNITION

by

AMY YATES

Under the Direction of Guantao Chen, PhD

ABSTRACT

A set of subsets is said to have the Helly property if the condition that each pair of subsets has a non-empty intersection implies that the intersection of all subsets has a non-empty intersection. In 1966, Gallai noticed that the set of all longest paths of a connected graph is pairwise intersecting and asked if the set had the Helly property. While it is not true in general, a number of classes of graphs have been shown to have the property. In this dissertation, we show that K_4-minor-free graphs, interval graphs, circular arc graphs, and

the intersection graphs of spider graphs are classes that have this property.

The accuracy of facial recognition algorithms on images taken in controlled conditions has improved significantly over the last two decades. As the focus is turning to more unconstrained or relaxed conditions and toward videos, there is a need to better understand what factors influence performance. If these factors were better understood, it would be easier to predict how well an algorithm will perform when new conditions are introduced.

Previous studies have studied the effect of various factors on the verification rate (VR), but less attention has been paid to the false accept rate (FAR). In this dissertation, we study the effect various factors have on the FAR as well as the correlation between marginal FAR and VR. Using these relationships, we propose two models to predict marginal VR and demonstrate that the models predict better than using the previous global VR.

INDEX WORDS: Helly Property, longest paths, facial recognition.

TITLE: INTERSECTION OF LONGEST PATHS IN GRAPH THEORY AND PREDICTING PERFORMANCE IN FACIAL RECOGNITION

by

AMY YATES

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

in the College of Arts and Sciences

Georgia State University

2016

TITLE: INTERSECTION OF LONGEST PATHS IN GRAPH THEORY AND PREDICTING PERFORMANCE IN FACIAL RECOGNITION

by

AMY YATES

Committee Chair: Guantao Chen

Committee: Zhongshan Li

Hendricus van der Holst

Yi Zhao

Changyong Zhong

Electronic Version Approved:

Office of Graduate Studies

College of Arts and Sciences

Georgia State University

December 2016

DEDICATION

This dissertation is dedicated to my parents who have helped countless times over the years and have always been encouraging.

ACKNOWLEDGEMENTS

This dissertation work would not have been possible without the support of many people. I want to express my gratitude to my family, my friends, and my advisor Professor Guantao Chen. Without your support over the years, I would have never gotten this far.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

- GSU - Georgia State University
- NIST - National Institute of Standards and Technology
- PaSC - Point-and-Shoot Challenge

PART 1

INTRODUCTION

My time in graduate school has been split between Georgia State University (GSU) and the National Institute of Standards and Technology (NIST). While at GSU, I focused on structural graph theory and worked on the intersection of longest paths in connected graphs. While at NIST, I have been working in the field of facial recognition. This dissertation is a result of my years at both these institutions and reflects the work I have done at both.

1.1 Graph Theory

A well-known fact in structural graph theory is that in a connected graph, any two longest paths contain a common vertex. In 1966, Gallai [1] questioned if, in a connected graph, there was a vertex common to all longest paths. This is analogous to asking if the set of longest paths, in a connected graph, has the *Helly property*.

In 1923, Helly [2] proved that for every finite family of at least $n + 1$ convex sets of $\mathbb{R}^n$ such that every $n + 1$ sets have a nonempty intersection, the entire family has a nonempty intersection. This has led to various theorems and properties [3]. A family of sets has the *Helly property* if every subfamily of pairwise intersecting members has an element common to all members.

Helly's theorem and the Helly property has been used numerous times in graph theory and various other disciplines [4]. Amenta [5] proved that every generalized linear programming problem implies a Helly theorem but that the reverse was not necessarily true.

The Helly property has been applied to many properties of graphs and hypergraphs. Dourado, Protti, and Szwarcfiter [4] did an extensive survey on the Helly property when applied to graphs and hypergraphs.

Jamison and Nowakowski [6] proved that the Helly number of the convex sets of a

connected graph is equal to the clique number of the graph. The Helly number of a family of sets is the minimum number n such that every subfamily, which has the property that the intersection of every n or fewer sets is nonempty, has a nonempty intersection itself. A convex set S of a graph G is a subset of the vertices, $S \subseteq V(G)$, such that for any two vertices $x, y \in S$, all the vertices on every shortest (x, y)-path belong to S.

Bretto, Ubéda, and Žerovnik [7] characterized the strong Helly hypergraphs, hypergraphs in which the edges of every partial subhypergraph has the Helly property. In their paper, they also found an algorithm to determine if a hypergraph is strong Helly in polynomial time.

Daligault, Gonçalves, and Rao [8] proved that diamond-free graphs, graphs with no induced subgraph isomorphic to K_4 with the deletion of an edge, are Helly circle graphs. A Helly circle graph is a graph whose vertices correspond to the chords of a circle such that every set of three pairwise intersecting chords have a point common to all chords.

Lin, Soulignac, Szwarcfiter [9] partially characterized the classes of normal Helly, proper Helly, and unit Helly circular arc graphs through forbidden induced subgraphs for each class. A Helly circular arc graph is a graph whose vertices correspond to the arcs of a circle that has the Helly property. A Helly circular arc graph is normal when at least three arcs are required to cover the corresponding circle. The graph is proper when when no arc is a subset of another, and the graph is unit when each arc is of the same length. As the authors explained, these characterizations imply algorithms to recognize the classes in linear time with the input being a circular arc graph. Recently, Cao, Grippo, and Safe [10] completely characterized normal Helly circular arc graphs through forbidden induced subgraphs and found an algorithm to recognize a normal Helly circular arc graph in linear time, answering questions in [9].

Bonomo [11] characterized the Helly circular arc graphs that are self-clique. Bonomo, Chudnovsky, and Durán [12] character the Helly circular arc graphs that are clique-perfect, graphs in which the minimum number of vertices needed to intersect all cliques and the maximum number of pairwise disjoint cliques are the same. Bonomo, Durán, Grippo, and

Safe [13] completely characterized unit Helly circle graphs, Helly circle graphs in which the chords of the corresponding circle are of equal length. Joeris, Lin, McConnell, Spinrad, and Szwarcfiter [14] gave a characterization of Helly circular arc graphs, which lead to a recognition algorithm that runs in polynomial time.

For a family of sets F, a graph G is said to be F-Helly if the family F has the Helly property in G.

Hamelink [15] proved that if every subset of the cliques of a graph G has the Helly property, i.e. G is clique-Helly, then G is a clique graph, which is a graph whose vertices correspond to the cliques of another graph. A short time later, Roberts and Spencer [16] characterized clique graphs stating that G is a clique graph if and only if there is a set of complete subgraphs which covers the edges of G and every subset has the Helly property. Lin and Szwarcfiter [17] proposed algorithms for determining if a graph is clique-Helly or hereditary clique-Helly, improving the complexity from previous algorithms. These algorithms can be done in polynomial time.

Groshaus and Szwarcfiter [18] in 2007 gave two characterizations of biclique-Helly graphs; a biclique is a maximal set of vertices which induce a complete bipartite graph. A year later, the authors [19] characterized the class of hereditary biclique-Helly graphs, graphs whose induced subgraphs are also biclique-Helly. The authors also characterized hereditary open neighborhood-Helly graphs and hereditary closed neighborhood-Helly.

Bandelt and Prisner [20] proved that the following classes are equivalent: disk-Helly graphs, dismantable clique-Helly graphs, and clique-Helly graphs that are convergent to a single vertex. Disk-Helly graphs are graphs in which the family of disks has the Helly property. A disk with center $v \in V(G)$ and radius k is the set of all vertices whose distance from v is at most k. A graph is dismantable if recursively removing vertices whose closed neighborhoods are completely contained in the closed neighborhood of another vertex results in a graph with a single vertex. The clique operator K is defined as $K^0(G) = G$ and $K^i(G) = K(K^{i-1}(G))$ for $i \geq 1$ where $K(G)$ is the clique graph of G. A graph is convergent under the clique operator if $K^j(G) = K^{j-1}(G)$. If $K^j(G)$ is a graph with a single vertex,

then G is said to converge to a single vertex.

Bondy, Durán, Lin, and Szwarcfiter [21] characterized the clique-Helly self-clique graphs as the graphs whose clique matrices are quasi-symmetric, matrices whose row and column families are identical. A graph is self-clique if it is isomorphic to its clique graph. Larrión and Pizaña [22] characterized self-clique hereditary clique-Helly graphs.

In 2005, Dourado, Petito, and Teixeira [23] proved that both the clique-Helly and the hereditary clique-Helly sandwich problems are NP-complete. Given a graph G and spanning subgraph H, the property P sandwich problem is to find another spanning subgraph H^* of G such that the edge set of H^* is a superset of the edge set of H and such that H^* has property P.

When it came to Gallai's question on if connected graphs were longest path-Helly, Walther [24] showed that the answer was no three years later by finding a planar counterexample on 25 vertices. Walther and Voss [25] and Zamfirescu [26] independently found a counterexample on 12 vertices, seen in Figure 1.1; Zamfirescu conjectured that no counterexample on 11 or less vertices exists. This is the smallest known connected graph where the set of all longest paths does not have the Helly property. Figure 1.2 from Schmitz and Werner [27] shows a planar counterexample.

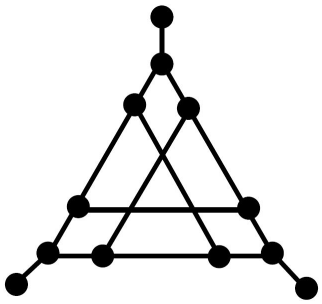

Figure (1.1) The 12-vertex counterexample

Counterexamples were found for 2-connected planar, 3-connected planar, and non-

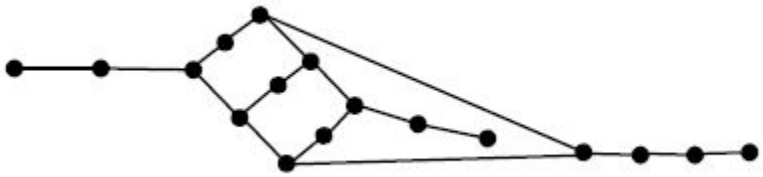

Figure (1.2) The 17-vertex planar counterexample

planar graphs as well. In 1972, Zamfirescu [28] found a 2-connected planar counterexample on 82 vertices. In 1974, Grünbaum [29] found a 3-connected planar counterexample on 484 vertices and a 3-connected non-planar counterexample on 324 vertices. In 1976, Zamfirescu [26] found smaller counterexamples. Zamfirescu found a 2-connected planar counterexample on 32 vertices and a 3-connected counterexample on 36 vertices.

Nadeem, Shabbir, and Zamfirescu [30] found a counterexample on 46 vertices embedded in the square lattice, a counterexample on 94 vertices embedded in the hexagonal lattice, a 2-connected counterexample on 126 vertices embedded in the square lattice, and a 2-connected counterexample on 244 vertices embedded in the hexagonal lattice. They conjectured that these counterexamples are minimal. In [31], Bashir and Tudor showed that Schmitz's counterexample in Figure 1.2 is not embeddable in the square lattice but that it *is* embeddable in the cubic lattice, and thus there is a counterexample on 17 vertices embeddable in the cubic lattice.

In 2012, Dino Jumani and Zamfirescu [32] found graphs which are embeddable into the triangular lattice but whose longest paths do not share a common vertex; the connected counterexample is on 30 vertices, and the 2-connected counterexample is on 92 vertices. The triangular lattice is an infinite planar graph in which every face is a triangle. In 2013, Nadeem, Shabbir, and Zamfirescu [33] found many graphs which are embeddable into the planar square lattice but whose longest paths do not share a common vertex. The planar square lattice is an infinite planar graph in which every face is a square.

Recently, Shabbir [34] found a counterexample on 58 vertices embedded on in a hexago-

nal lattice on the Klein bottle, a planar counterexample on 17 vertices embedded in a square lattice on the Klein bottle, a planar 2-connected counterexample on 80 vertices embedded in a square lattice on the Klein bottle, a counterexample on 12 vertices in a triangular lattice on the Klein bottle, and a planar 2-connected counterexample on 48 vertices in a triangular lattice on the Klein bottle. Bashir, Nadeem, and Shabbir [35] found connected graphs such that every pair of vertices is missed by a longest path in the triangular, square, and hexagonal lattices. They also found such graphs in some lattices embedded on the torus, Möbius strip, and the Klein bottle.

Any hypotraceable graph, a graph that does not have a Hamiltonian path but any vertex-deleted subgraph does, is a counterexample [36]. Araya and Wiener [37] proved the existence of a cubic planar hypotraceable graph on 340 vertices; they also proved such graphs existed on $2n$ vertices for every $n \geq 178$. Thomassen [36] found hypotraceable graphs on 34, 37, 39, and 40 vertices and showed that for all $n \geq 42$, a hypotraceable graph exists on n vertices.

Although the Helly property on the set of longest paths is not true in general, it has been shown to be true in certain classes of graphs. For example, in 1990, Klavžar and Petkovšek [38] proved that all longest paths in connected split graphs, block graphs, and cacti share a vertex. A split graph is a graph whose vertices can be partitioned into an independent set and a clique. A block graph is the intersection graph for the blocks of another graph. A cactus is a graph where no two cycles share an edge. In 2004, Balister, Györi, Lehel, and Schelp [39] established a similar result for interval graphs and circular arc graphs. Their proof for circular arc graphs had a gap that was recently closed by Joos [40].

Recently, Chen [41] proved the conjecture to be true when the matching number of a graph, the maximum size of a set of independent (or non-adjacent) edges, is at most 3. Chen also asked if the conjecture is true when the matching number is at most 5, pointing out that the graph in Figure 1.1 has a matching number of 6. Instead of looking at the set of all longest paths, Axenovich [42] proved in 2009 that any three longest paths in outerplanar graphs have a common vertex. Tutte [43] proved that 4-connected planar graphs are Hamiltonian, and

therefore, all longest paths contain all vertices and trivially have the Helly property. Tutte modified the methods of Whitney used to prove that planar triangulations in which the edges of every C_3 are the bounds of a face is Hamiltonian.

In 2011, the Helly property for the set of all longest paths was confirmed for outerplanar graphs by de Rezende, Fernandes, Martin, and Wakabayashi [44], strengthening the result by Axenovich in [42]. Later, in 2013, the same authors [45] also proved that all longest paths share a vertex in 2-trees, which are chordal graphs. In the same paper, they questioned whether the Helly property holds for series-parallel graphs, which are 2-connected graphs that can be reduced to a K_2 by a series of deletions of parallel edges and contracting edges.

1.2 Facial Recognition

Over the years, the application and need for automatic face recognition has grown. There are numerous applications in law enforcement. In the last few years, a need has arisen even in social media. Social media platforms, such as Facebook and Google+, can automatically tag friends and family in uploaded images. Cell phones can automatically tag contacts in images.

Still, determining if two faces are of the same identity is a difficult task. Despite many automated systems existing, the general face recognition problem is still unsolved. Predicting how a system will perform in new settings is still challenging. Since 1993, the error rate of facial recognition systems on frontal images, such as mugshots or taken in a controlled studio, has decreased dramatically [46]. However, when the image conditions are less and less constrained, performance decreases [47].

To help advance the state of automatic face recognition systems, various datasets and challenges have been issued. Starting in 1993, the Face Recognition Technology (FERET) program built database of 8525 images of 884 individuals [48]; the images were taken under controlled conditions. In 2001, the BIOID database was introduced and consisted of 1521 frontal, grayscale images of 23 individuals [49].

In 2003, Carnegie Mellon University (CMU) introduced the Pose, Illumination, and

Expression (PIE) Database [50]. The database consists of over 40,000 images of 68 individuals of various poses, illumination conditions, and expressions. Seven years later, CMU introduced the Multi-PIE database, which has over 750,000 images of 337 individuals [51]. This database addressed shortcomings of CMU's PIE database such has the low number of expressions captured and only one recording session. The database also includes high resolution images.

In 2005, the Face Recognition Grand Challenge (FRGC) was introduced [52]. The dataset consists of over 50,000 images of over 400 individuals; the frontal images are 2D, 3D, controlled illumination, and uncontrolled illumination. In 2006, the Face Recognition Vendor Test (FRVT 2006) used images taken under controlled conditions in studio lighting and along with a probe set of frontal images in uncontrolled lighting [53].

In 2007, the University of Massachusetts released the Labeled Faces in the Wild (LFW) database of over 13,000 images of over 5700 individuals [54]. The images are unconstrained images of celebrities taken under a variety of conditions such as lighting, location, and camera.

In 2010, NIST ran the Multiple-Biometric Evaluation (MBE) [55] using mugshots and frontal images taken under controlled conditions; some of the images came from the FRVT 2006.

In 2011, the Good, the Bad, & the Ugly (GBU) Face Challenge was introduced [56]. The challenge dataset has three partitions each with 2170 nominally frontal images of 437 individuals. The partitions were made from the FRVT 2006 dataset. The partitions contain the same set of individuals and were created based on how well or difficult the matched pairs were matched correctly.

In 2013, Sgroi, Bowyer, Flynn, and Phillips [57] introduced the Strong, Neutral, or Weak (SNoW) Face Impostor Pairs problem. The problem dataset was constructed similarly to the GBU dataset. The SNoW dataset has three partitions created from the FRVT 2006 dataset. The partitions were created based on how well or difficult the non-matched pairs were matched incorrectly.

In 2013, the Point-and-Shoot Challenge (PaSC) was introduced [47]. The challenge dataset has 9376 images of 293 individuals taken from various cameras, views, and conditions. The dataset also includes 2802 videos of 265 individuals.

Beveridge, Givens, Phillips, Draper, and Lui [58] used generalized linear mixed models to study factors that affect performance on images from the Face Recognition Vendor Test 2006. They looked at 50 factors, including gender, race, whether the subject was wearing glasses, age, focus, resolution, and many more.

Beveridge, Givens, Phillips, and Draper [59] used generalized linear mixed models to analyze factors that affect performance on images from the Face Recognition Grand Challenge. They looked at numerous factors including gender, race, age, expression, whether the subject was wearing glasses, image size, resolution, and many others. Many of these factors are very costly to annotate for each image.

Beveridge, Givens, Phillips, Draper, Bolme, and Lui [60] used generalized linear mixed models to analyze factors on images of the Face Recognition Vendor Test 2006. They looked at the factors gender, race, age, whether the subject was wearing glasses, and whether the image was indoors or not.

O'Toole, Phillips, An, and Dunlop [61] used FRVT 2006 images to investigate the effects of race and gender on the performance of algorithms. They showed a decrease in performance when the non-matching pairs of images were restricted by these demographic factors.

1.3 Dissertation Results

The first result of this dissertation, as stated in Theorem 1 below, we show that Gallai's question has an affirmative answer for connected K_4-minor-free graphs, another distinguished subclass of planar graphs. The class of K_4-minor-free graphs contains the class of series-parallel graphs. Series-parallel graphs are 2-connected, and based on the operations performed to obtain a series-parallel graph, such graphs have no K_4 minor. On the other hand, graphs with no K_4 minor can be 1-connected. Consequently, the Helly property holds for outerplanar graphs (graphs with no K_4 minor or $K_{2,3}$ minor) and series-parallel graphs,

which answers the question proposed in [45] positively.

The next result, in Section 6, establishes the property for interval graphs, and the third result, in Section 7, establishes the property for the intersection graphs of spider graphs. We use Lemma 7 to prove these results. The final result in Section 8 proves that the set of longest paths for circular arc graphs has the Helly property. Though Balister, Györi, Lehel, and Schelp [39] already proved the results for interval and circular arc graphs, we use new techniques.

In face recognition, predicting how a system will perform in new settings is a challenging problem. Without explicitly testing the new settings, a common method is to use the overall performance of the system on previously known settings. However, is there a way to better model the performance without needing the time to identify and label the individuals in the videos? In Section 9, we show that a better model does indeed exist using the Point-and-Shoot Face Recognition Challenge (PaSC) dataset.

Additionally, we investigate the impostor distribution and its relationship with the genuine distribution over the levels of various factors. A setting may be thought to be "easy" if it has a high amount of correct matches and a low amount of incorrect matches. Do such easy settings exist?

In this dissertation, we investigate both of these questions over four different state of the art, independently developed algorithms on the PaSC data set.

PART 2

DEFINITIONS

2.1 Graph Theory

A *graph* G is an ordered pair $(V(G), E(G))$ where $V(G) = \{v_1, v_2, ..., v_n\}$ is a set whose elements are referred to as *vertices* and where $E(G) \subseteq V(G)^2$ is a set whose elements are unordered pairs of vertices, called *edges*. $V(G)$ is called the *vertex set* of G, and $E(G)$ is called the *edge set* of G. When the graph G is unambiguous, these sets are denoted as V and E, respectively.

Edges are explicitly written as $e = uv$ where $u, v \in V(G)$ are the *endpoints* of the edge. If two vertices, u and v, are the endpoints of an edge in G, then u and v are said to be *adjacent*. A *loop* is an edge whose endpoints are the same vertex. If two (or more) edges contain the exact same set of endpoints, the edges are said to be *multiple edges*, *multi-edges*, or *parallel edges*.

In this dissertation, graphs contain no loops, but they may contain multi-edges. A graph that contains no loops or multi-edges is said to be *simple*.

A *walk*, W, is a sequence of vertices, written $W = w_1w_2...w_m$, such that $w_i \in V(G)$ for $i = 1, 2, ..., m$ such that $w_iw_{i+1} \in E(G)$ for $i = 1, 2, ..., m-1$. A walk, $P = p_1p_2...p_m$, whose vertices are distinct, i.e. $p_i \neq p_j$ if $i \neq j$, is called a *walk*. A *subpath* is a subsequence of a path P. The *length* of a path is the number of edges in the path. A walk, $C = c_1c_2...c_m$, such that $C' = c_1c_2...c_{m-1}$ is a path and $c_1 = c_m$ is called a *cycle*. Two paths $P = p_1p_2...p_k$ and $Q = q_1q_2...q_l$ are said to be *internally vertex-disjoint* if $k = l$, $\{p_1, p_k\} = \{q_1, q_l\}$, and $\{p_2, p_3, ..., p_{k-1}\} \cap \{q_2, q_3, ..., q_{l-1}\} = \emptyset$. That is, two paths are internally vertex-disjoint if no internal vertices, non-endvertices, are shared between the two paths.

A *complete graph* is a simple graph in which every pair of vertices is adjacent. Up to isomorphism, there is only one complete graph on n vertices, denoted K_n.

Let $G = (V(G), E(G))$ and $H = (V(H), E(H))$ be graphs for the following definitions.

A graph is said to be *connected* if there exists a path between any two distinct vertices in the vertex set. A graph is said to be *2-connected* if it is isomorphic to K_2 or if there exist two vertex-disjoint paths between any two distinct vertices in the vertex set. A connected graph without any cycles is called a *tree*. If $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$, then H is a *subgraph* of G. A connected subgraph of a tree is called a *subtree*.

Let $uv \in E(G)$. To *subdivide* the edge means to add a new vertex w to the vertex set of G and to replace the edge uv with the edges uw and wv. To *contract* the edge means to delete the edge uv then to identify the vertices u and v. This means that, in the resulting graph G', the vertices u and v are replaced by one vertex w and for each vertex $x \in V(G) \setminus \{u, v\}$, the edge $wx \in E(G')$ exists for every edge $ux \in E(G)$ and $vx \in V(G)$, possibly resulting in multiple edges.

If there exists a mapping $\phi : V(G) \to V(H)$ such that if $uv \in E(G)$ then $\phi(u)\phi(v) \in E(H)$, then G is said to be *homomorphic* to H. H is said to be a *minor* of G if H can be obtained from G after a series of contractions of edges and deletions of vertices and edges. If no such minor exists, then G is said to be *H-minor-free*.

A *series-parallel graph* is a graph that can be obtained by starting with a K_2 and performing the following operations in some order. (1) Add a parallel edge to an existing edge. (2) Subdivide an edge. Series-parallel graphs are exactly the K_4-minor-free graphs [62].

An *intersection graph* is a graph whose vertices correspond to subsets of a set such that two vertices are adjacent if and only if their corresponding subsets have nonempty intersection. A *chordal graph* is a graph with no chordless cycles with length greater than three; a *chord* is an edge between any two nonconsecutive vertices on a cycle. In fact, Gavril proved that a chordal graph is the intersection graph of a set of subtrees of a tree T [63]. An *interval graph* is a graph whose vertices correspond to a set of real open intervals such that two vertices are adjacent if their corresponding intervals have a nonempty intersection. An interval graph can also be defined as the intersection graph of a path graph, so an interval graph is a chordal graph. A *circular arc graph* is a graph whose vertices correspond to a set

of open arcs on a circle such that two vertices are adjacent if their corresponding arcs have a nonempty intersection.

2.2 Facial Recognition

Given two overlapping distributions of scores, genuine and impostors, a *receiver operating characteristic (ROC)* can be calculated. Given a threshold, all scores above the threshold are classified as matches and those below by non-matches. If a score from the genuine distribution is above the threshold, the score is declared a *genuine match*. If a score from the impostor distribution is above the threshold, the score is declared a *false alarm*. The percentage of scores in the genuine distribution that are classified as genuine matches is called the *verification rate (VR)*. The percentage of scores in the impostor distribution that are classified as false alarms is called the *false accept rate (FAR)*. By varying this threshold, a set of ordered pairs of these rates, (FAR, VR), can be obtained. Plotting these values produces an *ROC curve* and shows the trade-off "between true positive rate and false positive rate of a classifier" [64].

Our results are reported on participants in the Face and Gesture 2015 Person Recognition Evaluation [65], and in this competition, the participants followed the PaSC protocol. In the protocol for the PaSC, algorithms are given two videos and then return a number measuring the degree of similarity between the subjects in the pair of videos. Hence, in calculating and predicting performance, we compare videos in pairs.

In measuring performance, we are observing how often an algorithm correctly declares the same person to be in two videos. We are also interested in how often the algorithm incorrectly believes two different people from videos are the same person. However, we are not interested in the overall performance of the algorithm. Instead, we are more interested in how the performance changes over levels of a factor. Later in this disseration, for a set of videos of a factor-level, we are predicting how well an algorithm will correctly match videos of the same person (marginal VR). In our prediction, we use how often the algorithm incorrectly declared different people to be the same (marginal FAR). We then compare our

predicted performance to the actual observed performance.

The focus of analysis in this dissertation is on performance when comparing videos for a factor-level. Presented with two faces from videos x and y, an algorithm A returns a similarity score, $s_A(x, y)$, for video-pair (x, y). The similarity score denotes how similar the faces are estimated to be; a higher similarity score indicates a higher likelihood of the two faces belonging to the same subject.

To make a decision, a threshold τ_g is set so that every video-pair score at least as large τ_g is declared a match and every score below the threshold is considered a non-match. We denote the set of all video-pairs by V, and we note that V is partitioned into two sets M and I in the following way. Let M denote the set of video-pairs that are genuine matches, i.e. of the same subject, and let I denote the set of video-pairs that are impostors, i.e. of different subjects. The verification rate (VR) is the ratio of correctly matched pairs to the set of all genuine matches, and the false accept rate (FAR) is the ratio of the incorrectly matched pairs to the set of all impostors. Given a threshold τ_g, these are explicitly defined as

$$\mathrm{VR}(s_A(V), \tau_g) = \frac{|\{(x, y) \in M \mid s_A(x, y) \geq \tau_g\}|}{|M|} \tag{2.1}$$

$$\mathrm{FAR}(s_A(V), \tau_g) = \frac{|\{(x, y) \in I \mid s_A(x, y) \geq \tau_g\}|}{|I|} \tag{2.2}$$

where, again, V is the set of all video-pairs.

Generally, the threshold τ_g is set to specify the FAR at a certain instance. In our dissertation, we select τ_g so that $\mathrm{FAR}(s_A(V), \tau_g) = 0.10$. For PaSC, the standard for reporting VR is FAR $= 0.01$. However, we shifted the threshold to have enough false matches for analysis.

Nonetheless, the analysis in this dissertation is not focused on the overall performance over the set of all video-pairs. Rather, for this dissertation, as previously mentioned, the analysis is centered on performance when comparing video-pairs of factor levels. As an example, if the factor is gender, then we know from Section 9.1.4 that the factor-levels are

both-female (F/F), both-male (M/M), or female-male (F/M). Note female-male only arises for impostor-pairs. Let $F_i \subset V$ denote the set of video-pairs for the ith level of some factor (e.g. $F_{M/M}$ with the gender factor). With τ_g set so that the global FAR is 0.10 as desired, the marginal rates for factor-level F_i are calculated the following way:

$$\mathrm{VR}(s_A(F_i), \tau_g) = \frac{|\{(x,y) \in F_i \cap M \mid s_A(x,y) \geq \tau_g\}|}{|F_i \cap M|} \tag{2.3}$$

$$\mathrm{FAR}(s_A(F_i), \tau_g) = \frac{|\{(x,y) \in F_i \cap I \mid s_A(x,y) \geq \tau_g\}|}{|F_i \cap I|} \tag{2.4}$$

PART 3

DECOMPOSITION OF K_4-MINOR-FREE GRAPHS

A connected graph can be decomposed into blocks and cut-vertices. Tutte [66] introduced a method to decompose 2-connected graphs into 3-connected blocks: *bonds*, *cycles*, and *3-connected blocks*. Here, we modify this decomposition for K_4-minor-free graphs. The following observations imply that there are no 3-connected blocks (aside from K_3) when applying Tutte's decomposition to a 2-connected K_4-minor-free graph.

Lemma 1. *Let G be a 2-connected K_4-minor-free graph. If a subgraph H of G is a Θ-graph consisting of three internally vertex-disjoint paths $P_1^{[u,v]}$, $P_2^{[u,v]}$, and $P_3^{[u,v]}$, then $P_1^{(u,v)}$, $P_2^{(u,v)}$, and $P_3^{(u,v)}$ are in three different components of $G - \{u, v\}$, provided they are not empty.*

Proof. Suppose to the contrary there is a path $Q^{[x,y]}$ connecting $x \in V(P_1^{(u,v)})$ and $y \in V(P_2^{(u,v)})$ such that $Q^{[x,y]} \cap P_3^{(u,v)} = \emptyset$. Then by contracting $P_1^{(u,v)}$ to u, $P_2^{(u,v)}$ to y, $Q^{(x,y)}$ to x, and $P_3^{(u,v)}$ to v, we get a K_4, giving a contradiction. □

As a result of Lemma 1, $\{u, v\}$ is a 2-separation of G.

Lemma 2. *Let G be a 2-connected series-parallel graph but not a bond, and let $e = \{u, v\}$ be an arbitrary edge of G. If $\{u, v\}$ is not a 2-separation of G, then $G - e$ is no longer 2-connected. In particular, if G has at least 4 vertices, G is not 3-connected.*

Proof. Since $\{u, v\}$ is not a 2-separation of G and $|E(G)| \geq 3$, e is the only edge between u and v. Suppose on the contrary that $G - e$ is still 2-connected. By Menger's Theorem, there exist two internally vertex-disjoint $\{u, v\}$-paths P and Q. Then $P \cup Q \cup \{e\}$ is a Θ-graph. By Lemma 1, $P^{(u,v)}$ and $Q^{(u,v)}$ are in two different components of $G - \{u, v\}$, which in turn shows that $\{u, v\}$ is a 2-separation of G, a contradiction. □

Lemma 3. *Let G be a 2-connected graph, and let $\{u, v\}$ be a 2-separation of G. Then, there is no 2-separation $\{x, y\}$ such that x and y are in different $\{u, v\}$-bridges of G.*

Proof. Since $\{u, v\}$ is a 2-separation of G, there are three internally vertex-disjoint (u, v)-paths in G, so for any two distinct vertices x and y in different bridges of $\{u, v\}$, $G - \{x, y\}$ does not separate u and v. As each graph induced on the vertex set of a $\{u, v\}$-bridge together with $\{u, v\}$ is 2-connected, $\{x, y\}$ is not a 2-cut of G, which in turn shows that $\{x, y\}$ is not a 2-separation of G. □

In the following, we apply Tutte Decomposition Algorithm (TDA) (see chapter 3 in [66]) specifically for 2-connected K_4-minor-free graphs. Given a 2-connected graph G, we apply TDA to produce a set $\mathcal{D}(G)$ of 3-blocks, a set $\phi(G)$ of virtual edges, and a rooted tree $T_3(G)$ with vertex set $\mathcal{D}(G)$. This algorithm is applied on an ordered pair (G, e) where e is an arbitrary edge of G, but both $\mathcal{D}(G)$ and $\phi(G)$ are independent from the selection of edge e as Tutte pointed out in [66].

Tutte Decomposition Algorithm (TDA) Let G be a nontrivial 2-connected K_4-minor-free graph, and let $e \in E(G)$ be an arbitrary edge of G with end vertices u and v. We perform the following operations.

O-0 If G is a cycle, let $T_3(G) := K_1$ with vertex set $\mathcal{D}(G) := \{G\}$ and $\phi(G) := \emptyset$. We let G be the root of $T_3(G)$. Otherwise, we perform the following operations.

O-A If $\{u, v\}$ is a 2-separation of G, let $G_1, G_2, \cdots, G_k$ be all non-trivial bridges of edge e. For $i = 1, 2, \cdots, k$, add a *virtual edge* e_i between u and v, and let $\overline{G_i} := G_i + e_i$. Let B_{uv} be a bond with vertex set $\{u, v\}$ and edge set $\{e, e_1, e_2, \cdots, e_k\}$. Clearly, each $\overline{G_i}$ is a 2-connected K_4-minor-free graph with $|E(\overline{G_i})| < |E(G)|$ and $\{u, v\}$ is not a 2-separation of $\overline{G_i}$. By applying TDA to the ordered pair $(\overline{G_i}, e_i)$ and to the resulting smaller graphs, we obtain $T_3(\overline{G_i})$, $\mathcal{D}(\overline{G_i})$, and $\phi(\overline{G_i})$. Let $\mathcal{D}(G) := (\bigcup_{i=1}^{k} \mathcal{D}(\overline{G_i})) \cup \{B_{uv}\}$ and $T_3(G)$ be a tree obtained from vertex set $\mathcal{D}(G)$ and by adding an edge between B_{uv} and the root of $T_3(\overline{G_i})$ for each $i = 1, 2, \cdots, k$. Set the root of $T_3(G)$ as B_{uv}. Let $\phi(G) := (\bigcup_{i=1}^{k} \phi(\overline{G_i})) \cup \{e_1, e_2, \ldots, e_k\}$ be the set of virtual edges.

O-B If $\{u, v\}$ is not a 2-separation of G, then $G - e$ is no longer 2-connected by Lemma 2. That means $G - e$ is a nontrivial block chain, say $G_1 v_1 G_2 v_2 \cdots v_{k-1} G_k$ with $u \in V(G_1)$,

$v \in V(G_k)$, $u \neq v_1$, and $v \neq v_{k-1}$. Designate $u = v_0$ and $v = v_k$. For $i = 1, 2, \cdots, k$, if $G_i \cong K_2$ with $V(G_i) = \{v_{i-1}, v_i\}$, let $e_i = v_{i-1}v_i$. Otherwise, let $F_{i,1}, F_{i,2}, \ldots, F_{i,k_i}$ be all nontrivial bridges of $\{v_{i-1}, v_i\}$ in G_i, add k_i+1 virtual edges e_i and $e_{i,j}$ between v_{i-1} and v_i for $j = 1, 2, \cdots, k_i$, and let $B_{v_{i-1}v_i} = E(v_{i-1}, v_i) \cup \{e_i, e_{i,1}, \ldots, e_{i,k_i}\}$ and $G_{i,j} = F_{i,j} \cup \{e_{i,j}\}$ for $j = 1, 2, \ldots, k_i$. Let C_{uv} be the cycle induced by $\{e, e_1, e_2, \cdots, e_{k-1}\}$. For each pair (i, j) with $0 \leq i \leq k-1$ and $1 \leq j \leq k_i$, it is readily seen that $G_{i,j}$ is a 2-connected K_4-minor-free simple graph. Moreover, $\{v_{i-1}, v_i\}$ is not a 2-separation of $G_{i,j}$.

By applying TDA to all pairs $(G_{i,j}, e_{i,j})$ such that $G_i \neq e_i$ and the resulting smaller graphs, we obtain $T_3(G_{i,j})$, $\mathcal{D}(G_{i,j})$, and $\phi(G_{i,j})$ for $G_i \neq e_i$. Let

$$\mathcal{D}(G) := \{C_{uv}\} \cup \left(\cup_{i:G_i \neq e_i} \cup_{j=1}^{k_i} \mathcal{D}(G_{i,j})\right) \cup \left(\cup_{i:G_i \neq e_i} B_{v_{i-1}v_i}\right).$$

Let $T_3(G)$ be obtained from $\cup_{i=1}^{k}(\cup_{j=1}^{k_i} T_3(G_{i,j})$ (where the union is taken over these $G_i \neq e_i$) by adding a vertex C_{uv} and $B_{v_{i-1}v_i}$ for each i such that $G_i \neq e_i$ and an edge between C_{uv} and each $B_{v_{i-1}v_i}$ and by adding an edge between $B_{v_{i-1}v_i}$ to the root of each $T_3(G_{i,j})$. Set C_{uv} as the root of $T_3(G)$. Let $\phi(G) := \cup_{i:G_i \neq e_i} \left(\{e_i\} \cup (\cup_{j=1}^{k_i} \phi(G_{i,j})) \cup \{e_{i,j}\}\right)$ be the virtual edge set.

Lemma 4. *Let G be a 2-connected simple K_4-minor-free graph and $A = \{x, y\}$ be a set of two vertices of G. Then, A is the vertex set of a bond $B \in \mathcal{D}(G)$ if and only if A is a 2-separation of G.*

Proof. We prove the lemma by induction on $|V(G)|$. Lemma 4 is clearly true if $|V(G)| = 1$, for then G is a cycle as there is no 2-separation of G nor bond in $\mathcal{D}(G)$. Now, we assume that $|V(G)| > 1$, so G is not a cycle and either **O-A** or **O-B** was applied.

If **O-A** was applied, then $\{u, v\}$ is a 2-separation and $B_{uv} \in \mathcal{D}(G)$. By Lemma 3, $\{u, v\}$ are in the same $V(\overline{G_i})$ for some $i \in \{1, 2, \cdots, k\}$. Additionally, $\{x, y\}$ is a 2-separation in $\overline{G_i}$ if and only if $\{x, y\}$ is a 2-separation of G with $x, y \in V(\overline{G_i})$ and $\{x, y\} \neq \{u, v\}$. Thus, inductively, the proof follows.

If **O-B** was applied, the bonds are $B_{v_iv_{i+1}}$, and these resulted in $G_{i,j}$ for each $G_i \neq e_i$ with $j = 1, 2, \cdots, k_i$. Because G is not a cycle, there is at least one such i. Since each G_i such that $G_i \neq e_i$ is 2-connected, there are two internally vertex-disjoint (v_{i-1}, v_i)-paths P_1 and P_2 in G_i. Since these two paths and a (v_{i-1}, v_i)-path along the other direction of C_{uv} form a Θ-graph, $P_1 v_{i-1}, v_i)$ and $P_2(v_{i-1}, v_i)$ are in different bridges of $\{v_{i-1}, v_i\}$ in G_i, which in turn shows that $\{v_{i-1}, v_i\}$ is a 2-separation of G. Since $G_{i,j}$ such that $G_i \neq e_i$ is 2-connected and v_{i-1} and v_i are adjacent in $G_{i,j}$ for each $j = 1, 2, \cdots, k_i$, a pair of vertices in G_i form a 2-separation in G_i if and only if they form a 2-separation in G, are in $V(G_i)$, and are not in $V(C_{uv})$. Inductively, we can show that Lemma 4 holds in this case. □

Applying Lemma 4, we can show that $\mathcal{D}(G)$, $\phi(G)$, and $T_3(G)$ are independent from the choice of the edge uv and are uniquely determined by G. In fact, Tutte [66] showed that this statement is true for every 2-connected simple graph without the condition of being K_4-minor-free.

For each vertex $v \in V(G)$, let $\mathcal{D}_v(G)$ denote the set of 3-blocks containing v and $T_3[\mathcal{D}_v(G)]$ be the subgraph induced by $\mathcal{D}_v(G)$.

Lemma 5. *Let G be a 2-connected K_4-minor-free graph. For each vertex $v \in V(G)$, then the subgraph $T_3[\mathcal{D}_v(G)]$ is a subtree of $T_3(G)$.*

Proof. Since $T_3(G)$ is a tree, we only need to show that the subgraph induced by $\mathcal{D}_v(G)$ is connected. Otherwise, assume that there exist two 3-blocks $A, B \in \mathcal{D}_v(G)$ such that not all of the internal vertices of the (A, B)-path $D_1(= A)D_2 \ldots D_{m-1}D_m(= B)$ in $T_3(G)$ are in $\mathcal{D}_v(G)$. Since $T_3(G)$ is a bipartite graph with one class containing all bonds and the other containing all cycles, the path is an alternating path of bonds and cycles. Recall $V(B) \subset V(C)$ if bond B and cycle C are adjacent in $T_3(G)$. Since $v \in V(D_{i-1})\ V(D_i)$ for some $i > 1$, D_{i-1} is a cycle and D_i is a bond, and let $V(D_i) = \{w, z\}$. Because $v \notin V(D_i)$ but $v \in V(D_m)$, then $i < m$. As $T_3(G)$ is a tree, $D_1, D_2, ..., D_{i-1}$ and $D_{i+1}, D_{i+2}, ..., D_m$ are in two different components of $T_3(G) - D_i$. From **O-A**, the connected components of $G - \{w, z\}$ correspond to some nontrivial $\{w, z\}$-bridge of G. The 3-block D_i in $T_3(G)$ has

a unique neighbor for each nontrivial $\{w, z\}$-bridge. Therefore $\cup_{j=1}^{i-1} V(D_j)$ and $\cup_{j=i+1}^{m} V(D_j)$ are in two different components of $G - \{w, z\}$. However v is both sets, a contradiction. □

We now expand our consideration from 2-connected graphs to connected graphs. Let G be a connected K_4-minor-free graph. We obtain a decomposition tree, denoted as T_G, by the following steps.

Step 1. Decompose G into blocks and cut vertices (see [66]) and obtain a block-cut vertex tree, say $T_2(G)$, by the following description.

$T_2(G)$ is a bipartite graph with two partitions (U, V) such that for vertices in U, there is a 1-1 correspondence to blocks of G and such that for vertices in V, there is a 1-1 correspondence to cut vertices of G. For any $u \in U$ and $v \in V$, $uv \in E(T_2(G))$ if and only if the block corresponding to u contains the cut vertex corresponding to v in G. It is easy to see that $T_2(G)$ is a tree. Note that each block with at least 3 vertices is a 2-connected K_4-minor-free graph but not 3-connected by Lemma 2.

For notation simplicity, for each element $X \in U$, we use X either as a vertex of $T_2(G)$ or a 2-connected subgraph of G, but the meaning will be clear from the context.

Step 2. For each nontrivial 2-connected K_4-minor-free $X \in U$, we apply TDA on X and obtain $T_3(X)$, the rooted tree of X, with vertex set $\mathcal{D}(X)$ and virtual edge set $\phi(X)$. Notice that if $X = K_2$, then $T_3(X) = K_1$, which is a single-vertex graph; otherwise $\mathcal{D}(X)$ consists of cycles and bonds only by the assumption that G is K_4-minor-free.

For each $X \in U$ and a cut vertex $v \in X$, recall $\mathcal{D}_v(X) = \{B \in \mathcal{D}(X) \mid v \in B\}$, which we will call *$v$-blocks* in X. By Lemma 5, $T_3[\mathcal{D}(X_v)]$ is a subtree of $T_3(X)$.

Let

$$\mathcal{D}(G) = \bigcup_{X \in U} \mathcal{D}(X),$$

which is a set consisting of K_2s, cycles, and bonds obtained TDA applied in Step 2. Recall that V is the set of cut-vertices of G mentioned in Step 1. We modify $T_2(G)$ into a graph T_G with vertex set $V \cup \mathcal{D}(G)$ through the step below:

Step 3. For each $X \in U \subset V(T_2(G))$, replace X by its Tutte decomposition rooted tree $T_3(X)$ from Step 2; then for each $v \in V$, if $vX \in E(T_2(G))$, X is a block containing v. In this case, we let v be adjacent to a vertex in $T_3([\mathcal{D}_v X])$. (Note that to which specific vertex in $T_3([\mathcal{D}_v X])$ we join v is not essential to our proof.) Denote the resulted tree by T_G and call it a *decomposition tree* of G.

Given a subgraph H of G, let $V_{T_G}(H) = \{X \in V(T_G) \,|\, X \cap V(H) \neq \emptyset\}$ and $T_H = T_G[V_{T_G}(H)]$. Particularly, when $H = \{v\}$, i.e. a single vertex, we simply denote $T_{\{v\}}$ by T_v. The following lemma implies that if H is connected, then T_H is connected.

Lemma 6. *Let G be a connected K_4-minor-free graph. The following two statements hold.*

(1) For each vertex $v \in V(G)$, T_v is a subtree in T_G;

(2) For every edge $e = uv \in E(G)$, $T_u \cap T_v \neq \emptyset$.

Proof. (1). If v is not a cut vertex of G, then v is contained in exactly one block of G. We then know T_v is connected by Lemma 5. Assume v is a cut vertex. Then for each block X of G which contains v, $T_G[\mathcal{D}_{Xv}]$ is a subtree. Let T_v be the graph obtained by taking the union of all $T_G[\mathcal{D}_{Xv}]$ and adding v and edges joining v to a vertex of $T_G[\mathcal{D}_{Xv}]$, for each of the subtree $T_G[\mathcal{D}_{Xv}]$. It is easy to see T_v is a connected graph.

(2). For each edge $e \in E(G)$, there exists exactly one block, say B, of G containing e. If $B = e$, then $e \in T_u \cap T_v$. Otherwise, there exists at least 3 vertices in B. If $\{u, v\}$ is not a 2-separation of G, there is a unique cycle in $\mathcal{D}(B)$ containing e; otherwise, there is a bond in $\mathcal{D}(B)$ containing e. In either case, we have $T_u \cap T_v \neq \emptyset$. □

PART 4

K_4-MINOR-FREE GRAPHS

Theorem 1. *Let G be a connected graph with no K_4-minor. Then all longest paths share a common vertex in G.*

Proof. Let G be a connected simple K_4-minor-free graph and $\mathcal{L}$ be the set of all longest paths in G. We prove Theorem 1 by a sequence of claims.

Claim 1. *We may assume that there exists a cycle $C \in \mathcal{D}(G)$ such that $P \cap C \neq \emptyset$ for each $P \in \mathcal{L}$.*

Proof. Let T_G be a decomposition tree of G. For each longest path P, recall $V_{T_G}(P) = \{X \in V(T_G) \,|\, X \cap V(P) \neq \emptyset\}$ and $T_P = T_G[V_{T_G}(P)]$. By Lemma 6, T_P is connected and thus a subtree of T_G. For any two longest paths P and Q in $\mathcal{L}$, we have $T_P \cap T_Q \neq \emptyset$ since $P \cap Q \neq \emptyset$. Let $T_{\mathcal{L}} = \{T_P \,|\, P \in \mathcal{L}\}$. It is well-known that *a family of subtrees of a tree has the Helly property* (see problem 18 on p. 49 of [67]), so there is a vertex $B \in V(T_G)$ such that $B \in \bigcap_{P \in \mathcal{L}} T_P$. By the construction of T_G, there are four possibilities of B: a cut-vertex of G, a block K_2 of G, a bond, or a cycle from $\mathcal{D}(G)$. We may assume that B is a cut-vertex, a cut-edge, or a bond.

If B is a cut-vertex of G, then $B \subset \bigcap_{P \in \mathcal{L}} P$, so Theorem 1 holds.

If $B = xy$ is a cut-edge of G, we may assume, without loss of generality, $x \in \bigcap_{P \in \mathcal{L}} P$, so Theorem 1 holds. We may assume this, for otherwise, there exists two longest paths $P, Q \in \mathcal{L}$ such that $x \in V(P)$ but $y \notin V(P)$ and $x \notin V(Q)$ with $y \in V(Q)$. Since $P \cap Q \neq \emptyset$, there is a vertex $z \in P \cap Q$. Then $\{x, y, z\}$ contains a triangle-minor, which contradicts xy being a cut edge in G.

Suppose B is a bond. Let C be a cycle adjacent to B in $T(G)$. Since G is a simple graph, we have $V(B) \subseteq V(C)$, which in turn gives Claim 1. □

In what follows, the notation C is reserved for the cycle $C \in \mathcal{D}(G)$ such that every longest path contains a vertex of C.

Let $uv \in E(C)$. If $\{u, v\}$ is a 2-cut in G, then $uv \in E(C)$ is a virtual edge while the possible real edge between u and v is in the corresponding bond. Following TDA, $\{u, v\}$ is a 2-separation of G. In this case, let G_{uv} be the subgraph of G obtained by deleting all components of $G - \{u, v\}$ containing a vertex of $C - \{u, v\}$. Since $\{u, v\}$ is a 2-separation, there are two (u, v)-paths $R^{[u,v]}$ and $S^{[u,v]}$ in G_{uv}. By Lemma 1, $R^{(u,v)}$ and $S^{(u,v)}$ are in different components of $G - \{u, v\}$. We call $R^{[u,v]}$ and $S^{[u,v]}$ *connectors* of G_{uv}.

The following two claims follow directly from TDA.

Claim 2. *For any two distinct edges $uv, pq \in E(C)$, we have $V(G_{uv}) \cap V(G_{pq}) \subseteq \{u, v\} \cap \{p, q\}$ and $E(G_{uv} - \{u, v\}, G_{pq} - \{p, q\}) = \emptyset$ (see Figure 4.1).*

Claim 3. *If $P^{[u,v]}$ is a path in G with $P^{[u,v]} \cap C = \{u, v\}$, then u and v are two consecutive vertices on C.*

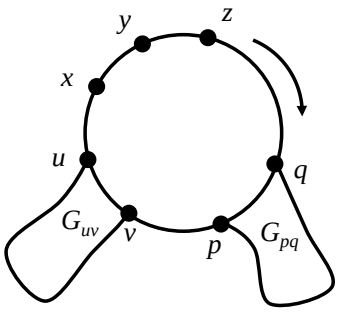

Figure (4.1) Structure of the cycle C

Claim 4. *If $P \in \mathcal{L}$ is a longest path in G such that P has at least one end, say u, on C, then both its predecessor u^- and successor u^+ along C are also on P.*

Proof. Suppose, on the contrary, that we assume $u^+ \notin V(P)$. If $uu^+ \in E(G)$, the $P \cup \{uu^+\}$ is a longer path, a contradiction. Otherwise, consider $R^{[u,u^+]}$ and $S^{[u,u^+]}$. Since $R^{(u,u^+)}$ and

$S^{(u,u^+)}$ are in different components of $G - \{u, v\}$, one of them, say $R^{(u,u^+)}$ is vertex-disjoint from P. Then $P \cup R^{[u,u^+]}$ is a longer path than P, giving a contradiction. □

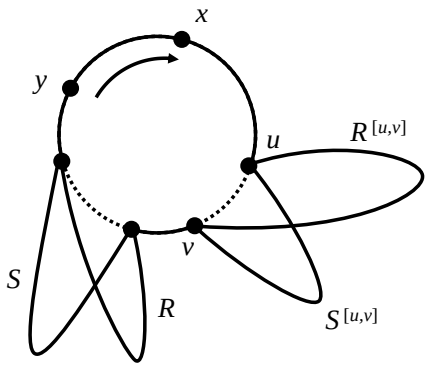

Figure (4.2) Two (u, v)-paths

For any two vertices $x, y \in C$, we use $\overrightarrow{C}_{\Pi}^{[x,y]}$ (resp. $\overleftarrow{C}_{\Pi}^{[x,y]}$) to denote a path obtained from $\overrightarrow{C}^{[x,y]}$ (resp. $\overleftarrow{C}^{[x,y]}$) by replacing each edge $uv \in E(C)$ by $R^{[u,v]}$ or $S^{[u,v]}$ whenever $\{u, v\}$ is a 2-separation.

Let $\mathcal{L}_1 = \{P \in \mathcal{L} \,|\text{P has at least one end vertex on}\, C\}$.

Claim 5. *For any $P \in \mathcal{L}_1$, $V(C) \subset V(P)$, so $V(C) \subset \bigcap_{P\in\mathcal{L}_1} V(P)$.*

Proof. Let $P \in \mathcal{L}_1$ with end vertex u on C, and set $Q = P - \{u\}$. By Claim 4, Q is a path containing both u^+ and u^- but not u. By Claim 3, traveling along Q from u^+ to u^- one must go through all vertices in $V(C) - u$, so $V(C) \subset V(P)$. □

Following Claim 5, we may assume $\mathcal{L}_2 := \mathcal{L} \setminus \mathcal{L}_1 \neq \emptyset$. If $|\mathcal{L}_2| \leq 1$, then Theorem 1 holds. We may assume $|\mathcal{L}_2| \geq 2$. Moreover, we have $\bigcap_{P\in\mathcal{L}} P \cap C = \bigcap_{P\in\mathcal{L}_2} P \cap C$. Notice that each path in $\mathcal{L}_2$ has exactly two tails of C.

Claim 6. *If $|P \cap C| \geq 2$ for every $P \in \mathcal{L}_2$, then $\bigcap_{P\in\mathcal{L}} P \neq \emptyset$.*

Proof. Let L be a longest tail of C, and let z be the origin (the common vertex of L and C) of L. We claim $z \in \bigcap_{P\in\mathcal{L}} P$. Suppose this is not true. By Claim 5, there exists $P \in \mathcal{L}_2$ such that $z \notin P$. Let P' and P'' be the two tails of P on C with origins u_1 and u_2, respectively.

Because $|P \cap C| \geq 2$, we know $u_1 \neq u_2$. If $P' \cap L \neq \emptyset$, then $u_1 z \in E(C)$ and both P' and L are subgraphs of $G_{u_1 z}$ by Claim 2, which in turn shows that $P'' \cap L = \emptyset$. Similarly, if $P'' \cap L \neq \emptyset$, then $P' \cap L = \emptyset$. We assume, without loss of generality, $P'' \cap L = \emptyset$. We also assume that along C the segment $\overrightarrow{C}^{[u_1,u_2]}$ does not contain vertex z. Then, $P^{[u_1,u_2]} \cap L = \emptyset$. Note that paths $P'' \cup P^{[u_2,u_1]}$, $\overleftarrow{C}_{\Pi}^{[u_1,z]}$, and L are internally vertex-disjoint paths. Concatenating these paths together, we obtain a longer path, which gives a contradiction. □

Hence, we assume there exists $P \in \mathcal{L}_2$ sharing exactly one vertex with C. We let $P \cap C = \{v\}$, and we denote $P' := P^{[v,v_1]}$ and $P'' := P^{[v,v_2]}$ as the two tails of P on C, both with origin v. Notice that $P = P' \cup P''$.

Claim 7. $v \in \bigcap_{P \in \mathcal{L}} P$.

Proof. Suppose to the contrary that there exists $Q \in \mathcal{L}_2$ such that $v \notin V(Q)$. By Claim 5, neither of the two ends of Q is on C. Thus Q has two tails, denoted as $Q' = Q^{[w_1,u_1]}$ and $Q'' = Q^{[w_2,u_2]}$, of C with origins w_1 and w_2, respectively. We assume, without loss of generality, the segment $\overrightarrow{C}^{[w_1,w_2]}$ does not contain v. Note that $w_1 = w_2$ if and only if Q and C share exactly one vertex.

We will distinguish a few cases according to which one of the four tails P', P'', Q' and Q'' is the longest. By the symmetry of P' and P'' and the symmetry of Q' and Q'', we only need to consider two cases according to whether P' or Q' is the longest one among the four tails.

First we assume that $|P'| \geq |Q'|$. If $w_1 \neq w_2$, then $P' \cap Q' \neq \emptyset$ and $P' \cap Q'' \neq \emptyset$. If $P' \cap Q'' = \emptyset$, then consider the path $\overrightarrow{C}_{\Pi}^{[v,w_1]}$ that does not intersect with P' except at v. We note that $\overrightarrow{C}_{\Pi}^{[v,w_1]}$ intersects $Q^{[w_1,w_2]} \cup Q''$ only at w_1 as $w_1 \neq w_2$. Thus, P', $\overrightarrow{C}_{\Pi}^{[v,w_1]}$, and $Q^{[w_1,w_2]} \cup Q'$ are internally vertex-disjoint paths. Thus by concatenating these paths, we have a path that is longer than Q, a contradiction. Similarly, we cannot have $P' \cap Q' = \emptyset$.

Therefore, $P' \cap Q' \neq \emptyset$ and $P' \cap Q'' \neq \emptyset$. However, this leads to a contradiction. If $P' \cap Q' \neq \emptyset$, then P' is in a $\{v, w_1\}$-bridge different from the one Q'' is in, which implies that Q'' should not intersect P'. Thus, if $w_1 \neq w_2$, then it cannot be that $|P'| \geq |Q'|$.

If $w_1 = w_2$ and $|P'| \geq |Q'|$, then we can interchange P and Q. Therefore, we assume that $|Q'| > |P'|$, we may have $w_1 = w_2$ or $w_1 \neq w_2$.

We claim that $P' \cap Q' \neq \emptyset$ and $P'' \cap Q' \neq \emptyset$. Otherwise, say P' and Q' are disjoint. Assume, without loss of generality, $\overleftarrow{C}^{[v,w_1]} = x_0(= v)x_1 \dots x_m(= w_1)$ contains at least three vertices. We may assume that both x_0x_1 and $x_{m-1}x_m$ are virtual edges because otherwise, $P'\overleftarrow{C}_{\Pi}^{[x_0,x_m]}Q'$ is longer than P. Since two (x_0, x_1)-paths $R^{(x_0,x_1)}$ and $S^{(x_0,x_1)}$ are in two different bridges of $\{x_0, x_1\}$, we may assume $R^{[x_0,x_1]}$ and P' only share a common vertex x_0. For the similar reason, we may assume that $R^{[x_{m-1}x_m]}$ and Q' only share a common vertex x_m. Then, P', $R^{(x_0,x_1)}$, $\overleftarrow{C}_{\Pi}^{[x_1,x_{m-1}]}$, $R^{(x_{m-1},x_m)}$, and Q' are internally vertex-disjoint. By concatenating these paths, we get a path longer than P, which gives a contradiction.

Hence, $Q' \cap P' \neq \emptyset$ and $Q' \cap P'' \neq \emptyset$. Along with TDA, this indicates that $w_1v \in E(C)$.

Let x (respectively y) be the first vertex along $P'^{[v,v_1]}$ (respectively $P''^{[v,v_2]}$) intersecting Q', that is, $Q' \cap P'^{[v,x]} = \{x\}$ and $Q' \cap P''^{[v,y]} = \{y\}$. Moreover, we assume that x is between w_1 and y on Q'.

Since $Q'^{[w_1,x]} \cup P'^{[x,v]}$ and $Q'^{[w_1,y]} \cup P''^{[y,v]}$ are two (w_1, v)-paths in G_{w_1v}, and the three tails Q', P', and P" are in the same$\{v, w_1\}$-bridge in G. We note that if $Q'' \cap P' \neq \emptyset$, then $w_2 = w_1$.

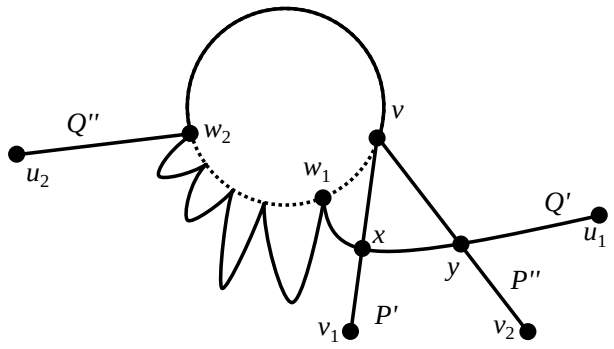

Figure (4.3) Illustration of x and y along with tails P', P'', Q', and Q''.

Let Θ_{vx} be the union of the paths $P'^{[v,x]}$, $P''^{[v,y]} \cup Q'^{[y,x]}$, and $Q'^{[x,w_1]} \cup Q^{[w_1,w_2]} \cup \overrightarrow{C}_{\Pi}^{[w_2,v]}$. Clearly, Θ_{vx} is a Θ-graph. Applying Lemma 1 three times to Θ_{vx}, we deduce $P''^{[y,v_2]} \cap$

$Q'^{[w_1,x]} = \emptyset$, $Q'' \cap P'' = \emptyset$, and $Q'' \cap P'^{[v,x]} = \emptyset$.

We claim $Q'^{(x,u_1]} \cap P'^{(x,v_1]} \neq \emptyset$. Otherwise, let $R_1 = P'^{[x,v_1]} \cup Q'^{[x,u_1]}$ and $R_2 = P''^{[v_2,v]} \cup P'^{[v,x]} \cup Q'^{[x,w_1]} \cup Q^{[w_1,w_2]} \cup Q''^{[w_2,u_2]}$ be two walks. Since we assume $Q'^{(x,u_1]} \cap P'^{(x,v_1]} = \emptyset$, R_1 is a path. Note that $P''^{[v_2,v]} \cup P'^{[v,x]} = P^{[v_2,x]}$ and $Q'^{[x,w_1]} \cup Q^{[w_1,w_2]} \cup Q''^{[w_2,u_2]} = Q^{[x,u_2]}$. Applying Lemma 1 to Θ_{vx}, we have $P^{[v_2,x]}$ and $Q^{[x,u_2]}$ are internally vertex-disjoint, so R_2 is also a path.

Following definition, we have

$$|R_1| + |R_2| = |P| + |Q|.$$

By Claim 1 and the fact that $R_1 \cap C = \emptyset$, we know R_1 is not a longest path, i.e. $R_1 \notin \mathcal{L}$, so $|R_2| > |P|$, which also gives a contradiction.

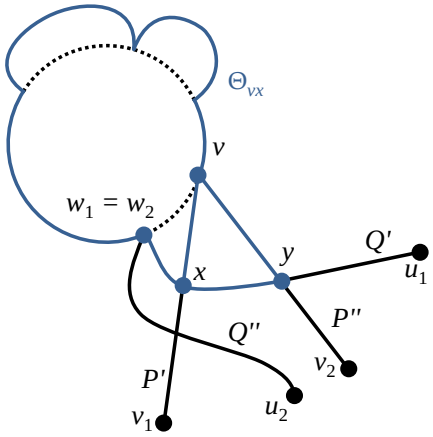

Figure (4.4) Illustration of the Θ-graph

Since $Q'^{(x,u_1]} \cap P'^{(x,v_1]} \neq \emptyset$, applying Lemma 1 to Θ_{vx} again, we get $P'^{(x,v_1]} \cap Q'^{[w_1,x)} = \emptyset$. Since $R^{(v,w_1)}$ and $S^{(v,w_1)}$ belong to two different bridges of $G_{vw_1} - \{v, w_1\}$, we may assume that $R^{(v,w_1)}$ is not in the bridge of $\{v, w_1\}$ containing x. Let $\overrightarrow{C}^{[w_1,v]}$ be the segment of $C - vw_1$. Clearly, $|\overrightarrow{C}^{(v,w_1)}| \geq 1$ since $|C| \geq 3$.

Let $R_1 := P'^{[v_1,x]} \cup Q'^{[x,w_1]} \cup \overrightarrow{C}_{\Pi}^{[w_1,v]} \cup P''$. Since $(P'^{[v_1,x]} \cup P''^{(v,v_2]}) \cap Q'^{(x,w_1)} = \emptyset$ and all these three paths are internally vertex-disjoint from $\overrightarrow{C}_{\Pi}^{[w_1,v]}$, R_1 is indeed a path.

We now define a walk R_2 as follows:

- If $R^{(v,w_1)} \cap Q'' = \emptyset$, let $R_2 := Q'^{[u_1,x]} \cup P'^{[x,v]} \cup R \cup Q^{[w_1,w_2]} \cup Q''^{[w_2,u_2]}$;

- If $R^{(v,w)} \cap Q'' \neq \emptyset$ (in this case, $w_2 = w_1$ is adjacent to v in C), let $R_2 := Q'^{[u_1,x]} \cup P'^{[x,v]} \cup \overrightarrow{C}_{\Pi}^{[w_1,v]} \cup Q''^{[w_2,u_2]}$.

Note that $Q^{[w_1,w_2]} \cup Q''^{[w_2,u_2]} = Q^{[w_1,u_2]}$, which are written separately in the definition of R_2 for the purpose of emphasizing their locations. Since $(Q'^{[u_1,x)} \cup Q^{[w_1,u_2]}) \cap R^{(v,w_1)} = \emptyset$ in the first case and $(Q'^{[u_1,x)} \cup Q^{[w_1,u_2]}) \cap \overrightarrow{C}_{\Pi}^{[w_1,v]} = \emptyset$ in the second case and these corresponding paths are internally vertex-disjoint from $P'^{[v,x]}$, R_2 is also a path.

By counting vertices in R_1 and R_2, we get

$$|R_1| + |R_2| \geq |P| + |Q| + |\overrightarrow{C}^{(w_1,v)}| > |P| + |Q|,$$

since $|\overrightarrow{C}^{(w_1,v)}| = |C| - 2 > 0$, which gives a contradiction to assumption that both P and Q are longest paths. □

This contradiction completes our proof. □

PART 5

INTERSECTING SUBTREE LEMMA

The Helly property is based off Helly's theorem [2] which, for 1-dimensional space, states that a set of convex, pairwise intersecting subsets has a non-empty intersection. For a graph $G = (V(G), E(E(G))$, a convex set can be defined as a set of vertices $S \subseteq V(G)$ such that for any pair of vertices $u, v \in S$, all vertices that lie on a shortest path between u and v belong to S [6]. From here, it is clear that any set of pairwise intersecting subtrees of a tree has the Helly property; however, for completeness and clarity, a proof is given below.

Lemma 7 (Intersecting Subtree Lemma)**.** *Let T be a tree, and let $\mathcal{T}$ be a set of connected subtrees of T. If $V(T_i) \cap V(T_j) \neq \emptyset$ for every $T_i, T_j \in \mathcal{T}$, then $\bigcap_{T_i \in \mathcal{T}} V(T_i) \neq \emptyset$. That is, $\mathcal{T}$ has the Helly property.*

Proof. Let v be the root of T. We start by defining an ordering $\prec$ of $V(T)$ where $x \prec y$ if and only if x is on the unique path connecting v and y in T.

With this definition, it is clear that $x \prec x$ for all $x \in V(T)$. If $x \prec y$ and $y \prec x$, then it must be that $x = y$ because T is a tree. Additionally, if $x \prec y$ and $y \prec z$, then we see that $x \prec z$. Thus, we can see that $(T, \prec)$ is a poset with v as the minimum element of T.

For any two subtrees, T_i, T_j, in $\mathcal{T}$, let $h(T_i, T_j)$ denote the minimum element of $V(T_i) \cap V(T_j)$. Among all the distances $h(T_i, T_j)$, let x be the maximal of all $h(T_i, T_j)$. Without loss of generality, we can assume that $p \in V(T_1) \cap V(T_2)$, so $x = h(T_1, T_2)$.

It must be that x is a minimum point for T_1 or for T_2, otherwise contradicting our choice of x. Assume that x is neither the minimum point for T_1 nor for T_2. Then, there exists $y \in V(T_1)$ such that $y \prec x$, and there exists $z \in V(T_2)$ such that $z \prec x$. Because T is a tree, there is only one path connecting v and x; therefore, it must be that either $z \prec y$ or $y \prec x$. Assume $z \prec y$. Because T_2 is connected and $z \prec y \prec x$, it must be that $y \in V(T_2)$, contradicting the fact that $x = h(T_1, T_2)$.

Let x be the minimum point for T_1. We claim that $x \in V(T_i)$ for each $T_i \in \mathcal{T}$.

Suppose that there exists $T_i \in \mathcal{T}$ such that $x \notin V(T_i)$, then because $V(T_i) \cap V(T_1) \neq \emptyset$ and $x \prec y$ for all $y \in V(T_1)$, it must be that $x \prec h(T_i, T_1)$. Because x is the maximal of these elements, it must be that $x = h(T_i, T_1)$. Hence, $x \in V(T_i)$. □

Recall that a chordal graph is the intersection graph of subtrees of some tree [63]. Let G be a chordal graph whose vertices correspond to subtrees of a tree T. For a connected subgraph H of G, we say that the *support of* H, denoted $Supp(H)$, is the union of the subtrees corresponding to the vertices in $V(H)$. By Lemma 7, if $\mathcal{H}$ is a set of subgraphs, if $Supp(H_i \cap H_j) \neq \emptyset$ for all $H_i, H_j \in \mathcal{H}$, then $Supp(\mathcal{H}) = \{Supp(H) | H \in \mathcal{H}\}$ has a non-empty intersection.

PART 6

INTERVAL GRAPHS

Theorem 2. *If G is a connected interval graph, then there is a vertex common to all longest paths of G.*

Proof. For every vertex $v \in V(G)$, we denote the interval mapped to v by $I_v = (a_v, b_v)$. Let $\mathcal{P}$ be the set of all longest paths of G. We note that for a path P, $Supp(P) = \bigcup_{v \in V(P)} I_v$.

Because we know that for any two longest paths P_i and P_j, there is a common vertex, i.e. $V(P_i) \cap V(P_j) \neq \emptyset$, this implies that $Supp(P_i \cap P_j) \neq \emptyset$. Thus, applying Lemma 7, we know that $\mathbb{R} \supseteq \bigcap_{P \in \mathcal{P}} Supp(P) \neq \emptyset$. For brevity, let $S = \bigcap_{P \in \mathcal{P}} Supp(P)$. Note that S is an open interval as it is the intersection of open intervals.

Let $x = \inf S$. We take the vertex v to be such that $\inf I_v \leq x < \sup I_v$ with maximum $\sup I_v$. We know that such a v exists as x is the infimum of S, so there must be some interval I_v satisfying these inequalities. We claim that $v \in V(P)$ for all $P \in \mathcal{P}$. We also note that for some $R \in \mathcal{P}$, $x = \inf Supp(R)$ else x would be smaller.

Suppose there exists $Q \in \mathcal{P}$ such that $v \notin V(Q)$. As $x = \inf S$, then $\inf Supp(Q) \leq x$. To prove the claim, we consider two exclusive cases.

Case 1. There exists $y \in V(Q)$ such that $a_y \geq x$.

Then there must exist two consecutive vertices $y, z \in V(Q)$ such that we see $a_y \leq x \leq a_z$. As y and z are consecutive on Q, we know that $I_y \cap I_z \neq \emptyset$, i.e. $a_z < b_y$. Because of our choice of v, we know that $b_y \leq b_v$. Combining these inequalities, we see that $a_y \leq x \leq a_z < b_y \leq b_v$.

As $x \in I_v$, we can see $I_v \cap I_y \neq \emptyset$ and $I_v \cap I_z \neq \emptyset$. Therefore, inserting v between y and z in Q results in a path longer than Q, a contradiction.

Case 2. For all $y \in V(Q)$, $a_y < x$.

Then $V(Q) \cap V(R) = \emptyset$ because for all $z \in V(R)$, $a_z \geq x$. However, as Q and R are longest paths, it must be that $V(Q) \cap V(R) \neq \emptyset$, a contradiction. $\square$

PART 7

SUBTREES OF SPIDER GRAPHS

Theorem 3. *If G is a connected intersection graph for a spider graph, then there is a vertex common to all longest paths of G.*

Proof. For every vertex $v \in V(G)$, we denote the subtree mapped to v by T_v. Let $\mathcal{P}$ be the set of all longest paths of G. Because we know that for any two longest paths P_i and P_j, there is a common vertex, i.e. $V(P_i) \cap V(P_j) \neq \emptyset$, this implies that $Supp(P_i \cap P_j) \neq \emptyset$. Thus, applying Lemma 7, we know that $\bigcap_{P \in \mathcal{P}} Supp(P) \neq \emptyset$. For brevity, let $S = \bigcap_{P \in \mathcal{P}} Supp(P)$. Note that S is a subtree as it is the intersection of subtrees.

Let T be the spider graph from which the subtrees corresponding to $V(G)$ come. Let $c \in V(T)$ be the center of the spider, the one vertex of T such that $d(c) \geq 3$.

Case 1. We have $c \notin S$.

As $c \notin S$, then S is confined entirely to one leg, denoted L, of the spider graph T. Let p be the point of S closest to c. We orient T so that p is the left-most point of S and so that the other legs and c are to the left p and come "before" it. We note that for some $R \in \mathcal{P}$, p is the left-most point of $Supp(R)$ else p would be closer to c or $c \in S$.

We take the vertex v to be such that T_v contains vertices coming before p and $|T_v \cap S|$ is maximum. We claim that $v \in S$. Suppose there exists $Q \in \mathcal{P}$ such that $v \notin V(Q)$. We note that $Supp(Q)$ either contains points before p or has its left-most point as p.

If there exists $x \in V(Q)$ such that T_x contains no points coming before p, then there must be two consecutive vertices $y, z \in V(Q)$ such that T_y contains points coming before p and T_z contains no points coming before p. Hence, as $T_y \cap T_z \neq \emptyset$, we know that T_y contains points coming after p as well, so we also have $T_v \cap T_y \neq \emptyset$ and $T_v \cap T_z \neq \emptyset$. Therefore, inserting v between y and z in Q results in a path longer than Q, a contradiction.

If for every $x \in V(Q)$, T_x contains points coming before p, then $V(Q) \cap V(R) = \emptyset$

because for all $y \in V(R)$, T_y contains no points coming before p. However, as Q and R are longest paths, it must be that $V(Q) \cap V(R) \neq \emptyset$, a contradiction.

Case 2. We have $c \in S$.

First, we define a *tail* of a path. Let Q be a subpath of a path P such that either Q is an endpoint, v_0, of P such that $c \in V(T_{v_0})$ or Q contains an endpoint of P and the other endpoint, v_m, of Q is the only vertex of Q such that $c \in V(T_{v_m})$. Each longest path P has $c \in Supp(P)$, so each longest path has two tails.

Let R be a longest tail among all tails of longest paths. As only one endpoint of R contains c, we assume that all "non-center" vertices of R are contained to one leg L of T.

Let v be a vertex of G such that $c \in V(T_v)$ and $|T_v \cap L|$ is maximum. We claim that $v \in V(P)$ for all $P \in \mathcal{P}$. Suppose there exists $Q \in \mathcal{P}$ such that $v \in V(Q)$. We consider two cases.

Case 2.1 Q contains an "inverval vertex" of R but not c, i.e. there exists $w \in V(Q)$ such that $T_w \subseteq L$.

Then, we must have two consecutive vertices $x, y \in V(Q)$ such that $c \in T_x$ and $T_y \subseteq L$ since every longest path goes through the center. Then, $T_v \cap T_x \neq \emptyset$ and $T_v \cap T_y \neq \emptyset$. Therefore, $xv, vy \in E(G)$, and inserting v between x and y results in a path longer than Q, a contradiction.

Case 2.2 Q does not contain an "interval vertex" of R, i.e. there does not exist $w \in V(Q)$ such that $T_w \subseteq L$.

Let $R = z_1, z_2, ..., z_m$. Let $Q^* = q_1, q_2, ..., q_k$ be a tail of Q with $c \in T_{q_1}$. We replace Q^* in Q by $q_1, v, z_2, z_3, ..., z_m$ to obtain Q'. We know that $z_i \notin V(Q)$ for $2 \leq i \leq m$ because of the assumption for the case. We also are assuming that $v \notin V(Q)$. Thus, Q' is a path.

Now, we look at the length of $|Q'|$. Recall that by our choice of R, we have that $|R| \geq |T|$.

$$|Q'| = |Q| - (|T| - 1) + 1 + (|R| - 1) = |Q| - |T| + |R| + 1 \quad \geq |Q| + 1 \tag{7.1}$$

This is a contradiction.

□

PART 8

CIRCULAR ARC GRAPHS

Let G be a circular arc graph whose vertices correspond to the open arcs of a circle C. For every $v \in V(G)$, let A_v denote the open arc on C corresponding to v. If there exists $v \in V(G)$ such that $A_v = C$, then clearly v is on every longest path, and the set of longest paths of G has the Helly property. Therefore, assume no such v exists. We let C have an orientation so that l_v and r_v are the two endpoints of arc A_v, respectively. Again, we let $\mathcal{P}$ denote the set of longest paths of G.

Lemma 8 (Balister, Győri, Lehel, Schelp [39])**.** *Let $X = \{x_1, x_2, ..., x_{t+1}\}$ be a set of real numbers, and let $J_1, J_2, ..., J_t$ be a sequence of open real intervals with $x_k, x_{k+1} \in J_k$ for every $1 \leq k \leq t$. If $x_{i_1} < x_{i_2} < ... < x_{i_{t+1}}$ are the elements of X in increasing order, then the intervals have a permutation $J_{j_1}, J_{j_2}, ..., J_{j_t}$ such that $x_{i_k}, x_{i_{k+1}} \in J_{j_k}$ for every $1 \leq k \leq t$.*

Lemma 9. *Let $P \in \mathcal{P}$ and $u, v \in V(G)$ with $A_u \subseteq A_v$. If $u \in V(P)$, then $v \in V(P)$.*

Proof. Suppose that $v \notin V(P)$. If u is an endvertex of P, then v can be appended to the beginning or end of P, extending it, a contradiction. Thus, u is not an endvertex of P.

Let w be the vertex consecutively after u in P. As $A_u \subseteq A_v$, because $A_u \cap A_w \neq \emptyset$, this implies that $A_v \cap A_w \neq \emptyset$. Therefore, v can be inserted between u and w, extending P, a contradiction. □

Theorem 4. *If G is a connected circular arc graph, then there is a vertex common to all longest paths of G.*

Proof. Let G be a circular arc graph, and let C be the circle from which the arcs that are associated with $V(G)$ come. For every vertex $v \in V(G)$, we denote the arc mapped to v by $A_v = (l_v, r_v)$. Let $\mathcal{P}$ be the set of all longest paths of G. We note that for a path P, $Supp(P) = \bigcup_{v \in V(P)} A_v$.

As Balister, Győri, Lehel, and Schelp noted [39], if $\bigcup_{v \in V(G)} A_v$ is not the entire circle, then G can be thought of as an interval graph, so the proof is given by Theorem 2. If there is a vertex $v \in V(G)$ such that $A_v = C$, then we are done as v must be on all longest paths.

Let $A_v = (l_v, r_v)$ be an arc such that the A_v-dominated path $P^{A_v} = vv_2...v_l$ is maximum over all arc-dominated paths. As P^{A_v} is A_v-dominated, this means that $A_{v_i} \subseteq A_v$ for $i = 2, 3, ..., l$. We claim that $v \in P$ for all $P \in \mathcal{P}$. From our choice of A_v, we know that there does not exist an arc A such that $A \supseteq A_v$ by Lemma 9.

Suppose there exists $Q \in \mathcal{P}$ such that $v \notin V(Q)$. Let $Q = q_1q_2...q_m$. We know that $V(Q) \cap V(P^{A_v}) = \emptyset$. If there exists $v_i \in V(P^{A_v})$ such that $v_i \in V(Q)$, then by Lemma 9 it must be that $v \in V(Q)$, a contradiction.

We pick $X = x_1, x_2, ..., x_{m+1}$ with $X \subseteq C \setminus [l_v, r_v] = (r_v, l_v)$ such that $x_i, x_{i+1} \in A_{q_i}$ for $i = 1, 2, ..., m$. From Lemma 8, we can rearrange the elements of X so that $x_{i_1} < x_{i_2} < ... < x_{i_{m+1}}$ and $x_{i_{k+1}}$ is immediately clockwise from x_{i_k} for $k = 1, 2, ..., m$. Then we can rearrange the path $Q = q_1q_2...q_m$ to obtain $Q^* = q_{j_1}q_{j_2}...q_{j_m}$ so that $V(Q) = V(Q^*)$ and $x_{i_k}, x_{i_{k+1}} \in A_{q_{j_k}}$ for $k = 1, 2, ..., m$. We know that $A_{q_{j_1}} \cap A_v = \emptyset$ and $A_{q_{j_m}} \cap A_v = \emptyset$. We consider two cases.

Case 1. There exists $q_{j_k} \in V(Q^*)$ such that $l_v \in A_{q_{j_k}}$.

Let j_k be the largest index such that $l_v \in A_{q_{j_k}}$. If $j_k = m$, then Q^*v is a longer path, so we assume that $j_k < m$. As $l_v \notin A_{q_{j_t}}$ for all $k < t \leq m$, we know that $x_{i_k} < x_{i_t} < r_{j_t} < l_v < r_{j_k}$ for all $t = k+1, k+2, ..., m$. We claim that we may assume that $A_{q_{j_t}} \subseteq A_{q_{j_k}}$ for $t = k+1, k+2, ..., m$.

Suppose there exists $t \in k+1, k+2, ..., m$ such that $A_{q_{j_t}} \nsubseteq A_{q_{j_k}}$. Then, we can see that $l_{q_{j_t}} < l_{q_{j_k}} < x_{i_k} < x_{i_{k+1}} \leq x_{i_t} < x_{i_{t+1}} < r_{q_{j_t}} < l_v < r_{q_{j_k}}$. Hence, $x_{i_k}, x_{i_{k+1}} \in A_{q_{j_t}}$ and $x_{i_t}, x_{i_{t+1}} \in A_{q_{j_k}}$. This implies that $x_{i_k} \in A_{q_{j_{k-1}}} \cap A_{q_{j_t}}$, $x_{i_{k+1}} \in A_{q_{j_{k+1}}} \cap A_{q_{j_t}}$, and $x_{i_t} \in A_{q_{j_{t-1}}} \cap A_{q_{j_k}}$. Also, if $t < m$, we have $x_{i_{t+1}} \in A_{q_{j_{t+1}}} \cap A_{q_{j_k}}$. Therefore, we can switch q_{j_k} and q_{j_t} to have a path that is a reordering of $V(Q)$ satisfying the conditions of Lemma 8.

Thus, we have that $A_{q_{j_t}} \subseteq A_{q_{j_k}}$ for $t = k+1, k+2, ..., m$. Now, consider two subpaths of Q^*, $Q_1^* = q_{j_1}q_{j_2}...q_{j_{k-1}}$ and $Q_2^* = q_{j_k}q_{j_{k+1}}...q_{j_m}$. As we showed earlier, we know that Q_2^*

is a $A_{q_{j_k}}$-dominated path. Based on our choice of v, this means that $|Q_2^*| \leq |P^{A_v}|$. As $A_{q_{j_k}} \cap A_v \neq \emptyset$, consider the walk $R = q_{j_1}q_{j_2}...q_{j_k}vv_2...v_l$. Because $V(Q^*) \cap V(P^{A_v}) = \emptyset$, R is a path. We see that $|R| = |Q_1^*| + 1 + |P^{A_v}| > |Q_1^*| + |P^{A_v}| \geq |Q_1^*| + |Q_2^*| = |Q^*|$. This means that $|R| > |Q^*|$, a contradiction.

Case 2. There does not exist $q_{j_k} \in V(Q^*)$ such that $l_v \in A_{q_{j_k}}$.

Let j_k be the largest index such that the distance from $r_{q_{j_k}}$ to l_v going clockwise is shortest. Now, as G is connected and $\bigcup_{v \in V(G)} A_v$ is the entire circle, we know there is a $q_{j_k}v$-path (a path connecting q_{j_k} and v) intersecting the entire closed arc $[r_{q_{j_k}}, l_v]$. Let $S = s_1s_2...s_b$ be such a path of shortest length, so $q_{j_k}s_1s_2...s_bv$ is a path. We note that $|S| \geq 1$.

Clearly, $V(Q^*) \cap V(S) = \emptyset$ and $V(P^{A_v}) \cap V(S) = \emptyset$, else S could be shorter.

If $j_k = m$, then $Q^*SP^{A_v}$ is a longer path, so we assume that $j_k < m$. We claim that we may assume that $A_{q_{j_t}} \subseteq A_{q_{j_k}}$ for $t = k+1, k+2, ..., m$.

Suppose there exists $t \in k+1, k+2, ..., m$ such that $A_{q_{j_t}} \nsubseteq A_{q_{j_k}}$. Then, we can see that $l_{q_{j_t}} < l_{q_{j_k}} < x_{i_k} < x_{i_{k+1}} \leq x_{i_t} < x_{i_{t+1}} < r_{q_{j_t}} < r_{q_{j_k}} < l_v$. Hence, $x_{i_k}, x_{i_{k+1}} \in A_{q_{j_t}}$ and $x_{i_t}, x_{i_{t+1}} \in A_{q_{j_k}}$. This implies that $x_{i_k} \in A_{q_{j_{k-1}}} \cap A_{q_{j_t}}$, $x_{i_{k+1}} \in A_{q_{j_{k+1}}} \cap A_{q_{j_t}}$, and $x_{i_t} \in A_{q_{j_{t-1}}} \cap A_{q_{j_k}}$. Also, if $t < m$, we have $x_{i_{t+1}} \in A_{q_{j_{t+1}}} \cap A_{q_{j_k}}$. Therefore, we can switch q_{j_k} and q_{j_t} to have a path that is a reordering of $V(Q)$ satisfying the conditions of Lemma 8.

Thus, we have that $A_{q_{j_t}} \subseteq A_{q_{j_k}}$ for $t = k+1, k+2, ..., m$. Now, consider two subpaths of Q^*, $Q_1^* = q_{j_1}q_{j_2}...q_{j_{k-1}}$ and $Q_2^* = q_{j_k}q_{j_{k+1}}...q_{j_m}$. As we showed earlier, we know that Q_2^* is a $A_{q_{j_k}}$-dominated path. Based on our choice of v, this means that $|Q_2^*| \leq |P^{A_v}|$.

Consider the walk $R = q_{j_1}q_{j_2}...q_{j_k}s_1s_2...s_bvv_2...v_l$. Because $V(Q^*) \cap V(P^{A_v}) = \emptyset$, $V(Q^*) \cap V(S) = \emptyset$, and $V(P^{A_v}) \cap V(S) = \emptyset$, R is a path. We see that $|R| = |Q_1^*| + 1 + |S| + |P^{A_v}| > |Q_1^*| + |P^{A_v}| \geq |Q_1^*| + |Q_2^*| = |Q^*|$. This means that $|R| > |Q^*|$, a contradiction.

□

PART 9

PREDICTING VIDEO FACE RECOGNITION PERFORMANCE IN NEW SETTINGS

9.1 PaSC Challenge and Data Set

The Point-and-Shoot Face Recognition Challenge (PaSC) was designed to advance the development of face recognition algorithms on videos taken with digital point and shoot cameras, particularly for handheld cameras found in cell phones [68]. A brief overview of the PaSC is given in Section 9.1.1. The rest of this section introduces the factors discussed in this dissertation: the location factor (Section 9.1.2), video-based factors (Section 9.1.3), and demographic factors (Section 9.1.4).

9.1.1 Data Set

A key feature of the PaSC is that the data was collected following a statistical experimental design; we call this a designed data collection. With a designed data collection, we can analyze how factors, such as location, pose, or face size, effect performance. In our analysis, we focus on the effect of factors, e.g. location and sensor. This is possible because videos in the PaSC are taken from six locations with six sensors, five of those being handheld.

In the video portion of the PaSC, 2802 videos of 265 subjects were taken over 7 different weeks at the University of Notre Dame in the spring semester of 2011. The videos show people carrying out tasks rather than looking into a camera. Collection was carried out according to a plan–a script–in which generally a person entered a scene, approached some designated spot, carried out an action, and then left the scene. The videos typically begin as the person is moving into the scene and terminate as the person is leaving.

Out of seven total weeks of collection, each subject is present in videos for at least four of the weeks, implying the differences in weeks' performances is not due to the subjects. (Hand-

held) Video length ranges roughly between 50 and 400 frames with most videos containing between 200 and 250 frames, and the resolutions ranged between 640×480 to 1280×720.

There were six different locations with six different sensors. Five of the sensors were handheld, and these varied by week. Additionally, data was collected by a tripod-mounted sensor, and this sensor filmed the same actions at the same location and time as the handheld sensor of the week.

Figure (9.1) Sampled portions of video frames from PaSC videos
These portions indicate some of the situations that make recognition challenging. Courtesy of Beveridge et al. [69].

Figure 9.1 shows a sample of frames from PaSC videos from different locations. Characterizing the videos are four primary factors: location, action being performed, video camera (sensor), and person in the video (subject).

9.1.2 Location Factor

One aspect the design of this data set allows us to analyze is how an algorithm performs when restricted to pairs of videos from certain locations. During each week, the videos were collected with a new combination of location and action taking place, for example picking up a newspaper in an office. No combination of location and action was repeated on subsequent weeks. Table 9.1 shows a summary of the location, handheld camera, and action combinations.

Each location and action combination was captured on a specific week by two different

Table (9.1) Location, camera, and action combinations

The abbreviations for the location is in the right column.

Sensor	Location	Action	Abbrev.
Flip Mino F360B	canopy	golf swing	Ca
Kodak Zi8	canopy	bag toss	Ca
Samsung M. CAM	office	pickup newspaper	Pa
Sanyo Xacti	lab 1	write on easel	Ea
Sanyo Xacti	lawn	blow bubbles	Bu
Nexus Phone	hallway	ball toss	Ba
Kodak Zi8	lab 2	pickup phone	Ph

cameras, one being handheld. Consequently, each video depicts a single subject at a certain location doing a specific action captured by one particular sensor, e.g. for a specific subject, there is exactly one video depicting the subject on the lawn blowing bubbles captured by a Sanyo Xacti. There is also a video of the subject blowing bubbles on the lawn captured by the tripod-mounted sensor, a Panasonic HD700. From Table 9.1, it is clear that the handheld sensors are confounded with the locations and actions.

In the findings below, the influence that location and action combinations exert over performance is strong, and the abbreviations introduced in Table 9.1 will be used when reporting results. Therefore here, briefly, is a bit more information about each. The **canopy** (Ca) was a white pop-up material structure setup outside in bad weather. Two actions were carried out on different days. The first was swinging a golf club, and the second was tossing a bean bag. The **office** (Pa) was a large well-lit room where a subject picked up and looked at a newspaper. In **Lab 1** (Ea) each subject wrote on a large floor standing easel set out in a large open lab space. The **lawn** (Bu) was an open grassy area in a plaza with bright sun. Subjects approached a table and blew bubbles. The **hallway** (Ba) was an interior space of an older building with relatively dark stone walls where subjects threw a toy basketball. In **lab 2** (Ph) a subject picked up a phone in a relatively cluttered lab area.

Figure 9.2 shows four zoomed-in clips from four different videos. The upper left clip is from the **office**. The upper right is from the **canopy**. The lower left is from **lab 2**, and the lower right is from the **lawn**. These frames are characteristic in several respects, for example

Figure (9.2) Clips of two people sampled from four PaSC handheld videos
All four videos were taken at different locations: two outdoors and two indoors.

suggesting the range of lighting conditions and also the fact that in general subjects are not attending to the camera.

As videos are compared in pairs, the location factor is defined by location-pairs, i.e. the locations of the videos for a given pair. In total, there are 22 location-pairs. For 6 pairings the videos are from the same location and collected in the same week; these only include impostor pairs, i.e. pairs of videos of different people. However, we focus mainly on cross-week comparisons, i.e. video-pairs in which the weeks of capture are different. There are 16 cross-week location-pairs. In 15 of the cross-week location-pairs, the videos were collected at different locations from different weeks, but for one pair, the videos were collected at the same location (canopy) on different weeks.

9.1.3 Video-Based Factors

Location, action, and sensor are not the only factors effecting performance. Another class of factors effecting performance comes directly from the videos themselves; that is,

these factors, called video-based factors, are dependent on the video from which they are estimated. As we show later in Section 9.4, video-based factors can encode properties of a location-pair. For our work, we measure this encoding by looking at aggregate statistics of video-based factors from all video-pairs of the location-pair.

We consider three video-based factors: face size, face confidence, and yaw. The factors come as extensions of image-based factors. Estimated by the Pittsburgh Pattern Recognition (PittPatt) face recognition SDK 5.2.2, face size is the number of pixels between the eyes, face confidence is PittPatt's self-assessment of how certain the algorithm was in detecting the true face, and yaw is the measurement of how far the face was turned to the left or right.

The image-based factors face size, face confidence, and yaw are estimated for the frames of the videos. For each video, the mean of each factor is calculated. This provides the factors for each video. However, we concentrate on comparing pairs of videos. As each video in a pair has its own set of video factors, we need to combine these paired factors. For the face size factor, we always use the smaller of the two values for each pair of videos; we do the same for the face confidence factor. On the other hand, for the yaw factor, we always use the difference of the two values.

The real-valued factors are converted to levels by ordering video-pairs from smallest to largest factor value and then dividing them into n equal sized bins. The result is n levels ranging from smallest to largest factor value. The PittPatt SDK 5.2.2 software estimated these factors for the frames of the videos, and the generalizations to videos and video-pairs follow the methods of Lee et al. [70]. For more details on the extension see Section A.

9.1.4 Demographic Factors

The demographic factors, specifically gender and race, are encoded based on the possible entries for each subject in a pair of videos. For gender, each subject can be male or female, so the levels are female-male (F/M), male-male (M/M), and female-female (F/F). For race, the majority of subjects are Caucasian or Asian, and we do not have enough subjects of any other race to perform analysis, so the levels are Caucasian-Asian (C/A), Caucasian-

Caucasian (C/C), and Asian-Asian (A/A).

9.2 Algorithms

Our analysis is performed on the four top performers in the Face and Gesture 2015 Person Recognition Evaluation [65]. The algorithms were state of the art and developed independently by four different research groups from four different countries on four different continents. Each algorithm is very different in how it computes a similarity score (the degree of similarity between two faces in two videos). This independence provides evidence that our conclusion will generalize to algorithms not included in this study.

Several other conditions were adopted to make sure the results were not tuned to the PaSC data set. First, the algorithms were not trained on subjects in the PaSC. Second, the algorithms were not trained on imagery from locations that are included in the PaSC. Third, cohort or gallery normalization using the PaSC imagery was not allowed.

The algorithms were developed by the Chinese Academy of Science (CAS), the Stevens Institute of Technology (SIT), The University of Ljubljana (Ljub), and the University of Technology Sydney (UTS).

9.3 Impostor-Pair Analysis for Location-Pairs

9.3.1 Range of Marginal FARs over Location-Pairs

It is well known that location significantly effects algorithm performance. The design of the PaSC data set enabled us to characterize the impact of location on performance. Previous studies have investigated the effect of location on verification rates [58], [70]. We proceed by examining the effect of location on the FAR and then look at the relationship between FAR and VR.

Since comparisons are between two videos, we look at performance for location-pairs. For the four algorithms in our study, we computed the FAR for the 22 location-pairs as described in Section 2.2. Figure 9.3 demonstrates how location factors effect FAR (upper

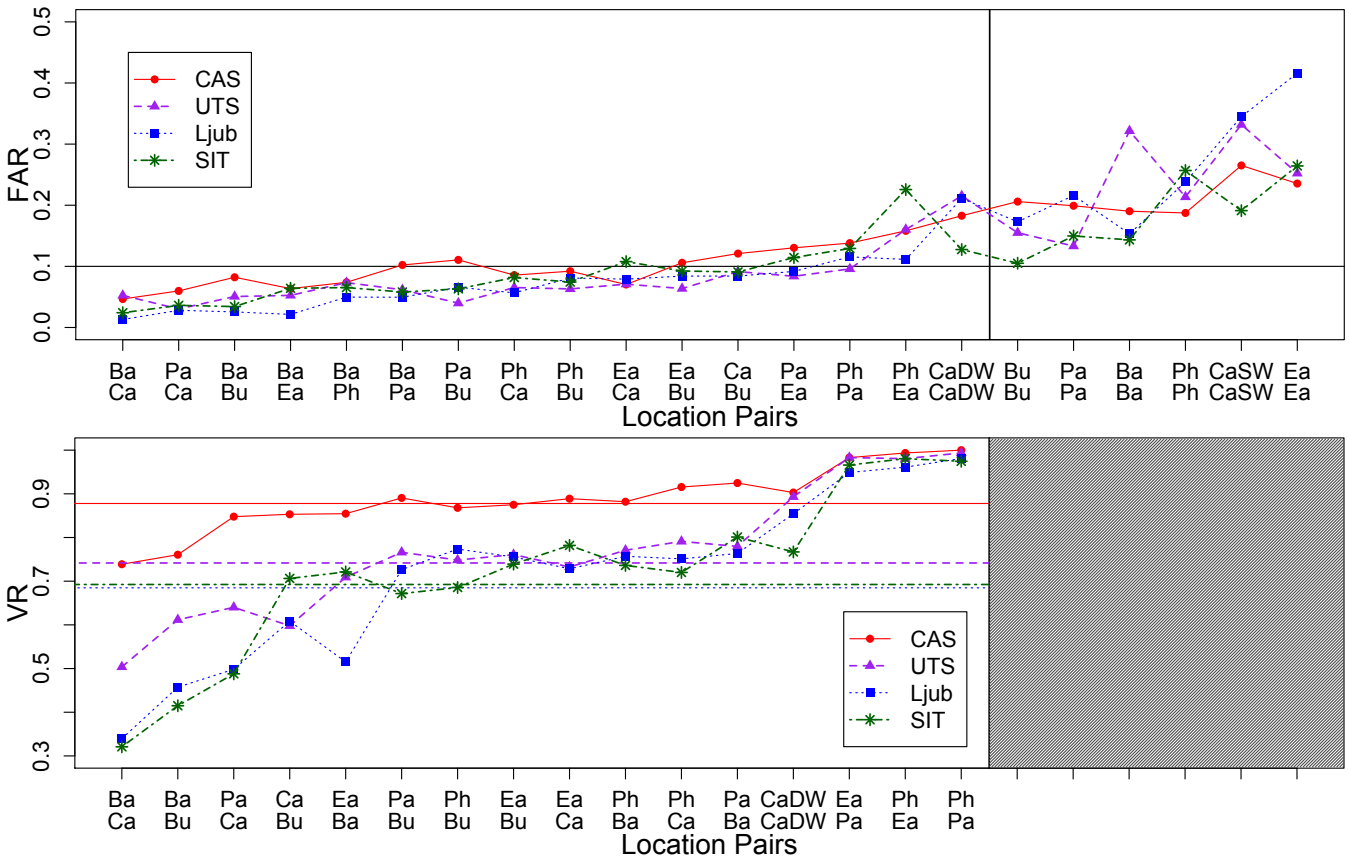

Figure (9.3) FAR and VR of each location-pair on handheld video-pairs for each algorithm

The FAR and VR are ordered by the mean rate over all the algorithms. The top graph is on FAR. The horizontal line corresponds to the global FAR = 0.10, and the vertical line between pairs CaDW-CaDW and Bu-Bu separates the pairs into cross-week (left) and same week. The bottom graph is on VR. The horizontal lines correspond to the global VR for each algorithm when the global FAR = 0.10. There are no same-week pairs for matches.

graph) and VR (lower) for the four algorithms on handheld video-pairs when the global FAR is set to 0.10. Along the horizontal axes are the pairs of locations described in Section 9.1.2. All 22 pairs are present in the upper graph, but only the 16 cross-week pairs are present in the lower graph because the same-week comparisons only contain impostor pairs. The vertical axes show the marginal FAR and VR values, respectively, using a τ_g that corresponds to a global FAR of 0.10. The location pairs are ordered by the mean rate over all the algorithms for both graphs. In the top graph, all location pairs to the left of the vertical line (from pairs Ba-Ca to CaDW-CaDW) are cross-week pairs; CaDW signifies canopy videos taken in different weeks. All pairs to the right consist of video-pairs taken in the same week. Figure B.1 in Section B demonstrates a similar effect in the tripod video-pairs.

The principal finding is that location exerts a dramatic influence over the impostor distribution and hence the marginal FAR. For handheld video-pairs, Algorithm Ljub has the greatest range in FAR from 0.01 to 0.42, and CAS has the smallest range from 0.05 to 0.24; for tripod video-pairs, Ljub still has the greatest range in FAR from 0.02 to 0.39, and CAS has the smallest range from 0.03 to 0.22. Table 9.2 shows the ranges for the cross-week location-pairs over both sets of video-pairs. For the handheld video-pairs, the FAR for the four algorithms CAS, UTS, Ljub, and SIT varies by a factor of 3.6, 7.33, 21, and 11.5, respectively. For the tripod video-pairs, the FAR for the algorithms CAS, UTS, Ljub, and SIT varies by a factor of 4.33, 7.67, 9, and 7, respectively. Prior work has already suggested the importance of location [58], [70]; this is the first clear evidence of how significantly it effects the impostor distribution.

Table (9.2) The cross-week ranges of location-pair marginal FAR location-pairs over both sets of video-pairs

Algorithm	Handheld	Tripod
CAS	$0.05 - 0.18$	$0.03 - 0.13$
UTS	$0.03 - 0.22$	$0.03 - 0.23$
Ljub	$0.01 - 0.42$	$0.02 - 0.18$
SIT	$0.02 - 0.23$	$0.03 - 0.21$

A related finding is the importance of the cross-week versus same-week distinction. For both sets of video-pairs, the mean cross-week marginal FAR averaged over the algorithms was 0.09 compared to 0.21 for same-week pairs. A recent related result on still face image by Sgori et al. [57] also showed higher FAR values for same day image-pairs compared to different day image-pairs. One important conclusion is that the presence of impostor pairs in a data set taken at the same time biases upward the expected FAR for the data set as a whole.

9.3.2 Do VR and FAR Track Together?

We will now look at the relationship between the location-pair FARs and VRs for the cross-week pairs. From Figures 9.3 and B.1, we can see that their appears to be some correlation between the marginal FARs and VRs. In both figures, most of the location-pairs do not change their placements in the orderings by more than three. Scatterplots in Figure 9.4 relate marginal VR to marginal FAR, eqs. 2.3 and 2.4, for the 16 cross-week location-pairs over the different sensor-pairs. The horizontal axis is the FAR on a log-scale, and the vertical axis is the VR on a linear scale. The points represent location-pairs over different sensor-pairs, and the line is a linear regressor. For all four algorithms, the regression line suggests a linear relationship between log(FAR) and VR. In other words, a location-pair that has a higher marginal VR will likely have a higher marginal FAR. Unfortunately, this linear relationship suggests that finding a location-pair that is easier than others is unlikely. We say a location-pair is easier if it has both a higher VR and a lower FAR than other pairs.

9.4 Impostor-Pair Analysis for Video-Based Factors

The impact of image- and video-based factors on verification rates has been extensively studied; however, their impact on the FAR has not been examined. We first look at the relationship between FAR and VR for three video-based factors and then investigate if there is an interaction between location-pairs and the video-based factors.

Figure 9.5 shows the trade-off between FAR and VR for face size. The procedure

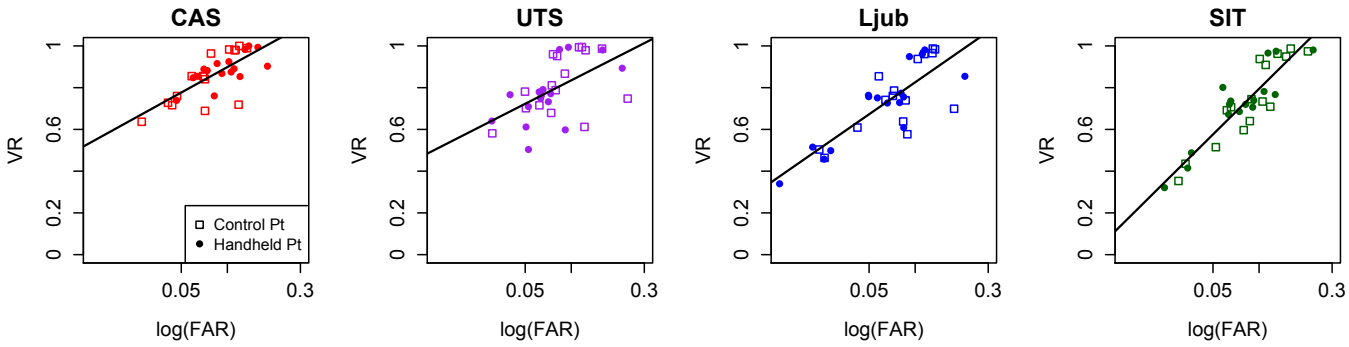

Figure (9.4) Scatterplots of VR vs log(FAR) of location-pairs over different sensor-pairs
The legend in the first graph applies to all.

described at the end of Section 9.1.3 for creating factor levels through sorting and binning was used to create 10 face size factor levels: smallest faces to largest faces. Each point in Figure 9.5 is plotted according to the average marginal VR and FAR for all those video-pairs at one face size level. A trend similar to that seen for location factors is evident, changes in face size associated with higher marginal VR correlate with higher marginal FAR. There is a similar relationship for yaw and face size; see Figure B.2 in Section B for the corresponding scatterplots.

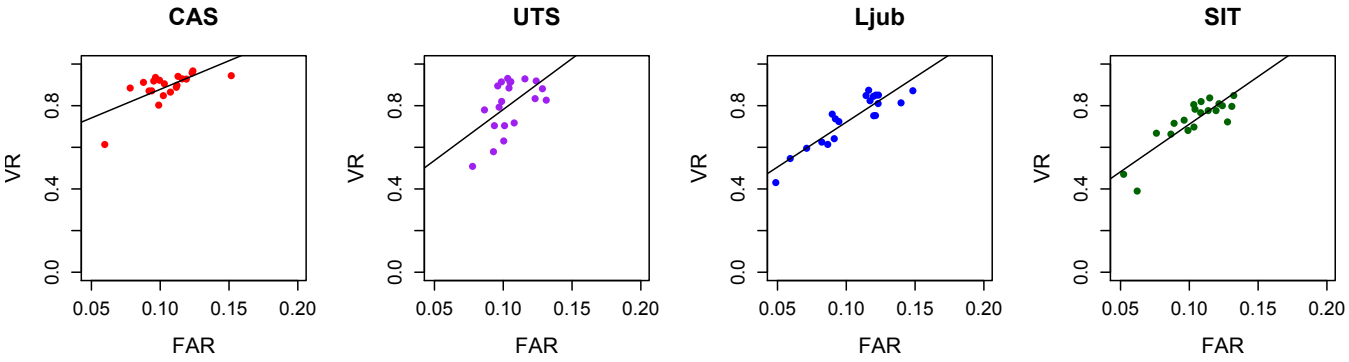

Figure (9.5) Scatterplots of VR vs FAR for Face Size over different sensor-pairs, divided into 10 bins, fitted with a linear regressor for each algorithm
Thresholds set to global FAR = 0.10.

Figure 9.6 highlights possible interactions between location and video factors for Algorithm Ljub. Like the scatterplots in Figure 9.4, each point corresponds to a location-pair and sensor-pair. Unlike in Figure 9.4, in Figure 9.6 circle size varies and is proportional the mean video factor for a location-pair. For the yaw-factor, all the circles are about the same size, which means that yaw does not interact with the location-pair. In contrast, a clear interaction effect between location and face size is evident: location-pairs with smaller VR and FAR tend to have small circle sizes and hence smaller mean face sizes. Figure 9.6 also suggests some interaction between location and face confidence.

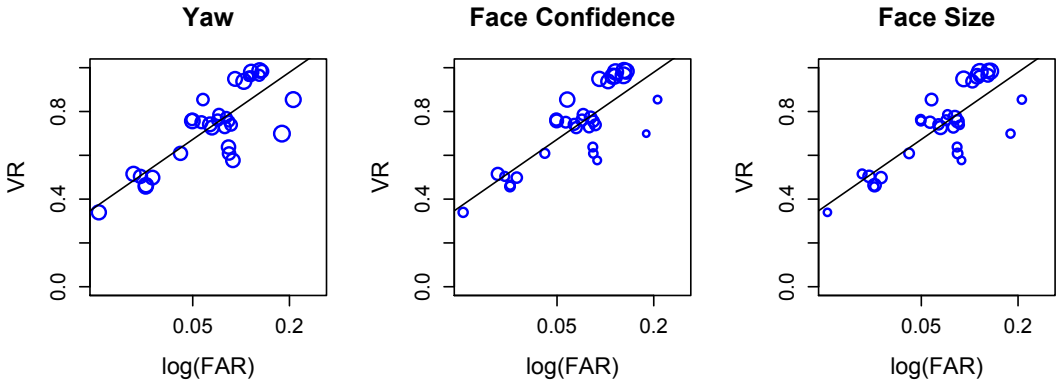

Figure (9.6) Interactions between Algorithm Ljub location-pairs from Figure 9.4 and each of the three video-based factors: yaw, face confidence, and face size. Each panel looks at the interaction for the factor in its title. The size of each circle is proportional to the mean of the factor for each location-pair.

This analysis was repeated for Algorithms SIT, UTS, and CAS, and the conclusions were the same. A complete set of plots for this analysis are in Figure B.3 in Section B. Across all four algorithms for all three video factors, we saw a trade-off between VR and FAR for different levels of each factor. Further analysis suggested an interaction between location and both face size and face confidence with face size having a larger interaction.

9.5 Impostor-Pair Analysis for Demographic Factors

It is known that gender and race effect the performance of algorithms [71], [72]. Figure 9.7 shows the effect of gender and race on the marginal FAR for Algorithm Ljub for handheld (top row) and tripod (bottom row) video-pairs. The corresponding results for all four algorithms are reported in Figure B.4 in Section B. The results show that cross-gender and cross-race impostor-pairs have a lower FAR. This is consistent with O'Toole et al. [71].

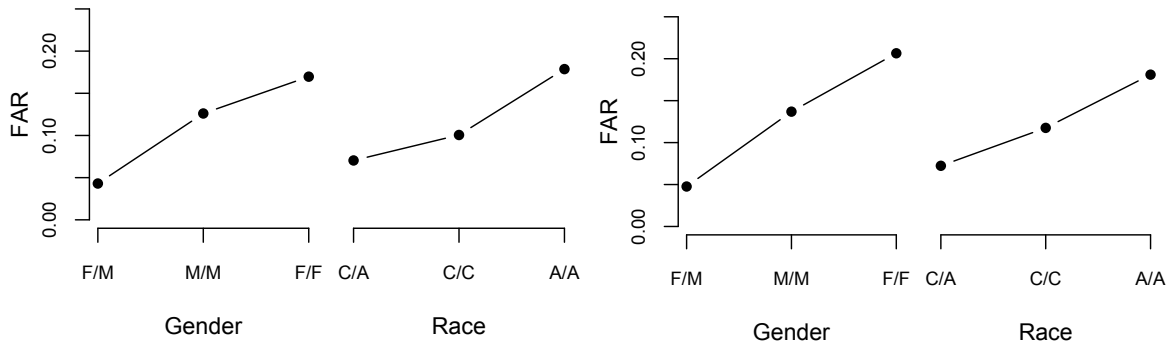

Figure (9.7) FAR for demographic factors for Ljub

The top graph is on handheld video-pairs, and the bottom graph is on tripod video-pairs. The FAR for each factor-level is reported for a global FAR = 0.10. For gender, there are three factor levels: female-male (F/M), male-male (M/M), and female-female (F/F). For race, there are three factor-levels: Caucasian-Asian (C/A), Caucasian-Caucasian (C/C), and Asian-Asian (A/A).

9.6 Impostor-Pair Analysis for Subject Identities

Subject identity as a factor has been studied fairly extensively, often under the heading "The Biometric Zoo" [73], [74]. However, defining factor levels based upon identity is problematic, so instead we move to the more interesting question of whether the marginal VR for a person correlates with the marginal FAR. In other words, do we see for people the same connection between VR and FAR as found for the other factors addressed above? To answer this question, for each algorithm, the 265 subjects are rank ordered by marginal

FAR and marginal VR. Spearman's rank correlation coefficient for these tests on handheld video-pairs are 0.14, 0.17, 0.24 and 0.35 for Algorithms CAS, UTS, Ljub, and SIT, respectively. Spearman's rank correlation coefficient for these tests on tripod video-pairs are 0.30, 0.32, 0.23 and 0.33 for Algorithms CAS, UTS, Ljub, and SIT, respectively. In short, unlike the other factors studied, VR and FAR are not strongly correlated for people. This finding is consistent with previous zoo studies on unconstrained face recognition [75].

9.7 Predicting Performance

9.7.1 Models

From all that we have learned from the marginal FAR of location pairs, a natural question to ask is if this knowledge would help in predicting performance of a new location. Given a location pair such that no knowledge of performance is known for one of the locations, how well can performance (marginal VR) be predicted? We know that there is a wide range of potential marginal VR as seen in the bottom graphs of Figures 9.3 and B.1. If we merely take the global VR as a predictor of performance, these figures demonstrate just how much error there can be.

Figure 9.4 also illustrates the range of potential marginal VR, showing scatterplots of VR vs log(FAR) of location-pairs over different sensor-pairs. Recall that additionally, a linear regressor is fit to the points for each algorithm. Observe the ranges of the marginal VR for the location-pairs of the four algorithms. For the algorithm SIT, the range is from 0.32 to 0.99 when the global FAR is set to 0.10 by eq. 2.2.

What if, instead of one new location, two locations are new and compared against each other? How well can we accurately predict performance of this entirely new pair? Is it even possible to predict the performance with the same technique used when only one location is new? Which factors should be included in a model?

We started with a very simple model. As explained below, Linear Model 1 uses only the FAR of a location-pair to predict what the observed VR will be. Simply knowing how

many false positives are in the set of video-pairs for a location-pair can indicate how well the algorithm will perform for those video-pairs. Additionally knowing some more information on the video-pairs, i.e. the video-based factors from Section 9.1.3, a better prediction can be made using Linear Model 2.

In Figure 9.4, a simple linear regressor is fit solely to the marginal verification and false accept rates of the location-pairs. The linear regressor is given by

$$\text{VR} = \alpha + \beta \log(\text{FAR}). \tag{9.1}$$

This is Linear Model 1.

Video-based factors are not incorporated into Linear Model 1. However, as we noted earlier, there is interaction between location and two video-based factors. There is interaction between location and face size, there is less interaction between location and face confidence, but there is no interaction seen between location and yaw.

To find a second model that utilizes video-based factors, we removed each location and partitioned the subjects into training and testing sets. On the remaining video-pairs that had both subjects in the training set, we fit models on the marginal VR (eq. 2.3) using marginal FAR (eq. 2.4) as well video-based factors from Section 9.1.3 and any relevant two-way interaction terms for each location-pair; we only kept terms that were significant ($p < 0.05$).

Many models resulted, and they performed robustly the same across the algorithms indicating that specifically which terms are in the model is not highly significant. With a set of second models being robustly the same in terms of prediction performance, we chose for Linear Model 2 to be given by

$$\text{VR} = \alpha + \beta_1 \log(\text{FAR}) + \beta_2 \,\text{Yaw} + \beta_3 \,\text{FC} + \beta_4 \,\text{Yaw} * \log(\text{FAR}) \tag{9.2}$$

where Yaw is the mean yaw and FC stands for the mean face confidence for the video-pairs of the location-pair. We use these models in the method described below in Section 9.7.2 for

predicting performance.

9.7.2 Prediction Procedure

In order to predict how well a set of videos of a location-pair might perform, we do the following. There are sixteen cross-week location-pairs over different sensor-pairs. For each location-pair L_i, one of the locations is randomly dropped. There will be no location-pair (no video) containing the dropped location; this location will be new. On the video-pairs of the remaining cross-week location-pairs, the subjects are partitioned into two sets: training and testing. Only video-pairs with both subjects in the training set are used.

With the video-pairs of the training set subjects, the global threshold τ_g is set so that the global FAR (eq. 2.2) is 0.10. The global VR (eq. 2.1) is calculated; this is denoted as VR_g. For the extant location-pairs, none of which use the new location, the marginal values are calculated over the different sensor-pairs, and these are used to fit the regression models from Section 9.7.1.

Using τ_g, the observed marginal VRs of the location-pair L_i are calculated over sensor-pairs, using eq. 2.3; we denote this by vr_i. Furthermore, the marginal FARs, far_i, are also calculated, using eq. 2.4. With the marginal values, a regression line can predict the observed verification rate. This predicted VR is $\widehat{\mathrm{vr}}_i = f(\mathrm{far}_i)$ where the function f is Linear Model 1 (eq. 9.1) or Linear Model 2 (eq. 9.2).

The root mean square error (RMSE) is used to determine the standard deviation between the predicted VR and the observed VR (vr_i). When using the global rate, VR_g, to predict the observed VR, the RMSE is denoted by $\mathcal{G}$. When using the VR predicted by a regression line, $\widehat{\mathrm{vr}}_i$, the RMSE is denoted by $\mathcal{E}$. Equations 9.3 and 9.4 formally express the definitions, respectively.

$$\mathcal{G} = \sqrt{\frac{\sum_{i=1}^{n}(\mathrm{VR}_g - \mathrm{vr}_i)^2}{n}} \tag{9.3}$$

$$\mathcal{E} = \sqrt{\frac{\sum_{i=1}^{n}(\widehat{\mathrm{vr}}_i - \mathrm{vr}_i)^2}{n}} \tag{9.4}$$

9.8 Results of Prediction

Are these models better than using the global VR? In order to test the models from Section 9.7.1, we implemented the procedure from Section 9.7.2 100 times with one location being new for each location-pair. Then, in order to test if the method was valid for two new locations, we ran the procedure another 100 times, but this time, both locations of a location-pair were new.

Figure 9.8 shows scatterplots of VR vs log(FAR) of location-pairs, the results of one iteration of the process of Section 9.7.2 with only one location new. To aid with legibility, only half the sensor-pairs are plotted. In both columns, the solid points represent the observed VR. The open squares, seen in the left column, represent the VR predicted from Linear Model 1 while the asterisks in the right column represent the VR predicted from Linear Model 2. In each scatterplot, vertical lines connect verification rates belonging to the same location-pair. In general, the VR predicted by Linear Model 2 is much closer to the observed VR than the VR predicted by Linear Model 1.

After 100 iterations of the Section 9.7.2 process, Figure 9.9(a) displays the mean RMSEs, equations 9.3 and 9.4, of predicting the observed VR with the previous global VR, with the VR produced from Linear Model 1, and with the VR produced from Linear Model 2 over all location-pairs and sensor-pairs. The bars extend one standard deviation. For Algorithms Ljub and SIT, the mean RMSEs from forecasting using Linear Model 1 are much lower than using the global VR, which are over 0.21. For the algorithms CAS and UTS, using Linear Model 1 is still better than using the global VR, which have mean RMSEs over 0.12, but the gap is not as large as it is for the other two algorithms.

The second linear model predicts the observed VR even better than the first linear model. The RMSEs from Linear Model 2 are much smaller than those from Linear Model 1 and definitely from those using the global VR. In fact, the means from Linear Model 2 are below 0.05 across three of the algorithms: CAS, Ljub, and SIT. The mean RMSE of Algorithm UTS is under 0.09, which is much smaller than it was from using the global VR

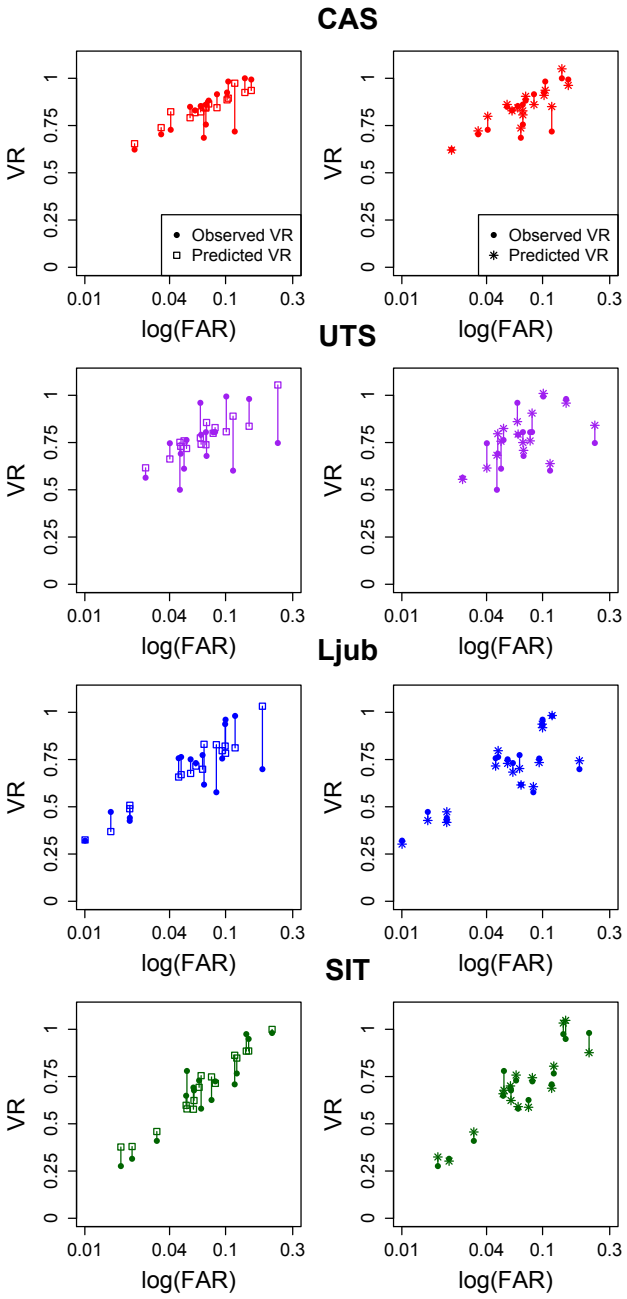

Figure (9.8) Scatterplots of VR vs. log(FAR) with the observed VR of location-pairs and the predicted VR from Linear Models 1 (left) & 2 (right), only half the sensor-pairs plotted for legibility. The legend for the top scatterplot of each column (for CAS) applies to all the scatterplots for the column.

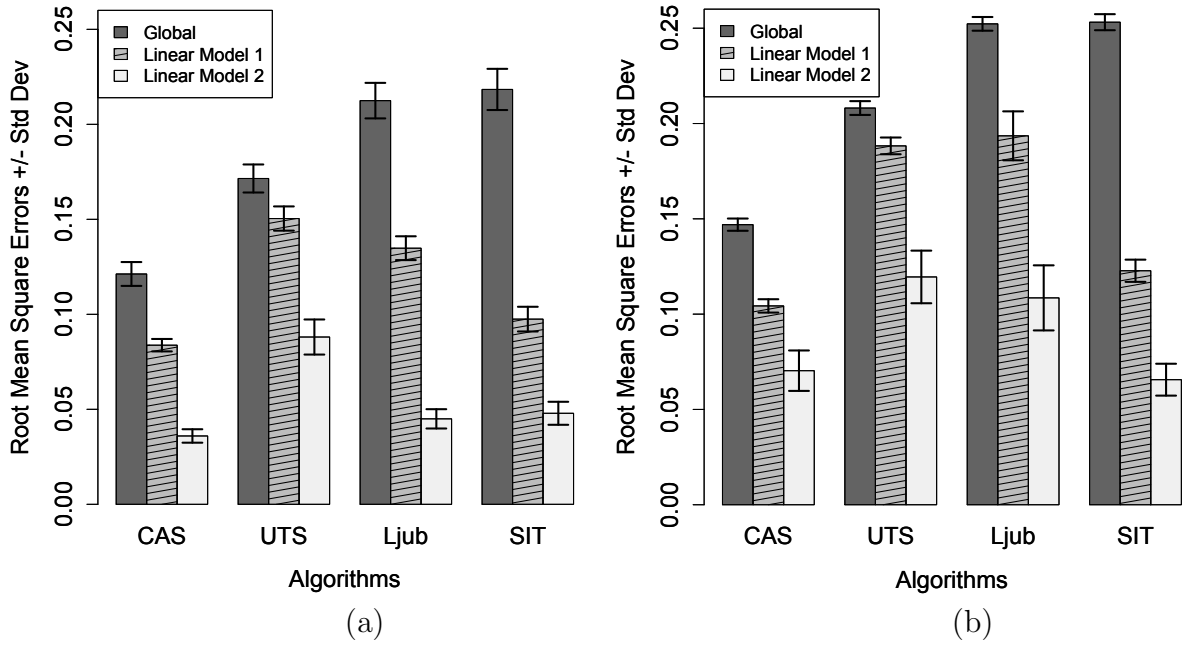

Figure (9.9) Bar plots of the mean RMSEs with standard deviation bars
In (a), one location is new. In (b), two locations are new.

or Linear Model 1 VR.

After 100 iterations, Figure 9.9(b) displays the mean RMSEs of predicting the observed VR with the global VR, with the VR produced from Linear Model 1, and with the VR produced from Linear Model 2 as in Figure 9.9(a), but in Figure 9.9(b), instead of one location being new, now both locations are new. Again, in general forecasting with Linear Model 1 is better than simply using the global VR. Using the global VR, Algorithm CAS has a mean RMSE of 0.15, and UTS has a mean RMSE of over 0.20. Algorithms Ljub and SIT have mean RMSEs over 0.25. For the algorithm SIT, the mean RMSE of Linear Model 1 less than half the mean RMSE of the global VR prediction. For Algorithms CAS, UTS, and Ljub, Linear Model 1 is still better than the previous global VR, but the differences are not as large as it is for SIT.

The second linear model still does even better than the first. There is a little more variability than before, but that is not surprising as now both locations are new. The mean

RMSEs are under 0.12 for Algorithms UTS and Ljub, and the mean RMSEs are below 0.08 for Algorithms CAS and SIT.

PART 10

CONCLUSIONS

We have proven that the set of longest paths in K_4-minor-free graphs, interval graphs, intersection graphs of spider graphs, and circular arc graphs have the Helly property. In proving that K_4-minor-free graphs have a vertex common to all longest paths, we proved the property for series-parallel graphs and outerplanar graphs, affirmatively answering a question in [45]. We used Lemma 7 for the supports of chordal graphs subclasses. In the future, we would hope that Lemma 7 for other subclasses of chordal graphs or perhaps even the entire class of chordal graphs.

We have shown that location and video factors effect the FAR for four algorithms on the video portion of the PaSC face recognition challenge. Surprisingly, for location and video-based factors there was a clear relationship between VR and FAR. For these factors, one level is not better than another; there is a trade-off between VR and FAR. An increase (resp. decrease) in the FAR results in an increase (resp. decrease) in the VR. Our results illuminate a path for better understanding the performance of face recognition algorithms in unconstrained scenarios. The results underscore a need to better control a tendency of current algorithms to increase impostor scores in favorable settings as defined by higher true-match scores. These results also establish a foundation for better modeling of distributional changes conditioned on measurable, knowable, attributes of target application locations, and consequently bring us closer to the goal of predicting performance in new settings.

Using these results, we were able to suggest two models for predicting the marginal VR when a new location is introduced. Previously, the global VR was sometimes used to predict the observed VR of a new location-pair, but as we showed, it does not predict the VR very well. There is a lot of variability in marginal VR across location-pairs, the RMSEs of prediction using global VR are quite large. When both locations are new, the RMSEs using

global VR are even larger. Better models than using the global VR are needed to capture the variability of the marginal VRs.

We have presented two models for predicting the marginal VR of a new location. The first model uses only the marginal FAR, and the second uses the marginal FAR as well as two video-based factors: yaw and face confidence. Both methods are better than simply using the previous global VR, but the second model came the closest to predicting the observed VR. Even with two new locations, the second model is much better than using the global VR. The algorithms on which we tested were from four different groups on four different continents, implying that our results will generalize well. Future work would include testing the models on new data separate from the PaSC dataset.

REFERENCES

[1] T. Gallai, "On directed paths and circuits," in *Theory of Graphs (Proc. Colloq., Tihany, 1966)*. New York: Academic Press, 1968, pp. 115–118.

[2] E. Helly, "ber mengen konvexer krper mit gemeinschaftlichen punkte." *Jahresbericht der Deutschen Mathematiker-Vereinigung*, vol. 32, pp. 175–176, 1923. [Online]. Available: http://eudml.org/doc/145659

[3] R. Wenger, "Helly-type theorems and geometric transversals," in *Handbook of Discrete and Computational Geometry, chapter 4*. CRC Press, 1997, pp. 63–82.

[4] M. C. Dourado, F. Protti, and J. L. Szwarcfiter, "Complexity aspects of the helly property: Graphs and hypergraphs."

[5] N. Amenta, "Helly theorems and generalized linear programming," in *Proceedings of the Ninth Annual Symposium on Computational Geometry*, ser. SCG '93. New York, NY, USA: ACM, 1993, pp. 63–72. [Online]. Available: http://doi.acm.org.ezproxy.gsu.edu/10.1145/160985.161000

[6] R. E. Jamison and R. Nowakowski, "A helly theorem for convexity in graphs," *Discrete Mathematics*, vol. 51, no. 1, pp. 35 – 39, 1984. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0012365X84900219

[7] A. Bretto, S. Ubda, and J. ?erovnik, "A polynomial algorithm for the strong helly property," *Information Processing Letters*, vol. 81, no. 1, pp. 55 – 57, 2002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020019001001867

[8] J. Daligault, D. Gonalves, and M. Rao, "Diamond-free circle graphs are helly circle," *Discrete Mathematics*, vol. 310, no. 4, pp. 845 – 849, 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0012365X09004580

[9] M. C. Lin, F. J. Soulignac, and J. L. Szwarcfiter, "Normal helly circular-arc graphs and its subclasses," *Discrete Applied Mathematics*, vol. 161, no. 7?8, pp. 1037 – 1059, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0166218X12004295

[10] Y. Cao, L. N. Grippo, and M. D. Safe, "Forbidden induced subgraphs of normal helly circular-arc graphs: Characterization and detection," *Discrete Applied Mathematics*, pp. –, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0166218X15004308

[11] F. Bonomo, "Self-clique helly circular-arc graphs," *Discrete Mathematics*, vol. 306, no. 6, pp. 595 – 597, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0012365X06000549

[12] F. Bonomo, M. Chudnovsky, and G. Durn, "Partial characterizations of clique-perfect graphs ii: Diamond-free and helly circular-arc graphs," *Discrete Mathematics*, vol. 309, no. 11, pp. 3485 – 3499, 2009, 7th International Colloquium on Graph TheoryICGT 05. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0012365X07010527

[13] F. Bonomo, G. Durn, L. N. Grippo, and M. D. Safe, "Partial characterizations of circle graphs," *Discrete Applied Mathematics*, vol. 159, no. 16, pp. 1699 – 1706, 2011, 8th Cologne/Twente Workshop on Graphs and Combinatorial Optimization (CTW 2009). [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0166218X10002398

[14] B. L. Joeris, M. C. Lin, R. M. McConnell, J. P. Spinrad, and J. L. Szwarcfiter, "Linear-time recognition of helly circular-arc models and graphs," *Algorithmica*, vol. 59, no. 2, pp. 215–239, 2009. [Online]. Available: http://dx.doi.org/10.1007/s00453-009-9304-5

[15] R. C. Hamelink, "A partial characterization of clique graphs," *Journal of*

Combinatorial Theory, vol. 5, no. 2, pp. 192 – 197, 1968. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0021980068800559

[16] F. S. Roberts and J. H. Spencer, "A characterization of clique graphs," *Journal of Combinatorial Theory, Series B*, vol. 10, no. 2, pp. 102 – 108, 1971. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0095895671900700

[17] M. C. Lin and J. L. Szwarcfiter, "Faster recognition of clique-helly and hereditary clique-helly graphs," *Information Processing Letters*, vol. 103, no. 1, pp. 40 – 43, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020019007000415

[18] M. Groshaus and J. L. Szwarcfiter, "Biclique-helly graphs," *Graphs and Combinatorics*, vol. 23, no. 6, pp. 633–645. [Online]. Available: http://dx.doi.org/10.1007/s00373-007-0756-6

[19] ——, "On hereditary helly classes of graphs." *Discrete Mathematics & Theoretical Computer Science (DMTCS)*, vol. 10, no. 1, pp. 71 – 78, 2008. [Online]. Available: http://ezproxy.gsu.edu/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=a9h&AN=41020742&site=ehost-live&scope=site

[20] H.-J. Bandelt and E. Prisner, "Clique graphs and helly graphs," *Journal of Combinatorial Theory, Series B*, vol. 51, no. 1, pp. 34 – 45, 1991. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0095895691900044

[21] A. Bondy, G. Durn, M. C. Lin, and J. L. Szwarcfiter, "Self-clique graphs and matrix permutations," *Journal of Graph Theory*, vol. 44, no. 3, pp. 178–192, 2003. [Online]. Available: http://dx.doi.org/10.1002/jgt.10496

[22] F. Larrin and M. Pizaa, "On hereditary clique-helly self-clique graphs," *Discrete Applied Mathematics*, vol. 156, no. 7, pp. 1157 – 1167, 2008, {GRACO} 20052nd Brazilian Symposium on Graphs, Algorithms and Combinatorics. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0166218X07002983

[23] M. C. Dourado, P. Petito, and R. B. Teixeira, "Helly property and sandwich graphs," *Electronic Notes in Discrete Mathematics*, vol. 22, pp. 497 – 500, 2005, 7th International Colloquium on Graph Theory. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1571065305052583

[24] H. Walther, "Über die Nichtexistenz eines Knotenpunktes, durch den alle längsten Wege eines Graphen gehen," *J. Combinatorial Theory*, vol. 6, pp. 1–6, 1969.

[25] H. Walter and H. J. Voss, *Über Kreise in Graphen*. VEB Dt. Verlag der Wissenschaften, 1974.

[26] T. Zamfirescu, "On longest paths and circuits in graphs," *Math. Scand.*, vol. 38, no. 2, pp. 211–239, 1976.

[27] W. Schmitz, "Über längste Wege und Kreise in Graphen," *Rend. Sem. Mat. Univ. Padova*, vol. 53, pp. 97–103, 1975.

[28] T. Zamfirescu, "A two-connected planar graph without concurrent longest paths," *J. Combinatorial Theory Ser. B*, vol. 13, pp. 116–121, 1972.

[29] B. Grünbaum, "Vertices missed by longest paths or circuits," *J. Combinatorial Theory Ser. A*, vol. 17, pp. 31–38, 1974.

[30] F. Nadeem, A. Shabbir, and T. Zamfirescu, "Planar lattice graphs with gallai's property," *Graphs and Combinatorics*, vol. 29, no. 5, pp. 1523–1529, 2012. [Online]. Available: http://dx.doi.org/10.1007/s00373-012-1177-8

[31] Y. Bashir and T. Zamfirescu, "Lattice graphs with gallai's property," vol. 56, pp. 65–71, 2013. [Online]. Available: http://ssmr.ro/bulletin/volumes/56-1/node8.html

[32] A. D. Jumani and T. Zamfirescu, "On longest paths in triangular lattice graphs," vol. 89, pp. 269–273, 2012.

[33] F. Nadeem, A. Shabbir, and T. Zamfirescu, "Planar lattice graphs with gallais property," *Graphs and Combinatorics*, vol. 29, no. 5, pp. 1523–1529, 2013. [Online]. Available: http://dx.doi.org/10.1007/s00373-012-1177-8

[34] A. Shabbir, "Fault-tolerant designs in lattice networks on the klein bottle," vol. 2, no. 2, pp. 99–109, 2014. [Online]. Available: http://www.ejgta.org/index.php/ejgta/article/view/52

[35] Y. Bashir, F. Nadeem, and A. Shabbir, "Highly non-concurrent longest paths in lattices," vol. 40, pp. 21–31, 2016. [Online]. Available: http://journals.tubitak.gov.tr/math/abstract.htm?id=17802

[36] C. Thomassen, "Hypohamiltonian and hypotraceable graphs," *Discrete Mathematics*, vol. 9, no. 1, pp. 91 – 96, 1974. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0012365X74900740

[37] M. Araya and G. Wiener, "On cubic planar hypohamiltonian and hypotraceable graphs," vol. 18, 2011. [Online]. Available: http://www.combinatorics.org/ojs/index.php/eljc/article/view/v18i1p85/0

[38] S. Klavžar and M. Petkovšek, "Graphs with nonempty intersection of longest paths," *Ars Combin.*, vol. 29, pp. 43–52, 1990.

[39] P. Balister, E. Győri, J. Lehel, and R. Schelp, "Longest paths in circular arc graphs," *Combin. Probab. Comput.*, vol. 13, no. 3, pp. 311–317, 2004. [Online]. Available: http://dx.doi.org/10.1017/S0963548304006145

[40] F. Joos, "A note on longest paths in circular arc graphs," *Discussiones Mathematicae Graph Theory*, vol. 35, pp. 419–426.

[41] F. Chen, "Nonempty intersection of longest paths in a graph with a small matching number," *Czechoslovak Mathematical Journal*, vol. 65, no. 2, pp. 545–553, 2015. [Online]. Available: http://dx.doi.org/10.1007/s10587-015-0193-2

[42] M. AXENOVICH, "When do three longest paths have a common vertex?" *Discrete Mathematics, Algorithms and Applications*, vol. 01, no. 01, pp. 115–120, 2009. [Online]. Available: http://www.worldscientific.com/doi/abs/10.1142/S1793830909000038

[43] W. T. Tutte, "A theorem on planar graphs," *Transactions of the American Mathematical Society*, vol. 82, no. 1, pp. 99–116, 1956.

[44] S. de Rezende, C. Fernandes, D. Martin, and Y. Wakabayashi, "Intersection of longest paths in a graph," *Electronic Notes in Discrete Mathematics*, vol. 38, no. 0, pp. 743 – 748, 2011, the Sixth European Conference on Combinatorics, Graph Theory and Applications, EuroComb 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1571065311001909

[45] S. F. de Rezende, C. G. Fernandes, D. M. Martin, and Y. Wakabayashi, "Intersecting longest paths," *Discrete Mathematics*, vol. 313, no. 12, pp. 1401 – 1408, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0012365X13001015

[46] P. J. Phillips, J. R. Beveridge, B. A. Draper, G. Givens, A. J. O'Toole, D. Bolme, J. Dunlop, Y. M. Lui, H. Sahibzada, and S. Weimer, "The good, the bad, and the ugly face challenge problem," *Image and Vision Computing*, vol. 30, no. 3, pp. 177 – 185, 2012, best of Automatic Face and Gesture Recognition 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0262885612000091

[47] J. Beveridge, P. Phillips, D. Bolme, B. Draper, G. Givens, Y. M. Lui, M. Teli, H. Zhang, W. Scruggs, K. Bowyer, P. Flynn, and S. Cheng, "The challenge of face recognition from digital point-and-shoot cameras," in *Biometrics: Theory, Applications and Systems (BTAS), 2013 IEEE Sixth International Conference on*, Sept 2013, pp. 1–8.

[48] P. J. Phillips, H. Wechsler, J. Huang, and P. Rauss, "The FERET database and evaluation procedure for face-recognition algorithms," *Image and Vision Computing Journal*, vol. 16, no. 5, pp. 295–306, 1998.

[49] O. Jesorsky, K. J. Kirchberg, and R. Frischholz, "Robust face detection using the hausdorff distance," in *Proceedings of the Third International Conference on Audio- and Video-Based Biometric Person Authentication*, ser. AVBPA '01. London, UK, UK: Springer-Verlag, 2001, pp. 90–95. [Online]. Available: http://dl.acm.org/citation.cfm?id=646073.677460

[50] T. Sim, S. Baker, and M. Bsat, "The CMU Pose, Illumination, and Expression database," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 1615–1618, 2003.

[51] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-pie," *Image and Vision Computing*, vol. 28, no. 5, pp. 807 – 813, 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0262885609001711

[52] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, "Overview of the Face Recognition Grand Challenge," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2005, pp. 947–954.

[53] P. J. Phillips, W. T. Scruggs, A. J. O'Toole, P. J. Flynn, K. W. Bowyer, C. L. Schott, and M. Sharpe, "FRVT 2006 and ICE 2006 large-scale results," National Institute of Standards and Technology, Tech. Rep. NISTIR 7408, 2007. [Online]. Available: http://iris.nist.gov/

[54] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled Faces in the Wild: a database for studying face recognition in unconstrained environments," University of Massachusetts, Amherst, Tech. Rep. 07-49, 2007.

[55] P. J. Grother, G. W. Quinn, and P. J. Phillips, "MBE 2010: Report on the evaluation of 2D still-image face recognition algorithms," National Institute of Standards and Technology, NISTIR 7709, 2010.

[56] P. J. Phillips, J. R. Beveridge, B. A. Draper, G. Givens, A. J. O'Toole, D. S. Bolme, J. Dunlop, Y. M. Lui, H. Sahibzada, and S. Weimer, "An introduction to the Good, the Bad, and the Ugly Face Recognition Challenge problem," in *Proceedings Ninth IEEE International Conference on Automatic Face and Gesture Recognition*, 2011.

[57] A. Sgroi, K. W. Bowyer, P. Flynn, and P. J. Phillips, "SNoW: understanding the causes of strong, neutral, and weak face impostor pairs," in *IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, 2013.

[58] J. R. Beveridge, G. H. Givens, P. J. Phillips, B. A. Draper, and Y. M. Lui, "Focus on quality, predicting FRVT 2006 performance," in *Proceeding of the Eighth International Conference on Automatic Face and Gesture Recognition*, 2008.

[59] J. R. Beveridge, G. H. Givens, P. J. Phillips, and B. A. Draper, "Factors that influence algorithm performance in the Face Recognition Grand Challenge," *Computer Vision and Image Understanding*, vol. 113, pp. 750–762, 2009.

[60] J. R. Beveridge, G. H. Givens, P. J. Phillips, B. A. Draper, D. S. Bolme, and Y. M. Lui, "FRVT 2006: Quo vadis face quality," *Image and Vision Computing Journal*, vol. 28, no. 5, pp. 732–743, 2010.

[61] A. J. O'Toole, P. J. Phillips, X. An, and J. Dunlop, "Demographic effects on estimates of automatic face recognition performance," in *Proceedings, Ninth International Conference on Automatic Face and Gesture Recognition*, 2011.

[62] R. Duffin, "Topology of series-parallel networks," *Journal of Mathematical Analysis and Applications*, vol. 10, no. 2, pp. 303 – 318, 1965. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0022247X65901253

[63] F. Gavril, "The intersection graphs of subtrees in trees are exactly the chordal graphs," *Journal of Combinatorial Theory, Series B*, vol. 16, no. 1, pp. 47 – 56, 1974. [Online]. Available: http://www.sciencedirect.com/science/article/pii/009589567490094X

[64] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*. Pearson, 2006.

[65] J. R. Beveridge, H. Zhang, B. A. Draper, P. J. Flynn, Z. Feng, P. Huber, J. Kittler, Z. Huang, S. Li, Y. Li, M. Kan, R. Wang, S. Shan, X. Chen, H. Li, G. Hua, V. Štruc, J. Križaj, C. Ding, D. Tao, and P. J. Phillips, "Report on the FG 2015 video person recognition evaluation," in *Proceedings Eleventh IEEE International Conference on Automatic Face and Gesture Recognition*, 2015.

[66] W. T. Tutte, *Graph theory*, ser. Encyclopedia of Mathematics and its Applications. Cambridge: Cambridge University Press, 2001, vol. 21, with a foreword by Crispin St. J. A. Nash-Williams, Reprint of the 1984 original.

[67] L. Lovász, *Combinatorial problems and exercises*, 2nd ed. Amsterdam: North-Holland Publishing Co., 1993.

[68] J. R. Beveridge, P. J. Phillips, D. S. Bolme, B. A. Draper, G. H. Givens, Y. M. Lui, M. N. Teli, H. Zhang, W. T. Scruggs, K. W. Bowyer, P. J. Flynn, and S. Cheng, "The challenge of face recognition from digital point-and-shoot cameras," in *IEEE Conference on Biometrics: Theory, Applications and Systems*, 2013.

[69] J. R. Beveridge, H. Zhang, P. Flynn, Y. Lee, V. E. Liong, J. Lu, M. Angeloni, T. Pereira, H. Li, G. Hua, V. Struc, J. Krizaj, and P. J. Phillips, "The IJCB 2014 PaSC video face and person recognition competition," in *Proceedings of the International Joint Conference on BIometrics*, 2014.

[70] Y. Lee, P. J. Phillips, J. J. Filliben, J. R. Beveridge, and H. Zhang, "Generalizing face quality and factor measures to video," in *International Joint Conference on Biometrics (IJCB)*, 2014.

[71] A. J. O'Toole, P. J. Phillips, X. An, and J. Dunlop, "Demographic effects on estimates of automatic face recognition performance," *Image and Vision Computing*, vol. 30, pp. 169–176, 2012.

[72] G. H. Givens, J. R. Beveridge, P. J. Phillips, B. A. Draper, Y. M. Lui, and D. S. Bolme, "Introduction to face recognition and evaluation of algorithm performance," *Computational Statistics and Data Analysis*, vol. 67, pp. 236–247, 2013.

[73] G. Doddington, W. Ligget, A. Martin, M. Przybocki, and D. Reynolds, "Sheeps, goats, lambs, and wolves: A statistical analysis of speaker performance in the NIST 1998 recognition evaluation," in *Proceedings ICSLP '98*, 1998.

[74] N. Yager and T. Dunstone, "The biometric menagerie," *IEEE Trans. Pattern Analysis Machine Intelligence*, vol. 32, no. 2, pp. 220–230, 2010.

[75] M. N. Teli, J. R. Beveridge, P. J. Phillips, G. H. Givens, D. S. Bolme, and B. A. Draper, "Biometric zoos: Theory and experimental evidence," in *International Joint Conference on Biometrics*, 2011.

Appendix A

COMPUTING VIDEO-BASED FACTORS

The image-based factors face size, face confidence, and yaw are well established, but defining these factors for a video is not obvious. To extend these factors for an entire video, we follow the method of Lee et al. [70], which has been shown to be effective.

To explain, let $g_{\mathrm{FS}}(f)$ be the face size for image f; the following method applies to face confidence and yaw. Suppose that video x has the frames $\{f_1, f_2, \ldots, f_m\}$, having face size values $\{g_{\mathrm{FS}}(f_1), g_{\mathrm{FS}}(f_2), \ldots, g_{\mathrm{FS}}(f_m)\}$. To find the face size as a video factor, we take the mean value of the frames. Therefore, the face size for video x is $h_{\mathrm{FS}}(x) = \mathrm{mean}\{g_{\mathrm{FS}}(f_1), g_{\mathrm{FS}}(f_2), \ldots, g_{\mathrm{FS}}(f_m)\}$.

For the video-based factors for video-pairs, we have the following. Let (x, y) be the video-pair. For both face size and face confidence, the minimum value of the two videos is taken under the assumption that the smaller value is more indicative of recognition impediments. Hence, we have the video-based factors are $k_{\mathrm{FS}}(x, y) = \min\{h_{\mathrm{FS}}(x), h_{\mathrm{FS}}(y)\}$ for face size and $k_{\mathrm{FC}}(x, y) = \min\{h_{\mathrm{FC}}(x), h_{\mathrm{FC}}(y)\}$ for face confidence. For yaw, the absolute difference is taken so that larger factor values result for video-pairs in which the faces generally have less similar viewpoints. Therefore, we have $k_{\mathrm{Yaw}}(x, y) = |h_{\mathrm{Yaw}}(x) - h_{\mathrm{Yaw}}(y)|$ as the video-based factor for yaw.

Appendix B

LARGE GRAPHS

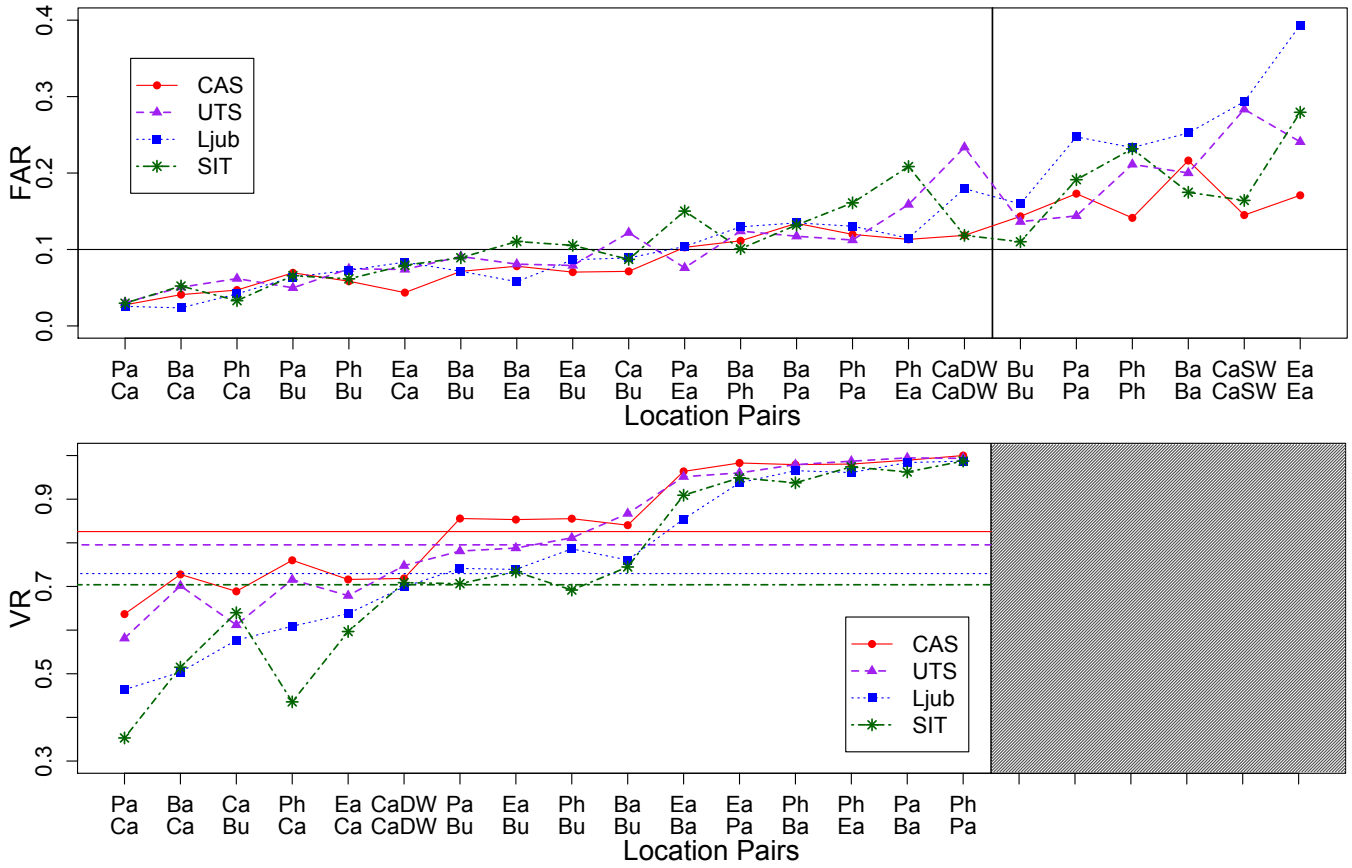

Figure (B.1) FAR and VR of each location-pair on tripod video-pairs for each algorithm

The FAR and VR are ordered by the mean rate over all the algorithms. The top graph is on FAR. The horizontal line corresponds to the global FAR = 0.10, and the vertical line between pairs CaDW-CaDW and Bu-Bu separates the pairs into cross-week (left) and same week. The bottom graph is on VR. The horizontal lines correspond to the global VR for each algorithm when the global FAR = 0.10. There are no same-week pairs for matches.

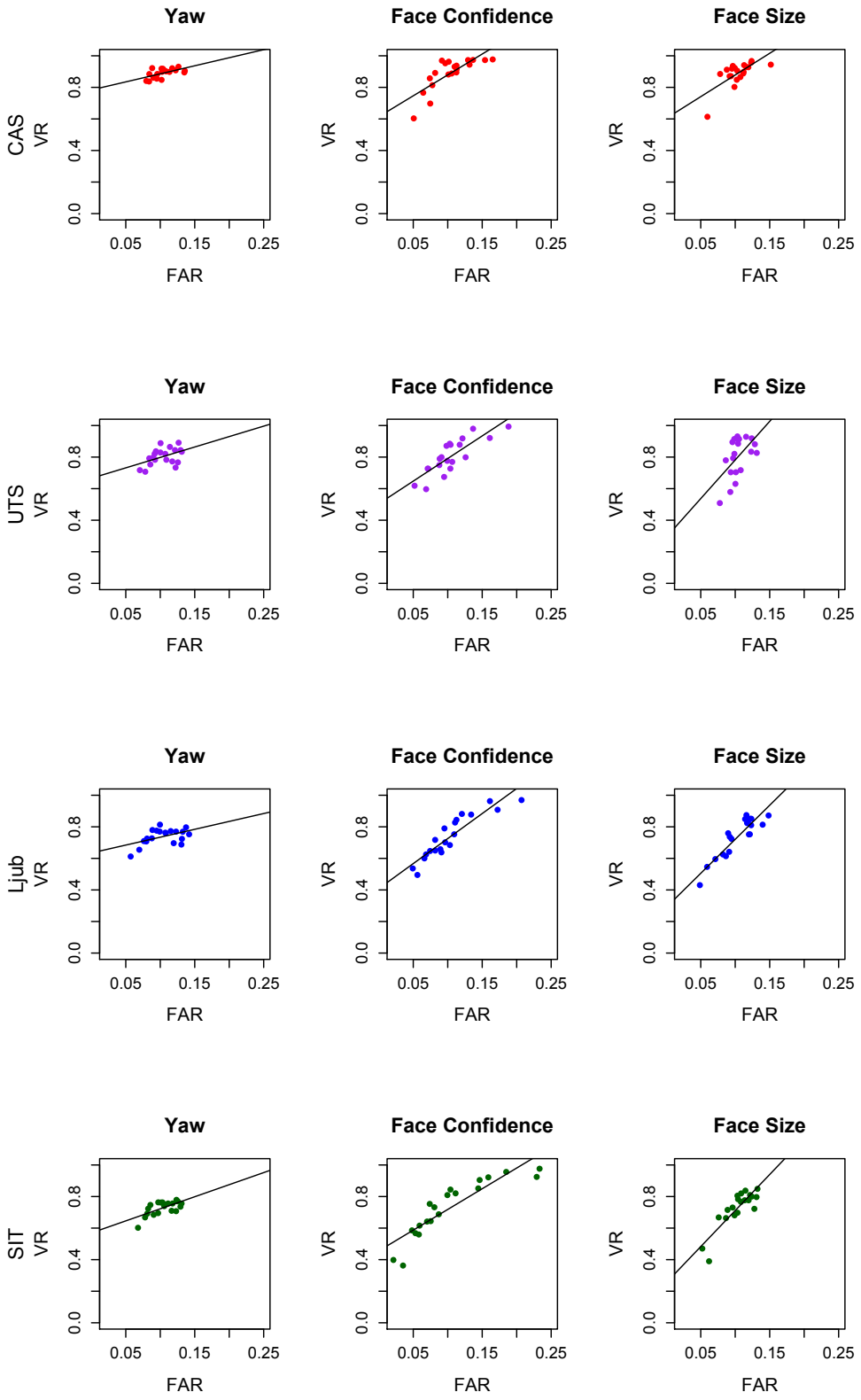

Figure (B.2) Scatterplots of VR vs FAR for video-based factors over different sensor-pairs, fitted with a linear regressor for each algorithm

There are twelve scatterplots, one for each algorithm and video-based factor. One column for each video factor and one row for each algorithm. All video-based factors are divided into 10 bins. Thresholds set to global FAR = 0.10.

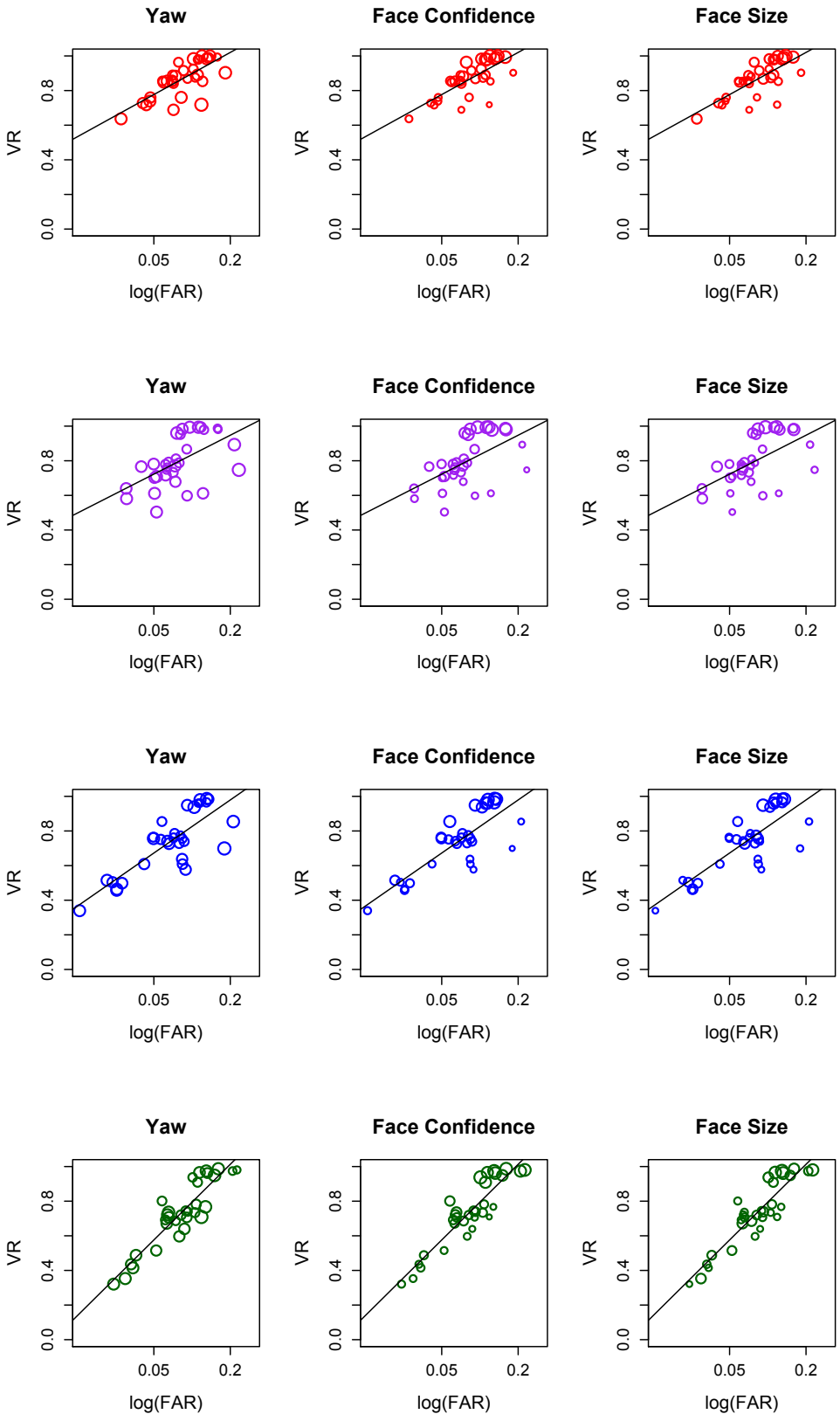

Figure (B.3) Interactions between location-pairs and video-based factors

There are twelve scatterplots, one for each algorithm and video-based factor. One column for each video factor and one row for each algorithm. Each panel looks at the interaction for an algorithm between location-pairs and the factor in its title. The size of each circle is proportional to the mean of the video factor for each location-pair.

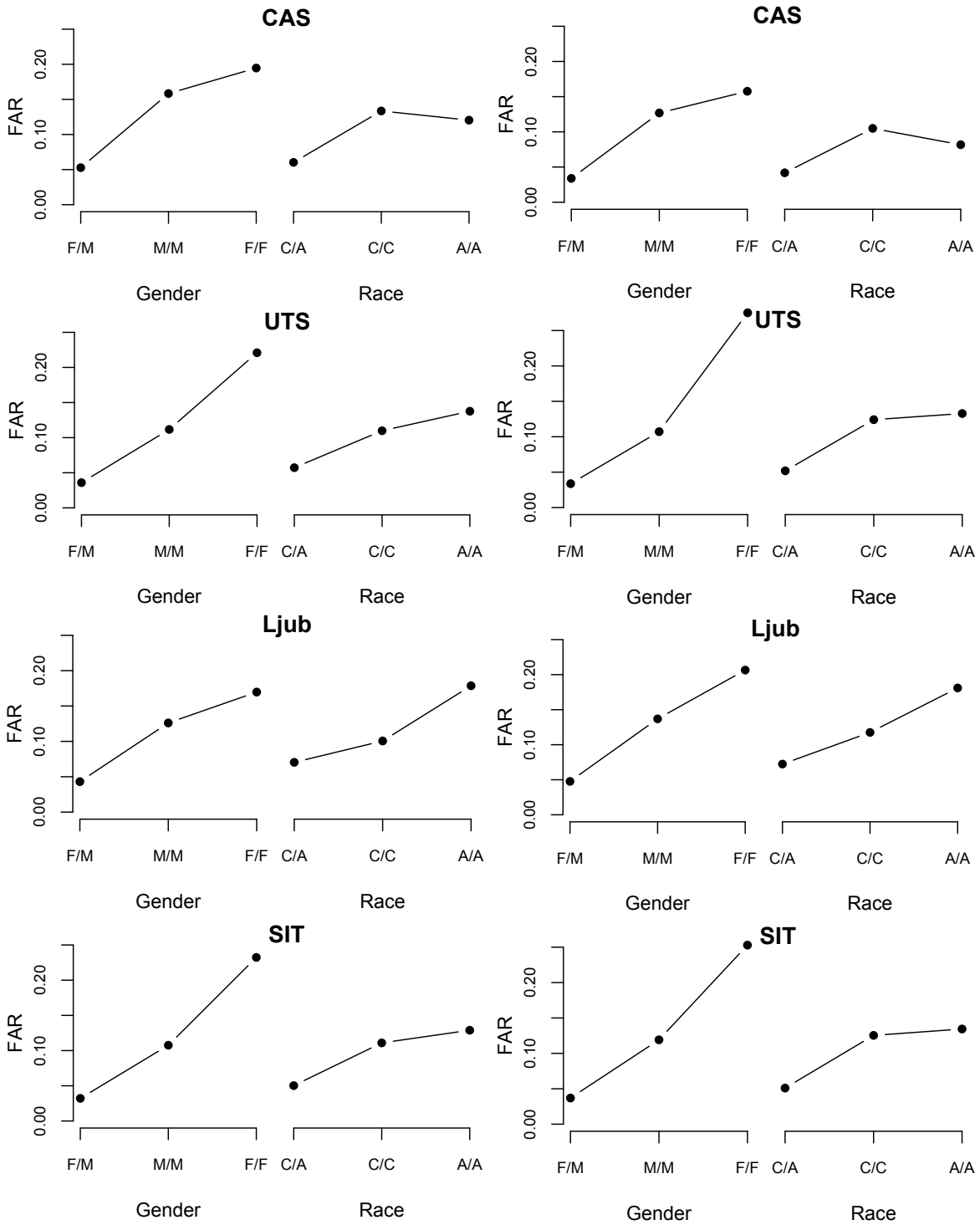

Figure (B.4) FAR for demographic factors for Algorithms CAS, UTS, Ljub, and SIT

The left column is on handheld video-pairs, and the right column is on tripod video-pairs. The FAR for each factor-level is reported for a global FAR = 0.10. For gender, there are three factor levels: female-male (F/M), male-male (M/M), and female-female (F/F). For race, there are three factor-levels: Caucasian-Asian (C/A), Caucasian-Caucasian (C/C), and Asian-Asian (A/A).