BOSTON UNIVERSITY

GRADUATE SCHOOL OF ARTS AND SCIENCES

Dissertation

PERSONALIZED FACE AND GESTURE ANALYSIS USING HIERARCHICAL NEURAL NETWORKS

by

AJJEN DAS JOSHI

B.A., Connecticut College, 2012
M.S., Boston University, 2014

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy
2018

© Copyright by
AJJEN DAS JOSHI
2018

Approved by

First Reader
Margrit Betke, PhD
Professor of Computer Science

Second Reader
Stan Sclaroff, PhD
Professor of Computer Science

Third Reader
Kate Saenko, PhD
Associate Professor of Computer Science

Acknowledgments

First and foremost, I would like to thank my dearest parents. I will forever be grateful to them for their infinite love, support and encouragement throughout my life. They have always prioritized my education and development above everything. Whatever success I achieve in life, I will always owe it to their sacrifices and blessings.

I would also like to express my sincerest gratitude to my advisors, Prof. Margrit Betke and Prof. Stan Sclaroff, for their unending patience and careful guidance. I feet doubly fortunate to be advised by two people, who I will always look up to. I have learnt so much from both of them: in research, in teaching and in mentoring others.

I would like to thank my thesis committee members: Prof. Abraham Matta, Prof. Kate Saenko and Prof. Jacob Whitehill. I am grateful to them for reading my thesis and providing insightful feedback, including during the defense.

During the course of my PhD, I was fortunate to have two fruitful internship experiences, which were a vital part of my growth. I would like to thank my research mentors during my internships: Dr. Hanspeter Pfister and Dr. Soumya Ghosh at Disney Research and Masha Shugrina at Adobe Research.

Much of the work presented here are results of collaborations: with Camille Monnier on the work pertaining to gesture recognition, with Dr. Linda Tickle-Degnen, Dr. Sarah Gunnery and Dr. Terry Ellis on the work regarding facial expressivity prediction, and with Dr. Beverly Woolf, Danielle Allessio, Dr. John Magee and Dr. Jacob Whitehill on the work concerning student learning outcome prediction. I wish to thank all of them for their valuable feedback and expert advice.

I would like to thank my fellow labmates at Boston University's Image and Vision Computing research group: Dr. Wenxin Feng, Dr. Andrew Kurauchi and Elham Saraee with whom I had the opportunity to collaborate with on numerous projects not included in

this thesis. I am also grateful to Dr. Ashwin Thangali, Dr. Zheng Wu, Dr. Danna Gurari, Dr. Diane Theriault, Dr. Shugao Ma, Dr. Jianming Zhang, Dr. Qinxun Bai, Dr. Fatih Cakir, Dr. Mehrnoosh Sameki, Dr. Mikhail Breslav, Dr. Kun He and Sarah Bargal for many interesting conversations about research and other topics over the last six years.

Finally, I would like to say thank you to Kripa, for your unconditional love and support, and your undying belief in me. With you, no mountain is too high to conquer.

The research reported in this thesis was supported in part by National Science Foundation grants 0910908, 1551572, 0855065 and 1337866 and US Navy contract N00014-13-P-1152. The content of this thesis is solely the responsibility of the author and does not necessarily reflect the views of the National Science Foundation.

PERSONALIZED FACE AND GESTURE ANALYSIS USING HIERARCHICAL NEURAL NETWORKS

(Order No.)

AJJEN DAS JOSHI

Boston University, Graduate School of Arts and Sciences, 2018

Major Professor: Margrit Betke, Professor of Computer Science

ABSTRACT

The video-based computational analyses of human face and gesture signals encompass a myriad of challenging research problems involving computer vision, machine learning and human computer interaction. In this thesis, we focus on the following challenges: a) the classification of hand and body gestures along with the temporal localization of their occurrence in a continuous stream, b) the recognition of facial expressivity levels in people with Parkinson's Disease using multimodal feature representations, c) the prediction of student learning outcomes in intelligent tutoring systems using affect signals, and d) the personalization of machine learning models, which can adapt to subject and group-specific nuances in facial and gestural behavior. Specifically, we first conduct a quantitative comparison of two approaches to the problem of segmenting and classifying gestures on two benchmark gesture datasets: a method that simultaneously segments and classifies gestures versus a cascaded method that performs the tasks sequentially. Second, we introduce a framework that computationally predicts an accurate score for facial expressivity and validate it on a dataset of interview videos of people with Parkinson's disease. Third, based on a unique dataset of videos of students interacting with MathSpring, an intelligent tutoring system, collected by our collaborative research team, we build models to predict learning outcomes from their facial affect signals. Finally, we propose a novel solution to a relatively unexplored area in automatic face and gesture analysis research: personalization of models to individuals and groups. We develop hierarchical Bayesian neural networks to overcome the challenges posed by group or subject-specific variations in face and gesture signals. We successfully validate our formulation on the problems of personalized subject-specific gesture classification, context-specific facial expressivity recognition and student-specific learning outcome prediction. We demonstrate the flexibility of our hierarchical framework by validating the utility of both fully connected and recurrent neural architectures.

Contents

1 Introduction **1**

1.1 Gesture Spotting and Recognition 3

1.2 Facial Expressivity Prediction 5

1.3 Learning Outcomes Prediction using Affect 7

1.4 Personalization 9

1.5 Contributions 11

1.6 Roadmap of Thesis 12

1.7 List of Related Papers 15

2 Related Work **16**

2.1 Gesture Spotting and Recognition 16

2.2 Facial Expressivity Prediction 19

2.3 Modeling Affect in Intelligent Tutoring Systems 21

2.4 Personalization 23

3 Comparing Random Forest Approaches to Segmenting and Classifying Gestures **25**

3.1 System Overview 26

3.1.1 Training 26

3.1.2 Testing 31

3.1.3 Evaluation 34

3.2 Datasets 35
3.2.1 NATOPS 35
3.2.2 ChaLearn 36
3.3 Experiments 36
3.4 Summary 44

4 Predicting Active Facial Expressivity in People with Parkinson's Disease 46
4.1 Dataset 47
4.1.1 Exploratory Data Analysis 48
4.2 System Overview 50
4.2.1 Input 50
4.2.2 Feature Extraction 51
4.2.3 Feature Importance 53
4.3 Experiments 53
4.3.1 Geometric Features 54
4.3.2 Action Unit and Audio Features 57
4.4 Summary 59

5 Affect-driven Learning Outcomes Prediction in Intelligent Tutoring Systems 61
5.1 Dataset Collection and Annotation 62
5.2 Exploratory Data Analysis 64
5.2.1 Label Distribution 64
5.2.2 Problem Times Analysis 64
5.2.3 Action Unit Analysis 66
5.3 Baseline Models 69
5.4 Feature Representation 69

5.5 Experiments 70

5.5.1 Effort Prediction 73

5.5.2 How long to make Accurate Predictions? 73

5.5.3 Interventions 75

5.6 Summary 76

6 Hierarchical Bayesian Neural Networks 77

6.1 Model 77

6.2 Scalable Learning and Inference 82

6.3 Local Reparameterization 84

6.4 Predictions 85

6.5 Personalization 86

6.6 Active Learning 87

6.7 Hierarchical Bayesian Recurrent Neural Networks 88

6.8 Summary 88

7 Applications of Hierarchical Bayesian Neural Networks to Problems in Face and Gesture Analysis 90

7.1 Subject-specific Gesture Recognition 91

7.1.1 Datasets 91

7.1.2 Experiments 91

7.2 Context-sensitive Facial Expressivity Prediction 104

7.2.1 Context 104

7.2.2 Dataset 106

7.2.3 Experiments 106

7.3 Student-specific learning outcome prediction 108

7.3.1 Dataset 109
7.3.2 Experiments 109
7.3.3 Summary 110

8 Conclusions and Future Work **112**
8.1 Contributions 112
8.2 Strengths, Limitations and Future Research Directions 115
8.2.1 Gesture Spotting and Recognition 115
8.2.2 Predicting Active Facial Expressivity in People with Parkinson's Disease 117
8.2.3 Affect-driven Learning Outcomes Prediction in Intelligent Tutoring Systems 118
8.2.4 Hierarchical Bayesian Neural Networks and Applications 120

Bibliography **122**

Curriculum Vitae **135**

List of Tables

3.1 Average Classification accuracy on all 24 gestures of the NATOPS dataset 39

3.2 Performance comparison on pairs of similar gestures in the NATOPS dataset with other approaches 42

3.3 Average classification accuracy on all 20 gestures of the ChaLearn dataset 43

3.4 Jaccard Index scores on the 2014 ChaLearn Gesture Recognition Challenge 44

5.1 Multi-class effort prediction results 71

List of Figures

1.1 A sequence of frames representing an instance of a gesture performed in the Italian language (An example taken from the ChaLearn gesture recognition dataset). 3

1.2 A sequence of frames representing a variety of expressions displayed by a student while working on math problems during a session with an intelligent tutoring system. 8

3.1 Pipeline view of training our gesture recognition framework that performs simultaneous spotting and classification 26

3.2 Pipeline view of training our cascaded gesture recognition framework that first spots a gesture before classifying it 27

3.3 Pipeline view of testing our gesture recognition framework that performs simultaneous spotting and classification 32

3.4 Pipeline view of testing our cascaded gesture recognition framework that first spots a gesture before classifying it 33

3.5 An example illustration of the Jaccard Score 34

3.6 RGB, Depth, and User-Mask Segmentation of a subject performing gesture ‘I Have’ in the NATOPS dataset 37

3.7 RGB, Depth, and User-Mask Segmentation of a subject performing gesture ‘sonostufo’ in the ChaLearn dataset 37

3.8 Some pairs of similar gestures in the NATOPS dataset 40

3.9 Confusion Matrix for pairs of similar gestures in the NATOPS dataset . . 41
3.10 Plot comparing the Jaccard index scores of training the simultaneous and sequential classifiers with number of training iterations 43

4.1 For each class in the dataset, the average Action Unit Occurence (AUO) is plotted for 17 different AUs. For each subplot, the x-axis represents the 4 facial expressivity classes whereas the y-axis represents the mean AUO score. 49
4.2 Geometric features, which capture facial dynamics, computed by our system 51
4.3 (Left) The mean MAE-scores and their standard deviations for models trained using different geometric feature sets (MS, ES, MS+ES). (Right) The mean R^2 scores and their standard deviations for models trained using different geometric feature sets (MS, ES, MS+ES). 56
4.4 Bar graph displaying features with 10 highest feature importance scores. (std: standard deviation, pf: peak frequency, Q: quartile) 56
4.5 (Left) The mean MAE-scores and their standard deviations for models trained using different audio-video feature sets (AU, MFCC, AU+MFCC). (Right) The mean R^2 scores and their standard deviations for models trained using different audio-video feature sets (AU, MFCC, AU+MFCC). 58
4.6 Bar graph displaying aggregated feature importance scores for the different action units. 58

5.1 Example images of the Learning Outcomes Prediction dataset collected from different modalities 62
5.2 Distribution of dataset according to effort labels 65
5.3 Average time for problem completion according to effort labels 65

5.4 Average Action Unit Occurrence distributed according to effort labels . . 67
5.5 Average Action Unit Occurrence distributed according to effort labels for the first 10 seconds of the input 68
5.6 Confusion matrix for 7-class effort prediction 71
5.7 Mean F-scores for multiclass classifiers trained with no oversampling (NOS), and various resampling methods. 72
5.8 Mean F-scores for one-vs-all binary classifiers trained for different problem outcome labels. 72
5.9 Multi-class model performance for effort prediction when data is observed for different time scales. 74
5.10 Performance of binary classification models for effort prediction when data is observed for different time scales. 74
5.11 Performance of multiclass classifier (Mean accuracy 0.35) for effort prediction of all classes except SOF when the first 30 seconds of data are observed. 75

6.1 Given input examples produced by g groups, we train a classifier using a hierarchical framework, where $\mathcal{W}_g$ is the set of group-specific weights parameterizing a Bayesian neural network. 78
6.2 Given few instances of training data from a new group, we personalize our model to learn weights specific to the new group. 78
6.3 Graphical model representation of our hierarchical Bayesian model. Shaded nodes indicate observed random variables. 81

7.1 Examples of gestures from MSRC-12 dataset (top left), ChaLearn 2013 dataset (top right) and the NATOPS dataset (bottom) 92

7.2 The mean logarithm of the expected lower bound (ELBO) versus the number of training epochs, for 15 random 75-25 splits of the data, when the model uses local reparameterization (lprm) and when it doesn't (no lrpm). 94

7.3 The mean logarithm of the expected lower bound (ELBO) versus the number of training epochs, for 15 random 75-25 splits of the ChaLearn dataset, when the model uses local reparameterization (lprm) and when it doesn't (no lrpm) for different HBNN architectures. 95

7.4 The mean F1-scores for different versions of our Hierarchical Bayesian gesture classifier. 97

7.5 The mean F1-scores for different personalization schemes plotted against the number of personalization instances per gesture. 99

7.6 The mean F1-scores for different personalization schemes plotted against number of personalization instances per gesture for different HBNN architectures for the ChaLearn dataset. 102

7.7 The mean F1-scores comparing feed-forward Hierarchical Bayesian Neural Networks (HBNN) with Hierarchical Bayesian Recurrent Neural Networks (HBRNN), for all three datasets. 103

7.8 Overview of our multimodal context-sensitive expressivity prediction model. 105

7.9 (Left) The mean F1-scores and their standard deviations for HBNN-C models trained using context (gender, sentiment) or no context (pooled). (Right) The mean MAE scores and their standard deviations for HBNN-R models trained using context (gender, sentiment) or no context (pooled). . 107

7.10 The mean F1-scores for a model trained with feed-forward neural architecture (HB-NN) and a recurrent neural architecture (HB-RNN) for multi-class expressivity classification. 107

7.11 The mean F1-scores and their standard deviations for an HBNN model trained with all data pooled together and a student-specific HBNN model. 109

List of Abbreviations

3D	3 Dimensional
ADASYN	Adaptive Synthetic Sampling
AR	Augmented Reality
ATS	Affective Tutoring Systems
ATT	Attempted
AU	Action Unit
AUC	Area Under the Curve
AUO	Action Unit Occurence
BALD	Bayesian Active Learning by Disagreement
Cat	Categorical
CNN	Convolutional Neural Networks
CRF	Conditional Random Field
DP	Dirichlet Process
DTW	Dynamic Time Warping
ELBO	Evidence Lower Bound
ES	Eye Shape
FACS	Facial Action Coding System
GRU	Gated Recurrent Unit
HBMR	Hierarchical Bayesian Multinomial Regression
HBNN	Hierarchical Bayesian Neural Network

HBNN-C	Hierarchical Bayesian Neural Network - Classification
HBNN-R	Hierarchical Bayesian Neural Network - Regression
HBRNN .	Hierarchical Bayesian Recurrent Neural Network
HCRF . . .	Hidden Conditional Random Field
HMM . . .	Hidden Markov Model
HOG	Histogram of Oriented Gradients
HS	Handshape
ICC	Intra-class Correlation Coefficient
ICRP	Interpersonal Communication Rating Protocol
ITS	Intelligent tutoring System
KL	Kullback-Leibler
k-NN	k Nearest Neighbors
lrpm	local reparameterization
LSTM . . .	Long Short Term Memory
MAE	Mean Absolute Error
MCMC . .	Markov Chain Monte Carlo
MFCC . . .	Mel Frequency Cepstral Coefficients
MS	Mouth Shape
MSE	Mean Squared Error
MSRC-12	Microsoft-12
NATOPS	Naval Air Training and Operating Procedures Standardization
NBMA . .	Naive Bayesian Model Averaging
NOS	No Oversampling
NOTR . . .	Not Read
OOB	Out Of Bag

PCA	Principal Component Analysis
PD	Parkinson’s Disease
pf	peak frequency
Q	Quartile
R^2	R Square
RAND ..	Random
RGB	Red Green Blue
RNN	Recurrent Neural Network
SHINT ..	Solved with Hint
SK	Skeletal
SMOTE .	Synthetic Minority Oversampling Technique
SOF	Solved on First Attempt
std	standard deviation
UAV	Unmanned Air Vehicles
VR	Virtual Reality
WBMA ..	Weighted Bayesian Model Averaging

Chapter 1

Introduction

The computational analysis of images and videos to extract useful information about humans is an integral research domain within computer vision and machine learning. This broad area of research has a wide array of applications, involving problems such as detecting and recognizing people [29] and faces in images [116], tracking them in video [61], detecting body parts and pose [133], determining what gesture [69], action [122] or activity [85] is being performed, inferring information about emotions [103] and facial expressions [23], among many others. In this thesis, we focus on applications related to the analysis of signals generated specifically by human gestures and faces.

In conjunction with speech, humans use gestures to communicate ideas, feelings and intentions. Kendon defines a gesture as ‘a label for actions that have the features of manifest deliberate expressiveness’ [60]. According to this definition, gestures are a voluntary and intentional movement of part of the body performed primarily for the purpose of expression.

Scholarly interest in gestures has been longstanding. For example, gestures were studied and conventionalized in Classical Antiquity, especially with regards to the important role they played in oration. In more recent research pursuits, gestures have been studied through the lens of anthropology, psychology, linguistics as well as computer science. Scholars have devised multiple methods of categorizing gestures. One such system of categorization, devised by McNeill [77], places gesture categories in a continuum based on its relationship to speech: on one end are gesticulations that are obligatorily accompanied by speech, whereas on the other end are pantomimes and sign language where speech is absent.

Computational advancements since the mid-20th century have enabled the development of intuitive and natural paradigms of instructing, communicating and interacting with computers. Because gestures are considered a universal and natural form of expression, they are often considered appropriate when designing new models for human computer interaction. In order for the distinction between human-human communication and human-machine communication to diminish, it is vital that algorithms learn to robustly spot and recognize human gestures.

In addition to speech and gesture, humans use another important channel to communicate and express themselves: their face. Faces often portray the hidden, internal state of the human mind. Humans not only use the face as a vehicle of expression and intent but also possess the ability to read and decipher the facial expressions of others. Because the face is such a rich source of information, equipping a computer with the ability to accurately read emotions and affect can have numerous benefits and applications. The field of 'Affective computing' delves on equipping a computer to "recognize and express emotions, develop its ability to respond intelligently to human emotions and enable it to regulate and utilize its emotions" [91].

In this thesis, we focus on the following challenges within face and gesture analysis: a) the classification of hand and body gestures along with the temporal localization of their occurrence in a continuous stream, b) the recognition of facial expressivity levels in people with Parkinson's Disease, c) the prediction of student learning outcomes in intelligent tutoring systems using affect signals, and d) the personalization of models that can adapt

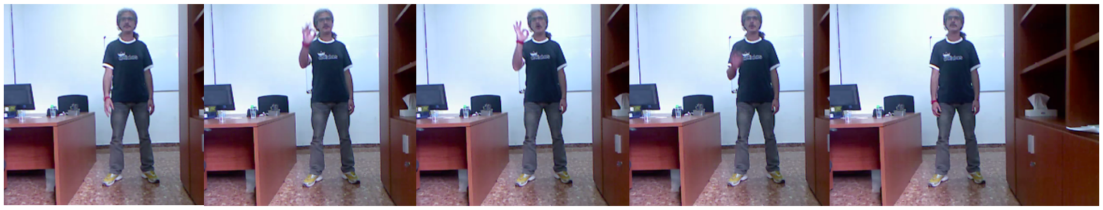


Figure 1.1: A sequence of frames representing an instance of a gesture performed in the Italian language (An example taken from the ChaLearn gesture recognition dataset).

to subject and group-specific nuances in facial and gestural behavior. We now introduce each of these problems in more detail.

1.1 Gesture Spotting and Recognition

The problem of spotting and recognizing meaningful gestures is a challenging and important research endeavor with a broad scope of applications, such as recognizing sign-language symbols, enabling video surveillance, establishing new interaction idioms in gaming and entertainment, and developing new modes of human-computer interaction, among others. In the first part of the thesis, we present a quantitative comparison of two methods to solve the problem of spotting and recognizing gestures from a continuous input stream.

A specific example of a gesture recognition application can be explored in the setting of a flight deck of an aircraft carrier. Deck officers use a vocabulary of gestures to communicate commands such as “All clear”, “Move ahead”, “Turn left/right”, “Slow down” etc. to aircraft pilots. However, the advent of unmanned air vehicles (UAVs) has engendered the need to create a system capable of communicating the same set of commands to these unmanned aircrafts. Equipping a UAV with a computer vision system capable of accurately and automatically recognizing the existing set of gestures while they are being performed by deck officers would provide the most efficient solution to this problem, as it would permit the continued operation of the current method of communication.

Another example of an application in gesture recognition lies in the domain of understanding the context provided by communication gestures. Human beings communicate with words as well as gestures. A computer vision system capable of deciphering the gestures used in specific languages, such as Italian, can provide contextual information that aids the task of translating a foreign language (Figure 1.1).

Gestures can be recognized from a wide array of sensors, ranging from RGB cameras that capture the motion of the human body to wearable devices with inertial motion units. With the popularity and availability of cameras capable of capturing depth information of a scene, gestures recognition datasets often contain 3D skeletal information of the user as well as intensity information from image frames. Designers of gesture recognition systems can therefore extract features from both skeletal as well as image data. We propose a random forest-based gesture classification model, where gestures are represented by a combination of both skeletal and image-based features.

It is important that the start and end points of a gesture be accurately identified in a continuous temporal stream, in order to maximize the probability of correctly estimating the gesture label. One approach in solving the segmentation and classification problem involves separating them into two sub-problems where the task of segmentation precedes the task of recognition. In this method, the focus is on first finding the gesture boundaries in time. The candidate gestures produced by the segmentation algorithm is then classified.

Another approach simultaneously performs the tasks of segmentation and classification. In methods such as this, gesture intervals for which above-threshold scores are given by the classifier are deemed to be the labeled and segmented gesture. Given a training set of multi-modal videos with multiple examples of all gestures in a gesture vocabulary, we provide a comparison of the two approaches based on random forest models, highlighting the strengths and weaknesses of each.

1.2 Facial Expressivity Prediction

The dynamics of the human face, like gestures, also plays an important role in enabling expressive communication and social interactions. The human face is "one of the most powerful channels of nonverbal communication" [31].

The computational analysis of facial expressivity can have a wide range of applications [88]. For example, automatic and accurate affect sensing can play a major role in diagnostic as well as treatment procedures in medical conditions where emotive, expressive and cognitive abilities are impaired. The development of such technologies can aid therapists and practitioners by helping them save valuable time otherwise devoted to laborious manual coding of patient observations. The impressive progress made in the field of automatic facial expression analysis [24, 41, 119] has spurred computational research in applications related to healthcare and behavioral psychology. In the second part of this thesis, our focus is on developing a machine learning model capable of predicting facial expressivity ratings of patients with Parkinson's disease (PD) from short interview videos.

PD affects over 10 million people worldwide and about 1% of people over 60 years old [1]. Patients with Parkinson's disease often have a reduced ability to exhibit spontaneous facial expression due to an increased rigidity of facial musculature, also known as facial masking [113] or facial bradykinesia [13]. The reduced ability in patients to express emotions can hinder aspects of their social life because they are often misperceived by others [113]. It is therefore important for clinicians and researchers to be able to objectively assess and quantify the level of active expressivity in the face, so they can measure facial masking as a symptom of PD and test whether interventions to improve facial masking are effective.

Facial expressivity is inherently more difficult to measure in people with PD because facial masking dims the clarity of muscle action shown in the face. Despite this difficulty, there are existing manuals for objectively measuring active expressivity in the Parkinsonian face, one of which is the Interpersonal Communication Rating Protocol (ICRP) [111], where active facial expressivity is among 20 indicators rated by trained experts along a 5-point Likert scale. Raters are trained to provide a "Gestalt" rating based on the intensity (strength of emotion or movement), duration (how long a behavior or movement lasts) and frequency (how often a behavior or movement lasts) of the expressive behavior observed. An active facial expressivity rating of 1 represents a person with "primarily one emotional expression plastered on the face, with low to no movement" whereas a rating of 5 is given to people with "highly active, animated, mobile and moving face with changing emotional expressions" [111].

As with other systems of manual coding, rating facial expressivity according to the ICRP brings forth challenges associated with scale and feasibility. Human coders have successfully coded facial expression in people with PD [54], but the costs associated with the manual assessment of all patients with PD can be prohibitively high. Comprehensive manual coding of 20 seconds of video can take upwards of an hour, and often two coders are needed to establish that the human coder is reliable.

Existing works involving computational analyses of facial emotions and expressivity of PD patients are mostly limited to pilot studies comparing facial characteristics and dynamics between a small group of PD patients and a separate control group [1, 9, 126]. An accurate machine learning model trained on an expertly annotated dataset and capable of generalizing to new data is, therefore, an attractive proposition. Here, we utilize a dataset of 772 short interview audio-video clips of PD patients and their corresponding facial expressivity labels to train a model that can accurately predict facial expressivity levels of new PD patients. For each video, we extract interpretable visual features from the face of the patients detected in the input frames as well as audio features from the raw audio to produce a multimodal feature descriptor, with which we train both classification and regression models and cross-validate them on held-out test sets.

1.3 Learning Outcomes Prediction using Affect

Automated affect analysis also has applications in the education domain. A interesting research direction is to inquire about how modeling student affect during digital learning experiences can be utilized to positively impact the student's overall learning experience. Intelligent tutoring systems (ITSs) have been developed with the aim of providing an individualized learning experience to users. One of the goals of an ITS is to build models of the student engaged in learning, so that the ITS can adapt its support mechanisms in order for the process of learning to be personalized [55]. It has been shown that students experience a variety of emotions, such as interest, flow, surprise, anger, boredom, frustration, confusion and anxiety, during learning [34]. Emotions felt and displayed by students have been shown to correlate well with their achievement in the learning task [90]. Equipping an ITS with the ability to interpret such affective signals could potentially enable it to monitor the students' progress, provide timely interventions as well as present appropriate affective reactions via a virtual tutor.

Affective Tutoring Systems (ATSs) use one or more sensors to observe the student in order to infer his or her emotional state while using the ATS. Ideally, ATSs can imitate a human teacher and adapt not only to the student's level of knowledge but also to his or her emotional state. Therefore, one of the primary capabilities of an ATS is to automatically recognize basic emotions, such as happiness, anger and disgust. Examples of ATSs with

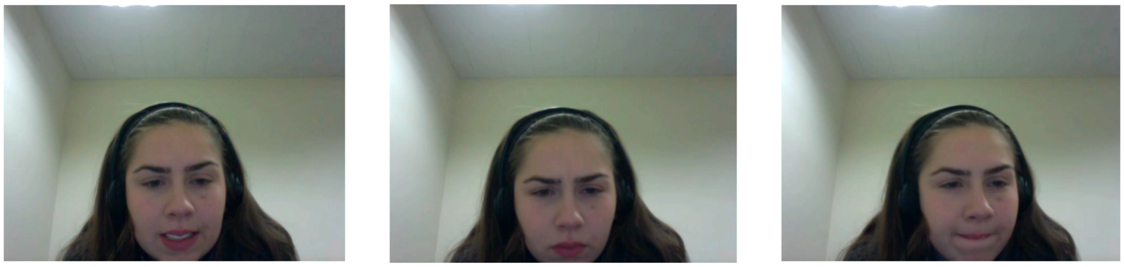


Figure 1.2: A sequence of frames representing a variety of expressions displayed by a student while working on math problems during a session with an intelligent tutoring system.

this feature include EER-Tutor [138] and FERMAT [139]. In addition to the basic emotions, some ITSs (e.g. AutoTutor [34], Guru Tutor [87]) possess models, which are trained to recognize learning-specific emotional states, such as engagement, concentration, confusion, boredom and frustration. Vision-based sensors such as webcams are suitable for capturing the facial dynamics of students as they are readily available in the most common platforms used for interacting with ITSs (e.g. phones, tablets, laptops) and are less invasive than other sensors, such as wearable devices that measure physiological signals like skin conductivity, heart rate, muscle activity or pressure-sensitive chairs that measure posture.

Most ATSs equipped with affect modeling capabilities attempt to predict the emotional state of users. However, in the third part of the thesis, our focus is instead on trying to directly predict the learning outcomes of students. That is, using facial features extracted from a video stream, we train classifiers that can directly predict the success or failure of a student's attempt to answer a question. In order to do so, our research collaborators collected a novel dataset of students interacting with MathSpring [4], a web-based mathematics ITS (Figure 1.2). An ATS's ability to directly predict learning outcomes can help improve the student's learning experience by enabling the ATS to provide interventions such as hints or messages of encouragement.

1.4 Personalization

In the final part of this thesis, we provide a treatment of the aforementioned problems through the lens of personalization. Humans, by nature, are unique and individualistic. Therefore, it is not surprising that human signals (such as the dynamics of skeletal joints when performing gestures, or movement of facial musculature when expressing emotions) exhibit a lot of variance. Consider, for example, a vocabulary of gestures used by members of a household to control a smart-home device. Although each individual may perform the gestures consistently, it is likely that the gestures are performed with user-specific idiosyncrasies which may lead to large inter-subject variations in gesture performance. Designing systems robust to such variations is a challenging problem. A generic gesture classifier, trained on examples of gestures pooled together from all subjects in the training set, is expected to be robust to variations with which gestures are performed by end-users. However, when the signal obtained from gestures performed by different users exhibit high variance, such systems have difficulty generalizing.

Personalizing gesture recognition systems using subject-specific training data provides a promising approach to alleviating such difficulties. We build hierarchical Bayesian classifiers that adapt to new subjects using subject-specific conditional distributions. Different from existing hierarchical Bayesian models, we parameterize the conditional distributions via multi-layered Bayesian neural networks. They allow us to learn potentially complex functional relationships between a subject's gestures and class labels from a modest number of training examples. Furthermore, by explicitly modeling uncertainty in weights, Bayesian neural networks are able to provide well calibrated estimates of posterior uncertainty along with predicted class labels. Leveraging recent progress on scalable stochastic variational inference, we develop algorithms for learning the posterior distribution over all network weights in the hierarchy. We further use the inferred posterior to drive active learning algorithms that guide interactive labeling of personalization gestures given a small pool of unlabeled subject-specific gestures.

First, we systematically test various aspects of the proposed models and algorithms on three challenging gesture recognition datasets — the MSRC-12 Kinect Gesture Dataset [43], the 2013 ChaLearn Gesture Challenge Dataset [40] and the NATOPS gesture dataset [105]. We find that even with relatively shallow two hidden layer networks, our approach is competitive with the *state-of-the-art* gesture personalization systems. We also empirically demonstrate that even with naive fully factorized variational inference, Bayesian neural networks provide uncertainty estimates that are useful for guiding active learning procedures. We then extend the functionality of our hierarchical framework by adding support to recurrent architectures and demonstrate their suitability in modeling signals of a sequential nature, such as gestures.

Second, we adapt the individual-specific hierarchical Bayesian framework to the problem of group-specific facial expressivity prediction. Most existing works on automated facial analysis train generic classifiers for the task-at-hand, ignoring additional context that can accompany the input. Contextual information can be derived, for example, from the identity of the patient, the gender of the patient, the mood of the patient during the time of the interview, etc. Here, we investigate whether contextual information can be leveraged to further improve the performance of the model. We experiment with two clearly defined notions of context: (1) *gender*: a variable indicating the gender (male or female) of the patient and, (2) *sentiment*: a variable indicating the sentiment (positive or negative) expressed during the interview. These variables are provided with the dataset and are utilized to divide the dataset into context-sensitive groups.

Instead of modeling individual subject-specific variances in gesture performance, we utilize the hierarchical Bayesian framework to capture the subtle context-sensitive group-specific variances in the input-expressivity mapping. We separate the training data into context-sensitive groups and train our hierarchical model using multimodal feature descriptors of each training video. In order to predict the facial expressivity score from a test video, we use the parameters of the trained model associated with the context-sensitive group to which the test video belongs.

Finally, we also evaluate our hierarchical model on the problem of personalized predictions of student outcomes. Because students vary significantly in how they display their emotional states during learning, we explore whether the learning outcome prediction performance can benefit from using personalized models.

1.5 Contributions

Here, we summarize the major contributions of this thesis:

- we present an analysis of methods for gesture spotting and classification by comparing a framework that employs a single multi-class random forest classification model to distinguish gestures from a given vocabulary in a continuous video stream with a framework that uses a cascaded approach,
- we present an interpretable system that computes facial expressivity scores in people with Parkinson's disease using multimodal audio-visual feature descriptors extracted from a video sequence,
- we develop models to predict learning outcomes based on affect signals extracted from the videos of a novel dataset of students interacting with MathSpring, an intelligent tutoring system, and

- we develop hierarchical Bayesian neural networks for personalized modeling of face and gesture signals in the presence of inter-group and inter-subject variations. We propose to utilize the inferred posterior to drive an active learning procedure for personalizing the model to new users. We evaluate the personalization framework to three tasks: subject-specific gesture recognition, context-specific facial expressivity prediction and student-specific learning outcome prediction.
- We also develop recurrent variants of our hierarchical Bayesian model and demonstrate its suitability in building personalized models involving sequential signals such as gestures.

1.6 Roadmap of Thesis

The rest of the thesis is organized as follows:

Chapter 2: Related Work

This chapter reviews the literature for relevant work on the problems of gesture segmentation and classification, facial emotion and expression prediction, video-based affect recognition in intelligent tutoring systems, as well as building personalized classifiers.

Chapter 3: Comparing Random Forest Approaches to Segmenting and Classifying Gestures

In this chapter, we compare two approaches to the problem of gesture localization and recognition: a method that performs the tasks of temporal segmentation and classification simultaneously with another that performs the tasks sequentially. We first test our proposed gesture recognition method on the NATOPS dataset of 9,600 gesture instances from a vocabulary of 24 aircraft handling signals, and present evaluations of our formulation in segmenting and recognizing gestures from a continuous stream on the ChaLearn dataset of 7754 gesture instances from a vocabulary of 20 Italian communication gestures.

Chapter 4: Predicting Facial Expressivity in People with Parkinson's Disease

In this chapter, we evaluate a framework that predicts a score for facial expressivity using a variety of feature descriptors. We experiment with a descriptor of geometric shape features of the face as well as multimodal feature representations consisting of Facial Action Units features combined with Mel Frequency Cepstral Coefficient features extracted from the audio stream. We train random forest classifiers and regressors based on the various features descriptors and present results based on evaluations of our formulation on a dataset of 772 20-second audio-video clips of interviews of people suffering from Parkinson's disease using 9-fold cross validation.

Chapter 5: Affect-driven Learning Outcomes Prediction in Intelligent Tutoring Systems

In this chapter, we investigate the problem of trying to predict the learning outcome of students attempting to solve individual problems based on signals extracted from their faces. Based on a novel dataset consisting of video-streams of students using MathSpring, we develop baseline models to predict problem solving outcomes, e.g. whether students are able to solve the problem at their first attempt or whether they require hints. We propose mechanisms by which we can use this model to improve the students' learning experience.

Chapter 6: Hierarchical Bayesian Neural Networks

In this chapter, we introduce hierarchical Bayesian neural networks to capture group-specific variations in human signals. We present algorithms for inferring the posterior distribution over all network weights in the hierarchy. We also develop methods for adapting our model to new groups when a small number of group-specific personalization data is available. We investigate active learning algorithms for interactively labeling personalization data in resource-constrained scenarios. Finally we extend the framework to support recurrent architectures, which are appropriate in modeling temporal tasks.

Chapter 7: Applications of Hierarchical Bayesian Neural Networks to Problems in Face and Gesture Analysis

In this chapter, we apply the hierarchical Bayesian model to explore whether personalization can be beneficial to any of the problems introduced in Chapters 3-5. Focusing first on the problem of gesture recognition where inter-subject variations are commonplace, we demonstrate the effectiveness of our proposed techniques by testing our framework on three widely used gesture recognition datasets.

We then adapt the hierarchical Bayesian neural network framework to enable the learning of facial expressivity model parameters that subtly adapt to pre-defined notions of context, such as the gender of the patient or the valence of the expressed sentiment. We present results based on evaluations of our formulation on a dataset of 772 20-second video clips of Parkinson's disease patients and demonstrate that training a context-specific hierarchical Bayesian framework yields an improvement in model performance in both multi-class classification and regression settings compared to the same model trained on all data pooled together.

Finally, we report results of experiments evaluating our hierarchical model on the problem of personalized predictions of student outcomes. We compare the performance of a generic model trained on data from all students pooled together versus that of a student-specific hierarchical Bayesian model.

Chapter 8: Conclusions and Future Work

In this chapter, we summarize the contributions we have made in the thesis and discuss the strengths and limitations of the proposed methods. We end this thesis with a discussion on future research directions and open problems.

1.7 List of Related Papers

Much of the work presented in this thesis has been published in the following journal, conference and workshop proceedings:

1. Joshi, A., Monnier, C., Betke, M., & Sclaroff, S. (2017). Comparing random forest approaches to segmenting and classifying gestures. Image and Vision Computing, 58, 86-95.

2. Joshi, A., Tickle-Degnen, L., Gunnery, S., Ellis, T., & Betke, M. (2016, June). Predicting Active Facial Expressivity in People with Parkinson's Disease. ACM International Conference on PErvasive Technologies Related to Assistive Environments (PETRA).

3. Joshi, A., Ghosh, S., Betke, M., & Pfister, H. (2016, December). Hierarchical Bayesian Neural Networks for Personalized Classification. Conference on Neural Information Processing Systems Workshop on Bayesian Deep Learning (NIPSW).

4. Joshi, A., Ghosh, S., Betke, M., Sclaroff, S., & Pfister, H. (2017, June). Personalizing Gesture Recognition Using Hierarchical Bayesian Neural Networks. IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

5. Joshi, A., Ghosh, S., Gunnery, S., Tickle-Degnen, L., Sclaroff, S., & Betke, M. (2018, May). Context-sensitive Facial Expressivity Prediction using Multimodal Hierarchical Bayesian Neural Networks. IEEE Conference on Automatic Face and Gesture Recognition (FG).

Chapter 2

Related Work

Here, we review the literature of work related to the problems of gesture spotting and recognition, facial emotion and expression analysis, and building personalized classifiers.

2.1 Gesture Spotting and Recognition

We begin by providing an overview of some of the important methods that have been used in gesture recognition and are relevant to our work. A more comprehensive survey of gesture recognition techniques can be found elsewhere [79], [96].

Nearest neighbor models have been used in gesture classification problems. Malassiotis et al.[74] used a k-NN classifier to classify static sign language hand gestures. A normalized cross-correlation measure was used to compare the feature vector of an input image with those in the k-NN model. Dynamic Time Warping (DTW) can be used to compute a matching score between two temporal sequences, a variant of which was used by Alon et al.[2]. A drawback of k-NN models is the difficulty in defining distance measures that clearly demarcate different classes of time series observations.

A Hidden Markov Model (HMM) is another widely used tool in temporal pattern recognition, having been implemented in applications of speech recognition, handwriting recognition, as well as gesture recognition. Starner et al.[108] employed an HMM-based system to recognize American Sign Language symbols. One difficulty while implementing HMMs is to determine an appropriate number of hidden states, which can be domain-dependent.

The Conditional Random Field (CRF), introduced by Lafferty et al.[68] is a discriminative graphical model with an advantage over generative models, such as HMMs: the CRF does not assume that observations are independent given the values of the hidden variables. Hidden Conditional Random Fields (HCRF) use hidden variables to model the latent structure of the input signals by defining a joint distribution over the class label and hidden state labels conditioned on the observations [93]. HCRFs can model the dependence between each state and the entire observation sequence, unlike HMMs, which only capture the dependencies between each state and its corresponding observation. Song et al. used a Gaussian temporal-smoothing HCRF [104] to classify gestures that combine both body and hand signals. They also presented continuous Latent Dynamic CRFs [106] to classify unsegmented gestures from a continuous input stream of gestures.

Random forest models perform well in many classification tasks, work efficiently on large datasets, and are very fast. Random forests have been applied to good effect in real-time human pose recognition [102], object segmentation [100], image classification [14], and sign language recognition [67] among others. Decision forests models have also been used variedly in gesture and action recognition tasks [45], [136], [137], [129], [18]. Miranda et al.[78] used a gesture recognition scheme based on decision forests, where each node in a tree in the forest represented a keypose, and the leaves of the trees represented gestures corresponding to the sequence of keyposes that constitute the gesture as one traverses down a tree from root to leaf. Demirdjian et al.[32] proposed the use of temporal random forests in order to recognize temporal events. Camgoz et al. [18] used random forests to perform gesture spotting and classification. In contrast to our work, they perform frame-level gesture classification by training a model where every individual frame is considered a separate training sample. Randomized decision forests have been shown to be robust to the effects of noise and outliers. Moreover, they generalize well to variations in data [16]. Thus, random forests are suitable for classification tasks involving data such as gestures because data collected by image and depth sensors can be sensitive to noise and their execution can exhibit a high level of variance.

More recently, deep learning approaches have gained popularity in gesture spotting and recognition tasks. Neverova et al. [83] presented a gesture localization and recognition scheme based on multi-modal deep learning operating at various spatial as well as temporal scales. Pigou et al. [92] presented an end-to-end neural network architecture incorporating temporal convolutions and bidirectional recurrence to perform gesture spotting and recognition. Molchanov et al. [80] utilized 3D convolutional neural networks to map depth and intensity input into accurate gesture labels. Du et al. [37] proposed a hierarchical recurrent neural network framework that learns intermediate representations from different parts of the input skeleton before hierarchically combining them to produce the final label prediction. Liu et al. [71] presented a tree-structure based traversal of the input skeleton to preserve the graphical structure of the human skeleton while training a Long Short Term Memory (LSTM) based recurrent neural network.

Cameras equipped with depth sensors combined with skeleton detection algorithms enable researchers to use features extracted from 3D joint positions in gesture and action recognition problems. Yao et al. [135] used concatenated raw coordinates of body joints for gesture classification whereas Xia et al. [127] employed histograms of 3D joint locations for the task of human action recognition. Raptis et al. [95] formulated an angular representation of user skeletons as features for the problem of dance gesture recognition. In some problems, it is advantageous to include in the feature representation, information that 3D body joint locations are unable to capture, e.g. hand shape. The salient properties of hand shape can be captured using image-based features such as Histograms of Oriented Gradients (HOG) [29]. Song et al. [104] combined features extracted from images of the user hands with joint features to classify gestures.

In this thesis, we focus on building concise feature representations using 3D skeletal joint-based features that capture global motion of the input gesture as well as appearance-based features in order to obtain a more granular representation of the hand shape. Although a random forest classifier does not explicitly model the inherent temporal nature of gestural data as done by graphical models, we aim to show that our proposed robust feature representations combined with the random forest model's generalization capacity can yield performance on par with those achieved by graphical models such as HMMs and HCRFs.

2.2 Facial Expressivity Prediction

Automatic analysis of facial expressions and affect has been an actively researched topic in the fields of computer vision and machine learning [31]. Many early works focused on the recognition of prototypic emotions from static images [89] or video [26]. A more detailed descriptor of the physical changes in the shape and texture of the face, named the Facial Action Coding System (FACS) was developed by Ekman and Friesen [38] to describe facial expressions in terms of anatomically defined Action Units (AUs). The problem of automatically identifying the presence [7] and intensity [86] of AUs from images [110] and video [21] has received a lot of attention in recent years. Progress in this field has led to development of several off-the-shelf applications [30, 8] capable of detecting AU presence and intensity values for several Action Units.

The dynamics of a person's face can provide information regarding the person's emotional state, intention and personality, as well as cognitive and biomedical status. The development of computational analyses techniques of facial expressions has opened avenues for researchers to view investigations of emotional and cognitive impairments using a computational lens. For example, Cohn et al. [25] conducted a feasibility study of detecting depression using facial actions and vocal prosody. Wang et al. [123] analyzed video-based facial expressions to study neuropsychiatric disorders such as Asperger's Syndrome and Schizophrenia.

In the context of Parkinson's disease, Wu et al. [126] conducted a preliminary study to quantify facial expressivity of patients with PD by comparing AU activations between a group of 7 Parkinson's patients and 8 control patients. The authors quantify facial expressivity by manually defining a mathematical formula based on automatically detected AUs, and demonstrate a significant difference in facial expressivity between the control group and the patients. Bandini et al. [9] reported, from a pilot experiment involving 4 patients and 4 people in a control group, that control subjects exhibit higher distances from a neutral face when expressing emotions compared to PD patients. Almutiry et al. [1] found that certain expressions, such as happiness and disgust, are most discriminative when comparing the expressive behavior of PD patients with healthy controls.

In this thesis, our focus is not on quantifying differences in facial expressivity characteristics between PD patients and individuals in a control group. Instead, we use a larger dataset of approximately 800 data points of 117 patient interview audio-video inputs and their corresponding expertly annotated facial expressivity labels to automatically *learn* a function that maps the multimodal input feature representation to the facial expressivity score.

2.3 Modeling Affect in Intelligent Tutoring Systems

Another potential application of applying emotion and expression analysis is in the domain of education. One of the drawbacks of large classroom models of pedagogy is the inability of the teacher to cater to the varying needs of individual students. One-on-one human tutoring, which overcomes this drawback, has been shown to be a more effective means of teaching [121].The goal of Intelligent Tutoring Systems (ITSs) is to similarly provide a platform capable of delivering a personalized learning experience as per the needs and requirements of the student [55]. A popular example of an ITS is MathSpring [4], formerly known as Wayang Outpost, which is a web-based ITS for learning mathematics concepts for middle and high school students.

An important source of information that enables human tutors to provide a personalized feedback is the affective state of the user. Students display a variety of emotions, such as interest, flow, surprise, anger, boredom, frustration, confusion and anxiety, during learning [34]. Emotions felt and displayed by students have been shown to correlate well with their achievement in the learning task [90]. In recent years, advances in computer vision and machine learning have led to the development of fast and robust facial expression analysis tools [8]. Many ITSs have incorporated variants of such emotion analysis capabilities, so that they can utilize the observed affective states of the users to provide more appropriate feedback via their user interfaces. Adapting to student affective states, such as uncertainty and confusion, as measured by ITSs have shown to improve their effectiveness [20, 33]. Grafsgaard et al. [52] showed that different facial expressions measured correspond to different learning experiences.

Student affect has been modeled using a variety of signals. For example, Wixon et al. [125] used student self-reports to build affect models. Corrigan et al. [28] trained detectors based on log data. A common signal channel used by ITSs to model affective states is camera-based captures of the user's face. For example, EER-Tutor [138] tracks the facial features of the users with a video camera to classify the user's face to states such as happy, smiling, angry and neutral. Similarly, the FERMAT tutor [139] uses a video stream of the user to classify the face into one of the seven basic emotions: angry, disgusted, scared, happy, sad, surprised and neutral. In contrast to the well-studied basic emotions, it has been shown that learning-centric emotions such as boredom and confusion feature more prominently during the process of interacting with ITSs [6, 36]. AutoTutor [34] uses a video camera as well as a pressure-sensitive chair to recognize learning-centric emotional states such as flow, confusion, boredom, frustration and eureka. Guru Tutor [87] utilizes a video camera and eye tracker to measure a student's level of interest and boredom.

Another key concept in student learning is that of engagement. Automatic engagement detection not only allows ITSs to adjust their teaching strategies in real-time, but also allows educational materials to be analyzed to determine which portions causes disengagement. Whitehill et al. [124] presented computer-vision based techniques for automatic engagement detect. D'Mello et al. [35] introduced an advanced, analytic and automated approach to measure engagement at fine-grained temporal resolutions. Kaur et al. [81] introduced a new dataset for student engagement detection and localization in the wild.

In this thesis, we ask the following question: given a video of a student interacting with an intelligent tutoring system, can we train a model to directly predict the learning outcome? Compared to existing works, which focus on modeling emotional states such as engagement or boredom, we wish to utilize facial affect signal to directly predict whether the student will successfully solve the problem. The ability to infer whether the student is going to solve the problem, require hints, or give up can then be used to provide appropriate interventions.

2.4 Personalization

Personalization approaches have been developed for speech [101], handwriting [27, 62], facial action unit recognition [21] and gestures [59]. Work on domain adaptation that either adapts model parameters [131] or feature representations [99] is closely related to these approaches. Our work draws on previous efforts in hierarchical Bayesian domain adaptation [42]. We extend this line of work by parameterizing group/domain-specific conditional distributions via more flexible Bayesian neural networks in place of simpler log-linear models.

A particular challenge faced by personalization systems is the small amounts of subject-specific data available for personalization. Yao et al. [135] tackled this by recasting the problem into one of selecting the best performing model from a portfolio of pre-trained models. Since no new learning occurs, the approach is very data efficient. However, they find it to be outperformed by baselines where the models are partially or fully re-trained given new personalization instances. We deal with data paucity by resorting to Bayesian neural networks.

Pioneering work on Bayesian neural networks can be traced back to [17, 73, 82]. Recent progress in deep learning along with advances in scalable inference has reinvigorated interest in them. Hierarchical Bayesian neural networks have previously been proposed [49, 70]. However, they rely on expensive Markov chain Monte-Carlo inference and fail to scale to even moderate sized architectures. In contrast, we exploit stochastic variational methods [12, 115] that scale to both large architectures and large datasets. Previous work has developed such algorithms for Bayesian neural network [12] and Bayesian logistic regression [115] models. We introduce a stochastic variational formulation for hierarchical Bayesian neural networks. Bayesian neural networks have been shown to better represent model uncertainty and are therefore appropriate in active learning scenarios

[44]. Likewise, we exploit the inferred posterior over weights to guide active learning [58] methods that significantly improve performance of the system in scenarios where labeling data is expensive.

Like in gesture recognition, most existing work on problems in affect and expression analysis focus on building generic and generalizable classifiers (e.g. [120, 63]). However, there have been some recent works focusing on personalization of classifiers, i.e. tailoring classifiers to adapt to individual variances, e.g. in modeling facial AU intensity [132, 21] and pain recognition [72]. Rudovic et al. [98] used a context-sensitive model to estimate AU intensity, where context is defined by *who*: the identity of the individual, *when*: the timing of the facial expressions and *how*: how the facial expressions change over time.

In the context of ITSs, Grafsgaard et al. [53] investigated how facial expression patterns differ by age. Similarly, Vail et al. [118] studied gender differences in facial expressions during learning.

Unlike previously proposed works that focus on a single personalization task, we propose, in this thesis, a generic personalization framework and validate it on three human signal analysis problems: subject-specific gesture recognition, context-specific facial expressivity prediction and student-specific learning outcome prediction.

Chapter 3

Comparing Random Forest Approaches to Segmenting and Classifying Gestures

Performing gesture recognition in untrimmed videos is a challenging problem: in addition to correctly identifying the gesture label, the gesture also needs to be accurately localized, i.e. the times at which in-vocabulary gestures start and end need to be determined. In this chapter, we compare two approaches: a method that performs the tasks of temporal segmentation and classification simultaneously with another that performs the tasks sequentially. The first method trains a single random forest model to recognize gestures from a given vocabulary, as presented in a training dataset of video plus 3D body joint locations, as well as out-of-vocabulary (non-gesture) instances. The second method employs a cascaded approach, training first a binary random forest model to distinguish gestures from background and a multi-class random forest model to classify segmented gestures. Given a test input video stream, both frameworks are applied using sliding windows at multiple temporal scales. We evaluated our formulation in segmenting and recognizing gestures on two different benchmark datasets: the NATOPS dataset of 9600 gesture instances from a vocabulary of 24 aircraft handling signals, and the ChaLearn dataset of 7754 gesture instances from a vocabulary of 20 Italian communication gestures.

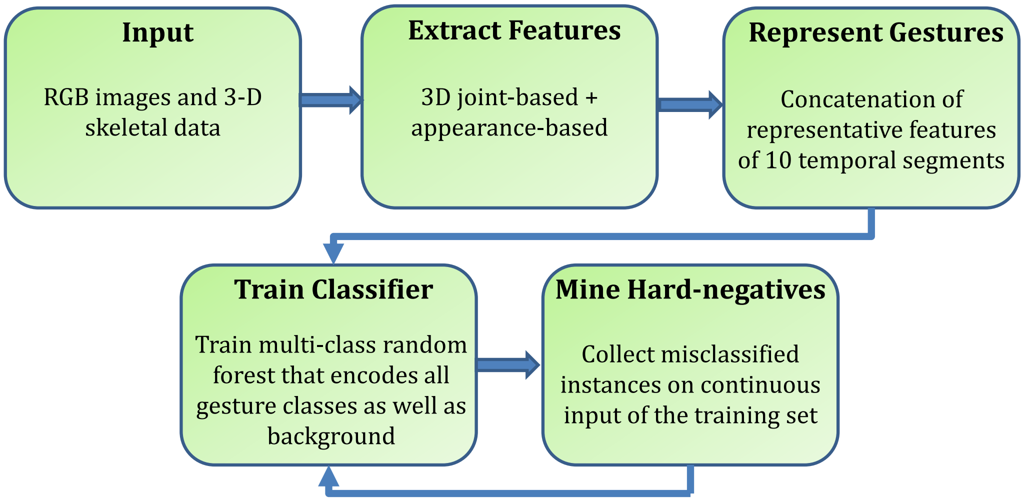

Input
RGB images and 3-D skeletal data
Extract Features
3D joint-based + appearance-based
Represent Gestures
Concatenation of representative features of 10 temporal segments
Train Classifier
Train multi-class random forest that encodes all gesture classes as well as background
Mine Hard-negatives
Collect misclassified instances on continuous input of the training set

Figure 3.1: Pipeline view of training our gesture recognition framework that performs simultaneous spotting and classification

3.1 System Overview

Here, we describe in detail the formulation of both gesture recognition systems. We first explain the differences in the procedures used in training our random forest frameworks, and then illustrate how the classifiers are used to spot and classify gestures from a continuous stream. Pictorial overviews of training the two frameworks are depicted in Figures 3.1 and 3.2.

3.1.1 Training

The training set of gestures used in our experiments is labeled with true temporal segmentation as well as classification values. That is, each video sample used in training is associated with a file that describes the class labels of the gestures that are present in the video, along with their start and end frames.

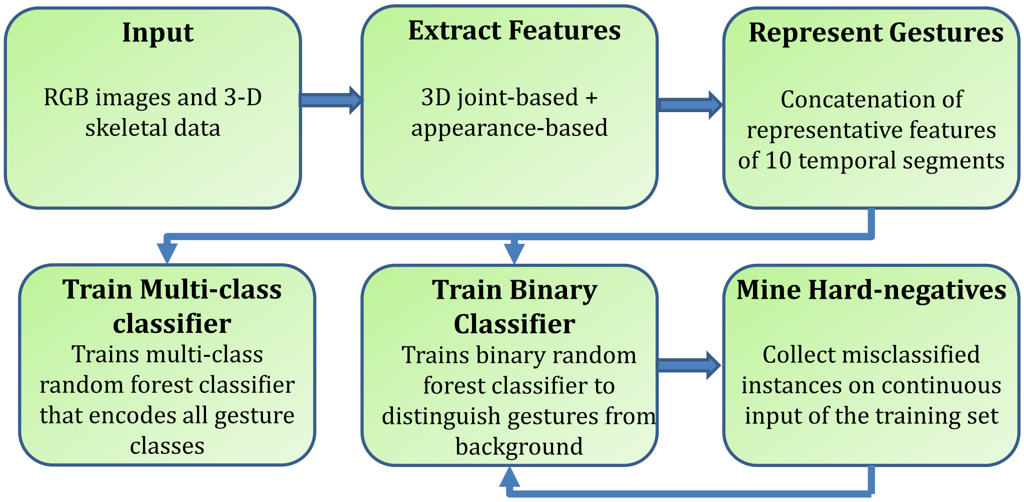

Input
RGB images and 3-D skeletal data
Extract Features
3D joint-based + appearance-based
Represent Gestures
Concatenation of representative features of 10 temporal segments
Train Multi-class classifier
Trains multi-class random forest classifier that encodes all gesture classes
Train Binary Classifier
Trains binary random forest classifier to distinguish gestures from background
Mine Hard-negatives
Collect misclassified instances on continuous input of the training set

Figure 3.2: Pipeline view of training our cascaded gesture recognition framework that first spots a gesture before classifying it

3.1.1.1 Simultaneous spotting and classification framework

Let n be the number of different gestures that are present in the gesture vocabulary. We trained an $n+1$-class random forest classifier using all examples of the n different gestures in the training set, as well as some randomly selected examples of non-gestures (found in intervals between two gestures). Non-gestural examples may contain a sequence of gestural silence, that is when the user is relatively static, or they may contain non-gestural movements, that is when the user is moving or performing out-of-vocabulary gestures.

3.1.1.2 Cascaded spotting and classification framework

For the cascaded framework, we trained a binary random forest classifier using all instances of the n different gestures in the training set as positive examples and an equivalent number of randomly selected instances of non-gestures (found in intervals between two gestures) as negative examples. This binary classifier was used during test time to distinguish a gesture from the background. Additionally, we trained an n-class random forest classifier using all examples of the n different gestures in the training set. This multiclass classifier was used during test time to predict the class label of a candidate gesture spotted by the binary classifier.

3.1.1.3 Feature Extraction

Each training example consists of a varying number of frames, each of which is described by a feature descriptor. In both frameworks, our system computes normalized positional and velocity features for nine different skeletal body joints (left and right shoulders, elbows, wrists and hands, as well as the head joint). Since gestures are performed by subjects with different heights, at different distances from the camera sensor, we first normalized the positional coordinates of the users' joints using the length of the user's torso as a reference. The normalized position vector for joint j at time t is:

$$\mathbf{W}_j(t) = \frac{\mathbf{W}^r_j(t) - \mathbf{W}^r_{hip}(t)}{l}, \tag{3.1}$$

where $\mathbf{W}^r_j(t)$ is the raw position vector for joint j at time t, $\mathbf{W}^r_{hip}(t)$ is the raw position vector for the hip joint at time t, and l is the length of the torso defined as:

$$l = \|(\mathbf{W}_{head} - \mathbf{W}_{hip})\|. \tag{3.2}$$

Our system uses the normalized positional coordinates (W_x, W_y, W_z) of these nine joints along with their rotational values (R_x, R_y, R_z, R_w), which are provided with the dataset, and computes values for their velocities $(W'_x, W'_y, W'_z, R'_x, R'_y, R'_z, R'_w)$.
Thus, there are 126 feature descriptors extracted from 3D skeletal data for every frame. In addition, we augment our skeletal feature vector with HOG features on 32x32 pixel

squares centered on the left and right hands. Each 32x32 pixel square window is divided into 4x4 cells. Each window is also divided into 3x3 overlapping blocks (each block contains 2x2 cells) to perform normalization.We obtained a dimensionality-reduced representation of the HOG features by performing Principal Component Analysis (PCA) and using the first 20 principal components for each hand. The first 20 components explained about half of the variance (0.44 %, 0.43 % for the left and right hands respectively) and were chosen so that the resulting feature space was a balanced combination of both the skeletal features obtained from joints as well as hand-appearance features obtained from HOG representations.Thus, every frame of every instance in our training set is represented by a 166 dimensional feature descriptor.

3.1.1.4 Gesture Representation

In order to remove the effects of noisy measurements, we first smoothed all features using a moving average filter spanning 5 frames. Smoothing features slightly improved classification accuracy (an increase in classifier accuracy of 1.4% on a validation set on the NATOPS dataset). Because instances of gestures and non-gestures in our training set are temporal sequences of varying length, there arises the need to represent every gesture with a feature vector of the same length. We achieved this by dividing the gesture into 10 equal-length temporal segments, and representing each temporal segment with a vector of the median elements of all features. Using 10 temporal segments provided a balance between keeping the feature representation concise, while encapsulating enough temporal information useful in discerning the gesture classes. The representative vectors of each temporal segment were then concatenated into a single feature vector.

3.1.1.5 Random Forest Training

We defined the training set as $\mathcal{D} = \{(\mathbf{X}_1, Y_1), ..., (\mathbf{X}_n, Y_n)\}$. Here, $(\mathbf{X}_1, ..., \mathbf{X}_n)$ corresponds to the uniform-length feature vector representing each gesture or non-gesture, and $(Y_1, ..., Y_n)$ represents their corresponding class labels.

A random forest classification model consists of several decision tree classifiers $\{t(\mathbf{x}, \phi_k), k = 1, \ldots\}$ [16]. Each decision tree $t(\mathbf{x}, \phi_k)$ in the forest is constructed until they are fully grown. Here $\mathbf{x}$ is an input vector and ϕ_k is a random vector used to generate a bootstrap sample of objects from the training set $\mathcal{D}$. The ideal number of trees in our random forest model was determined to be 500 by studying the Out-of-Bag (OOB) error rate in the training data.

Let d be the dimensionality of the feature vector of the inputs. At each internal node of the tree, m features are selected randomly from the available d, such that $m < d$. $m = \sqrt{d}$ provided the highest accuracy among other common choices for m (1, $0.5\sqrt{d}$, $2\sqrt{d}$, d). From the m chosen features, the feature that provides the most information gain is selected to split the node. Information gain (I) can be defined as:

$$I_j = H(S_j) - \sum_{k \epsilon (L,R)} \frac{|S_j^k|}{|S|} H(S_j^k), \tag{3.3}$$

where S_j is the set of training points at node j, $H(S_j)$ is the Shannon entropy at node j before the split, and S_j^L and S_j^R are the sets of points at the right child and left child respectively of the parent node j after the split.

The Shannon entropy can be defined as:

$$H(S) = -\sum_{c \epsilon C} p_c log(p_c), \tag{3.4}$$

where S is the set of training points and p_c is the probability of a sample being class c.

We trained and saved a random forest classification model based on the features that we extracted. There is a need to strengthen the classifier's ability to accurately detect intervals of non-gestures because the randomly chosen intervals of non-gestural examples fail to fully model the class of non-gestures. In order to achieve this, we applied the random forest model on continuous input of the training set and collected false positives and false negatives, which are examples of intervals from the training set that the classifier fails to classify correctly. The set of false positive and false negative instances is then added to the original training set, and the random forest is re-trained using the new extended set of training examples. This process of bootstrapping, as performed by Marin et al. [75], is performed iteratively until the number of false positives is reduced below a threshold.

3.1.2 Testing

The task during testing is to use our trained random forest model to determine the temporal segmentation of gestures in a continuous video and accurately classify the segmented gesture. A sample test video contains a number of frames, and the same features collected during training are computed for every frame. Unlike training videos, test videos do not contain information about where gestures start and end. Therefore, we perform multi-scale sliding window classification to predict the class labels of the gestures, as well as their start and end-points.

3.1.2.1 Multi-scale sliding window classification

We performed multi-scale sliding window classification to predict the class labels of the gestures, as well as their start and end points.

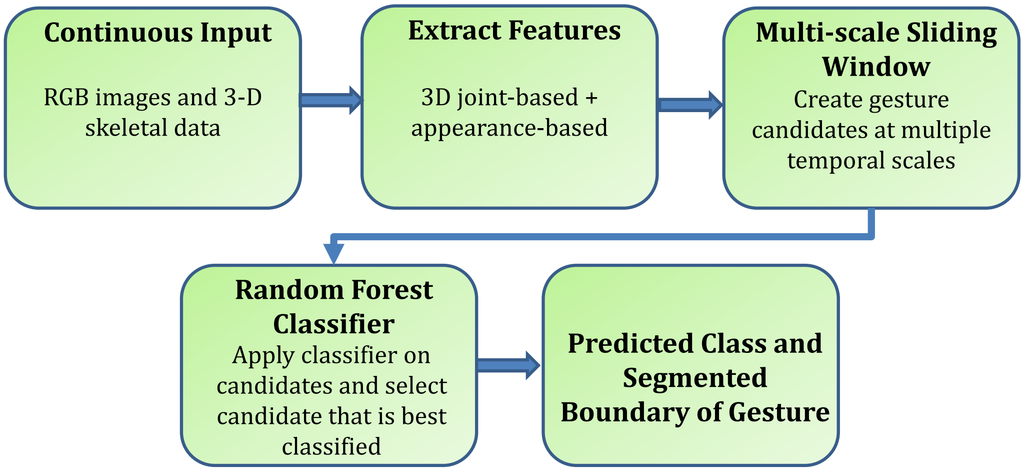

Continuous Input
RGB images and 3-D skeletal data
Extract Features
3D joint-based + appearance-based
Multi-scale Sliding Window
Create gesture candidates at multiple temporal scales
Random Forest Classifier
Apply classifier on candidates and select candidate that is best classified
Predicted Class and Segmented Boundary of Gesture

Figure 3.3: Pipeline view of testing our gesture recognition framework that performs simultaneous spotting and classification

For each input video, gesture candidates were constructed at different temporal scales. Let f_s be the number of frames in the shortest gesture in the training set and f_l be the number of frames in the longest gesture in the training set. Then, the temporal scales ranged from length f_s to length f_l, in increments of 5 frames. Let, $\mathcal{G} = \{g_1, ...g_n\}$ be the set of gesture candidates at different temporal scales. At each scale, a candidate gesture g_i was constructed by concatenating the feature vectors at an interval specified by the temporal scale, so that the dimensions of the feature vector matched those of the gestures used to train the classification model.

Within a buffer of length larger than the longest temporal scale, a sliding window was used to construct gesture candidates at each temporal scale. For a buffer of size b, the number of gesture candidates at scale s_i is equal to $b - s_i + 1$. We chose b to be 100 frames, which is marginally greater than the maximum length of a gesture in the training set. Overviews of training the two frameworks are depicted in Figures 3.3 and 3.4.

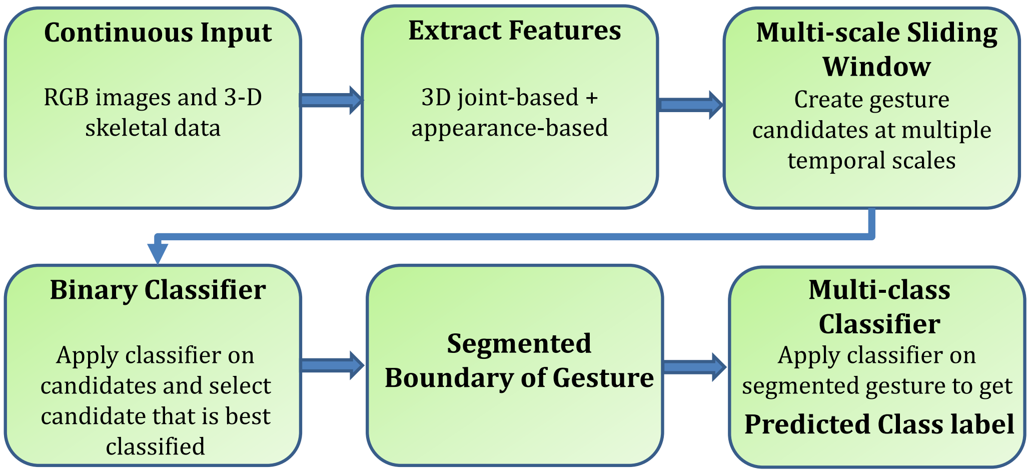

Continuous Input
RGB images and 3-D skeletal data
Extract Features
3D joint-based + appearance-based
Multi-scale Sliding Window
Create gesture candidates at multiple temporal scales
Binary Classifier
Apply classifier on candidates and select candidate that is best classified
Segmented Boundary of Gesture
Multi-class Classifier
Apply classifier on segmented gesture to get
Predicted Class label

Figure 3.4: Pipeline view of testing our cascaded gesture recognition framework that first spots a gesture before classifying it

3.1.2.2 Simultaneous spotting and classification framework

Gesture candidates generated by the sliding window within the temporal neighborhood defined by the buffer at each scale were classified by our trained random forest model and competed to generate a likely gesture candidate G_{s_i} at that scale. Since gesture candidates at the neighborhood of where the gesture is truly temporally located tend to be classified as the same gesture, we performed Non-Maxima Suppression to select the most likely gesture candidate.That is, for each scale s_i, $b - s_i + 1$ gesture candidates were generated and the one classified with the highest confidence (G_{s_i}) within a temporal neighborhood was selected. The confidence score is the percentage of decision trees that vote for the predicted class. Finally, the likely gesture candidates at the various scales competed to generate the final predicted gesture within the buffer.

Therefore, within the buffer, the scale of the final predicted gesture helps determine the segmentation boundaries of the gesture, whereas its class label is that which is predicted

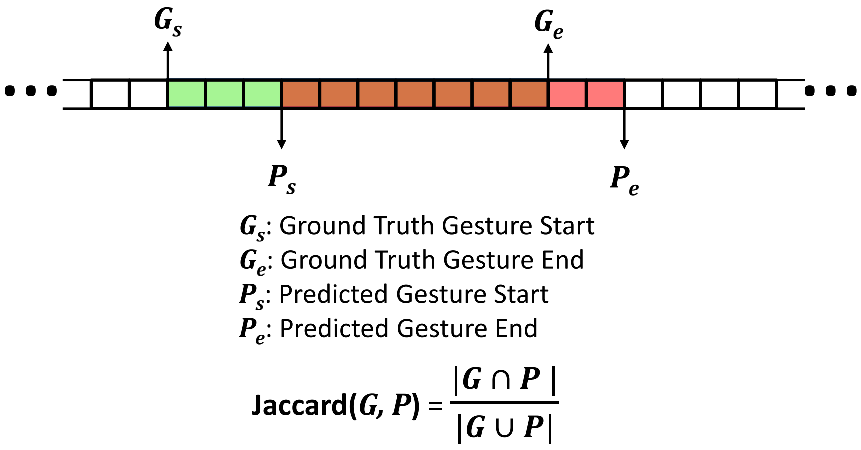

G_s
G_e
P_s
P_e
G_s: Ground Truth Gesture Start
G_e: Ground Truth Gesture End
P_s: Predicted Gesture Start
P_e: Predicted Gesture End
Jaccard$(G, P) = \frac{|G \cap P|}{|G \cup P|}$

Figure 3.5: An example illustration of the Jaccard Score

by the random forest classifier. The end point of the predicted gesture was chosen to be the start point of the new buffer. This process was then repeated until the end of the test video was reached.

3.1.2.3 Cascaded spotting and classification framework

In our cascaded framework, the multi-scale sliding window mechanism outputted whether the gesture candidate was of the gesture or background class, instead of predicting the final class label. Non-overlapping candidates predicted as gestures by the upper-level binary classifier were then given their final gesture label by the multi-class random forest classifier.

3.1.3 Evaluation

In order to evaluate the performance of our gesture spotting and classification frameworks, we use the Jaccard Index score. The Jaccard Index score, in the context of gesture spotting and recognition, is an intersection over union measure that incorporates the evaluation of the predicted gesture label as well as the predicted gesture start and end points [39] and is a common measure for such tasks [18], [83], [92]. For a given sequence of test frames that contains a gesture, the Jaccard Index score can be computed when the ground truth gesture label, the ground truth gesture start and end points, the predicted gesture label and the predicted gesture start and end points are given (as illustrated in Figure 3.5).

3.2 Datasets

Here, we describe in detail the nature of the datasets we have used to test our gesture recognition system.

3.2.1 NATOPS

The Naval Air Training and Operating Procedures Standardization (NATOPS) gesture vocabulary comprises of a set of gestures used to communicate commands to naval aircraft pilots by officers on an aircraft carrier deck. The NATOPS dataset [105] consists of 24 unique aircraft handling signals, which is a subset of the set of gestures in the NATOPS vocabulary, performed by 20 different subjects, where each gesture has been performed 20 times by all subjects. Thus, each gesture has 400 samples. The samples were recorded at 20 FPS using a stereo camera at a resolution of 320 x 240 pixels. The videos were recorded in such a way that the position of the camera and the subject relative to the camera was fixed, and changes in illumination and background was avoided. The dataset includes RGB color images, depth maps, and mask images for each frame of all videos. A 12 dimensional vector of body features (angular joint velocities for the right and left elbows and wrists), as well as an 8 dimensional vector of hand features (probability values for hand shapes for the left and right hands) collected by Song et al. [105] was also provided for all frames of all videos of the dataset. An example gesture for the NATOPS dataset is illustrated in Figure 3.6.

3.2.2 ChaLearn

The ChaLearn dataset was provided as part of the 2014 Looking at People Gesture Recognition Challenge [40]. The focus of the gesture recognition challenge was to create a gesture recognition system trained on several examples of each gesture category performed by various users. The gesture vocabulary contains 20 unique Italian cultural and anthropological signs. Gestural communication is a major part of communication in Italian culture, and developing systems to recognize such gestures is a task that can have many applications.

The development data used to train the recognition system contains a total of 7,754 manually labeled gestures. Additionally, a validation set with 3,363 labelled gestures was provided to test the performance of the trained classifier. During the final evaluation phase, another 2,742 gestures were provided. The gesture examples are contained in several video clips. Along with the RGB data, depth data, user mask data along with skeletal information was also provided. Skeletal information was contained in a .csv file, where world coordinates, rotation values and pixel coordinates were provided for 20 different joints of the user in each frame of the video clip. An example gesture for the ChaLearn dataset is illustrated in Figure 3.7.

3.3 Experiments

Here we describe the experiments performed to evaluate our gesture recognition system on the two datasets. We used the NATOPS dataset to evaluate our gesture classification system in a non-continuous setting. We used a set of gesture samples to train our gesture clas-

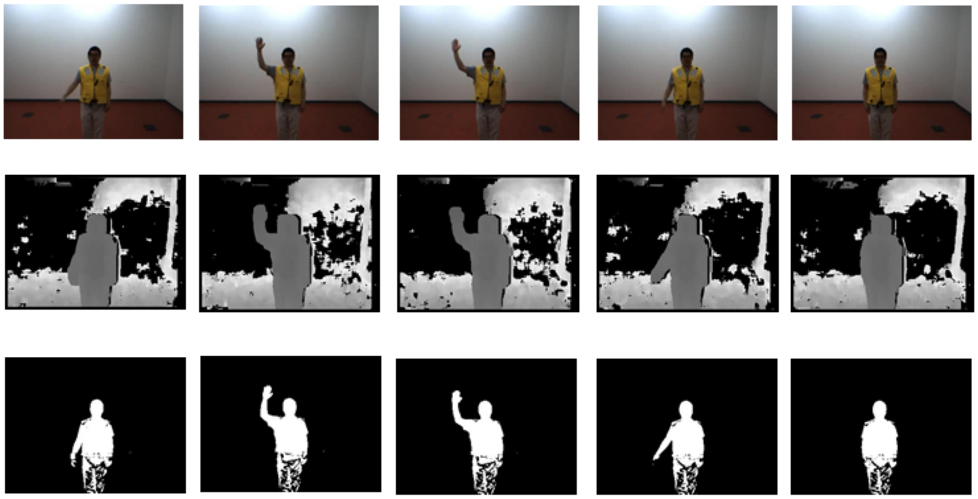


Figure 3.6: RGB, Depth, and User-Mask Segmentation of a subject performing gesture ‘I Have’ in the NATOPS dataset

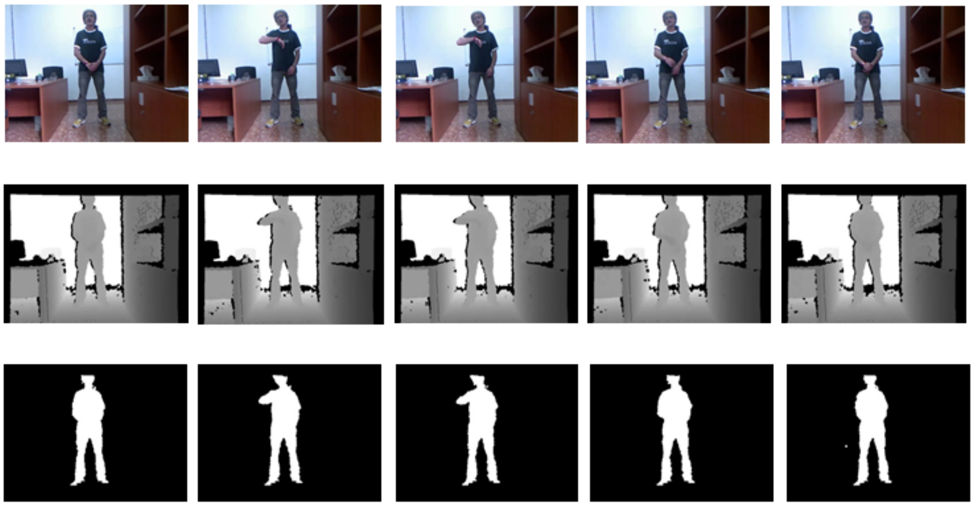


Figure 3.7: RGB, Depth, and User-Mask Segmentation of a subject performing gesture ‘sonostufo’ in the ChaLearn dataset

sifier, and tested its performance on a test-set of pre-segmented gestures. The ChaLearn dataset consists of training and test videos where the user performs both in-vocabulary and out-of-vocabulary gestures, with intervals of gestural silence or transitions. Thus, we used the ChaLearn dataset to test the performance of our system on continuous input.

The difference in evaluation metrics is a consequence of the differences in the nature of the datasets. The NATOPS dataset consists of pre-segmented gesture examples, hence the primary task is to formulate methods to do gesture classification. The Chalearn dataset consists of continuous videos where segments of gesture performance is interspersed with segments of non-gestures. Thus, the challenge is to both *spot* the gesture and *classify* the spotted gesture.

From the NATOPS dataset, we trained our gesture recognition model with the following features sets in order to formulate a good feature representation:

(a) 3D skeletal joints and hand-shape based feature set (SK+HS): This feature set [104] consists of 20 unique features for each timeframe for every gesture. The extracted features are angular joint velocities for the right and left elbows and wrists, as well as probability values of hand shapes for the left and right hands. Since each gesture instance is described by a single feature descriptor obtained by concatenating 10 representative feature vectors, the feature vector representing a gesture instance is of length 200.

(b) Appearance-based feature set (EOD): Each frame of the gesture instances is represented by a 400 dimensional feature vector, which was calculated using randomly pooled edge-orientation and edge-density features. Each gesture example is represented by a single-dimension feature vector of length 4000.

(c) EODPCA: In this feature representation, we reduced the above 4,000-d feature space

Table 3.1: Average Classification accuracy on all 24 gestures of the NATOPS dataset

Feature set	Average Classification Accuracy	Standard Deviation across subjects
Feature set a (SK+HS)	84.7%	5.1
Feature set b (EOD)	76.6%	8.4
Feature set c (EODPCA)	67.7%	9.5
Feature set d (SK+HS+EODPCA)	87.3%	4.9

into a 200-d feature space via Principal Component Analyis (PCA).

(d) SK+HS+EODPCA: This feature set was obtained by concatenating the 200-d 3D skeletal joints and hand-shape based (SK+HS) feature descriptor of a gesture with the corresponding dimensionality-reduced edge orientation and density (EOD-PCA) feature descriptor to form a 400-d feature vector for every gesture.

For each feature set described above, we trained random forests with 500 trees on 19 subjects and tested on the remaining subject in a leave-one-out cross-validation approach.

We computed the average recognition accuracy (averaged across all subjects and all gestures) of the random forest classifier on the four different feature sets (a) - (d) of the NATOPS dataset for all 20 test subjects each performing the 24 gestures in the vocabulary 3.1. The feature set containing 3D skeletal joints and hand-shape features (SK+HS) is correctly classified 84.77% of the time, whereas the feature set containing features based on edge density and orientation is correctly classified 76.63% of the time. This suggests, in our case, that 3D joint-based based features encode more class-discerning information than features based on edge density and orientation. However, the highest classification accuracy of 87.35% is achieved on the feature set that combines joint-based features with appearance-based features, suggesting the benefit of combining the two approaches of collecting features.

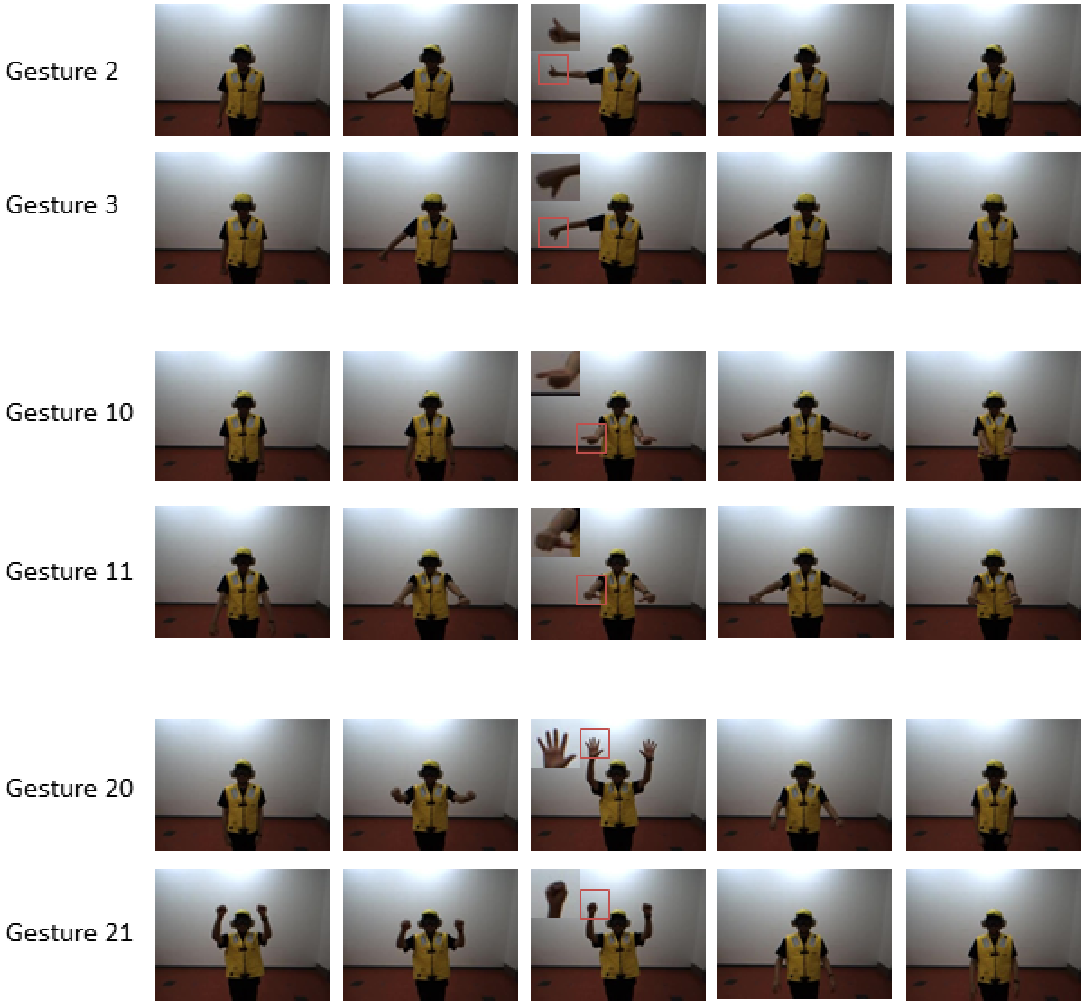
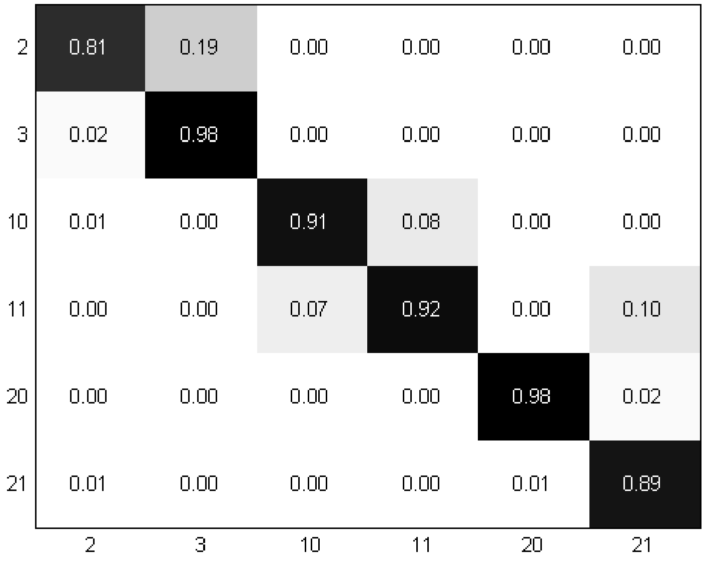


Figure 3.8: Some pairs of similar gestures in the NATOPS dataset



	2	3	10	11	20	21
2	0.81	0.19	0.00	0.00	0.00	0.00
3	0.02	0.98	0.00	0.00	0.00	0.00
10	0.01	0.00	0.91	0.08	0.00	0.00
11	0.00	0.00	0.07	0.92	0.00	0.10
20	0.00	0.00	0.00	0.00	0.98	0.02
21	0.01	0.00	0.00	0.00	0.01	0.89

Figure 3.9: Confusion Matrix for pairs of similar gestures in the NATOPS dataset

Gesture pairs (2,3), (10, 11) and (20, 21) were confused, often getting misclassified as the other 3.8. Figure 3.9 uses a confusion matrix to illustrate the misclassifications between these pairs of similar gestures.

We compared the classification performance of our random forest classifier with the performance of other classifiers that have been used on this dataset (Table 3.2). Our random forest approach on the challenging subset of similar gestures, tested on samples from 5 subjects as specified by Song et al. [107], yields results that exceeds those produced by the state-of-the-art (Linked HCRF) (Table 3.2). The graphical models presented by Song et al. [107] were trained using feature set a (SK+HS), whereas we use feature set d (SK+HS+EODPCA) to train our gesture recognition model.

From the ChaLearn dataset, we trained our gesture recognition model with the following feature sets:

(a) Raw 3D skeletal joint data (RAW): Features contain unedited raw skeleton data, that

Table 3.2: Performance comparison on pairs of similar gestures in the NATOPS dataset with other approaches (The HMM, HCRF, and Linked HCRF) presented by Song et al. [107].

Classifier	Average Classification Accuracy
HMM	77.6%
HCRF	78.0%
Linked HCRF	87.0%
Random Forest (our)	88.1%

is, each frame consists of 9 values for all 20 joints. The feature vector per frame has 180 dimensions, and per gesture has 1800 dimensions.

(b) Normalized skeletal joint positions and velocities (SKPV): This feature set contains normalized positional and velocity data for 9 joints. The feature vector per frame has 126 dimensions, and per gesture has 1260 dimensions.

(c) Normalized skeletal joint positions, velocities and accelerations (SKPVA): This feature set contains positional, velocity, and acceleration data for 9 joints. The feature vector per frame has 189 dimensions, and per gesture has 1890 dimensions.

(d) SK+HOGPCA: This feature set was obtained by concatenating the 1260-d feature vector of normalized skeletal joint positions and velocities (SK) with the 400-d feature vector of HOG data for 32x32 pixel squares around the left and right hands whose dimensionality has been reduced by PCA. The resultant feature vector per gesture example is 1660-d.

For each feature set described above, we trained random forests with 500 trees on gesture instances from the training and validation sets, and tested the performance of our classifier on the test dataset. The division of the data into training, validation and test sets has been described earlier [39].

Table 3.3: Average classification accuracy on all 20 gestures of the ChaLearn dataset

Feature set	Average Classification Accuracy
Feature set a (RAW)	81.4%
Feature set b (SKPV)	88.1%
Feature set c (SKPVA)	83.5%
Feature set d (SK+HOGPCA)	88.9%

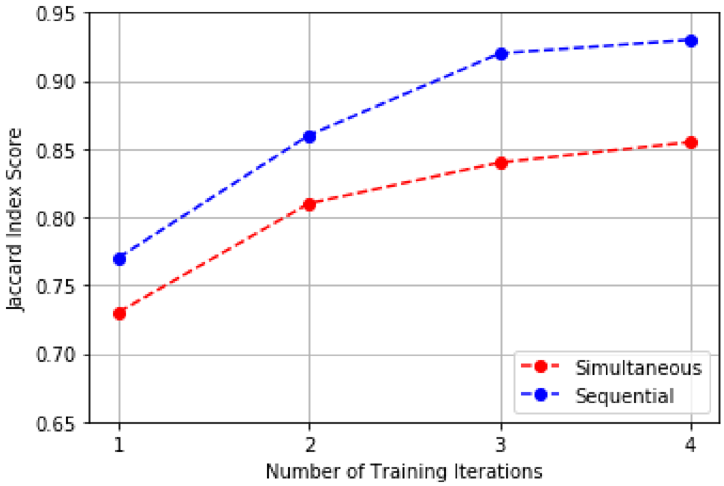


Figure 3.10: Plot comparing the Jaccard index scores of training the simultaneous and sequential classifiers with number of training iterations

The feature set that combines normalized positional and velocity information (SKPV), with HOG features of the hands (HOGPCA), is correctly classified correctly 88.91% of the time (Table 3.3, which is the highest average classification accuracy of all feature sets.

The iterative procedure of training a random forest improves its capacity to correctly classify and segment gestures for both methods. This is evident in the increase in Jaccard scores on the training sets (Figure 3.10).

Table 3.4 shows the Jaccard score of our method compared with the winning scores of the ChaLearn gesture recognition challenge. The competition winner used information from skeleton joints, intensity and depth videos in a deep neural network framework to

Table 3.4: Jaccard Index scores on ChaLearn Gesture Recognition Challenge 2014 [39]

Method	Jaccard Index Score
Deep Neural Network [83]	0.87
Simultaneous Spotting and Classification	0.68
Sequential Spotting and Classification	0.72

achieve a Jaccard score of 0.84 [84]. Our classifier achieves a good recognition accuracy of 88.91% on pre-segmented gestures. One benefit of using a cascaded gesture spotting and classification framework is that it enables separate evaluations of the spotting and classification schemes. The framework which performs spotting and classification simultaneously achieves a Jaccard score of 0.68 whereas the cascaded framework that first spots a gesture before classifying it achieves a score of 0.72.

3.4 Summary

Our method consists of first creating a uniform fixed-dimensional feature representation of all gesture samples, and then using all training samples to train a random forest. On a challenging subset of the NATOPS dataset, our approach yields results comparable to those produced by graphical models such as HCRFs. Although a random forest classifier does not explicitly model the inherent temporal nature of gestural data as done by graphical models, its performance in accuracy on this particular dataset exceeds that achieved by graphical models such as HMMs, and different variants of HCRFs, which are presented by Song et al. [107]. Additionally our experiments also show that classification accuracy was improved by combining 3D skeletal joint-based features with appearance-based features, thus underlying the importance of a well-chosen feature set for a classification task.

We have presented a comparison of random forest frameworks for a multi-gesture classification problem on a continuous setting. On the ChaLearn dataset, our classifier yields an average accuracy of 88.91 % when tested on a set of segmented gestures. However, the task of simultaneously detecting and classifying gestures is a more difficult challenge than classifying accurately segmented gestures. Doing gesture spotting and classification by employing a cascaded framework yields better results than doing simultaneous spotting and classification, suggesting that solving the two problems sequentially is advantageous, especially in datasets where gestures are separated by background.

Chapter 4

Predicting Active Facial Expressivity in People with Parkinson's Disease

Our capacity to engage in meaningful conversations depends on a multitude of communication signals, including verbal delivery of speech, tone and modulation of voice, execution of body gestures, and exhibition of a range of facial expressions. Among these cues, the expressivity of the face strongly indicates the level of one's engagement during a social interaction. It also significantly influences how others perceive one's personality and mood. Objective automated affect analysis systems can be applied to quantify the progression of symptoms in neurodegenerative diseases such as Parkinson's Disease (PD). PD hampers the ability of patients to emote by decreasing the mobility of their facial musculature, a phenomenon known as "facial masking."

In this chapter, we investigate how to computationally predict an accurate and objective score for facial expressivity of a person. We first present an exploratory Action Units-based analysis of a dataset of video clips of Parkinson's patients, attempting to spot trends in how various action unit occurrences vary across patients labeled with different expressivity levels. We then compare predictive models of facial expressivity based on different feature representations: first a baseline model trained on geometric shape features of the face, followed by models trained on more informative action unit features combined with audio features. We evaluated our formulation on a dataset of 772 20-second video clips using

9-fold cross validation. We also provide insight on both geometric as well as action unit features that are important in this prediction task by computing variable importance scores for our features.

4.1 Dataset

The dataset consists of 805 video samples. This dataset was originally collected by Tickle-Degnen et al. in a previous study to determine the effects of self-management rehabilitation on Health-Related Quality-Of-Living in Parkinson's disease [112]. Participants (N = 117) in this study were divided randomly into three groups based on the type of rehabilitation in a 6 week intervention program. All participants in this study had previously been diagnosed with Parkinson's disease by a movement disorder specialist and had the ability to understand and to communicate with personnel [112]. Patients were videotaped participating in standardized social interactions, where cameras were placed to show a mostly frontal face and torso view. From the videotapes, a 20 second representative segment consisting of patients speaking about a positive or negative experience was chosen for analysis. Each video was given 5-point Likert scale ratings for the variables of the ICRP, one of which measures active expressivity of the face, by at least four trained research assistants and a composite score for each variable was computed by taking the average of the scores provided by each rater. Using the intra-class correlation coefficient (ICC), the inter-rater reliability for the variable representing the active expressivity in the face was reported to be .89 (for n = 4 raters) and .67 (for n = 1 rater) [111], suggesting a reasonable level of agreement.

For our experiments, video samples where the subject's face could not be detected in a sufficient number (30) of frames due to occlusion or bad illumination were discarded while building our expressivity prediction model, reducing the size of the dataset from

805 to 772 video samples. This threshold was chosen to maintain a sufficient number of video-label pairs for training while ensuring that features could be extracted from each sample in order to contribute to building an accurate model. The ground truth expressivity labels $y_i \in \mathbb{R}$ for each video was taken as the average of 4 expert ratings.

4.1.1 Exploratory Data Analysis

Facial action units are components of the facial action coding system [38], which was developed to taxonomize human facial movements by their appearance on the face. Because Facial Action Units (AUs) are precisely and anatomically defined, they serve as good candidates to use as features in applications requiring interpretability. Since the ground truth expressivity labels are continuous values, we discretized them into 4 classes to aid our exploratory analyses. Classes 1, 2, 3 and 4 contain samples with facial expressivity ratings in the range [1, 2), [2, 3), [3, 4) and [4, 5] respectively. We visualized how often and with what intensity various action units occur on average for the different expressivity classes of the entire dataset. For each video, we aggregated AU presence values weighted by their respective intensity values and normalized them by the total number of frames in which the face was detected:

$$AUO_a = \frac{1}{N} \sum_j AUI_a^j \times AUP_a^j, \tag{4.1}$$

where, AUO_a represents the mean Action Unit Occurence for AU a of the video, N represents the number of frames in the video in which the face was detected and AUI_a^j and AUP_a^j represent the presence and intensity values of AU_a for frame j respectively.

For all videos belonging to a specific facial expressivity class, we computed the mean AUO for 17 AUs whose presence and intensities were detected by OpenFace and plot them (Figure 4.1). Although it is challenging for automatic AU recognition methods to

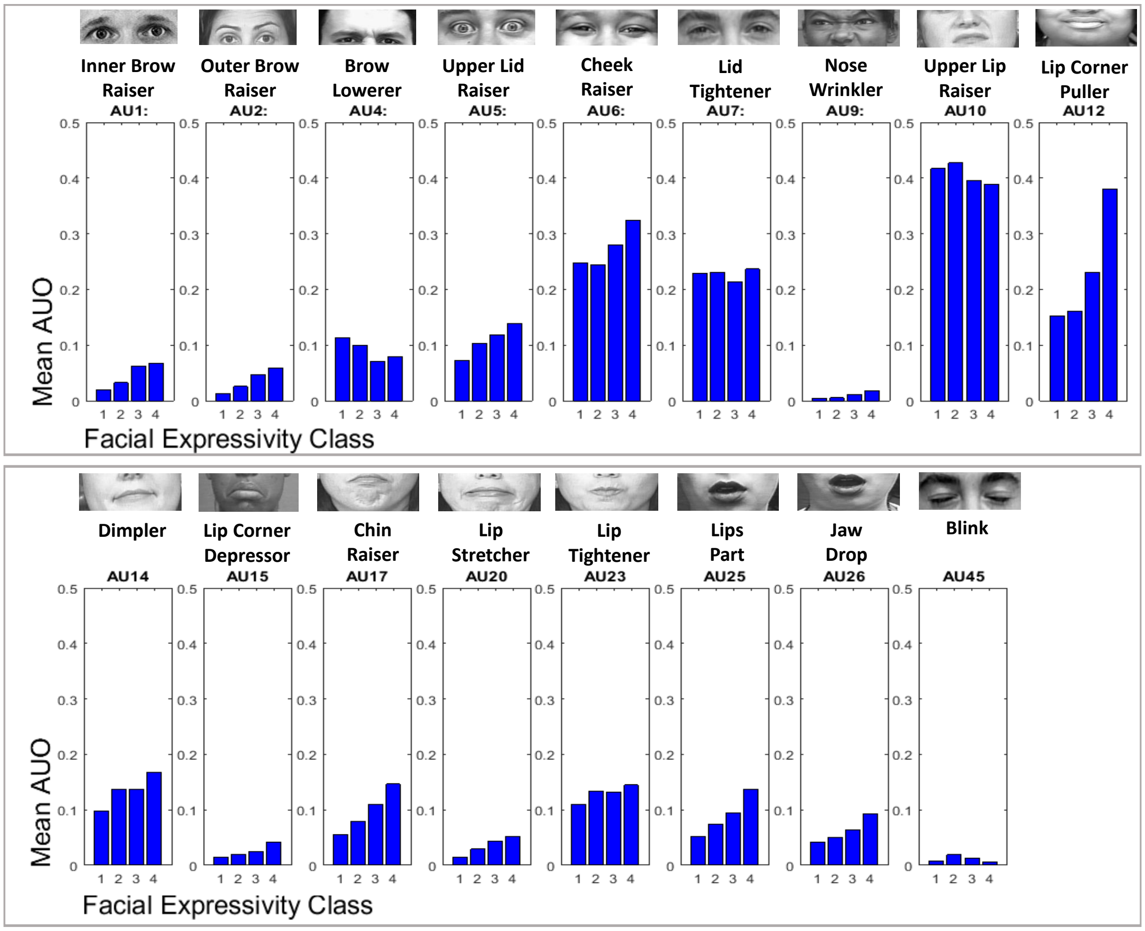


Figure 4.1: For each class in the dataset, the average Action Unit Occurence (AUO) is plotted for 17 different AUs. For each subplot, the x-axis represents the 4 facial expressivity classes whereas the y-axis represents the mean AUO score.

generalize well to datasets beyond the ones on which the models have been trained, we observed that they capture enough signal allowing the analysis of some interesting trends.

On average, we found that the $AUOs$ for several AUs increased when facial expressivity increases. For example, AUs corresponding to brow raising (AU1, AU2), lip corner pulling (AU12), chin raising (AU17), lip stretching (AU20) and jaw dropping (AU26) occured more frequently in videos with higher expressivity values. This indicates that in people deemed to have higher values of facial expressivity, certain Action Units are more frequently activated with higher intensities. For other action units, such as AUs corresponding to brow lowering (AU4), lip tightening (AU23) and blinking (AU45), a clear linear trend was absent. The AU representing upper lip raising (AU10) has the highest AUO values across all classes on average due to the fact that patients are speaking for the entire duration of the video clip.

4.2 System Overview

Here, we explain the elements of our expressivity prediction framework in detail:

4.2.1 Input

The input to our system consists of 20-second audio-video clips of interviews of subjects facing the camera. Most frames in the sequence of images in the video contain full frontal faces of the subject along with the torso. Videos where the frontal face of the subject cannot be detected in a significant number of contiguous frames due to occlusion by the hand or severe out-of-plane rotation of the head were discarded from the training and testing procedure in our framework.

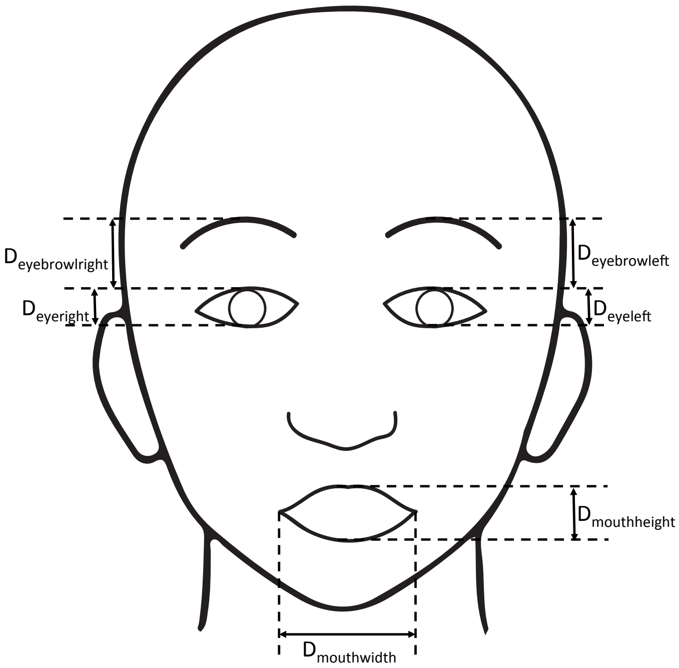


Figure 4.2: Geometric features, which capture facial dynamics, computed by our system

4.2.2 Feature Extraction

From the raw input, we experimented with several feature representations in order to train our expressivity prediction models.

4.2.2.1 Geometric Features

Aside from being informative about discriminative facial events, each geometric attribute has the advantage of being easily interpretable. Geometric features that measure the distance between the brows and the eyes, the height of the eye, the height of the mouth and the angle between the mouth corners have been commonly used in facial expression analysis [31]. Moreover, the facial dynamics associated with these features are studied by ICRP raters while determining the rating for facial expressivity. We aimed to not only maximize expressivity prediction accuracy but also provide insight on the geometric features that are most discriminative. Figure 4.2 illustrates the the geometric features that we computed in this framework.

In order to compute the aforementioned geometric features, we used a robust facial landmark tracker by Asthana et al. [5]. For every frame, the tracker outputs the x and y coordinates of 59 facial landmarks, as well as pitch, roll and yaw angles to describe head pose. Since, active expressivity of the face is a summary score, we extracted geometric features from the temporal signals associated with corresponding facial landmarks in a contiguous sequence of frames.

First, we set the landmark between the two eyes as the origin of the reference frame. To account for the variation in the distances between the subject and the camera and the dimensions of the faces of the different subjects, we normalized the coordinates of the landmarks by taking the inter-ocular distance of the subject as a reference. From the normalized coordinates of the facial landmarks, we extracted distances between certain facial landmarks to describe the dynamics of the eyes ($\mathbf{D}_{eyeright}$, $\mathbf{D}_{eyeleft}$), eyebrows ($\mathbf{D}_{eyebrowright}$, $\mathbf{D}_{eyebrowleft}$) and mouth ($\mathbf{D}_{mouthheight}$, $\mathbf{D}_{mouthwidth}$) at each frame (Figure 4.2).

For each of these distance signals, we computed their first derivatives. Finally, for each signal channel, we computed four quartile values, the standard deviation, and peak frequency, and concatenated them to form a single 72-dimensional representative feature vector.

4.2.2.2 Action Unit Features

Facial action units provide a more detailed description of the movements and dynamics of the face than geometric distances. In order to make use of this richer representation, we extracted 18 AU presence and 17 AU intensity values for every input frame using

OpenFace [8], an open-source library for action unit computation. In order to compute an aggregate feature representation from the per-frame AU presence and intensity values, we used statistics (mean, standard deviation, min and max) for each feature to produce a 140-dimensional visual feature representation.

4.2.2.3 Audio Features

We also wished to investigate whether facial expressivity is correlated with vocal qualities. In order to explore whether facial expressivity can be predicted from the raw audio, we extracted Mel-Frequence Cepstral Coefficient (MFFC) features using the Librosa library [76]. We also computed the same statistics from the MFCC features to obtain a 160-dimensional audio feature representation.

4.2.3 Feature Importance

Feature importance scores indicate how useful a given feature or attribute is in the classification or regression task. One simple and interpretable method of estimating the importance score for a feature is to measure the difference in model error before and after randomly permuting the values of the feature during training [16]. If the difference in error before and after the process of noising up the feature variable is large, one can assume that it plays an important role in the regression or classification task whereas if the difference is negligible, one can assume that the feature has little importance.

4.3 Experiments

Here, we provide a description of the experiments performed on the dataset to evaluate our facial expressivity prediction framework. We first obtained baseline results by training and testing our framework using geometric features. We improved upon those results by utilizing a combination of action unit based features as well as audio based features. For all experiments, we evaluated our models using subject-independent 9-fold cross-validation with the feature representations described above.

4.3.1 Geometric Features

(a) Mouth shape statistics feature set (MS): For each sample video, we computed, for every frame, distances to describe the movement of the mouth. The vector of distances $\mathbf{D}_{mouthheight}$ and $\mathbf{D}_{mouthwidth}$ captures the dynamics of the shape of the mouth. For each distance vector, we computed their first derivatives. Finally, for each signal channel, we computed four quartile values, the standard deviation, and peak frequency, and concatenated them to form a single representative 24-dimensional feature vector.

(b) Eye shape statistics feature set (ES): In this feature representation, we computed distances to capture the dynamics of eye and eyebrow movement. The vector of distances $\mathbf{D}_{eyeleft}$ and $\mathbf{D}_{eyeright}$ measures the movement of the left and right eyelids respectively. The vector of distances $\mathbf{D}_{eyebrowleft}$ and $\mathbf{D}_{eyebrowright}$ measures the raising of the left and right eyebrows. For each distance vector, we computed their first derivatives and for each signal channel, we computed four quartile values, the standard deviation, and peak frequency, and concatenated them to form a single representative 48-dimensional feature vector.

(c) Combined Geometric shape statistics feature set (MS+ES): In this feature representation, we concatenated the aforementioned feature vectors to produce a combined 72-dimensional feature vector.

For each feature set described above, we trained and tested each feature representation using a baseline regression model of random forests with 150 trees using 9-fold cross-validation. We determined the ideal number of trees in the forest by observing the average Out-Of-Bag (OOB) error rate while training our model with each feature set.

We computed the mean absolute errors and R^2 scores averaged over all folds along with their respective standard deviations for all feature sets. The Mean Absolute Error (MAE) is given by:

$$MAE = \frac{\sum_{i=1}^{N} |Y_i - Y_{pred_i}|}{N}, \tag{4.2}$$

where, Y_i and Y_{pred_i} correspond to ground truth and predicted scores and N is the number of test samples. The MAE score accounts for the average absolute error of the predicted scores.

The R^2 score is given by:

$$R^2 = (1 - \frac{\sum_{i=1}^{N} (Y_i - Y_{pred_i})^2}{\sum_{i=1}^{N} (Y_i - Y_{mean})^2}) \times 100, \tag{4.3}$$

where, Y_{mean} corresponds to the mean of the ground truth. The R^2 score is based on the ratio of the error made by the model to the error made by a baseline predictor that always predicts the mean score of the training data. The R^2 score gives a measure of the relative improvement in the Mean Square Error (MSE) of our regression model with respect to the baseline mean expressivity predictor.

Our analysis shows that the feature set containing eye shape statistics (ES) has the lowest mean absolute error of 0.56, and the highest R^2 score of 42.33, averaged over 9 folds (Figure 4.3). The feature set based only on mouth shape dynamics performs worse on both measures.

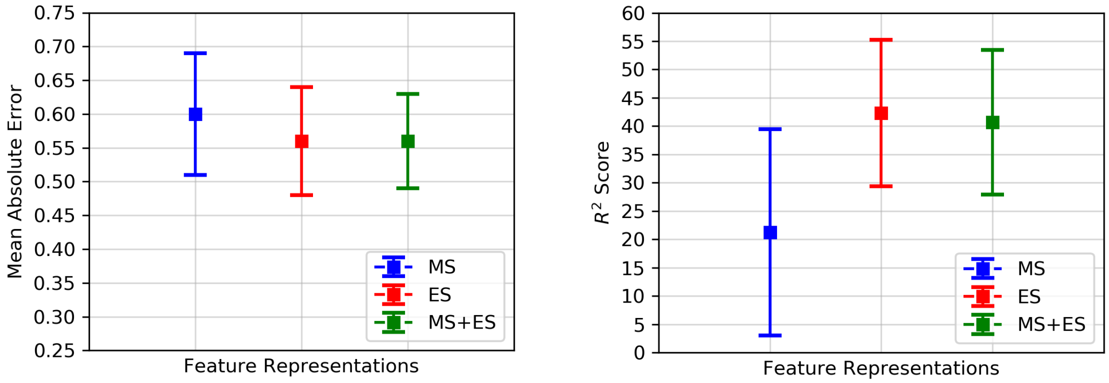


Figure 4.3: (Left) The mean MAE-scores and their standard deviations for models trained using different geometric feature sets (MS, ES, MS+ES). (Right) The mean R^2 scores and their standard deviations for models trained using different geometric feature sets (MS, ES, MS+ES).

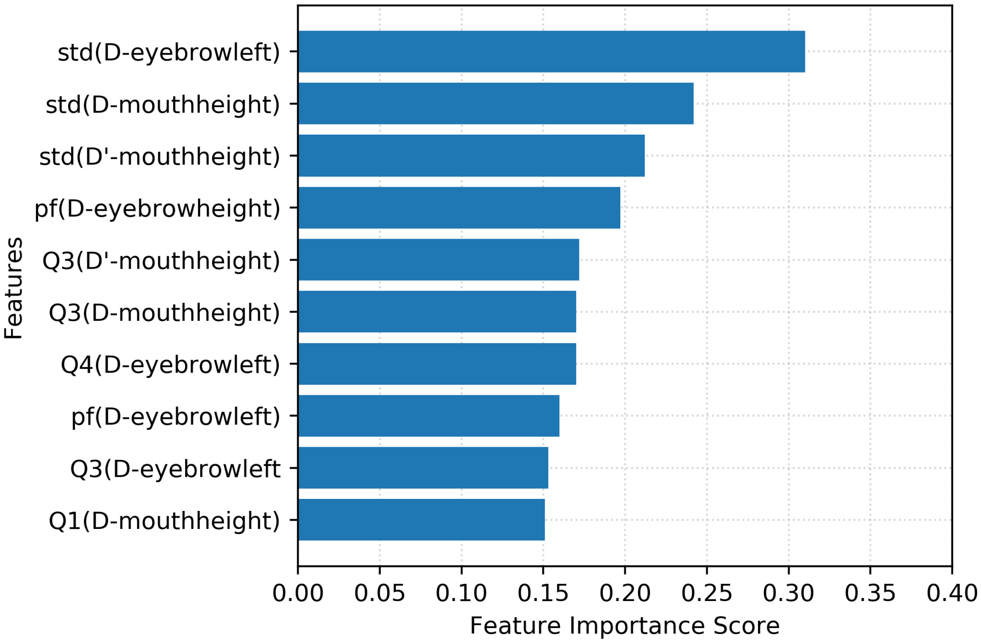


Figure 4.4: Bar graph displaying features with 10 highest feature importance scores. (std: standard deviation, pf: peak frequency, Q: quartile)

For the MS+ES feature set, we computed the feature importance estimates, averaged over all folds, and sorted them. The ten features with the highest importance scores are shown in Figure 4.4. The average difference in MSE before and after randomly permuting this set of features is the highest among all features. We observe that different attributes of $\mathbf{D}_{\text{eyebrowleft}}$, $\mathbf{D}_{\text{eyebrowright}}$ and $\mathbf{D}_{\text{mouthheight}}$ distance vectors populate this list, indicating their importance to the regression task.

4.3.2 Action Unit and Audio Features

For experiments with action unit and audio feature representations, we trained regression models using Hierarchical Bayesian Neural Networks (HBNN), a supervised machine learning framework which will be introduced in Chapter 6. We trained a model with 1 hidden layer containing 50 hidden nodes, and RMSprop [114] was used for optimization.

We trained regression models with all data pooled into one group and trained our model with visual features, consisting of Action Unit statistics (AU), audio features, consisting of MFCC statistics (MFCC), as well as a combined audio-video feature representation (AU+MFCC) (Figure 4.5).

We found that the model trained solely on AU features obtained a mean absolute error (MAE) of 0.51 and an R^2 score of 39.8. We note that this regression model performs better than the baseline random forest model trained on geometric features, as described above. We found that using features computed from the raw audio also led to reasonable model performance (0.62 mean MAE, 26.6 mean R^2 score). In instances where the video is missing, corrupted, or of low quality for automated facial analysis, expressivity could therefore be estimated solely from audio. However, using a combined multimodal feature representation of both video and audio features yielded the best performance (mean MAE-score of 0.49, mean R^2 score of 47.3).

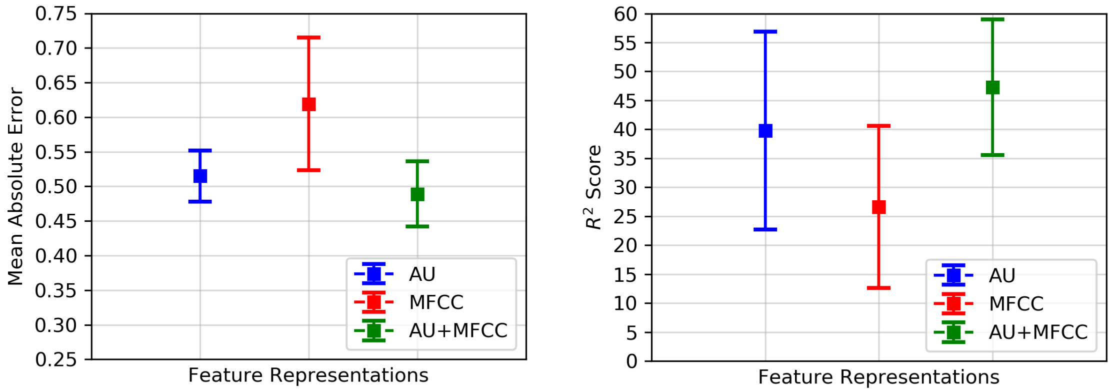


Figure 4.5: (Left) The mean MAE-scores and their standard deviations for models trained using different audio-video feature sets (AU, MFCC, AU+MFCC). (Right) The mean R^2 scores and their standard deviations for models trained using different audio-video feature sets (AU, MFCC, AU+MFCC).

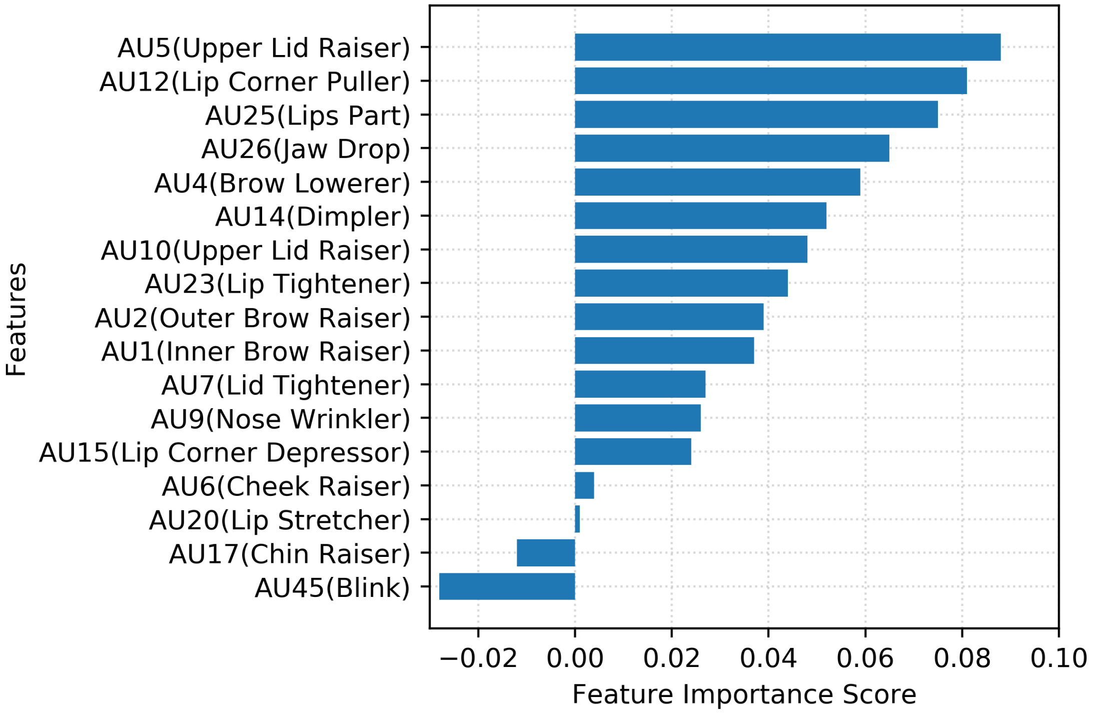


Figure 4.6: Bar graph displaying aggregated feature importance scores for the different action units.

Using the model trained on visual features (AU), we also computed an estimate of feature importance for all visual features over all folds (Figure 4.6). In order to obtain a heuristic of importance associated with each individual AU for the task of facial expressivity prediction, we averaged the scores for all features associated with any given action unit. For example, the importance score for AU5 (Upper Lid Raiser) is computed by taking the mean of all feature importance scores corresponding to the 8 features associated with AU5 (means, standard deviations, maxes and mins of AU5 presence and intensity values).

AU5 (Upper Lid Raiser), often associated with expressions displaying shock and anger, and AU12 (Lip Corner Puller), associated with expressions containing smiles, scored the highest in the AU importance heuristic, whereas AU45 (Blink) scored the lowest. It is interesting to note that AUOs computed for AU5 and AU12 (Figure 4.1) exhibited an increasing trend with higher expressivity, which is absent for AU45.

4.4 Summary

Automated assessment of facial expressivity in Parkinson's Disease patients has the potential to be a useful tool for clinicians in this field. However, most existing works in the domain are limited to small-scale pilot studies comparing the characteristics and dynamics of facial expressions exhibited by a small group of PD patients and a separate control group. In this work, we utilized a dataset of 772 short audio-video clips of 117 PD patients along with their facial expressivity labels to train a machine learning model capable of predicting the facial expressivity ratings of new audio-video clips.

We provided an exploratory analysis of facial expressivity in terms of how often various facial Action Units are activated in the videos in the dataset, weighing the activations by their intensities. We observed an increasing trend of AU occurences for several action units, such as AUs 1, 2, 5 and 12 with increasing facial expressivity. We also computed a heuristic importance score for each AU and found that AUs 5 and 12 were deemed most important in the expressivity prediction task, while AUs 17 and 45 were deemed the least important.

We presented a baseline random forest regression framework to predict active expressivity of the face. The method consists of first detecting facial landmarks for a sequence of continuous frames from an input video and extracting geometric shape features. Each input sample was represented by a feature vector computed from statistics of the geometric shape signals. A model trained on these features achieved an average mean absolute error of 0.56 and R^2 score of 40.68. Additionally, we computed importance scores for each feature to provide insight into what geometric shape features are most important in this challenging prediction task.

Finally, we demonstrated the utility of extracting features from not only the visual domain but also the audio in order to accurately predict facial expressivity, finding that a model trained on a combined audio-visual feature representation (MAE score of 0.49, R^2 score of 47.3) comfortably outperformed our geometric features-based baseline model, as well as models trained on features extracted from a single modality.

Chapter 5

Affect-driven Learning Outcomes Prediction in Intelligent Tutoring Systems

Affect-sensitive intelligent tutoring systems attempt to infer the emotional state of users from affect signals and utilize that information to provide responsive interventions that improve the students' learning experience. An important facet of ITS research, which can accelerate the development of effective affect analysis algorithms, is the availability of education domain-specific datasets that can be shared by researchers to develop, improve, and evaluate affect-sensing machine learning algorithms. Considering the dearth of large-scale, publicly available affect datasets in learning and education settings, our research collaborators first collected a facial affect dataset of videos of students working on math problems in MathSpring, a web-based ITS. We then processed this large-scale raw data into a supervised machine learning dataset, where each data instance corresponds to a short videoclip of a student working on a single problem and its corresponding label is the problem outcome.

While most works on affect analysis in educational settings propose methods to model student emotions, such as anger, surprise, engagement, frustration etc., we focus in this chapter on directly mapping useful representations of the facial affect input into predictions of problem outcome labels.

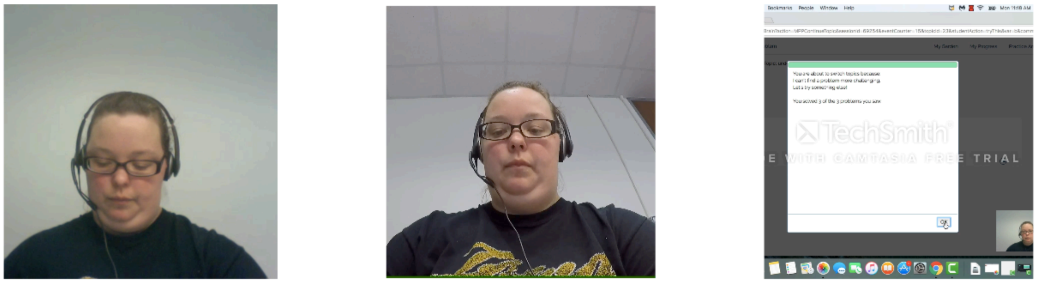


Figure 5.1: Example images of dataset collected from different modalities: A front facing webcam captures the subject while she looks at the screen (left), a secondary GoPro camera is placed at an angle on the laptop trackpad in order to capture the student's face when she faces down to work on the problem at hand (middle) and the MathSpring interface's clickstream logs capture the mouse coordinates of the user (right).

5.1 Dataset Collection and Annotation

The dataset was collected by our research collaborators and consists of video recordings of college students participating in math problem-solving sessions in MathSpring, a popular browser based ITS intended to aid students in the learning of mathematics concepts. A total of 30 students (4 males, 26 females) participated in the study, with several students taking part in multiple sessions, each of which lasted approximately one hour. In total, 38 student sessions were recorded, from which 1596 problem samples were extracted.

The data was collected in a classroom setting, where the students were asked to solve the MathSpring problems on a laptop, while being recorded by two cameras: the laptop webcam along with a GoPro camera placed on the trackpad of the laptop. The purpose of recording with the GoPro was to capture the faces of the participants while they were facing down, for example, when writing on a sheet of paper on the desk while working on the Math problem, and therefore not properly visible on the webcam. All participants provided consent, following due process, in order to facilitate publicly releasing the dataset to foster research in the area. In addition, the clickstream data of the users' mouse coordinates were also captured by the MathSpring interface (Figure 5.1). In this chapter, we only utilized the webcam stream, leaving multimodal affect analysis from all data streams for future work.

Each data instance consists of a video clip of the student working on a single problem. These were obtained by trimming the raw videos based on problem start and end times recorded in MathSpring's log file. Each data instance is associated with the problem-solving effort outcome label. The effort labels are enumerated below:

1. ATT (attempted): student did not see any hints but solved question after 1 incorrect attempt,
2. GIVEUP: student performed some action but did not solve problem at all,
3. GUESS: student did not see hints, but solved question after greater than 1 incorrect attempts,
4. NOTR (not read): student performed some action, but the first action was fast,
5. SHINT (solved with hint): student eventually got the correct answer after seeing one or more hints,
6. SKIP: student skipped problem with no action at all, and
7. SOF (solved on first attempt): student answered correctly in first attempt, without seeing any hints.

Different from existing affect-sensing ITSs, the primary challenge is to build models that attempt to directly predict the outcome of the problem as early as possible, based on the affect signals displayed in the face as captured by the camera. An accurate early prediction of problem outcome could then be used by the ITS to provide effective, appropriate and proactive interventions to improve the student's learning experience.

5.2 Exploratory Data Analysis

5.2.1 Label Distribution

We first plot the distribution of effort labels in the collected dataset (Figure 5.2). Among all the data instances, more than half of the data instances consist of problems solved at first attempt (SOF), whereas instances corresponding to all other labels correspond to less than half of the entire dataset. The class that occurs least frequently in this dataset are instances where the students give up (GIVEUP). One conclusion to be drawn from the distribution of effort labels is that the students in our experiments were not sufficiently challenged. This is in line with our initial expectations, as MathSpring's problem repository was designed primarily for middle and high school students, whereas the participants, who were recruited with the purpose of acquiring consent for public release of the affect dataset, were undergraduate students. This analysis can inform future data collection, in order to garner a more balanced label distribution with sufficient examples that span the effort label space.

5.2.2 Problem Times Analysis

We also plot the average time taken for problems to be completed for different labels in the effort axis (Figure 5.3). We can observe that students, on average, take the longest time while solving a problem with the help of hints (SHINT). Students, on average, take less time to solve a problem at first attempt (SOF) than solving a problem with hints (SHINT), solving a problem after multiple attempts (ATT), guessing an answer (GUESS) and giving

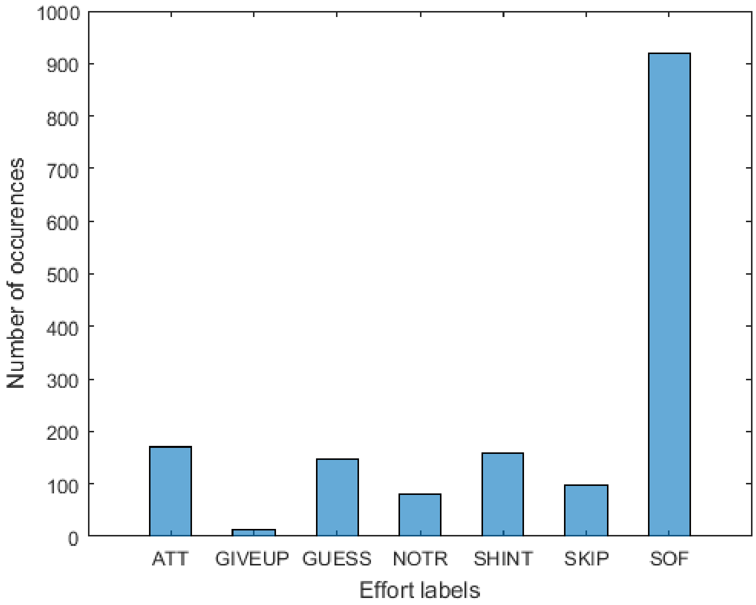


Figure 5.2: Distribution of dataset according to effort labels

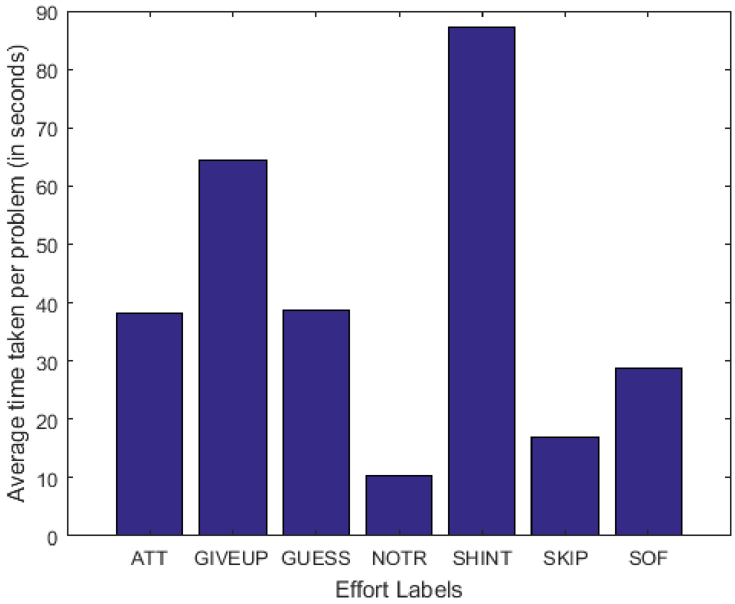


Figure 5.3: Average time for problem completion according to effort labels

up (GIVEUP). The label which, expectedly, corresponds to the least amount of time taken by the participant is when the ITS deems that the student has not read the problem (NOTR).

5.2.3 Action Unit Analysis

Because of the interpretability of Facial Action Units (AUs), we visualized how often and with what intensity various action units occur on average for the different effort and emotion classes of the entire dataset. For each data instance, we aggregated AU presence values weighted by their respective intensity values and normalized them by the total number of frames in which the face was detected:

$$AUO_a = \frac{1}{N} \sum_j AUI_a^j \times AUP_a^j, \tag{5.1}$$

where, AUO_a represents the mean Action Unit Occurence for AU a of the video, N represents the number of frames in the video in which the face was detected and AUI_a^j and AUP_a^j represent the presence and intensity values of AU_a for frame j respectively.

For all videos, we computed the mean AUO for 17 AUs whose presence and intensities were detected by OpenFace and plotted them separated by effort classes (Figure 5.4). From the mean AUO plots, we can observe some interesting average trends. For example, AUs 4 (Brow lowerer), 7 (Lid tightener) and 17 (Chin raiser) are activated comparatively highly across all effort labels, whereas AUs 2 (outer brow raiser), 5 (upper lid raiser) and 20 (lip stretcher) are not. It is interesting to note that AUs 2, 5 and 20 are associated with the emotions of fear and surprise. Another interesting trend is that the mean AUOs across all AUs are higher for instances labeled SHINT compared to inputs labeled SOF, indicating that participants offer more affective expressiveness when requiring hints to solve a problem compared to when they solve them at first attempt.

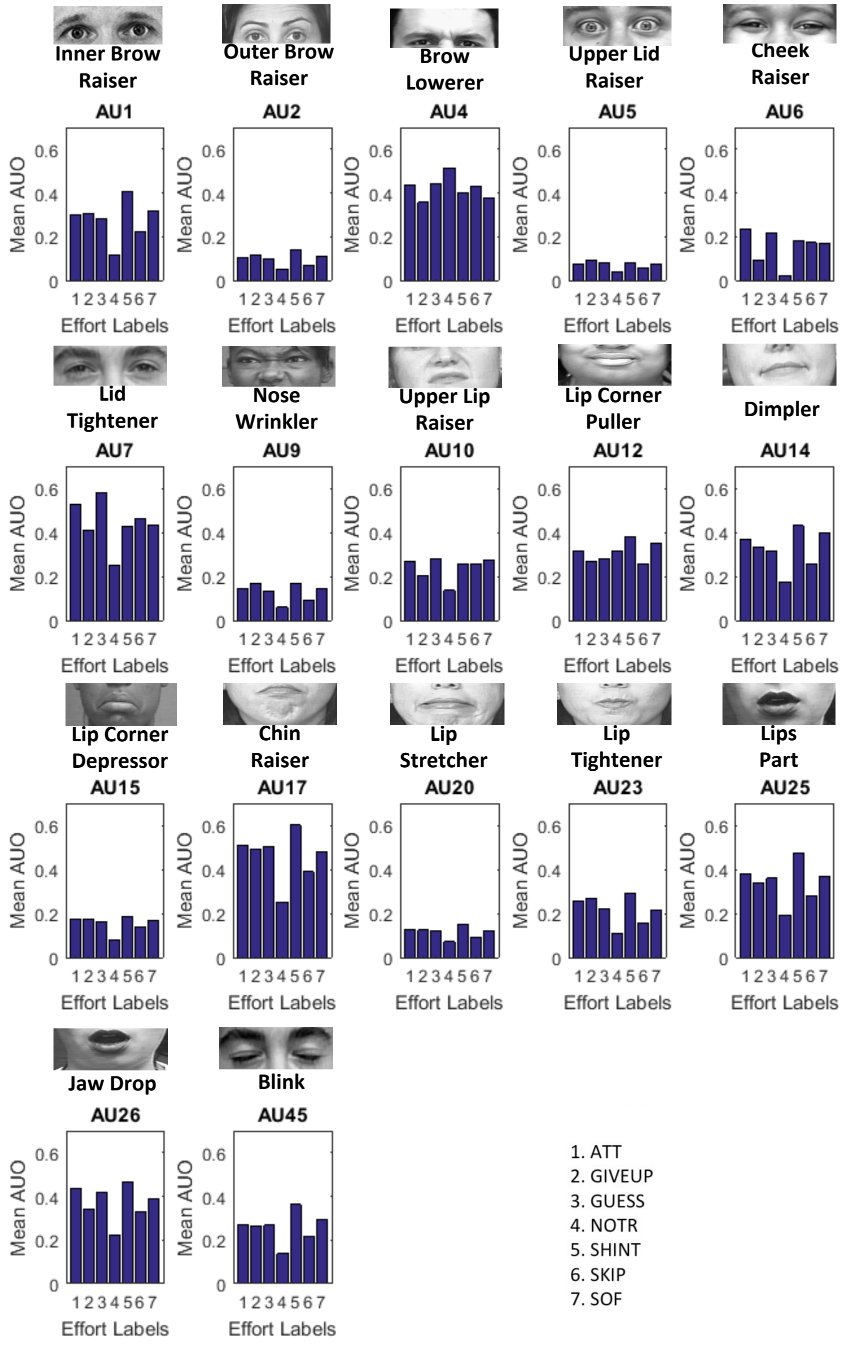


Figure 5.4: Average Action Unit Occurrence distributed according to effort labels

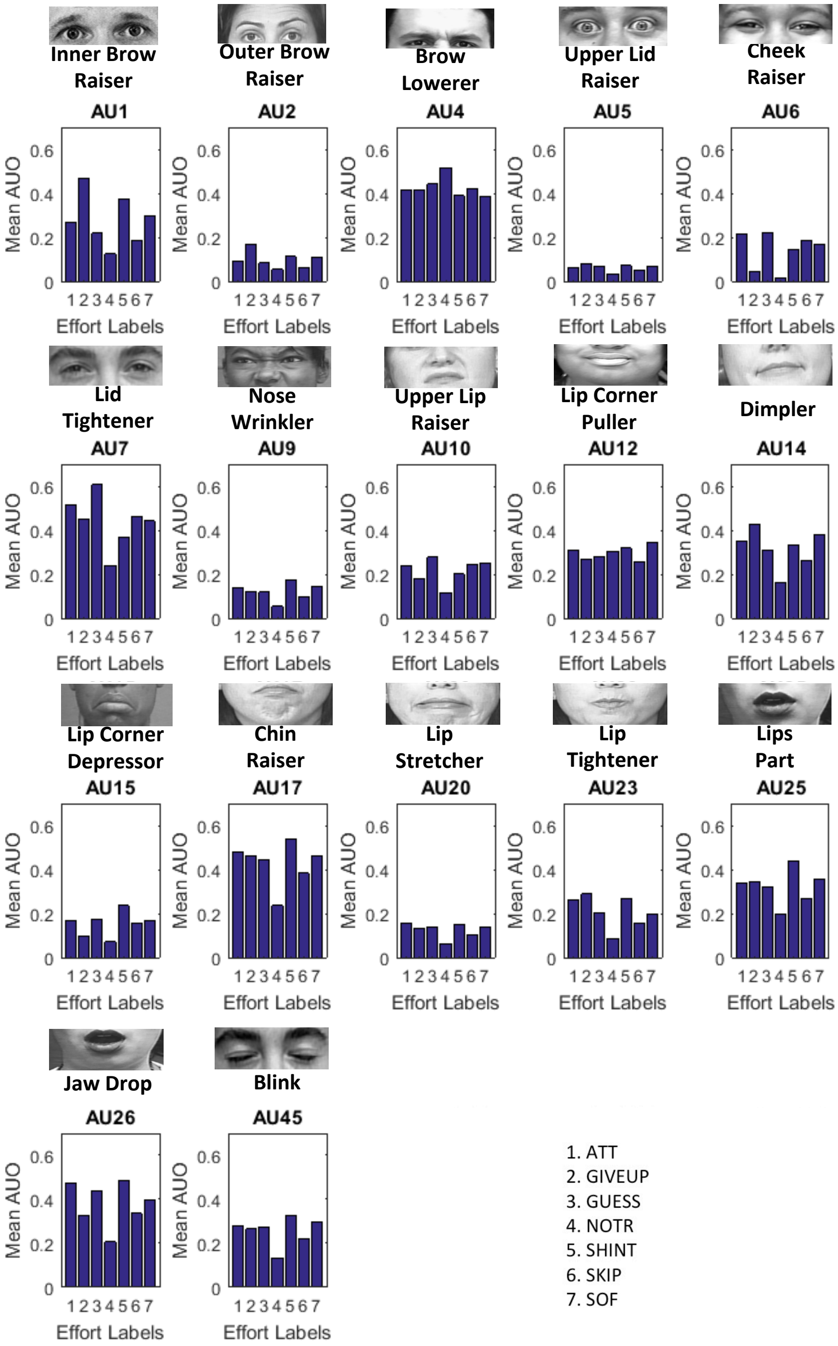


Figure 5.5: Average Action Unit Occurrence distributed according to effort labels for the first 10 seconds of the input

In order for the ITS to provide timely and proactive interventions, the ITS needs to forecast problem outcome labels as early as possible. We plotted the AUOs for the first 10 seconds of the input (Figure 5.5). Comparing Figures 5.4 and 5.5, we can observe that the AUOs when observing only the first 10 seconds of the data are not remarkably different to the AUOs when the entire input is observed. This indicates that the distributions of action unit activations, on average, do not vary significantly in the early and latter stages of the input. Therefore the problem outcome labels could be potentially predicted by training models that observe only a fraction of the input without significantly sacrificing accuracy.

5.3 Baseline Models

The input to our baseline models consists of variable-length webcam videoclips of participants working on MathSpring problems. As stated earlier, the corresponding GoPro video stream and MathSpring clickstream are not used in this work. A significant proportion of the frames contain full frontal faces of the subject, representing times when they are interacting with the ITS (i.e. reading the problem, thinking about the solution, answering the question and interacting with the on-screen educational avatar). For the baseline models, frames where the frontal face of the subject could not be detected due to occlusion by the hand or severe out-of-plane rotation of the head were discarded from the training and testing procedures.

5.4 Feature Representation

For each frame of all the videos in the dataset, 18 AU presence and 17 AU intensity values, along with head-pose and eye-gaze vectors, are extracted using OpenFace [8]. In order to compute an aggregate feature representation, we used statistics (mean, standard deviation, min and max) for each feature as well as statistics for their derivatives to produce a uniform length 376-dimensional feature representation. The derivatives capture the change in feature activations at each timestep. The mean, standard deviation, min and max are representative summary statistics of the variable-length features and concisely capture the distribution of values for each feature, which can be used by the classifier to distinguish samples from different classes. For our experiments, we trained a multi-layer perceptron with 2 hidden layers, each with 100 activation nodes, trained using Adam [64].

5.5 Experiments

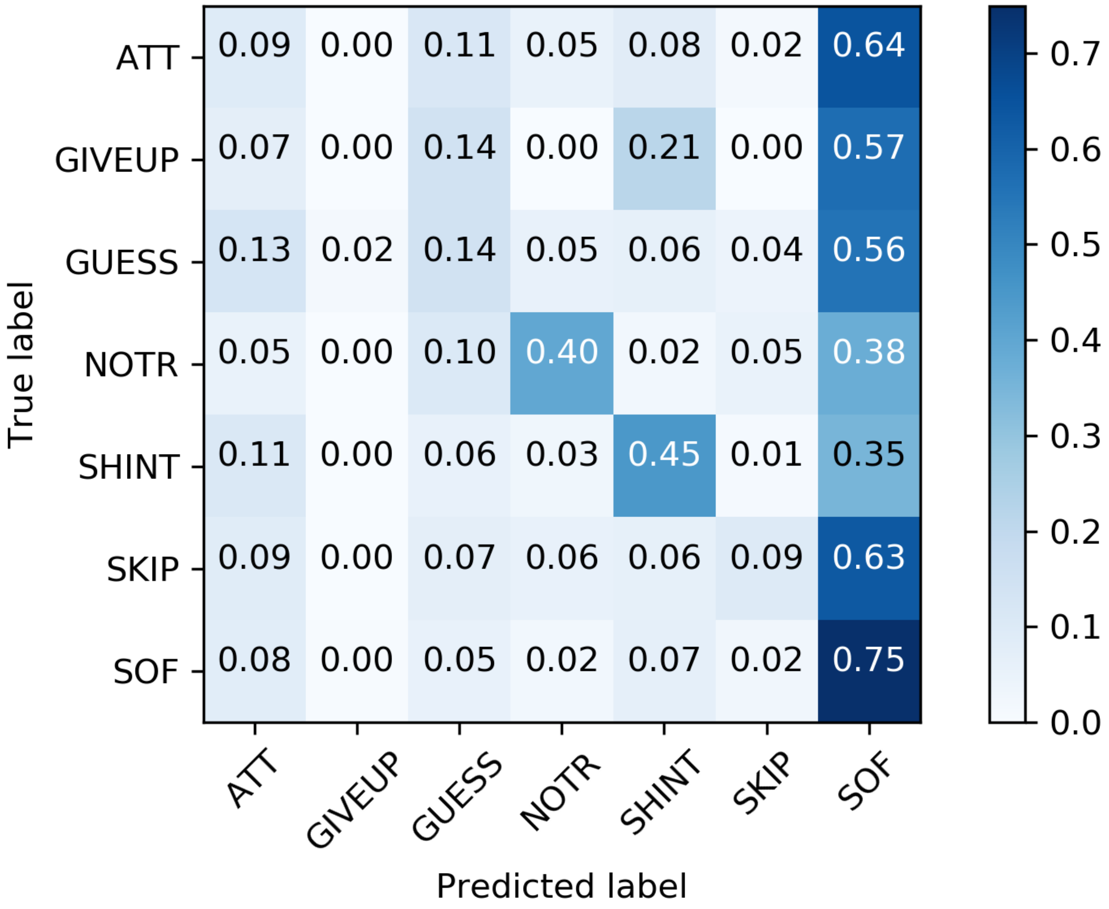
For all our experiments, we trained and tested our models on 5 random, stratified 75/25 splits of the data.

We first trained a model for the multi-class effort prediction task. Our baseline model achieves a mean accuracy of 0.54 and a mean F-score of 0.27 (Table 5.1). Given that our dataset is severely imbalanced with more than half the samples corresponding to the 'SOF' label, the predictions of our model are heavily biased towards that label, as is evident in the figure depicting the normalized confusion matrix (Figure 5.6).

One way to overcome the issue of data imbalance is to oversample the minority classes before training our classifier. We do so using three over-sampling techninques: RAND (Random), which randomly oversamples the under-represented classes with replacement, SMOTE (Synthetic Minority Oversampling Technique) [19], which generates new samples by interpolating in feature space, and ADASYN (Adaptive Synthetic Sampling) [56], which generates samples by interpolating next to original samples which are incorrectly classified by a k-NN classifier. We find that a model trained using RAND and SMOTE achieves slight improvement in classifier performance (Figure 5.7).

Table 5.1: Multi-class effort prediction results

Multiclass Classification Problem	**Accuracy**	**F-score**
Effort	0.54 ± 0.01	0.27 ± 0.01



	ATT	GIVEUP	GUESS	NOTR	SHINT	SKIP	SOF
ATT	0.09	0.00	0.11	0.05	0.08	0.02	0.64
GIVEUP	0.07	0.00	0.14	0.00	0.21	0.00	0.57
GUESS	0.13	0.02	0.14	0.05	0.06	0.04	0.56
NOTR	0.05	0.00	0.10	0.40	0.02	0.05	0.38
SHINT	0.11	0.00	0.06	0.03	0.45	0.01	0.35
SKIP	0.09	0.00	0.07	0.06	0.06	0.09	0.63
SOF	0.08	0.00	0.05	0.02	0.07	0.02	0.75

Figure 5.6: Confusion matrix for 7-class effort prediction

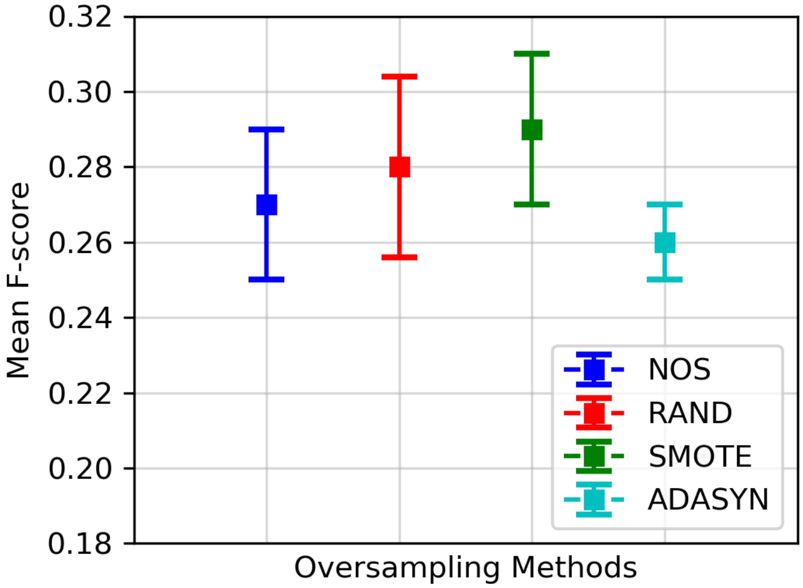


Figure 5.7: Mean F-scores for multiclass classifiers trained with no oversampling (NOS), and various resampling methods.

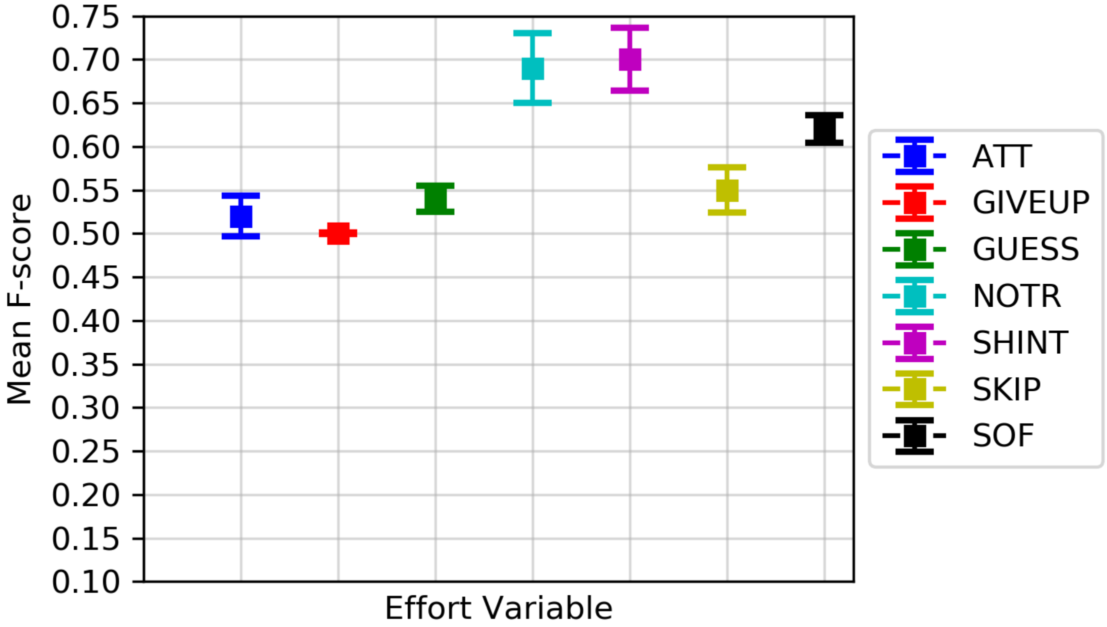


Figure 5.8: Mean F-scores for one-vs-all binary classifiers trained for different problem outcome labels.

5.5.1 Effort Prediction

Because oversampling does not yield significant model performance, we trained individual one-vs-all binary classifiers for all effort labels. A model capable of predicting each of these indicators can help the ITS make decisions with regard to providing proactive interventions. For example, a model that can successfully predict whether a student can answer the question on the first attempt can prompt the ITS to prevent displaying unnecessary hints, or to increase difficulty levels of subsequent questions. Similarly, if the student is predicted to require hints to solve the problem, the ITS can proactively offer a hint before the student asks for it. We find that models trained to predict SHINT, NOTR and SOF yield the best results, indicating that facial affect signals displayed during problem-solving corresponding to these labels are the most discriminative (Figure 5.8).

5.5.2 How long to make Accurate Predictions?

Ideally, an affect-sensitive model should be able to accurately predict the effort label of the user as early as possible, in order to enable quick and effective interventions by the ITS. Therefore, we test our classification models when only a fraction of the data is observed during test time. In order to do so, we first train models on the first 1, 5, 10 and 30 seconds, as well as the entire length of the input and test them on corresponding test conditions, for both multiclass and binary classification settings. In Figures 5.9 and 5.10, we plot the F-scores for the various problem outcome labels, as obtained by the models, when predicting problem outcomes after observing 1 second, 5 seconds, 10 seconds, 30 seconds as well as the entire length of the data instance, for the multiclass and binary classification settings respectively. We can observe that model performance, expectedly, increases as features computed from longer temporal sequences are available. Based on this dataset, our baseline models are better at accurately predicting SHINT, NOTR and SOF compared

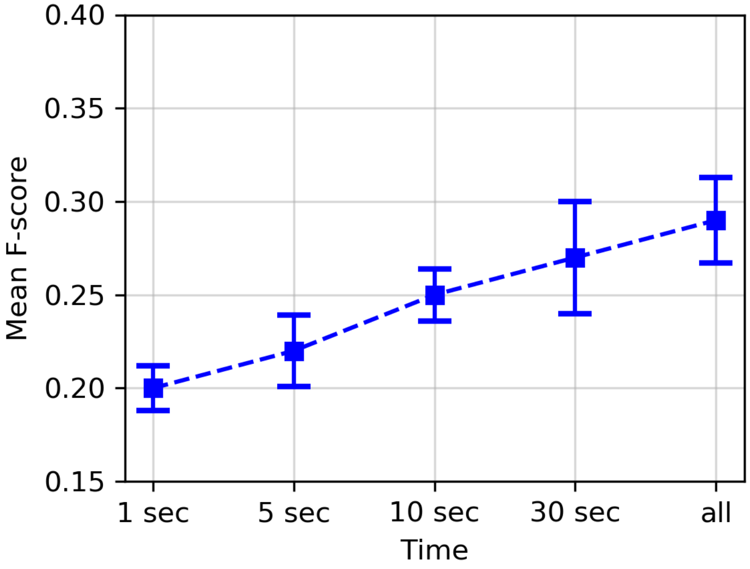


Figure 5.9: Multi-class model performance for effort prediction when data is observed for different time scales.

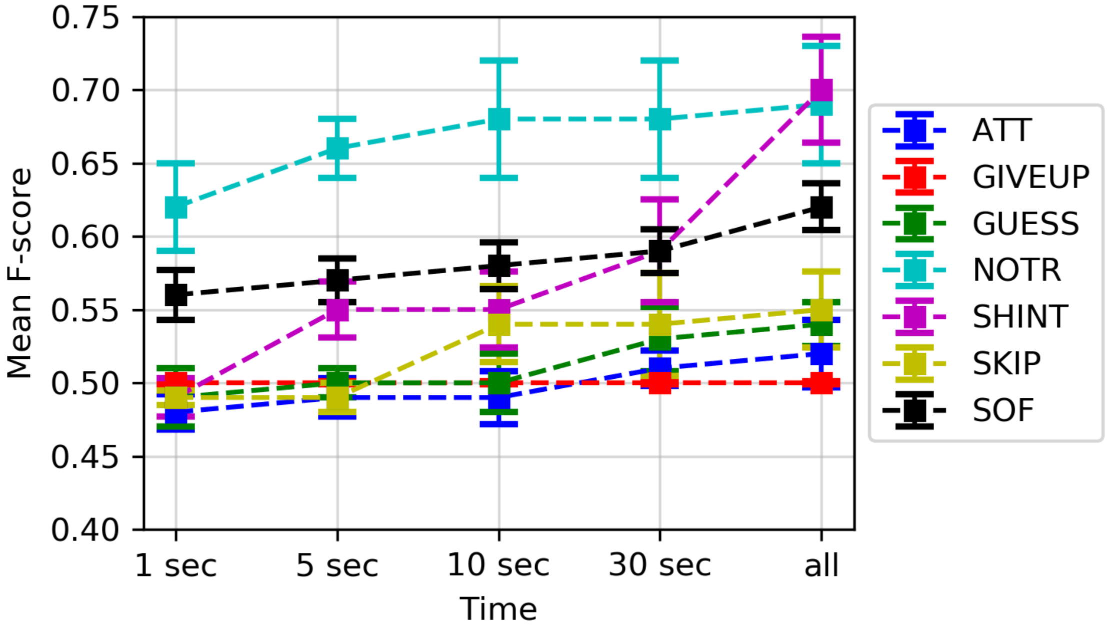


Figure 5.10: Performance of binary classification models for effort prediction when data is observed for different time scales.

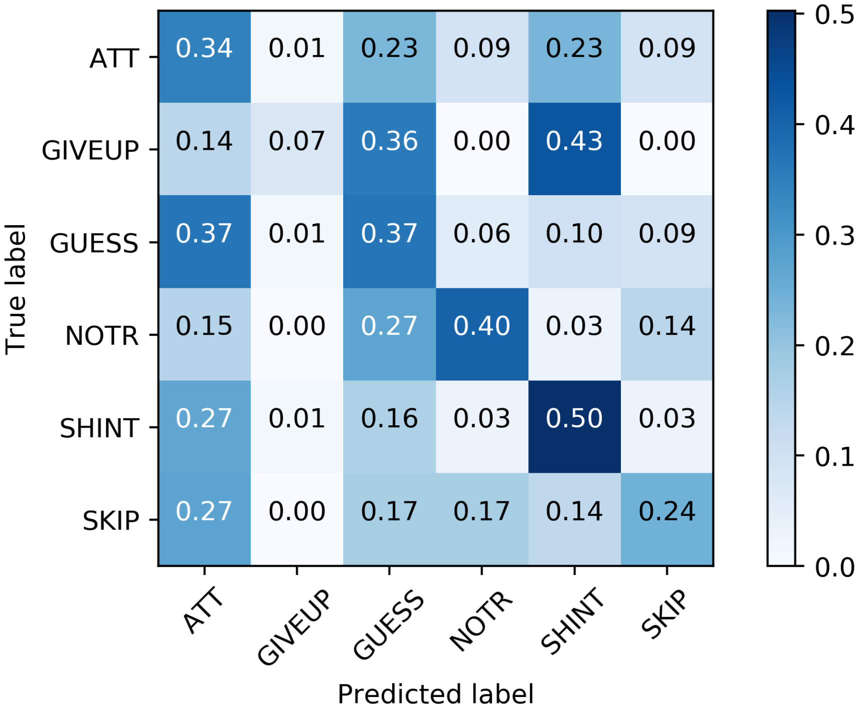


Figure 5.11: Performance of multiclass classifier (Mean accuracy 0.35) for effort prediction of all classes except SOF when the first 30 seconds of data are observed.

to predicting ATT, GIVEUP, GUESS and SKIP.

5.5.3 Interventions

This model can be integrated into the ITS and its predictions can be used to provide more proactive interventions. For example, if the SOF model of an ITS predicts that the user will solve the problem at the first attempt, the ITS can suppress any interventions but subsequently present problems that are more difficult and challenging. If the model predicts that the student will not solve the problem at first attempt, it can provide appropriate interventions based on the confidence of the other effort label models.

In order to investigate which non-SOF classes get confused with one another, we trained a multiclass classification model (using the same random 75/25 splits as used in the other experiments) to predict all problem outcome labels excluding SOF using only the first 30 seconds of the input data. The normalized confusion matrix for this model is plotted in Figure 5.11. We can see that the model is most confident in predicting SHINT, NOTR and GUESS. Possible interventions that the ITS can provide, given a confident prediction for SHINT and GUESS, are appropriate hints associated with the problem at hand.

One label that the model is currently inept at predicting is GIVEUP, which can be attributed to the paucity of examples corresponding to this label in the dataset. Given additional examples associated with the behavior of giving up while solving problems and an improved model capable of accurately forecasting this behavior, the ITS could intervene with combinations of hints and encouraging messages to help the student.

5.6 Summary

In this chapter, we first described the process with which a novel dataset used in this study was collected by our research collaborators and preprocessed to produce data instances corresponding to a single problem. We provided an exploratory analysis of the different problem outcome classes using average facial action unit activations and discussed a few observed trends. We then investigated the problem of trying to directly predict the learning outcome of students attempting to solve individual problems based on signals extracted from their faces. We developed baseline models to predict the problem outcome labels of students solving math problems, in both binary and multiclass classification settings. We also investigated how early problem outcome labels can be forecasted and provided a discussion of possible interventions that the ITS can provide.

Chapter 6

Hierarchical Bayesian Neural Networks

Building robust classifiers trained on data susceptible to group or subject-specific variations is a challenging pattern recognition problem. In this chapter, we develop hierarchical Bayesian neural networks to capture group-specific variations and share statistical strength across groups. Leveraging recent work on learning Bayesian neural networks, we build fast, scalable algorithms for inferring the posterior distribution over all network weights in the hierarchy. We also develop methods for adapting our model to new groups when a small number of group-specific personalization data is available. Finally, we investigate active learning algorithms for interactively labeling personalization data in resource-constrained scenarios. A pictorial example of our hierarchical personalization framework is illustrated in Figures 6.1 and 6.2.

6.1 Model

Given a dataset $\mathcal{D} = \{x_n, y_n\}_{n=1}^{N}$, containing N input $x_n \in \mathbb{R}^D$, and label $y_n \in \mathcal{Y}$ pairs, we aim to learn the functional mapping from inputs to class labels and to make class predictions for previously unseen inputs x_*. Further, we focus on the case where $\mathcal{D}$ is generated by G distinct groups, where groups may represent individuals or pre-defined clusters of data instances.

To preserve group-specific effects we endow each group with its own conditional dis-

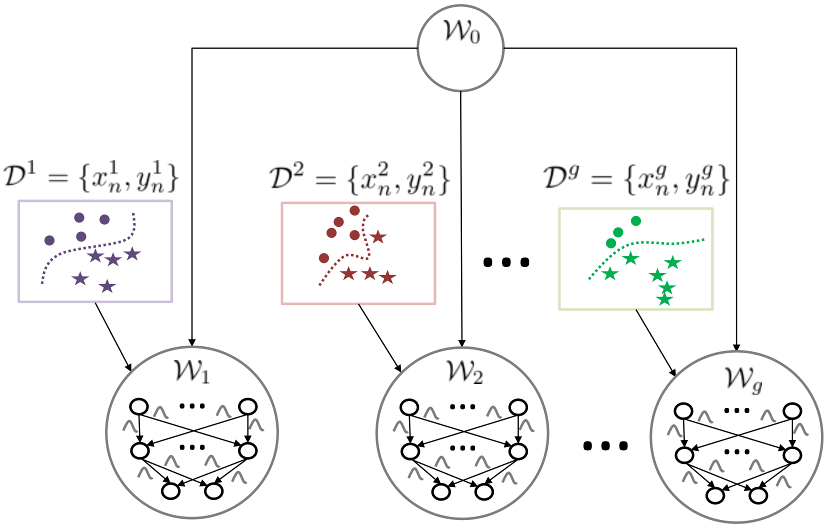

$\mathcal{W}_0$
$\mathcal{D}^1 = \{x_n^1, y_n^1\}$
$\mathcal{D}^2 = \{x_n^2, y_n^2\}$
$\mathcal{D}^g = \{x_n^g, y_n^g\}$
$\mathcal{W}_1$
$\mathcal{W}_2$
$\mathcal{W}_g$

Figure 6.1: Given input examples produced by g groups, we train a classifier using a hierarchical framework, where $\mathcal{W}_g$ is the set of group-specific weights parameterizing a Bayesian neural network. The different shapes correspond to different input classes and the different colors represent the groups who produced those examples.

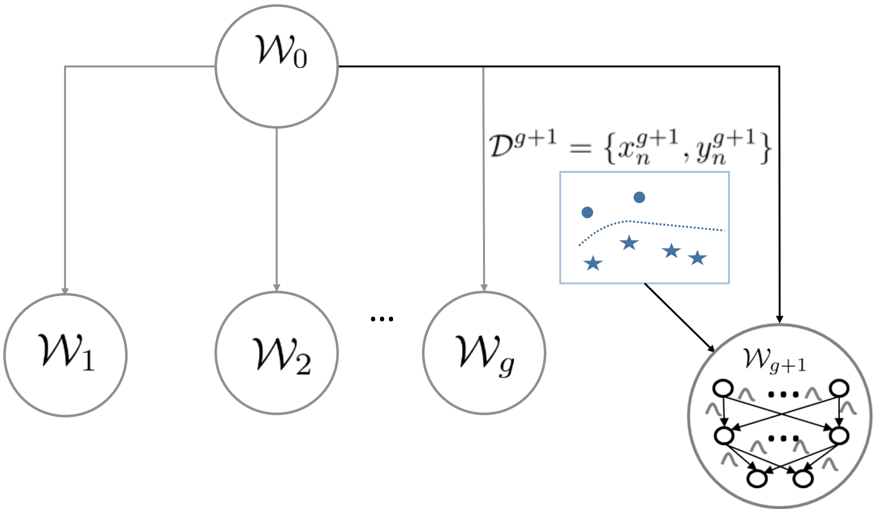


Figure 6.2: Given few instances of training data from a new group, we personalize our model to learn weights specific to the new group.

tribution, allowing the input-label mapping to vary among groups. The conditional distributions are parameterized via multi-layered feedforward neural networks, which enables the model to capture potentially complex mappings between inputs and labels. Assuming the distribution factorizes over data instances, we have,

$$p(\mathbf{y} \mid \mathcal{W}, \mathbf{z}, \mathbf{x}) = \prod_{n=1}^{N} \prod_{g=1}^{G} p(y_n \mid f(\mathcal{W}_g, x_n))^{\mathbf{1}[z_n = g]}. \tag{6.1}$$

Here, z_n is a G-dimensional categorical random variable indicating the group membership of data instance n. We assume that the group indicators $\mathbf{z} = \{z_n\}_{n=1}^{N}$ are observed during training. During testing we are able to reason about the class label y_* of a held-out feature x_* even when the corresponding group membership z_* is unobserved. We wish to learn $\mathcal{W} = \{\mathcal{W}_1, \ldots, \mathcal{W}_{\mathrm{G}}\}$, where $\mathcal{W}_g$ is the set of group-specific weights parameterizing a neural network f whose hidden layers employ rectified linear activations and whose output layer is constrained to be linear. We note here that the function f can be any differentiable function.

We place factorized Gaussian priors on $\mathcal{W}_g$ with independent group-specific variances to model our prior assumption that each group's functional mapping is an independently corrupted version of a common latent mapping (parameterized by $\mathcal{W}_0$),

$$p(\mathcal{W}_g \mid \mathcal{W}_0, \tau_g) = \prod_{l=1}^{L} \prod_{i=1}^{V_{l-1}} \prod_{j=1}^{V_l} \mathcal{N}(w_{ij,l}^{g} \mid w_{ij,l}^{0}, \tau_g^{-1}). \tag{6.2}$$

We further place uninformative priors — zero mean Gaussians with a large fixed variance τ_0^{-1} on the weight means $\mathcal{W}_0$,

$$p(\mathcal{W}_0 \mid \tau_0) = \prod_{l=1}^{L} \prod_{i=1}^{V_{l-1}} \prod_{j=1}^{V_l} \mathcal{N}(w_{ij,l}^{0} \mid 0, \tau_0^{-1}). \tag{6.3}$$

Here, V_l denotes the number of units in layer l and $l = 0$ corresponds to the input layer.

The group specific variances τ_g^{-1} control the amount of deviation from the mean exhibited by the group's input-label mapping. Specifying them manually can be difficult and authors in the past [135] have resorted to setting them via cross-validation. Although cross-validation procedures can be effective for simpler models, they are untenable here. Such a procedure would involve searching over G-dimensional continuous spaces, re-training the model for each parameter candidate. Instead, we place hyper-priors on the variances and infer them jointly with $\mathcal{W}$. The Gamma distribution is the conjugate prior over the precision of a Gaussian distribution and hence a popular choice [10]. However, recent work [46] has shown it to be unsuitable for specifying uninformative priors in hierarchical models. Following [46], we instead use the half-normal distribution with a large fixed variance v to specify uninformative priors over group-specific standard deviations $\tau_g^{-1/2}$,

$$p(\gamma_g \mid v) = \mathcal{N}(\gamma_g \mid 0, v); \quad \tau_g^{-1/2} = |\gamma_g|, \tag{6.4}$$

where we have introduced an auxiliary variable γ_g and used the property, if $a \sim \mathcal{N}(0, \sigma^2)$, then $|a| \sim$ Half-Normal$(0, \sigma^2)$. It also immediately follows that $\tau_g^{-1} = \gamma_g^2$. In the next section, we will see that the auxiliary variable formulation simplifies inference. Finally, we model the observed class labels as categorically distributed random variables,

$$y_n \mid \mathcal{W}, x_n, z_n \sim \text{Cat}(y_n \mid \mathcal{S}(f(\mathcal{W}_{z_n}, x_n))), \tag{6.5}$$

where $\mathcal{S}(a) = \exp\{a\}/\sum_k \exp\{a_k\}$ is the softmax function that maps the real valued output of f to the probability simplex. We can summarize the joint distribution specified

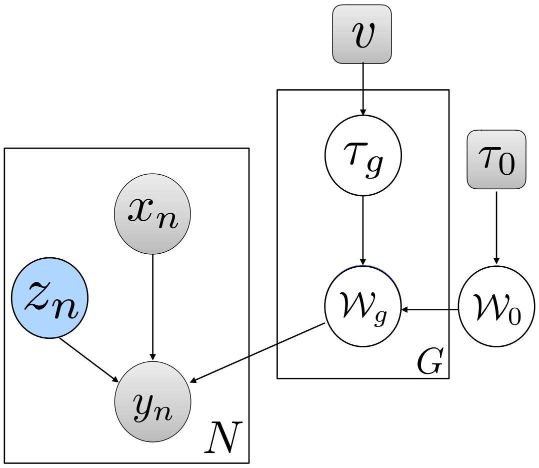


Figure 6.3: Graphical model representation of our hierarchical Bayesian model. Shaded nodes indicate observed random variables.

by the model as,

$$\begin{aligned} p(\mathcal{W}_0, \mathcal{W}, \mathcal{T}, \mathbf{y} \mid \mathbf{x}, \mathbf{z}, \tau_0, v) = {} & p(\mathcal{W}_0 \mid \tau_0^{-1}) \\ & \prod_{g=1}^{G} p(\gamma_g \mid v) p(\mathcal{W}_g \mid \mathcal{W}_0, \tau_g^{-1}) \\ & \prod_{n=1}^{N} \prod_{g=1}^{G} p(y_n \mid f(\mathcal{W}_g, x_n))^{\mathbf{1}[z_n = g]}, \end{aligned} \tag{6.6}$$

where $\mathcal{T} = \{\gamma_1, \ldots, \gamma_G\}$. The hierarchical Bayesian neural network explicitly captures inter-group variances by allowing the group-specific conditional distribution of data from different groups to systematically vary from each other. At the same time, they share statistical strength across groups — samples observed for a particular group not only provide information about that group's distribution but also about other group-specific distributions. A graphical representation of the model is depicted in Figure 6.3.

6.2 Scalable Learning and Inference

Learning our model involves inferring the posterior distribution $p(\mathcal{W}_0, \mathcal{W}, \mathcal{T} \mid \mathcal{D}, \mathbf{z}, \gamma_0, v)$ over model parameters. Unfortunately, the nonlinear activations employed by the networks in the hierarchy render this posterior intractable forcing us to resort to approximate inference techniques. Leveraging recent advances in scalable approximate Bayesian learning, we use variational inference to learn a tractable approximation to the posterior. We restrict the approximating family to the following form,

$$q(\mathcal{W}_0, \mathcal{W}, \mathcal{T} \mid \phi) = q(\mathcal{W}_0|\phi_0) \prod_{g=1}^{G} q(\mathcal{W}_g|\phi_g) q(\gamma_g|\phi_{\gamma_g}), \tag{6.7}$$

where $\phi = \{\phi_0, \phi_1, \ldots, \phi_G, \phi_{\gamma_1}, \ldots, \phi_{\gamma_G}\}$ represents the variational free parameters. We approximate the weight posteriors with fully factorized Gaussian distributions,

$$\begin{aligned} q(\mathcal{W}_0|\phi_0) &= \prod_{l=1}^{L} \prod_{i=1}^{V_{l-1}} \prod_{j=1}^{V_l} \mathcal{N}(w_{ij,l}^0 \mid \mu_{ij,l}^0, \psi_{ij,l}^0), \\ q(\mathcal{W}_g|\phi_g) &= \prod_{l=1}^{L} \prod_{i=1}^{V_{l-1}} \prod_{j=1}^{V_l} \mathcal{N}(w_{ij,l}^g \mid \mu_{ij,l}^g, \psi_{ij,l}^g). \end{aligned} \tag{6.8}$$

The auxiliary variable γ_g affects the model only through its absolute value $|\gamma_g|$. Thus, we can also restrict the posterior of γ_g to $q(\gamma_g|\phi_{\gamma_g}) = \mathcal{N}(\gamma_g \mid \mu_{\gamma_g}, \psi_{\gamma_g})$, a Gaussian family.

We optimize the variational parameters to minimize the Kullback-Leibler divergence KL$(q||p)$ between the true posterior and the variational approximation by maximizing the evidence lower bound (ELBO),

$$\mathcal{L}(\phi) = \mathbb{E}_{q_\phi}[\ln p(\mathcal{W}_0, \mathcal{W}, \mathcal{T}, \mathbf{y} \mid \mathbf{x}, \mathbf{z}, \gamma_0, v)] - \mathbb{E}_{q_\phi}[\ln q(\mathcal{W}_0, \mathcal{W}, \mathcal{T} \mid \phi)], \tag{6.9}$$

with respect to the variational free parameters ϕ.

The non-conjugacy between the neural network parameterized categorical distributions and the Gaussian priors cause the expectations in the ELBO to be intractable. This precludes the availability of traditional fixed point updates. Instead, following recent work [115, 12, 66, 94], we approximate the intractable expectations with unbiased Monte-Carlo estimates,

$$\begin{aligned} \hat{\mathcal{L}}(\phi) = \frac{1}{S} \sum_{s=1}^{S} \ln p(\mathcal{W}_0^s, \mathcal{W}^s, \mathcal{T}^s, \mathbf{y} \mid \mathbf{x}, \mathbf{z}, \gamma_0, v) - \mathbb{E}_{q_\phi}[\ln q(\mathcal{W}_0, \mathcal{W}, \mathcal{T} \mid \phi)], \\ \mathcal{W}_0^s, \mathcal{W}^s, \mathcal{T}^s \sim q(\mathcal{W}_0, \mathcal{W}, \mathcal{T} \mid \phi). \end{aligned} \tag{6.10}$$

The gradient $\nabla_\phi \mathcal{L}(\phi)$ is then approximated with the noisy but unbiased estimate $\nabla_\phi \hat{\mathcal{L}}(\phi)$. Computing $\nabla_\phi \hat{\mathcal{L}}(\phi)$ requires gradients with respect to the means and variances of the Gaussian variational approximations. The non-centered parameterization proposed in [66], $w \sim \mathcal{N}(\mu, \psi) \Leftrightarrow \epsilon \sim \mathcal{N}(0, 1), w = \mu + \psi^{1/2}\epsilon$, allows us to differentiate through the Monte-Carlo approximation,

$$\begin{aligned} \nabla_{\mu,\sigma} \mathbb{E}_{q_w}[g(w)] \Leftrightarrow & \nabla_{\mu,\psi} \mathbb{E}_{\mathcal{N}(\epsilon \mid 0,1)}[g(\mu + \psi^{1/2}\epsilon)] \\ = & \mathbb{E}_{\mathcal{N}(\epsilon \mid 0,1)}[\nabla_{\mu,\psi} g(\mu + \psi^{1/2}\epsilon)] \\ = & \frac{1}{S} \sum_{s} \nabla_{\mu,\psi} g(\mu + \psi^{1/2}\epsilon^s); \epsilon^s \sim \mathcal{N}(0, 1), \end{aligned} \tag{6.11}$$

for any differentiable function g. With the unbiased gradient estimates in hand, Equation 6.9 can be optimized through stochastic gradient ascent [15].

6.3 Local Reparameterization

Although stochastic gradient ascent is guaranteed to asymptotically converge to a local optimum, its non asymptotic performance is contingent on the variance of the unbiased gradient estimates. While the gradient estimate in Equation 6.11 has been previously used to learn Bayesian neural networks [12], we find the variance of this estimator too high to effectively learn our hierarchical model.

To address this issue, we note that the weights in a layer only influence the ELBO ($\mathcal{L}(\phi)$) through the layer's pre-activations. Instead of estimating the ELBO by sampling the variational posterior on the weights one could instead sample the implied variational distribution on the considerably smaller number of pre-activations. This is the "local reparameterization trick" introduced in [65], where the authors show that the corresponding gradient estimates have provably lower variance. For factorized Gaussian variational posteriors over weights, the corresponding pre-activation distributions are also easy-to-compute factorized Gaussians. The pre-activation b_{il}, of the i^{th} node of layer l is distributed as $\mathcal{N}(\mu_{w_{il}}^T a, \sigma_{w_{il}}^{2^T} a^2)$, where a is the input to layer l, $\mu_{w_{il}}$ and $\sigma^2_{w_{il}}$ are the means and variances of the variational posterior over weights associated with node i.

We find that local reparameterization provides significant computational cost savings, accuracy improvements and is crucial for effectively learning hierarchical Bayesian neural networks.

6.4 Predictions

Given a held-out input x_* from an observed group z_*, the posterior predictive distribution over classes is given by,

$$\begin{aligned}
&p(y_* \mid x_*, \mathcal{D}) \\
&= \int p(y_* | \mathcal{W}, z_*, x_*) p(\mathcal{W}_0, \mathcal{W}, \mathcal{T} \mid \mathcal{D}) d\mathcal{W}_0 d\mathcal{W} d\mathcal{T} \\
&\approx \int p(y_* \mid \mathcal{W}, z_*, x_*) q(\mathcal{W}_{z_*} \mid \hat{\phi}_{z_*}) d\mathcal{W}_{z_*},
\end{aligned} \tag{6.12}$$

where the approximation in the second line follows from the variational approximation and $\hat{\phi}_{z_*}$ denotes the optimal variational parameters. In our experiments, we evaluate the integral using a Monte-Carlo estimate.

Next, we consider the case when group (z_*) and class (y_*) memberships are both unobserved and need to be inferred. Classifying x_* involves performing an additional inference of its group membership. Since this inference needs to be performed at test time for each data instance, it is imperative that the inference be fast. To facilitate fast and accurate inference of the group memberships, we use an inference network [97, 48] h_θ, another multi-layered fully connected neural network with weights θ and a G dimensional softmax output layer. We learn this inference network by utilizing all examples from the training set where z is observed. This inference network paramterizes the approximate posterior $q(z \mid x)$. Because z is observed during training, training of the group inference network can occur independently of other variational parameters. At test time, inferring a distribution over the unknown group memberships, $q(z_* \mid x_*, \hat{\theta}) = \text{Cat}(z_* \mid h_{\hat{\theta}}(x_*))$, simply involves a single forward pass through the network, where $\hat{\theta}$ denotes the estimated weights. Our use of an inference network is in sharp contrast to traditional mean field methods where each datapoint is assigned an independent variational parameter that is optimized via several iterations of expensive optimization, at test time. In the presence of a new group, we add an output node to the group inference network. However, we find that only updating the weights associated with the new node is sufficient and the network need not be re-trained.

Marginalizing over the joint posterior predictive distribution, we get the predictive distribution over class labels:

$$\begin{aligned} p(y_* \mid x_*, \mathcal{D}) &= \sum_{z_*=1}^{G} p(y_*, z_* \mid x_*, \mathcal{D}) \\ &= \sum_{z_*=1}^{G} \int p(y_* | \mathcal{W}, z_*, x_*) p(\mathcal{W}_0, \mathcal{W}, z_*, \mathcal{T} \mid \mathcal{D}) d\mathcal{W}_0 d\mathcal{W} d\mathcal{T} \\ &\approx \sum_{z_*=1}^{G} q(z_* \mid x_*, \hat{\theta}) \int p(y_* \mid \mathcal{W}, z_*, x_*) q(\mathcal{W}_{z_*} \mid \hat{\phi}_{z_*}) d\mathcal{W}. \end{aligned} \tag{6.13}$$

The integral over $\mathcal{W}$ is estimated via a Monte-Carlo approximation, $p(y_* \mid x_*) \approx \sum_{z_*=1}^{G} q(z_* \mid x_*, \hat{\theta}) \frac{1}{T} \sum_t p(y_* \mid \mathcal{W}^t, z_*, x_*), \mathcal{W}^t \sim q(\mathcal{W} \mid \hat{\phi}_{z_*}, \hat{\theta})$.

6.5 Personalization

In this section, we focus on incorporating data from a new, previously unseen group and adapting the model to the new group. We call this process personalization and focus on the cases when a small number of data instances from the new group are made available for training. Denoting data instances from new group $G+1$ as $\mathcal{D}_{G+1}$, we learn a group-specific model $\mathcal{W}_{G+1} \mid \mathcal{D}_{G+1}$. The learning can be performed efficiently by observing that $\{\mathcal{W}_g\}_{g=1}^{G+1}$ are conditionally independent given $\mathcal{W}_0$. Thus, given a model trained on $\mathcal{D}$, we only update $\mathcal{W}_{G+1}$ while keeping the estimates $\{\mathcal{W}_g\}_{g=1}^{G} \mid \mathcal{D}$ and $\mathcal{W}_0 \mid \mathcal{D}$ fixed. We could additionally update the posteriors $\{\mathcal{W}_g\}_{g=1}^{G} \mid \mathcal{D} \cup \mathcal{D}_{G+1}$ and $\mathcal{W}_0 \mid \mathcal{D} \cup \mathcal{D}_{G+1}$. However, typically only a small number of adaptation instances $\mathcal{D}_{G+1}$ are available — too few to have a sizeable effect on the posteriors $\{\mathcal{W}_g\}_{g=1}^{G} \mid \mathcal{D}$ and $\mathcal{W}_0 \mid \mathcal{D}$.

6.6 Active Learning

Collecting and labeling personalization inputs can be expensive. For example, consider a system designed to recognize specialized gestures such as those made by naval aircraft handlers onboard aircraft carriers. Not only is the process of collecting additional gestures likely to be challenging, labeling the gestures requires specialized domain knowledge and can be prohibitively expensive. To best utilize limited labeling resources, we next describe an active learning procedure to guide the selection of data instances to label, given a small pool of unlabeled adaptation examples.

Having access to the posterior distribution over weights, rather than just point estimates, allows us to use Bayesian active learning by disagreement (BALD) — a *state-of-the-art* active learning algorithm [58]. Given a pool of unlabeled inputs X_{pool} from group g and a model trained on $\mathcal{D}$, BALD sequentially selects inputs x_l, such that,

$$x_l = \underset{x \in \mathrm{X}_{pool}}{\operatorname{argmax}}\, \mathbb{H}[y \mid x, \mathcal{D}] - \mathbb{E}_{\mathcal{W}_g \sim p(\mathcal{W}_g \mid \mathcal{D})} \mathbb{H}[y \mid x, \mathcal{W}_g], \tag{6.14}$$

where $\mathbb{H}[t] = -\int p(t) \log p(t) dt$. As noted by Houlsby et al. [58], Eq. 6.14 lends itself to an intuitive explanation: BALD seeks a data instance x_l for which the model, averaging over all weights, is uncertain about y (high $\mathbb{H}[y \mid x, \mathcal{D}]$) but individual settings of the weights have high certainty in their predictions (low $\mathbb{E}_{\mathcal{W}_g \sim p(\mathcal{W}_g \mid \mathcal{D})} \mathbb{H}[y \mid x, \mathcal{W}_g]$) — i.e., when the posterior weights disagree the most. Approximation methods to efficiently evaluate Eq. 6.14 are available for certain classes of models, but do not extend to our multi-class classification problem. We therefore resort to a Monte-Carlo approach. We empirically found that, even with a modest number of samples, the approximations significantly improve upon selecting data instances uniformly at random.

6.7 Hierarchical Bayesian Recurrent Neural Networks

The proposed hierarchical Bayesian framework is, in theory, agnostic to the neural architecture used to learn the input-output mapping. For input data of sequential nature, such as gestures, where modeling temporal dynamics can be important, we extend the functionality of this framework by adding support for recurrent neural architectures. For this scenario, f can be defined to represent a recurrent neural network, e.g. a vanilla RNN of the form:

$$h_i = ReLU(w_{hh}^g h_{i-1} + w_{xh}^g x_i), \tag{6.15}$$

$$f = \mathcal{S}(w_{hy}^g h_T), \tag{6.16}$$

where, $\mathcal{W}_g = \{w_{xh}^g, w_{hh}^g, w_{hy}^g\}$ and $w_{xh}^g, w_{hh}^g, w_{hy}^g$ represent the input-to-hidden, hidden-to-hidden and hidden-to-output weight matrices respectively and h_i represents the hidden state at time i. The equations for the recurrent unit f can be replaced with those of more complex recurrent units such as the Long Short Term Memory (LSTM) [47], or the Gated Recurrent Unit (GRU) [22].

6.8 Summary

In this chapter, we developed hierarchical Bayesian neural networks for personalized modeling of face and gesture signals in the presence of inter-group and inter-subject variations.

We proposed a mechanism to utilize the inferred posterior to drive an active learning procedure for personalizing the model to new users. We also developed recurrent variants of our hierarchical Bayesian model as an alternative for building personalized models involving sequential signals such as gestures.

Chapter 7

Applications of Hierarchical Bayesian Neural Networks to Problems in Face and Gesture Analysis

In this chapter, we apply the hierarchical Bayesian model introduced in Chapter 6 to explore whether the problems introduced in Chapters 3 (gesture recognition), 4 (expressivity prediction) and 5 (learning outcome prediction) can benefit from personalization. Focusing first on the problem of gesture recognition where inter-subject variations are commonplace, we demonstrate the effectiveness of our proposed techniques by testing our framework on three widely used gesture recognition datasets.

We then adapt the hierarchical Bayesian neural network framework to enable the learning of facial expressivity model parameters that subtly adapt to pre-defined notions of context, such as the gender of the patient or the valence of the expressed sentiment. We present results based on evaluations of our formulation on a dataset of 772 20-second video clips of Parkinson's disease patients and demonstrate that training a context-specific hierarchical Bayesian framework yields an improvement in model performance in both multi-class classification and regression settings compared to the same model trained on all data pooled together. Finally, we evaluate our hierarchical model on the problem of personalized predictions of student outcomes.

7.1 Subject-specific Gesture Recognition

Here, we extensively evaluate our hierarchical personalization framework on the problem of subject-specific gesture recognition.

7.1.1 Datasets

We used three datasets to test our framework, all of which contain skeletal data of the subjects performing the gestures. The MSRC-12 Kinect Gesture Dataset contains 12 different gestures performed by 30 different subjects for a total of $\sim$4900 gesture instances (Figure 7.1 top left). The gestures were recorded using the Microsoft Kinect.

The 2013 Chalearn Gesture Challenge dataset contains examples of 20 gestures collected from 36 different subjects. Like Yao et al. [135], we experimented with the Training and Validation data containing $\sim$11000 samples. The gestures in the dataset, recorded using the Microsoft Kinect, represent common communication signals used in the Italian language (Figure 7.1 top right).

The NATOPS dataset [105] consists of 24 unique aircraft handling signals performed by 20 different subjects, where each gesture has been performed 20 times by all subjects (Figure 7.1 bottom). A 12-dimensional vector of body features (angular joint velocities for the right and left elbows and wrists), as well as an 8 dimensional vector of hand features (probability values for hand shapes for the left and right hands) collected by Song et al. [105] are provided as features for all frames of all videos in the dataset.

7.1.2 Experiments

For controlled comparisons with previous work [135], we used identical feature representations — raw x,y,z world coordinates for 20 body joints in the MSRC-12 and Chalearn datasets. For NATOPS, we used the 20 dimensional features made available in [105],

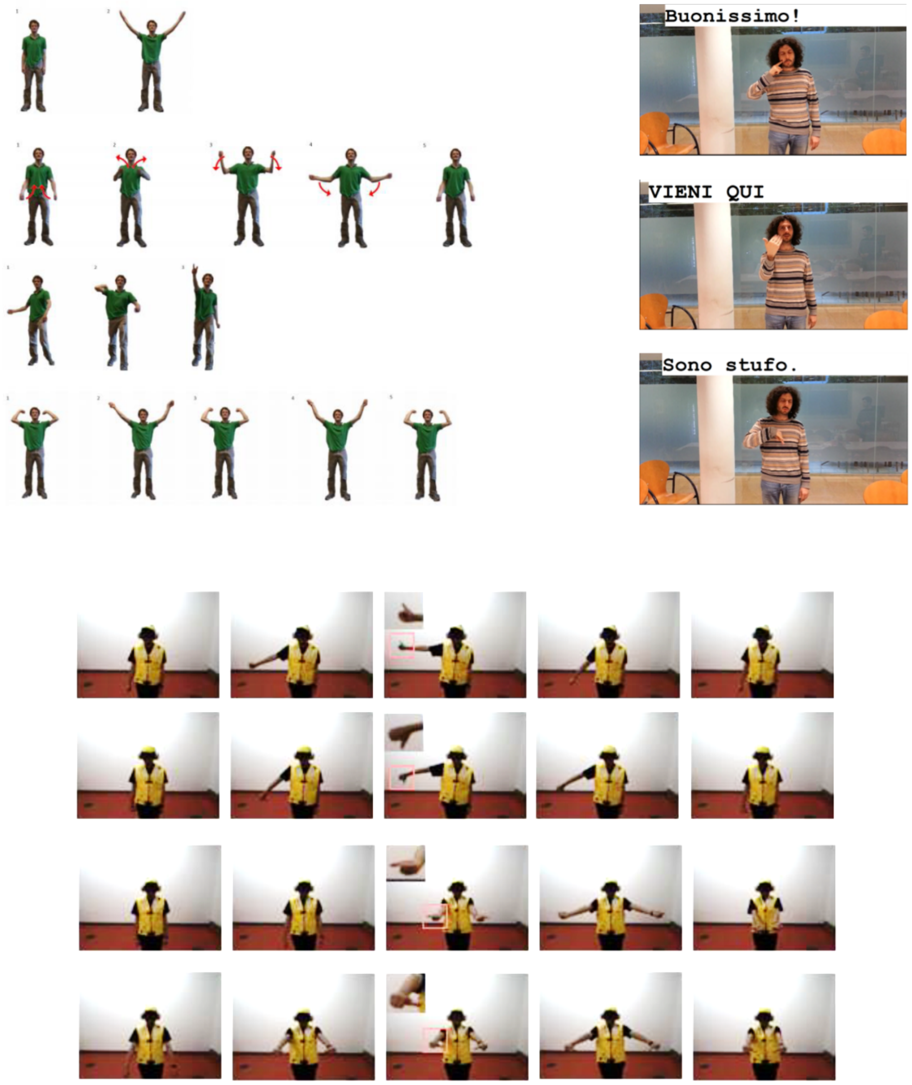


Figure 7.1: Examples of gestures from MSRC-12 dataset (top left), ChaLearn 2013 dataset (top right) and the NATOPS dataset (bottom)

per frame. We extracted frames by sampling uniformly in time and concatenated the per-frame features to produce 600-dimensional input feature vectors for all three datasets. This allowed us to use a common model architecture for the three different datasets. In our experiments, we trained a Hierarchical Bayesian Neural Network with varying number of hidden layers, each with 400 activation nodes. We set the hyper-parameters v to 100 and τ_0^{-1} to 1000 and used RMSprop [114] to optimize the ELBO.

7.1.2.1 Benefits of Local Reparameterization

For all three datasets, on fifteen random 75-25 split of the data, we trained a Hierarchical Bayesian Neural Network for 100 epochs, with and without using local reparameterization. When not using local reparameterization, we approximated the ELBO using 20 Monte Carlo samples whereas when using local reparameterization, we only used 1 sample. We plot the mean logarithm of the ELBO versus the number of training epochs (Figure 7.2) and observe that the ELBO curves for the model that employs local reparameterization is much higher than the model that doesn't, suggesting the model can learn a better approximation of its parameters much faster.

To investigate the effectiveness of the locally re-parameterized ELBO gradients, we trained an HBNN with 1, 2 and 3 hidden layers, each layer with 400 activation nodes, for 100 epochs replicated over 15 random 75/25 splits of the ChaLearn dataset. Fig. 7.3 displays the ELBO evolution over the course of training with and without local reparameterization (lprm). We found that for all three architectures, the models using locally re-parameterized gradients made better progress, achieving higher expected lower bounds with the gap in performance increasing with depth. This is not surprising, considering that the dimensionality of the space spanned by the network weights increases more rapidly than the dimensionality of the pre-activation space.

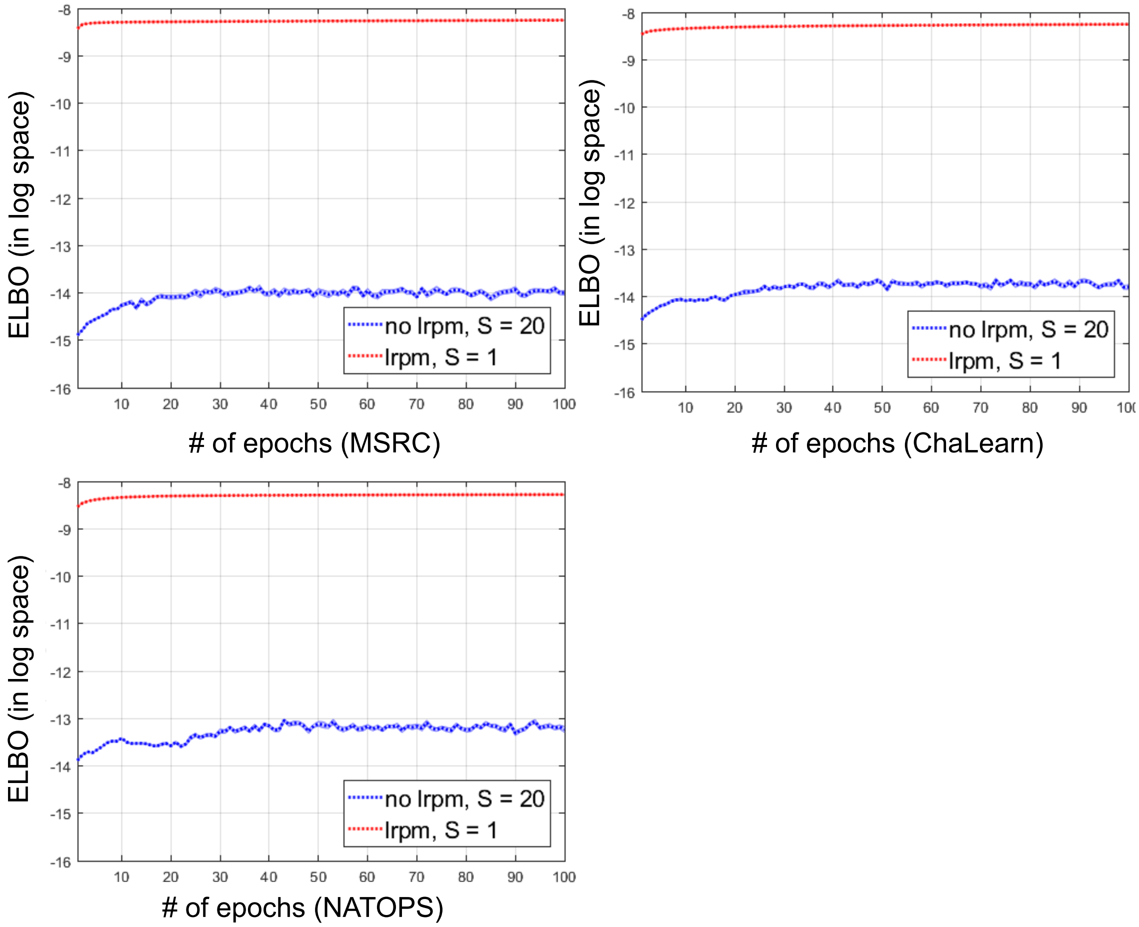


Figure 7.2: The mean logarithm of the expected lower bound (ELBO) versus the number of training epochs, for 15 random 75-25 splits of the data, when the model uses local reparameterization (lprm) and when it doesn't (no lrpm). For all three datasets, MSRC-12 (top left), ChaLearn 2013 (top right) and NATOPS (bottom), the model reaches a faster convergence when using local reparameterization.

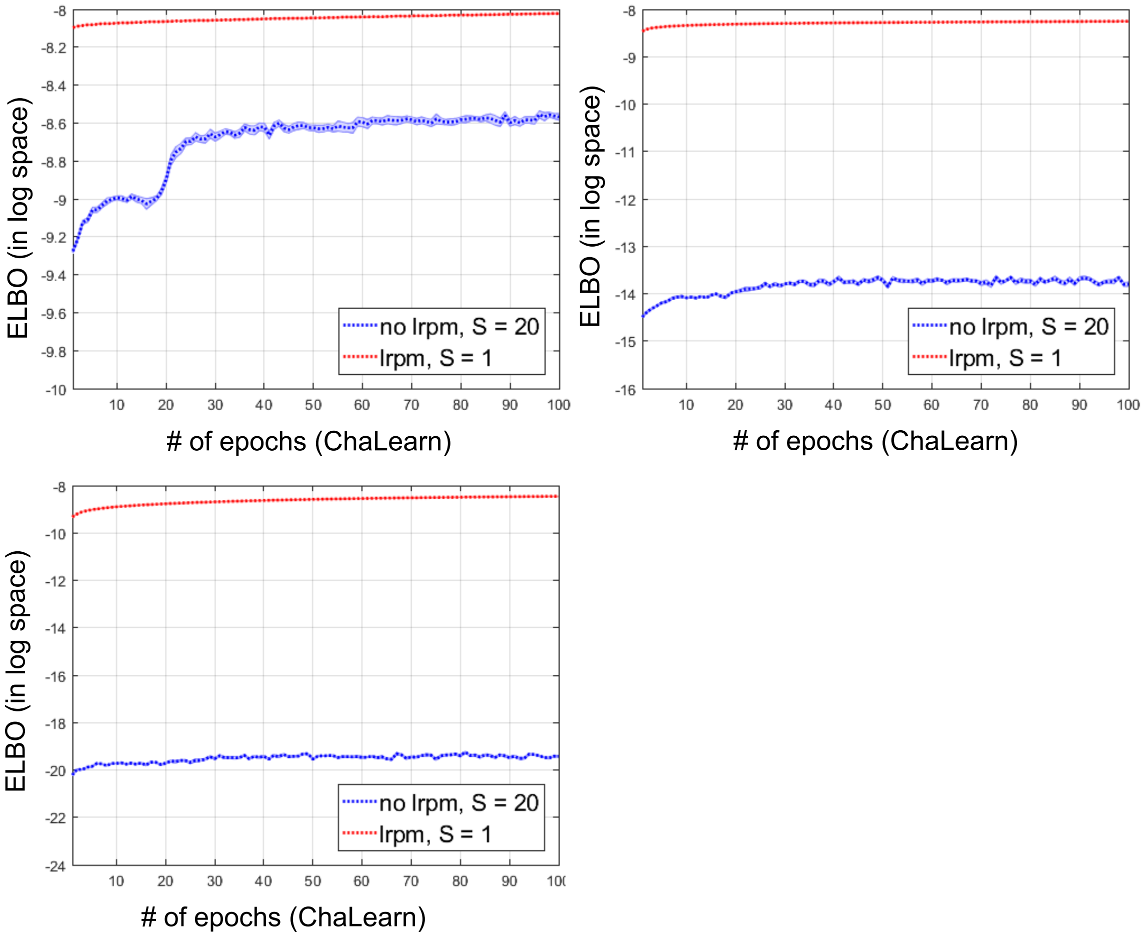


Figure 7.3: The mean logarithm of the expected lower bound (ELBO) versus the number of training epochs, for 15 random 75-25 splits of the ChaLearn dataset, when the model uses local reparameterization (lprm) and when it doesn't (no lrpm) for different HBNN architectures: HBNN with one hidden layer (top left), HBNN with two hidden layers (top right) and HBNN with three hidden layers (bottom).

7.1.2.2 Gesture Recognition

Next, we demonstrate the flexibility afforded by parameterizing the group-specific conditional distributions with Bayesian neural networks. For all datasets, we trained a HBNN with two hidden layers with 400 units each and benchmarked against two strong baselines: a multinomial regression version of our hierarchical Bayesian framework (HBMR), and a two hidden layer non-hierarchical Bayesian neural network that pools data from all subjects into a single group. We trained all models for 50 epochs on 5 random 75/25 replications of the data. Figure 7.4 presents the corresponding results. First, focusing on the case when subject memberships are known (HBNN-Known Z and HBMR-Known Z), we found that the non-linear HBNN models significantly improved upon their linear counterparts HBMR models across the three datasets. HBNNs also outperformed the non-hierarchical Bayesian neural networks on all three datasets clearly demonstrating the benefits of employing subject-specific models over pooled ones. Interestingly, HBMR only outperformed the non-hierarchical Bayesian neural network on the MSRC dataset. This suggests that compared to capturing complex non-linear relationships between gestures and labels, modeling subject-specific idiosyncrasies is less important for the NATOPS and Chalearn datasets. Further comparisons with existing gesture recognition systems are available in the supplement.

Unknown Subject Memberships. We studied the effectiveness of our proposed subject membership inference network. When the membership of a test gesture is unknown we compared two methods for predicting its class label — naive Bayesian model averaging (HBNN-NBMA) where we uniformly averaged the posterior predictive distributions of all subjects and, weighted Bayesian model averaging (HBNN-WBMA), where the weights were determined by the subject membership inference network.

On the MSRC-12 and NATOPS datasets, we found that HBNN-WBMA significantly

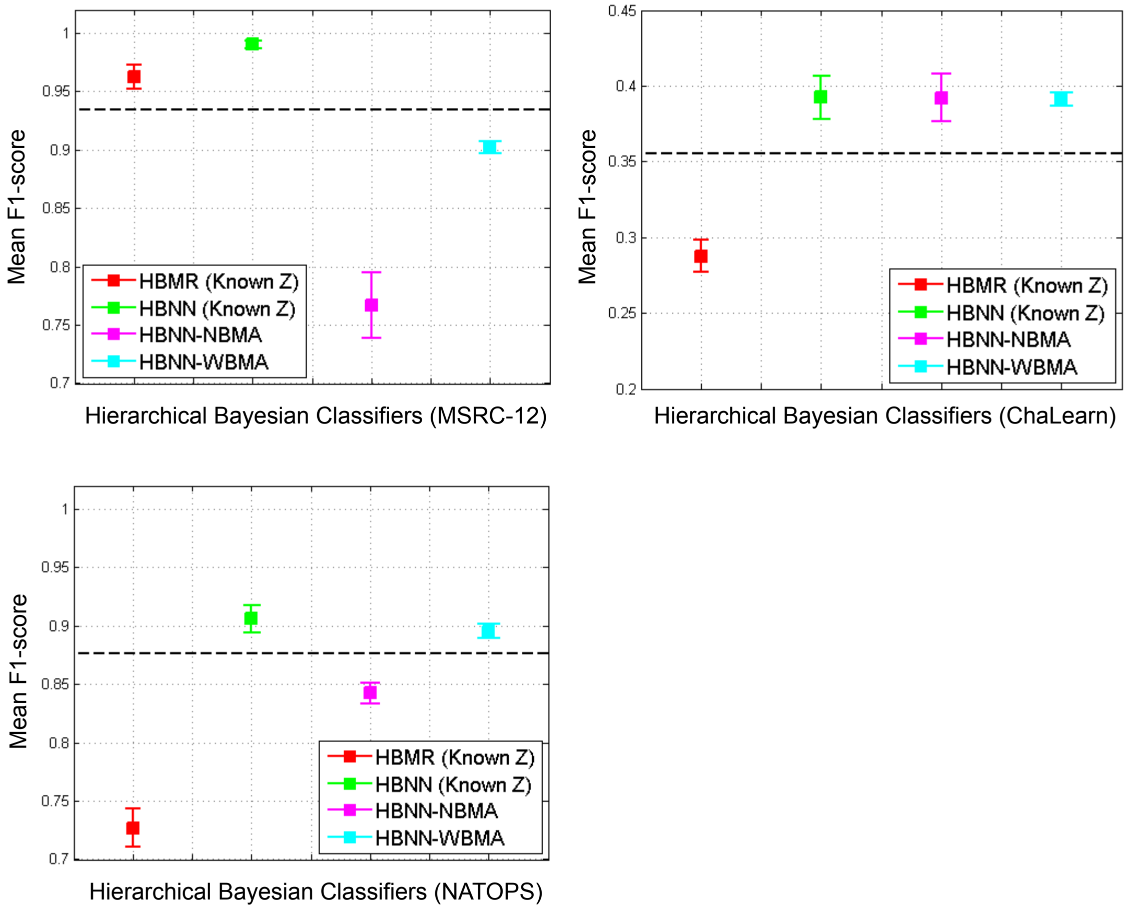


Figure 7.4: The mean F1-scores for different versions of our Hierarchical Bayesian gesture classifier. For all three datasets (MSRC-12 dataset (top left) and Chalearn 2013 dataset (top right) and NATOPS dataset (bottom)), we trained a Hierarchical Bayesian Multinomial Regression classifier (HBMR) and a Hierarchical Bayesian Neural Network (HBNN) and used them to predict the class labels of the test data. For HBNN, when group membership of the test data is known, we used the weights belonging to the corresponding group to make a prediction (HBNN (Known Z)). When group membership of the test data is unknown, we present results obtained with Naive Bayesian Model Averaging (HBNN-NBMA) and Weighted Bayesian Model Averaging (HBNN-WBMA). We compared our results with a baseline BNN trained with data from all subjects pooled into one group, whose mean is depicted in the figures as a dashed black line.

outperformed HBNN-NBMA. On ChaLearn, both methods performed similarly but HBNN-WBMA exhibited lower variance across splits. Together, these results demonstrate that the use of a recognition network is helpful when subject-memberships are not known at test time.

We note that *apriori* knowledge of the subject-membership of a gesture leads to better predictive performance on all but the ChaLearn dataset. The ChaLearn dataset is more challenging due to less rigidly defined gestures. This results in more variability in gestures and weakens our assumption that each subject performs a given gesture consistently and differently from other individuals. This may explain why knowing the subject memberships does not translate into significant performance improvements.

7.1.2.3 Personalization

We now present experiments demonstrating the personalization ability of HBNN models. Given a limited number of training instances from the new subject, we learned model parameters tuned to the subject. For all datasets, we used a leave-one-subject-out cross validation scheme, where we personalized models pre-trained on $G - 1$ subjects and used a pool of seven (fifteen for NATOPS) randomly selected gestures per class from the test subject for personalization. Both pre-trained and personalized models contained two layers, with 400 units each, and were trained for 50 epochs. We considered two schemes for incorporating gestures from the personalization pool: RAND, where data from the training pool of the test subject was added uniformly at random, and BALD where data from the training pool was selected using uncertainty-based sampling (Eq. 6.14). For each test subject, we repeated the experiment five times, randomly selecting the pool of personalization gestures in each replicate.

We benchmarked these methods against a strong non-personalized baseline — a non-

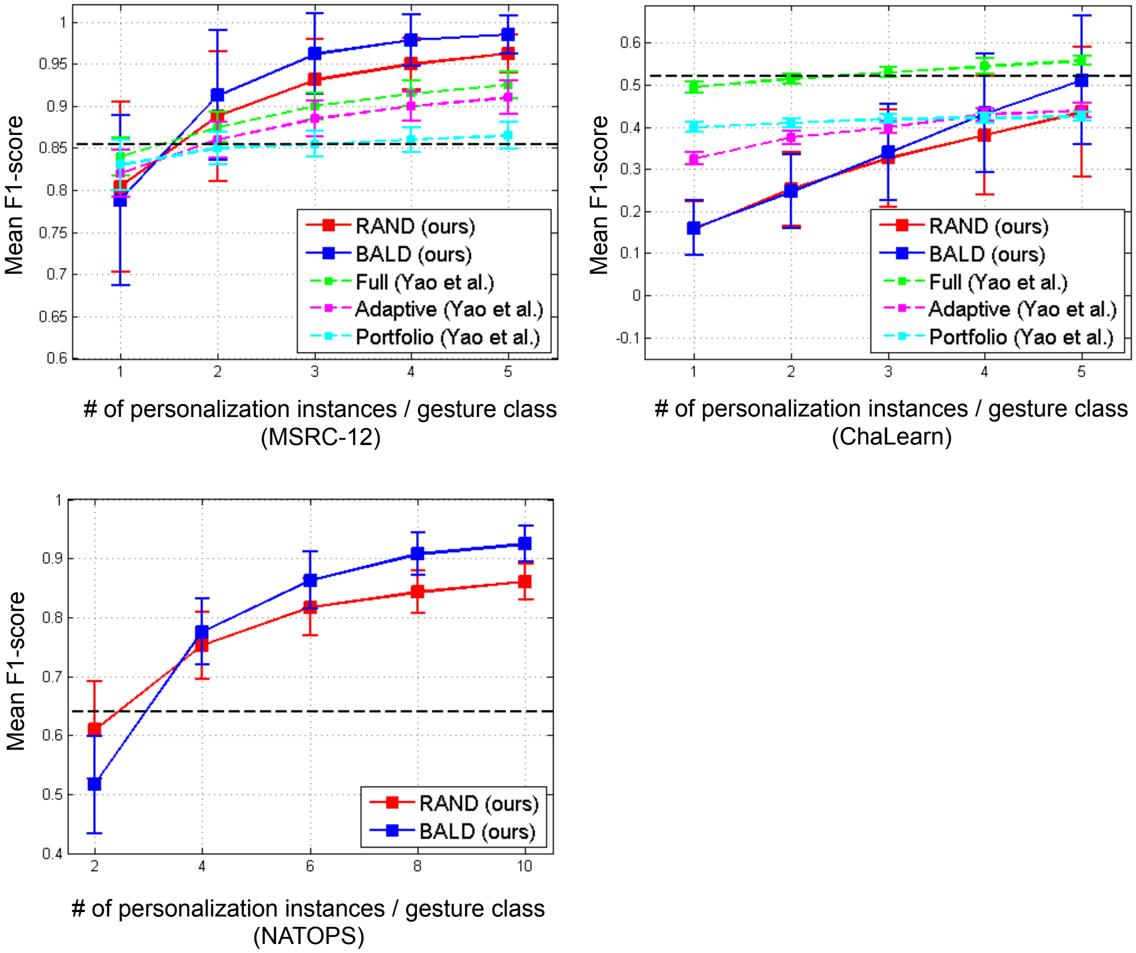


Figure 7.5: The mean F1-scores for different personalization schemes plotted against the number of personalization instances per gesture. We observe that personalization using BALD outperforms personalization using RAND when the number of personalization instances is greater than 1 for the MSRC-12 dataset (top left), 3 for the ChaLearn 2013 dataset (top right) and 4 for the NATOPS dataset (bottom). Our results also compare favorably with the personalization methods presented by Yao et al. [135], who reported their results for the MSRC-12 and ChaLearn 2013 datasets. We compare the personalization results with a baseline BNN trained with all training data pooled into one group, whose mean is depicted in the figures as a dashed black line.

hierarchical BNN (with two 400-unit hidden layers) trained with data from all subjects except the test (personalization) subject pooled together. The results in Fig. 7.5 show that with as few as two and three gesture examples per subject, HBNN outperformed the baseline on MSRC and NATOPS. On ChaLearn, BALD with five gesture examples per class performed as well as the non-personalized baseline.

It may be surprising to note that personalization baseline on ChaLearn (Figure 7.5) resulted in higher F1 scores than the non-personalized baseline presented in Figure 7.4. However, the baseline in Figure 7.4 corresponds to a model trained on samples from all subjects but with the training set size limited to 75% while the model in Figure 7.5 was trained on 35 out of 36 subjects corresponding to 97% of the dataset. For the ChaLearn data intra-subject variability in gestures dwarfs inter-subject variations. Thus, observing more of the dataset as opposed to gestures from the same subject leads to better performance. This is also why HBNNs need more (4) personalization examples for ChaLearn than the other datasets.

Comparing BALD with RAND, we found that BALD improves personalization performance on all three datasets, when the number of training instances exceeded one, three and four for MSRC, NATOPS and ChaLearn datasets. This is an interesting result which suggests that even our naive mean field approximation provides predictive uncertainty estimates of sufficient fidelity that lead to BALD's uncertainty based sampling outperforming RAND's uniform at random sampling. Moreover, our experiments suggest that when labeling resources are limited, BALD based active learning is an attractive option for building personalized classification systems. We do note that BALD and RAND perform similarly when very few personalization instances are available. This may be due to the uncertainty estimates being poor in the very few personalization instances regime.

We compared our approach to the existing *state-of-the-art* in gesture personalization

[135] on MSRC and ChaLearn datasets (Figure 7.5). Yao et al. [135] presented three personalization methods: *full personalization*, which refers to fully re-training random forest classifiers given personalization data, *adaptive personalization*, which refers to adapting the parameters of pre-trained random forests given personalization data, and a *portfolio* approach, where a library of random forest classifiers are pre-trained and the best performing portfolio member is used to classify data from a new subject. We observe that on MSRC, both RAND and BALD outperformed all of the competing methods when the number of personalization instances per gesture class is greater than two. On ChaLearn, BALD outperformed portfolio and adaptive schemes and is within noise of full personalization after observing five personalization instances.

7.1.2.4 Effects of Modifying Depth of Model

Given that the process of personalizing to a new subject usually entails the availability of very few number of training instances, we study the effects of modifying depth on model performance. For the Chalearn dataset, we used a leave-one-subject-out cross validation scheme, where we personalized models pre-trained on $G-1$ subjects with a pool of 7 randomly selected gestures per class from each test subject using HBNNs with 1 and 3 hidden layers. We plot the mean F1-scores for different personalization schemes against the number of personalization instances per gesture for the different HBNN architectures (Figure 7.6). We observe that models personalized using BALD outperforms models personalized using RAND for all architectures. The HBNN model with 1 hidden layer performs comparably to the best-performing HBNN with 2 hidden layers (Figure 7.5). However, the HBNN model with 3 hidden layers performs worse due to overfitting.

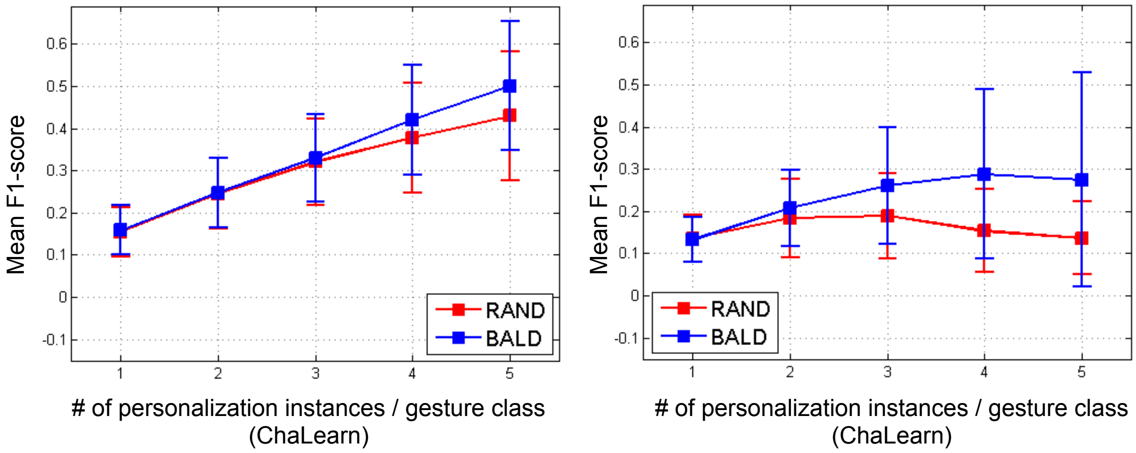


Figure 7.6: The mean F1-scores for different personalization schemes plotted against number of personalization instances per gesture for different HBNN architectures for the ChaLearn dataset: HBNN with one hidden layer (left), and HBNN with three hidden layers (right).

7.1.2.5 Hierarchical Bayesian Recurrent Neural Networks

Instead of concatenating features from all timesteps and feeding it into a fully connected architecture, we investigated the benefits of explicitly modeling temporal dynamics of the input gestures by training hierarchical Bayesian recurrent neural networks (HBRNNs). For each gesture dataset, we represented each input video with a 10 x 60-d feature representation and fed it into an HBRNN framework, with the RNN architecture corresponding to a GRU with a 100-dimensional hidden representation. For all three datasets, the HBRNN model performed comparably (MSRC-12 and NATOPS) or obtained a boost in performance (ChaLearn) (Figure 7.7) compared to the best performing fully connected HBNN model from earlier experiments (Figure 7.4). This demonstrates the benefits of using HBRNNs in modeling temporal signals such as gestures. In addition, we note that the HBRNN framework achieves this performance at a fraction of the parameter cost.

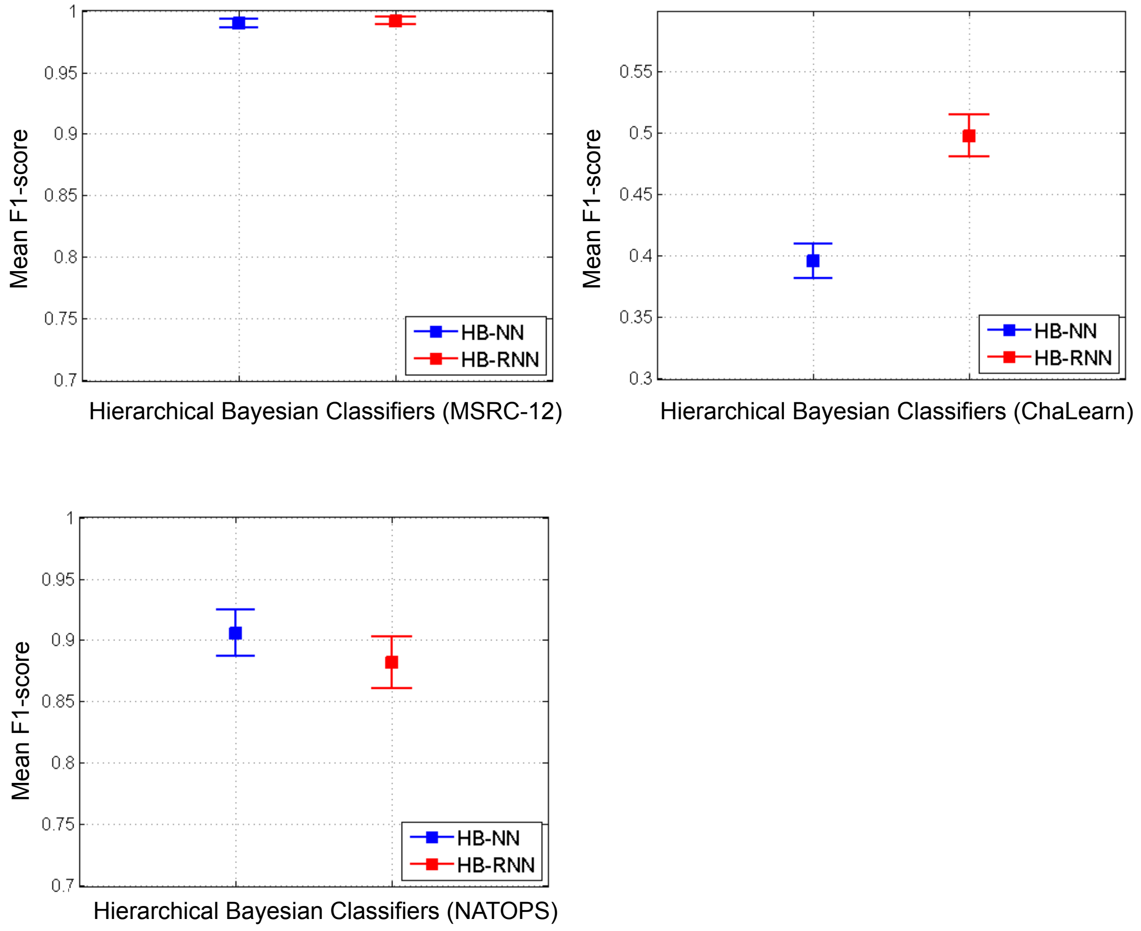


Figure 7.7: The mean F1-scores comparing feed-forward Hierarchical Bayesian Neural Networks (HBNN) with Hierarchical Bayesian Recurrent Neural Networks (HBRNN), for all three datasets (MSRC-12 dataset (top left) and Chalearn 2013 dataset (top right) and NATOPS dataset (bottom)).

7.2 Context-sensitive Facial Expressivity Prediction

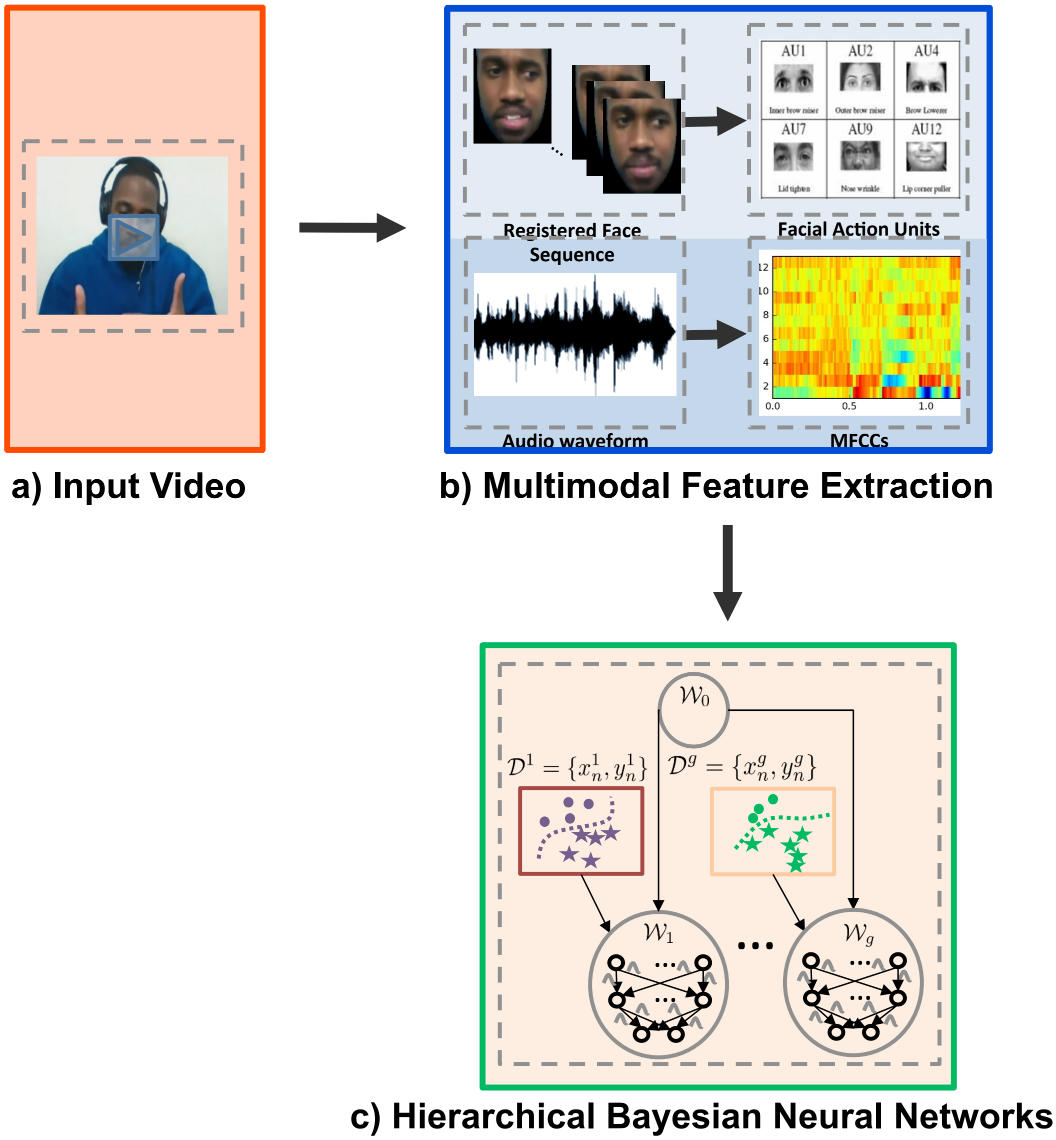
Now, we switch the focus of our hierarchical personalization framework from individual-specific gesture recognition to group-specific facial expressivity prediction of people with Parkinson's Disease (PD), where the groups are determined by factors such as gender or the valence of the sentiment being expressed by the PD patient.

7.2.1 Context

We experiment with two notions of context: gender (male and female) and sentiment (positive and negative) expressed in the interview. We wish to investigate whether dividing the dataset into context-sensitive groups and leveraging any variations inherent in the groups' input-label mapping can yield improvements in model performance. For example, previous research has indicated people display varying levels of expressive behavior when discussing positive experiences compared to when speaking about negative experiences [109]. Utilizing a framework that is capable of learning related but slightly different functions seems apt for such a scenario.

We assume we have access to context indicators, i.e. the subject's gender and the sentiment of the experience that the subject describes, for each video in both the training and test sets. This allows us to separate the dataset into context-specific groups (Figure 7.8).

The hierarchical Bayesian neural network learns the context-sensitive group-specific variances by allowing the group-specific conditional distribution of data from different groups to vary from each other, while allowing the sharing of statistical strength across groups.



a) Input Video

b) Multimodal Feature Extraction

c) Hierarchical Bayesian Neural Networks

Figure 7.8: Overview of our multimodal context-sensitive expressivity prediction model. From an input audio-video clip (a), we extract Facial Action Unit-based interpretable features as well as Mel-Frequency Cepstral Coefficient Features (b). We train a context-sensitive expressivity model by utilizing a hierarchical Bayesian neural network framework (c). Here $\mathcal{D}^1, ..., \mathcal{D}^g$ represents our dataset $\mathcal{D}$ divided into g context-sensitive groups, which we hypothesize to have subtly different input-label mappings. $\mathcal{W}_1, ..., \mathcal{W}_g$ represent the group-specific weights that parameterize the mapping between the input and the expressivity ratings.

7.2.2 Dataset

We used the same dataset of 772 audio-video clips of patient interviews as described in Chapter 4. The ground truth expressivity labels $y_i \in \mathbb{R}$ for each video was taken as the average of 4 expert ratings. In our experiments, we evaluated both regression and classification formulations in predicting the expert ratings. For classification experiments, we discretized the labels of the entire dataset into 4 classes. Classes 1, 2, 3 and 4 contain samples with facial expressivity ratings in the range [1, 2), [2, 3), [3, 4) and [4, 5] respectively.

7.2.3 Experiments

Here, we report results on experiments conducted to investigate the benefits of context-sensitive modeling for facial expressivity prediction.

7.2.3.1 Context-sensitive Modeling

For each context indicator (gender and sentiment), we first divided the training data into two groups (male and female for gender, positive and negative for sentiment). We trained our framework using this multi-group paradigm with the combined audio-video feature representation. During testing, we obtained the estimate of the expressivity rating of the test sample using the classification or regression parameters associated with its corresponding context indicator.

Compared to a baseline model (HBNN-C-pooled), which ignored context and was trained with the data from all groups pooled together (obtaining a mean F1-score of 0.50), we found that retaining contextual information provided by gender (HBNN-C-gender) yielded no empirical benefit in classifier performance (mean F1-score of 0.50). However, utilizing the context provided by sentiment (HBNN-C-sentiment) improved the performance of the model in the multiclass classification settings (mean F1-score of 0.55)

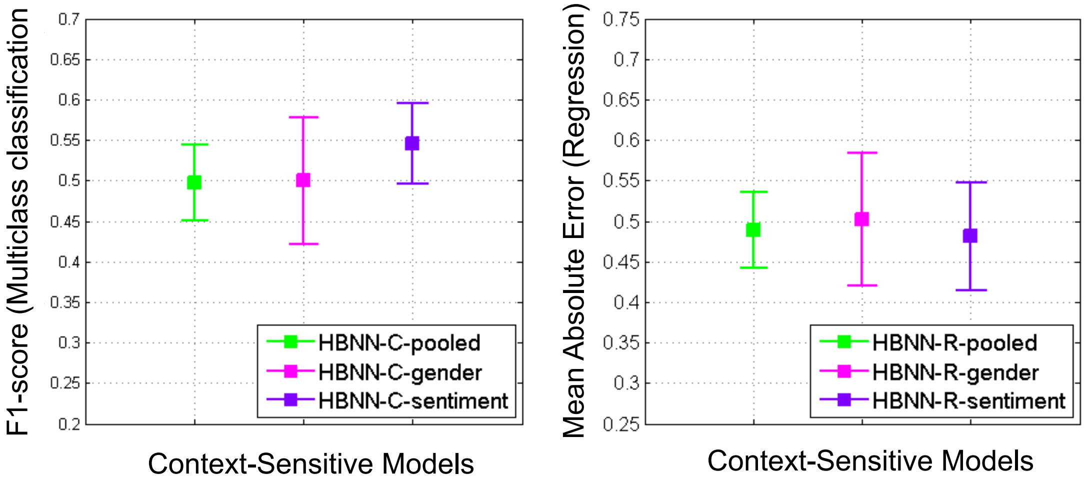


Figure 7.9: (Left) The mean F1-scores and their standard deviations for HBNN-C models trained using context (gender, sentiment) or no context (pooled). (Right) The mean MAE scores and their standard deviations for HBNN-R models trained using context (gender, sentiment) or no context (pooled).

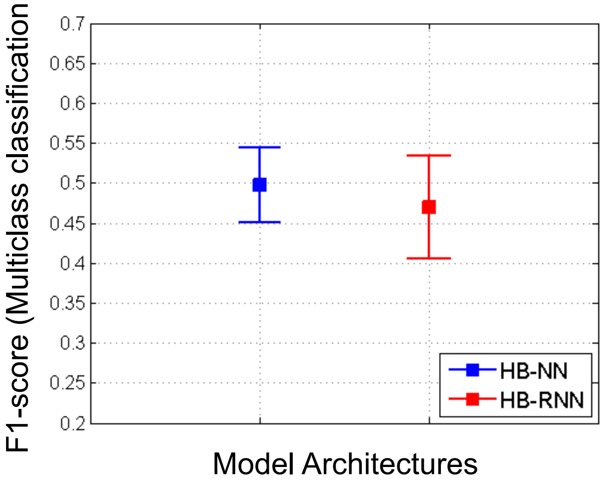


Figure 7.10: The mean F1-scores for a model trained with feed-forward neural architecture (HB-NN) and a recurrent neural architecture (HB-RNN) for multi-class expressivity classification.

(Figure 7.9). Similarly, the regression model that utilized context provided by sentiment (HBNN-R-sentiment) yielded a slightly improved MAE score of 0.48, outperforming the baseline model which obtained a mean MAE score of 0.49 (Figure 7.9). This suggests that the input-label mappings in the sentiment-driven context-sensitive groups may contain sufficient group-specific variance in order for the hierarchical framework to leverage it into improved model performance.

7.2.3.2 Modeling Temporal Dynamics

Although facial expressivity is coded as a “gestalt” score, we wished to explore whether preserving the temporal order and dynamics in the feature representations leads to any classification benefits. To answer this question, we represented each input video with a 10 x 300-d feature representation and fed it into a Hierarchical Bayesian Recurrent Neural Network (HBRNN) framework. However, we found no empirical benefit of explicitly modeling temporal dynamics as the mean F1-scores achieved by HBRNNs do not exceed the ones obtained using their non-recurrent counterparts (Figure 7.10).

We posit that because raters label expressivity as a summary score of the entire video, modeling the temporal order of events do not matter as much in this scenario. The challenges of training recurrent models on relatively small datasets, such as this, may be an additional reason why the HBRNN model fails to outperform the HBNN model in the task of facial expressivity prediction.

7.3 Student-specific learning outcome prediction

Finally we apply the hierarchical personalization framework to the problem of student-specific learning outcome prediction. Students may vary significantly in how they display their emotional states during learning, while engaged with and reacting to the ITS. We,

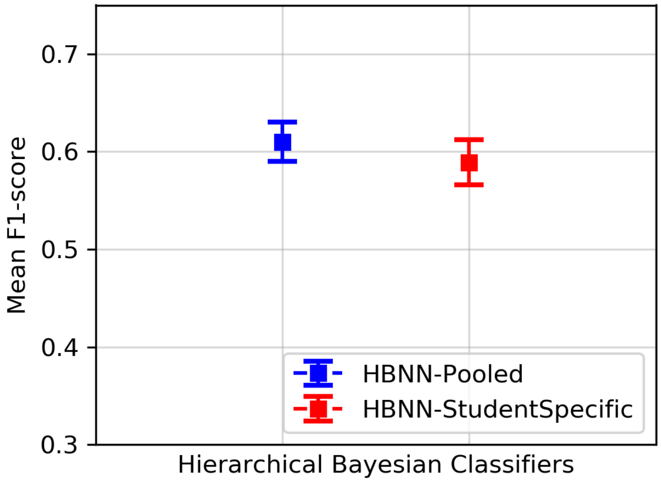


Figure 7.11: The mean F1-scores and their standard deviations for an HBNN model trained with all data pooled together and a student-specific HBNN model.

therefore, wish to explore whether there are any benefits of learning slightly different mappings to the learning outcome label for each individual student, as opposed to learning a generic classification function for all students pooled together.

7.3.1 Dataset

We used the same dataset of 1596 problem outcome video-clips of students engaged with MathSpring. Each input video is represented by the same action unit-based summary statistic feature descriptor, as described in Chapter 5.

7.3.2 Experiments

Here, we report results on experiments conducted to investigate the benefits of student-specific modeling for learning outcome prediction. We trained and tested all our models on 5 random, stratified 75/25 splits of the data. All HBNN models had 1 hidden layer with a 100 activation nodes and were trained for the SOF-vs-all binary classification task, as this was the only binary classification task where all students had examples for both classes.

Compared to a baseline model (HBNN-pooled), which was trained with the data from all 30 students pooled together (obtaining a mean F1-score of 0.61), we found that training a student-specific model did not bring any improvement in classifier performance (mean F1-score of 0.59).

The lack of improvement, we posit, can be attributed to a few reasons. First, some individuals do not possess a sufficient number of training examples. Moreover, for many students, their examples are highly imbalanced across the 2 classes (e.g. some students have solved almost all problems on first attempt). Second, the underlying assumption in our personalization framework that between-group variance is high and within-group variance is low, is not as strong as in the gesture recognition problem.

7.3.3 Summary

In this chapter, we first extensively evaluated the hierarchical Bayesian neural network model, introduced in Chapter 6, to the problem of personalized gesture recognition system. We illustrated the benefits of the hierarchical model over baselines that ignore subject-specific gesture variations and demonstrated the scalability of the model's capacity to learn complex feature-label mappings. We used the inferred posterior distributions over weights to guide active learning procedures for personalizing pre-trained models to new users. Our posterior driven active learning algorithm consistently outperformed selecting gestures at random as well as outperforming or being competitive with existing methods. We then extended the framework to support recurrent architectures, demonstrating their benefits in modeling gestures.

Second, we illustrated the benefits of using a framework that adapts to contextual information. Our hierarchical Bayesian model trained on a dataset divided according to the sentiment expressed in the interviews outperformed a baseline model that ignored this contextual information in both classification and regression scenarios. We then used the HBRNN framework to model temporal dynamics in expressivity prediction but found no empirical benefits compared to results obtained using fully-connected feed-forward architectures trained on aggregated features.

Third, we investigated whether there are any benefits of learning slightly different mappings from the raw video input to the learning outcome label for each individual student, as opposed to learning a generic classification function for all students pooled together but found no empirical benefits

Chapter 8

Conclusions and Future Work

In this thesis, we focused on the following challenges within face and gesture analysis: a) the classification of hand and body gestures along with the temporal localization of their occurrence in a continuous stream, b) the recognition of facial expressivity levels in people with Parkinson's Disease using multimodal feature representations, c) the prediction of student learning outcomes in intelligent tutoring systems using affect signals, and d) the personalization of models that can adapt to subject and group-specific nuances in facial and gestural behavior.

8.1 Contributions

Here, we summarize the major contributions of this thesis:

- We presented an analysis of methods for gesture spotting and classification by comparing two methods. The first method trains a single random forest model to recognize gestures from a given vocabulary, as presented in a training dataset of video plus 3D body joint locations, as well as out-of-vocabulary (non-gesture) instances. The second method employs a cascaded approach, training a binary random forest model to distinguish gestures from background and a multi-class random forest model to classify segmented gestures. Given a test input video stream, both frameworks are applied using sliding windows at multiple temporal scales. We evaluated our formulation in segmenting and recognizing gestures on two different benchmark datasets: the NATOPS dataset of 9600 gesture instances from a vocabulary of 24 aircraft handling signals, and the ChaLearn dataset of 7754 gesture instances from a vocabulary of 20 Italian communication gestures. The performance of our method compares favorably with state-of-the-art methods that employ Hidden Markov Models or Hidden Conditional Random Fields on the NATOPS dataset.

- We investigated how to computationally predict an accurate and objective score for facial expressivity in people with Parkinson's Disease. We first presented a baseline method that trains a random forest regressor based on geometric shape features of the face. We provided insight on the geometric features that are important in this prediction task by computing variable importance scores for our features. We then build improved models on more informative facial action unit-based features, providing interpretations based on their aggregated feature importance. We demonstrated the utility of extracting features from not only the visual domain but also the audio in order to accurately predict facial expressivity, finding that a model trained on a combined audio-visual feature representation outperformed models trained on features extracted from a single modality. We evaluated our formulation on a dataset of 772 20-second interview video clips of PD patients using 9-fold cross validation.

- We described the process with which a novel multimodal dataset used in this study was collected and annotated, with the aim of fulfilling an existing gap in affective tutoring systems literature: a benchmark, publicly available facial affect dataset in an educational setting. We provided an exploratory analysis of the different problem outcome classes using average facial action unit activations, discussing some interesting observed trends. Based on this novel dataset, we then developed baseline models to predict the problem outcome labels of students solving math problems, demonstrating its effectiveness in accurately forecasting several problem outcome labels.

- We developed hierarchical Bayesian neural networks for personalized modeling of face and gesture signals in the presence of inter-group and inter-subject variations. Leveraging recent work on learning Bayesian neural networks, we built variational inference-based fast, scalable algorithms for inferring the posterior distribution over all network weights in the hierarchy. We also developed methods for adapting our model to new groups when a small number of group-specific personalization data is available. We proposed to utilize active learning algorithms for interactively labeling personalization data in resource-constrained scenarios. We also implemented recurrent variants of our hierarchical Bayesian model, given their suitability in building models involving sequential signals.

- We applied our hierarchical Bayesian framework to three tasks: subject-specific gesture recognition, context-specific facial expressivity prediction and student-specific learning outcome prediction.

  First, we illustrated the benefits of the hierarchical model over baselines that ignore subject-specific gesture variations and demonstrated the scalability of the model?s capacity to learn complex feature-label mappings, testing our framework on three widely used gesture recognition datasets. We used the inferred posterior distributions over weights to guide active learning procedures for personalizing pre-trained models to new users, showing that our posterior driven active learning algorithm consistently outperformed selecting gestures at random. We demonstrated the suitability of applying hierarchical Bayesian recurrent neural networks in the gesture recognition task, achieving comparable or improved model performance at a fraction of the parameter cost.

Second, we illustrated the benefits of using a framework that adapts to contextual information, regarding the task of facial expressivity prediction. Our hierarchical Bayesian model trained on a dataset divided according to the sentiment expressed in the interviews outperformed baseline models that ignored this contextual information in both classification and regression scenarios.

Third, we applied our personalization framework to the problem of student-specific problem outcome prediction. However, unlike in subject-specific gesture recognition and context-specific expressivity prediction, we did not find empirical benefits of using our personalization framework over a generic classifier.

8.2 Strengths, Limitations and Future Research Directions

Here, we discuss the strengths of the methods we have proposed and address their weaknesses, suggesting ideas for research directions that could further improve our work.

8.2.1 Gesture Spotting and Recognition

We presented an analysis of methods for gesture spotting and classification by comparing a framework that employs a single multi-class random forest classification model to distinguish gestures from a given vocabulary in a continuous video stream with a framework that uses a cascaded approach. The strengths of the two methods we proposed lie in their simplicity to train and their capacity to generalize well to variations in user size, distance to the sensor, and speeds at which the gestures are performed, as well as our methods' robustness to the effects of sensor noise. One area of the framework that can be improved is the process of selecting and creating better feature sets. Many additional features, such as joint-pair distances used by Yao et al.[134], can be experimented with in order to improve the accuracy of our framework. Additionally, selecting a small group of features over an interval of frames to split a node in a decision tree, instead of selecting a single feature at a single frame, might be better suited to the purpose of learning complex spatio-temporal objects such as gestures. However, computing more features may hamper the random forest framework's speed during test time.

In gesture recognition, there are often ambiguities between similar gesture pairs in both datasets, which our random forest classifier cannot differentiate well. A potential idea for further exploration is to use another layer of tree-forest classifiers to identify the features that can differentiate the ambiguities in order to further refine classification results. In general, gesture classification can be performed in a hierarchical framework, where random forests at the top-most level will accurately separate a dynamically-defined set of super-classes, each of which will be subject to further classification by classifiers at subsequent layers, until all classes are well-separated.

Moreover, feature engineering approaches have generally been replaced by feature learning approaches across many large-scale computer vision tasks, including gesture recognition. Novel neural network architectures based on CNNs, LSTMs, 3D-CNNs and their unique combinations learn discriminative feature representations directly from input skeletal, RGB and depth data and have been shown to obtain good results on numerous gesture recognition benchmark datasets [83, 92, 80, 71]. For example, Neverova et al. [83] presented a gesture localization and recognition scheme based on a multimodal deep learning architecture that leverages audio signals to take advantage of the fact that gestures are often accompanied by speech or sounds.

Another drawback of our current approach lies in the use of a sliding window mechanism. Exhaustive, multi-scale sliding window search is not very computationally efficient and cannot predict flexible gesture boundaries. Workarounds to sliding window approaches have been proposed in the object detection [50, 57] and activity detection [128] literature. However, these approaches rely on the entire input (e.g. the complete input image or complete input video) being available to the algorithm during test time. In real-time gesture recognition applications, where the model must be able to respond with its prediction in real-time, sliding window approaches are still appropriate.

We should also note that deep learning approaches are not always suitable for gesture recognition applications. For one, gesture recognition applications often require low latency computations in resource-constrained devices, e.g. real-time gesture recognition in AR/VR settings, where only a fraction of the on-device computation resources can be devoted to real-time gesture recognition. Second, gesture recognition systems are often designed for specific applications, where data collection and annotation in a scale required for most deep learning methods can be prohibitively expensive.

8.2.2 Predicting Active Facial Expressivity in People with Parkinson's Disease

We presented an interpretable system that computes facial expressivity scores in people with Parkinson's Disease using multimodal audio-visual feature descriptors extracted from a video sequence. Automated assessment of facial expressivity in Parkinson's Disease patients has the potential to be a useful tool for clinicians in this field. Human coders have successfully coded facial expression in people with PD [54] but the costs associated with the manual assessment of all patients with PD can be prohibitively high. Comprehensive manual coding of 20 seconds of video can take upwards of an hour, and often two coders are needed to establish that the human coder is reliable. Most existing works in the domain of computational facial analysis of PD patients are limited to small-scale pilot studies comparing the characteristics and dynamics of facial expressions exhibited by a small group of PD patients against those of a separate control group. By utilizing a dataset of

772 short audio-video clips of 117 PD patients along with their facial expressivity labels, we demonstrated the feasibility of using a machine learning model in predicting the facial expressivity ratings of new audio-video clips.

A potential weakness of our current approach lies in the simplicity of our feature representation. Although summary statistics-based feature representations, such as the ones we have used, provide concise, easy-to-interpret features that was appropriate for our application, we forego a significant amount of signal from the raw input, which could potentially prove useful for more complex classification/regression frameworks. However, utilizing larger, complex models is challenging, given the relatively small size of our dataset (consisting of less than 700 training samples).

Considering that PD is widespread and affects millions of people around the world, the benefits of an accurate, interpretable and automated facial analysis for patients are beyond doubt. One avenue for further research is to extend this work on a larger scale. However, obtaining real patient data on a large scale can be a challenge. An interesting research question to ask then is: can the vast amounts of audio-video interview data widely and freely available in the Internet be leveraged to learn better facial expressivity models? With some expenses for expert annotation, one could train deep, multimodal models on the large, non-PD data and finetune them on the smaller target dataset of PD patients. Given that the distribution of the source domain of interview clips might differ from that of the target domain of interview clips of PD patients, domain adaptation methods might be useful [117].

8.2.3 Affect-driven Learning Outcomes Prediction in Intelligent Tutoring Systems

We investigated the problem of trying to predict the learning outcome of students from facial affect signals, based on a novel dataset of student videos interacting with MathSpring, a popular web-based ITS. The dataset was collected with the intention of releasing it as a benchmark affect dataset in an educational setting, the likes of which are currently missing in the literature. Based on this novel dataset, we developed models to directly predict the learning outcomes of the students from concise action unit-based feature representations that capture the facial affect dynamics of the input video. This is different from most existing work that maps the input video into the student's emotional state, such as happiness, anger and level of engagement.

While the results we provided are that of baseline models, there are several avenues for improvement. First, we have so far ignored two rich streams of information while building our predictive models: the GoPro video stream that captures the students' faces when they are facing down and therefore not visible in the webcam, and the mouse-coordinate clickstream which can often be very informative about the students' internal state. A multi-modal model that utilizes signals from all streams will probably result in better predictive performance.

Despite the relatively large size of the raw dataset, the problem outcome labels are quite sparse. It is therefore challenging to build accurate models that map very high dimensional, highly variable spatio-temporal affect signals into a single problem outcome label using only a few examples. Moreover, the raw input signals are mostly dominated by non-informative neutral facial expressions. Obtaining denser labels around times of high facial activity could help provide an improved understanding of the relationship between facial affective signals and the final problem outcome.

Finally, the biggest challenge in ATSs is to then utilize these affect-sensitive models to provide appropriate and effective interventions that quantifiably improve the learning experience. There have been some recent works that have ventured in this direction. For example, Gordon et al. [51] combined the valence and engagement values inferred from facial affect of students and inputted them into a social robot's reinforcement learning algorithm, which allowed the social robot tutor to personalize its motivational strategies according to the observed facial affect of students. In future work, our research team plans to provide personalized interventions in MathSpring based on the proposed affect analysis models, and then conduct experiments to validate the effectiveness of the interventions, as well as analyze affect signals that result after the interventions are applied.

8.2.4 Hierarchical Bayesian Neural Networks and Applications

Group-specific variations in data can pose a significant challenge to building robust and reliable classifiers. We developed hierarchical Bayesian neural networks for personalized modeling of face and gesture signals in the presence of inter-group and inter-subject variations. When group-membership labels are available, we showed that they can be leveraged to build group-specific models within a hierarchical framework. We demonstrated the utility of this hierarchical approach to three tasks: subject-specific gesture recognition, context-specific facial expressivity prediction and student-specific learning outcome prediction.

One drawback of this hierarchical framework is that it relies on group-membership labels being available during training. When group-membership labels are missing or corrupt and we have no prior knowledge regarding the number of groups that generated our data, is it possible to infer the group-membership labels directly from the data along with the rest of the model parameters? For example, such a mechanism would allow the model to automatically determine contextual clusters in the training data in the absence of pre-defined context labels.

One possible idea to meet this goal is by extending the current hierarchical Bayesian neural network formulation. By placing a Dirichlet Process (DP) prior on the random variables, which indicate the group membership label of a data instance, and learning its parameters, we can learn an effective grouping of the training data [3]. DP mixtures are a popular Bayesian nonparametric model used for clustering problems where the number of clusters need not be specified beforehand. Xue et al. [130] introduced a model similar to ours for the multi-task learning scenario in order to identify subgroups of related tasks. They showed their method worked better than baseline individual per-task models as well as a single model trained with data from all tasks. Recent advances in variational inference algorithms [11] for DP mixtures as an alternative to the more expensive MCMC-based methods point to the possibility of adding this extension to our current framework.

Another shortcoming of this framework is its size, in terms of the total number of model parameters, and time taken to optimize the models relative to comparable non-Bayesian counterparts. This makes scaling this framework to massive, modern end-to-end frameworks particularly challenging. A potential workaround would be to design hybrid networks: convolutional networks for feature learning that are shared across groups and combined with personalized hierarchical Bayesian fully connected or recurrent hierarchical Bayesian networks.

Other extensions to this framework would be to utilize it on personalization challenges in other domains. For example, an interesting application would be to build such systems that can personalize to various devices. Consider, for example, a future smart-home setting where the user interacts with an AI system through speech, emotions and gestures via various devices. A personalized multimodal framework could learn efficient recognition systems for each different device medium.

Bibliography

[1] Riyadh Almutiry, Samuel Couth, Ellen Poliakoff, Sonja Kotz, Monty Silverdale, and Tim Cootes. Facial behaviour analysis in Parkinson's disease. In *International Conference on Medical Imaging and Virtual Reality*, pages 329–339. Springer, 2016.

[2] Jonathan Alon, Vassilis Athitsos, Quan Yuan, and Stan Sclaroff. Simultaneous localization and recognition of dynamic hand gestures. In *Seventh IEEE Workshops on Application of Computer Vision, 2005. WACV/MOTIONS'05*, volume 2, pages 254–260. IEEE, 2005.

[3] Charles E Antoniak. Mixtures of Dirichlet Processes with applications to Bayesian nonparametric problems. *The Annals of Statistics*, pages 1152–1174, 1974.

[4] Ivon Arroyo, Carole Beal, Tom Murray, Rena Walles, and Beverly P Woolf. Web-based intelligent multimedia tutoring for high stakes achievement tests. In *International Conference on Intelligent Tutoring Systems*, pages 468–477. Springer, 2004.

[5] Akshay Asthana, Stefanos Zafeiriou, Shiyang Cheng, and Maja Pantic. Incremental face alignment in the wild. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1859–1866. IEEE, 2014.

[6] Ryan S Baker, Sidney K D'Mello, Ma Mercedes T Rodrigo, and Arthur C Graesser. Better to be frustrated than bored: The incidence, persistence, and impact of learners cognitive–affective states during interactions with three different computer-based learning environments. *International Journal of Human-Computer Studies*, 68(4):223–241, 2010.

[7] Tadas Baltrušaitis, Marwa Mahmoud, and Peter Robinson. Cross-dataset learning and person-specific normalisation for automatic action unit detection. In *Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on*, volume 6, pages 1–6. IEEE, 2015.

[8] Tadas Baltrušaitis, Peter Robinson, and Louis-Philippe Morency. OpenFace: an open source facial behavior analysis toolkit. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–10. IEEE, 2016.

[9] Andrea Bandini, Silvia Orlandi, Hugo Jair Escalante, Fabio Giovannelli, Massimo Cincotta, Carlos A Reyes-Garcia, Paola Vanni, Gaetano Zaccara, and Claudia Manfredi. Analysis of facial expressions in Parkinson's disease through video-based automatic methods. *Journal of Neuroscience Methods*, 281:7–20, 2017.

[10] Christopher M Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[11] David M Blei, Michael I Jordan, et al. Variational inference for Dirichlet Process mixtures. *Bayesian analysis*, 1(1):121–143, 2006.

[12] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 1613–1622, 2015.

[13] Matteo Bologna, Giovanni Fabbrini, Luca Marsili, Giovanni Defazio, Philip D Thompson, and Alfredo Berardelli. Facial bradykinesia. *J Neurol Neurosurg Psychiatry*, 84(6):681–685, 2013.

[14] Anna Bosch, Andrew Zisserman, and Xavier Muoz. Image classification using random forests and ferns. In *IEEE 11th International Conference on Computer Vision, 2007. ICCV 2007.*, pages 1–8. IEEE, 2007.

[15] Léon Bottou. Stochastic gradient learning in neural networks. *Proceedings of Neuro-Nimes*, 91(8), 1991.

[16] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[17] Wray L Buntine and Andreas S Weigend. Bayesian back-propagation. *Complex systems*, 5(6):603–643, 1991.

[18] Necati Cihan Camgöz, Ahmet Alp Kindiroglu, and Lale Akarun. Gesture recognition using template based random forest classifiers. In *Computer Vision-ECCV 2014 Workshops*, pages 579–594. Springer, 2014.

[19] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.

[20] Min Chi, Kurt VanLehn, Diane Litman, and Pamela Jordan. Empirically evaluating the application of reinforcement learning to the induction of effective and adaptive pedagogical strategies. *User Modeling and User-Adapted Interaction*, 21(1-2):137–180, 2011.

[21] Wen-Sheng Chu, Fernando De la Torre, and Jeffrey F Cohn. Selective transfer machine for personalized facial action unit detection. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3515–3522. IEEE, 2013.

[22] Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Workshop on Deep Learning, December 2014*, 2014.

[23] Ira Cohen, Nicu Sebe, Ashutosh Garg, Lawrence S Chen, and Thomas S Huang. Facial expression recognition from video sequences: temporal and static modeling. *Computer Vision and Image Understanding*, 91(1-2):160–187, 2003.

[24] Jeffrey F Cohn and Fernando De la Torre. Automated face analysis for affective computing. *The Oxford handbook of Affective Computing*, page 131, 2014.

[25] Jeffrey F Cohn, Tomas Simon Kruez, Iain Matthews, Ying Yang, Minh Hoai Nguyen, Margara Tejera Padilla, Feng Zhou, and Fernando De la Torre. Detecting depression from facial actions and vocal prosody. In *Affective Computing and Intelligent Interaction and Workshops, 2009. ACII 2009. 3rd International Conference on*, pages 1–7. IEEE, 2009.

[26] Jeffrey F Cohn, Adena J Zlochower, James J Lien, and Takeo Kanade. Feature-point tracking by optical flow discriminates subtle differences in facial expression. In *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, pages 396–401. IEEE, 1998.

[27] Scott D Connell and Anil K Jain. Writer adaptation for online handwriting recognition. *Pattern Analysis and Machine Intelligence, IEEE transactions on*, 24(3):329–346, 2002.

[28] Seth Corrigan, Tiffany Barkley, and Zachary Pardos. Dynamic approaches to modeling student affect and its changing role in learning and performance. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 92–103. Springer, 2015.

[29] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. CVPR 2005.*, volume 1, pages 886–893. IEEE, 2005.

[30] F. De la Torre, W.-S. Chu, X. Xiong, F. Vicente, X. Ding, and J. F. Cohn. Intraface. In *Automatic Face and Gesture Recognition*, 2015.

[31] Fernando De la Torre and Jeffrey F Cohn. Facial expression analysis. In *Visual Analysis of Humans*, pages 377–409. Springer, 2011.

[32] David Demirdjian and Chenna Varri. Recognizing events with temporal random forests. In *Proceedings of the 2009 International Conference on Multimodal Interfaces*, pages 293–296. ACM, 2009.

[33] Matt Dennis, Judith Masthoff, and Chris Mellish. Adapting performance feedback to a learners conscientiousness. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 297–302. Springer, 2012.

[34] S D'Mello, A Graesser, and RW Picard. Toward an affect-sensitive AutoTutor. *IEEE Intelligent Systems*, 4(22):53–61, 2007.

[35] Sidney D'Mello, Ed Dieterle, and Angela Duckworth. Advanced, analytic, automated (aaa) measurement of engagement during learning. *Educational Psychologist*, 52(2):104–123, 2017.

[36] Sidney D'Mello, Blair Lehman, Reinhard Pekrun, and Art Graesser. Confusion can be beneficial for learning. *Learning and Instruction*, 29:153–170, 2014.

[37] Yong Du, Wei Wang, and Liang Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1110–1118, 2015.

[38] Paul Ekman and Wallace V Friesen. *Facial action coding system*. Consulting Psychologists Press, 1977.

[39] Sergio Escalera, Xavier Baró, Jordi Gonzalez, Miguel A Bautista, Meysam Madadi, Miguel Reyes, V Ponce, Hugo J Escalante, Jamie Shotton, and Isabelle Guyon. ChaLearn looking at people challenge 2014: Dataset and results. In *Proceedings of the 2014 IEEE European Conference on Computer Vision (ECCV 2014) ChaLearn Workshop on Looking at People*. IEEE, 2014.

[40] Sergio Escalera, Jordi Gonzàlez, Xavier Baró, Miguel Reyes, Oscar Lopes, Isabelle Guyon, Vassilis Athitsos, and Hugo Escalante. Multi-modal gesture recognition challenge 2013: Dataset and results. In *Proceedings of the 15th ACM on International conference on multimodal interaction*, pages 445–452. ACM, 2013.

[41] Beat Fasel and Juergen Luettin. Automatic facial expression analysis: A survey. *Pattern Recognition*, 36(1):259–275, 2003.

[42] Jenny Rose Finkel and Christopher D Manning. Hierarchical Bayesian domain adaptation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 602–610. Association for Computational Linguistics, 2009.

[43] Simon Fothergill, Helena Mentis, Pushmeet Kohli, and Sebastian Nowozin. Instructing people for training gestural interactive systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1737–1746. ACM, 2012.

[44] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep Bayesian active learning with image data. In *International Conference on Machine Learning*, pages 1183–1192, 2017.

[45] Juergen Gall, Angela Yao, Nima Razavi, Luc Van Gool, and Victor Lempitsky. Hough forests for object detection, tracking, and action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(11):2188–2202, 2011.

[46] Andrew Gelman and Jennifer Hill. *Data Analysis Using Regression and Multilevel/Hierarchical models*. Cambridge University Press, 2006.

[47] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to forget: Continual prediction with LSTM. 1999.

[48] Samuel J Gershman and Noah D Goodman. Amortized inference in probabilistic reasoning. In *Proceedings of the 36th Annual Conference of the Cognitive Science Society*, 2014.

[49] Malay Ghosh, Tapabrata Maiti, Dalho Kim, Sounak Chakraborty, and Ashutosh Tewari. Hierarchical Bayesian neural networks: an application to a prostate cancer study. *Journal of the American Statistical Association*, 99(467):601–608, 2004.

[50] Ross Girshick. Fast R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1440–1448, 2015.

[51] Goren Gordon, Samuel Spaulding, Jacqueline Kory Westlund, Jin Joo Lee, Luke Plummer, Marayna Martinez, Madhurima Das, and Cynthia Breazeal. Affective personalization of a social robot tutor for children's second language skills. In *AAAI*, pages 3951–3957, 2016.

[52] Joseph Grafsgaard, Joseph B Wiggins, Kristy Elizabeth Boyer, Eric N Wiebe, and James Lester. Automatically recognizing facial expression: Predicting engagement and frustration. In *Educational Data Mining 2013*, 2013.

[53] Joseph F Grafsgaard, Seung Y Lee, Bradford W Mott, Kristy Elizabeth Boyer, and James C Lester. Modeling self-efficacy across age groups with automatically tracked facial expression. In *International Conference on Artificial Intelligence in Education*, pages 582–585. Springer, 2015.

[54] Sarah D Gunnery, Elena N Naumova, Marie Saint-Hilaire, and Linda Tickle-Degnen. Mapping spontaneous facial expression in people with Parkinson's disease: a multiple case study design. *Cogent Psychology*, page 1376425, 2017.

[55] Danita Hartley and Antonija Mitrovic. Supporting learning by opening the student model. In *International Conference on Intelligent Tutoring Systems*, pages 453–462. Springer, 2002.

[56] Haibo He, Yang Bai, Edwardo A Garcia, and Shutao Li. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*, pages 1322–1328. IEEE, 2008.

[57] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 2980–2988. IEEE, 2017.

[58] Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011.

[59] Ajjen Joshi, Soumya Ghosh, Margrit Betke, and Hanspeter Pfister. Hierarchical Bayesian neural networks for personalized classification. In *Neural Information Processing Systems Workshop on Bayesian Deep Learning*, 2016.

[60] Adam Kendon. *Gesture: Visible Action as Utterance*. Cambridge University Press, 2004.

[61] Saad M Khan and Mubarak Shah. A multiview approach to tracking people in crowded scenes using a planar homography constraint. In *European Conference on Computer Vision*, pages 133–146. Springer, 2006.

[62] Wolf Kienzle and Kumar Chellapilla. Personalized handwriting recognition via biased regularization. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 457–464. ACM, 2006.

[63] Minyoung Kim and Vladimir Pavlovic. Structured output ordinal regression for dynamic facial emotion intensity prediction. In *European conference on computer vision*, pages 649–662. Springer, 2010.

[64] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[65] Diederik P. Kingma, Tim Salimans, and Max Welling. Variational dropout and the local reparameterization trick. In *Advances in Neural Information Processing Systems*, 2015.

[66] Diederik P Kingma and Max Welling. Stochastic gradient VB and the variational auto-encoder. In *Second International Conference on Learning Representations, ICLR*, 2014.

[67] Alina Kuznetsova, Laura Leal-Taixé, and Bodo Rosenhahn. Real-time sign language recognition using a consumer depth camera. In *Computer Vision Workshops (ICCVW), 2013 IEEE International Conference on*, pages 83–90. IEEE, 2013.

[68] John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional Random Fields: Probabilistic models for segmenting and labeling sequence data. *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289, 2001.

[69] Rung-Huei Liang and Ming Ouhyoung. A real-time continuous gesture recognition system for sign language. In *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, pages 558–567. IEEE, 1998.

[70] Yulan Liang and Arpad G Kelemen. Hierarchical Bayesian neural network for gene expression temporal patterns. *Statistical Applications in Genetics and Molecular Biology*, 3(1):1–23.

[71] Jun Liu, Amir Shahroudy, Dong Xu, and Gang Wang. Spatio-temporal LSTM with trust gates for 3d human action recognition. In *European Conference on Computer Vision*, pages 816–833. Springer, 2016.

[72] Daniel Lopez-Martinez and Rosalind Picard. Multi-task neural networks for personalized pain recognition from physiological signals. *arXiv preprint arXiv:1708.08755*, 2017.

[73] David JC MacKay. A practical Bayesian framework for backpropagation networks. *Neural computation*, 4(3):448–472, 1992.

[74] Sotiris Malassiotis, Niki Aifanti, and Michael G Strintzis. A gesture recognition system using 3D data. In *Proceedings First International Symposium on 3D Data Processing Visualization and Transmission, 2002*, pages 190–193. IEEE, 2002.

[75] Javier Marin, David Vázquez, Antonio M López, Jaume Amores, and Bastian Leibe. Random forests of local experts for pedestrian detection. In *2013 IEEE International Conference on Computer Vision (ICCV)*, pages 2592–2599. IEEE, 2013.

[76] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. librosa: Audio and music signal analysis in Python. In *Proceedings of the 14th Python in Science Conference*, pages 18–25, 2015.

[77] David McNeill. *Gesture and Thought*. University of Chicago press, 2008.

[78] Leandro Miranda, Thales Vieira, Dimas Martinez, Thomas Lewiner, Antonio W Vieira, and Mario FM Campos. Real-time gesture recognition from depth data through key poses learning and decision forests. In *25th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), 2012*, pages 268–275. IEEE, 2012.

[79] Sushmita Mitra and Tinku Acharya. Gesture recognition: A survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 37(3):311–324, 2007.

[80] Pavlo Molchanov, Shalini Gupta, Kihwan Kim, and Jan Kautz. Hand gesture recognition with 3d convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–7, 2015.

[81] Aamir Mustafa, Amanjot Kaur, Love Mehta, and Abhinav Dhall. Prediction and localization of student engagement in the wild. *arXiv preprint arXiv:1804.00858*, 2018.

[82] Radford M Neal. *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media, 2012.

[83] Natalia Neverova, Christian Wolf, Graham Taylor, and Florian Nebout. Moddrop: adaptive multi-modal gesture recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence, 2015*, 2015.

[84] Natalia Neverova, Christian Wolf, Graham W Taylor, and Florian Nebout. Multiscale deep learning for gesture detection and localization. In *Computer Vision-ECCV 2014 Workshops*, pages 474–490. Springer, 2014.

[85] Bingbing Ni, Gang Wang, and Pierre Moulin. RGBD-HuDaAct: A color-depth video database for human daily activity recognition. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 1147–1153. IEEE, 2011.

[86] Jeremie Nicolle, Kevin Bailly, and Mohamed Chetouani. Facial action unit intensity prediction via hard multi-task metric learning for kernel regression. In *Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on*, volume 6, pages 1–6. IEEE, 2015.

[87] Andrew M Olney, Sidney D'Mello, Natalie Person, Whitney Cade, Patrick Hays, Claire Williams, Blair Lehman, and Arthur Graesser. Guru: A computer tutor that models expert human tutors. In *International Conference on Intelligent Tutoring Systems*, pages 256–261. Springer, 2012.

[88] Maja Pantic and Leon J. M. Rothkrantz. Automatic analysis of facial expressions: The state of the art. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(12):1424–1445, 2000.

[89] Maja Pantic and Leon JM Rothkrantz. Expert system for automatic analysis of facial expressions. *Image and Vision Computing*, 18(11):881–905, 2000.

[90] Reinhard Pekrun, Thomas Goetz, Lia M Daniels, Robert H Stupnisky, and Raymond P Perry. Boredom in achievement settings: Exploring control–value antecedents and performance outcomes of a neglected emotion. *Journal of Educational Psychology*, 102(3):531, 2010.

[91] Rosalind W Picard. *Affective Computing*. MIT Press, 1995.

[92] Lionel Pigou, Aäron van den Oord, Sander Dieleman, Mieke Van Herreweghe, and Joni Dambre. Beyond temporal pooling: Recurrence and temporal convolutions for gesture recognition in video. *arXiv preprint arXiv:1506.01911*, 2015.

[93] Ariadna Quattoni, Sybor Wang, L-P Morency, Michael Collins, and Trevor Darrell. Hidden conditional random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(10):1848–1852, 2007.

[94] Rajesh Ranganath, Sean Gerrish, and David M Blei. Black box variational inference. In *AISTATS*, pages 814–822, 2014.

[95] Michalis Raptis, Darko Kirovski, and Hugues Hoppe. Real-time classification of dance gestures from skeleton animation. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 147–156. ACM, 2011.

[96] Siddharth S Rautaray and Anupam Agrawal. Vision based hand gesture recognition for human computer interaction: A survey. *Artificial Intelligence Review*, 43(1):1–54, 2015.

[97] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of The 31st International Conference on Machine Learning*, pages 1278–1286, 2014.

[98] Ognjen Rudovic, Vladimir Pavlovic, and Maja Pantic. Context-sensitive dynamic ordinal regression for intensity estimation of facial action units. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(5):944–958, 2015.

[99] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *Computer Vision–ECCV 2010*, pages 213–226. Springer, 2010.

[100] Florian Schroff, Antonio Criminisi, and Andrew Zisserman. Object class segmentation using random forests. In *Proceedings of the British Machine Vision Conference*, 2008.

[101] Koichi Shinoda and Chin-Hui Lee. A structural Bayes approach to speaker adaptation. *Speech and Audio Processing, IEEE Transactions on*, 9(3):276–287, 2001.

[102] Jamie Shotton, Toby Sharp, Alex Kipman, Andrew Fitzgibbon, Mark Finocchio, Andrew Blake, Mat Cook, and Richard Moore. Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 56(1):116–124, 2013.

[103] Mohammad Soleymani, Maja Pantic, and Thierry Pun. Multimodal emotion recognition in response to videos. *IEEE Transactions on Affective Computing*, 3(2):211–223, 2012.

[104] Yale Song, David Demirdjian, and Randall Davis. Multi-signal gesture recognition using temporal smoothing hidden conditional random fields. In *2011 IEEE International Conference on Automatic Face & Gesture Recognition and Workshops (FG 2011)*, pages 388–393. IEEE, 2011.

[105] Yale Song, David Demirdjian, and Randall Davis. Tracking body and hands for gesture recognition: Natops aircraft handling signals database. In *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pages 500–506. IEEE, 2011.

[106] Yale Song, David Demirdjian, and Randall Davis. Continuous body and hand gesture recognition for natural human-computer interaction. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 2(1):5.1–5.28, 2012.

[107] Yale Song, L Morency, and Randall Davis. Multi-view latent variable discriminative models for action recognition. In *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2120–2127. IEEE, 2012.

[108] Thad Starner and Alex Pentland. Real-time american sign language recognition from video using hidden markov models. In *Motion-Based Recognition*, pages 227–243. Springer, 1997.

[109] Kayoko Takahashi, Linda Tickle-Degnen, Wendy J Coster, and Nancy K Latham. Expressive behavior in Parkinson's disease as a function of interview context. *American Journal of Occupational Therapy*, 64(3):484–495, 2010.

[110] Y-I Tian, Takeo Kanade, and Jeffrey F Cohn. Recognizing action units for facial expression analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2):97–115, 2001.

[111] Linda Tickle-Degnen. *The Interpersonal communication rating protocol: A manual for measuring individual expressive behavior*. Tufts University, 2010.

[112] Linda Tickle-Degnen, Terry Ellis, Marie H Saint-Hilaire, Cathi A Thomas, and Robert C Wagenaar. Self-management rehabilitation and health-related quality of life in Parkinson's disease: A randomized controlled trial. *Movement Disorders*, 25(2):194–204, 2010.

[113] Linda Tickle-Degnen and Kathleen Doyle Lyons. Practitioners impressions of patients with Parkinson's disease: the social ecology of the expressive mask. *Social Science & Medicine*, 58(3):603–614, 2004.

[114] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-RMSprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 2012.

[115] Michalis Titsias and Miguel Lázaro-Gredilla. Doubly stochastic variational Bayes for non-conjugate inference. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1971–1979, 2014.

[116] Matthew A Turk and Alex P Pentland. Face recognition using eigenfaces. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, pages 586–591. IEEE, 1991.

[117] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Computer Vision and Pattern Recognition (CVPR)*, volume 1, page 4, 2017.

[118] Alexandria K Vail, Joseph F Grafsgaard, Kristy Elizabeth Boyer, Eric N Wiebe, and James C Lester. Gender differences in facial expressions of affect during learning. In *Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization*, pages 65–73. ACM, 2016.

[119] Michel Valstar. Automatic facial expression analysis. In *Understanding Facial Expressions in Communication*, pages 143–172. Springer, 2015.

[120] Michel F Valstar and Maja Pantic. Fully automatic recognition of the temporal phases of facial actions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(1):28–43, 2012.

[121] Kurt VanLehn. The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. *Educational Psychologist*, 46(4):197–221, 2011.

[122] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Action recognition by dense trajectories. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3169–3176. IEEE, 2011.

[123] Peng Wang, Frederick Barrett, Elizabeth Martin, Marina Milonova, Raquel E Gur, Ruben C Gur, Christian Kohler, and Ragini Verma. Automated video-based facial expression analysis of neuropsychiatric disorders. *Journal of neuroscience methods*, 168(1):224–238, 2008.

[124] Jacob Whitehill, Zewelanji Serpell, Yi-Ching Lin, Aysha Foster, and Javier R Movellan. The faces of engagement: Automatic recognition of student engagementfrom facial expressions. *IEEE Transactions on Affective Computing*, 5(1):86–98, 2014.

[125] Michael Wixon and Ivon Arroyo. When the question is part of the answer: Examining the impact of emotion self-reports on student emotion. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 471–477. Springer, 2014.

[126] Peng Wu, Isabel Gonzalez, Georgios Patsis, Dongmei Jiang, Hichem Sahli, Eric Kerckhofs, and Marie Vandekerckhove. Objectifying facial expressivity assessment of Parkinson's patients: Preliminary study. *Computational and Mathematical Methods in Medicine*, 2014, 2014.

[127] Lu Xia, Chia-Chih Chen, and JK Aggarwal. View invariant human action recognition using histograms of 3d joints. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 20–27. IEEE, 2012.

[128] Huijuan Xu, Abir Das, and Kate Saenko. R-C3D: region convolutional 3d network for temporal activity detection. In *IEEE Int. Conf. on Computer Vision (ICCV)*, pages 5794–5803, 2017.

[129] Lijie Xu and Kikuo Fujimura. Real-time driver activity recognition with random forests. In *Proceedings of the 6th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, pages 1–8. ACM, 2014.

[130] Ya Xue, Xuejun Liao, Lawrence Carin, and Balaji Krishnapuram. Multi-task learning for classification with dirichlet process priors. *Journal of Machine Learning Research*, 8(Jan):35–63, 2007.

[131] Jun Yang, Rong Yan, and Alexander G Hauptmann. Cross-domain video concept detection using adaptive SVMs. In *Proceedings of the 15th International Conference on Multimedia*, pages 188–197. ACM, 2007.

[132] Shuang Yang, Ognjen Rudovic, Vladimir Pavlovic, and Maja Pantic. Personalized modeling of facial action unit intensity. In *International Symposium on Visual Computing*, pages 269–281. Springer, 2014.

[133] Yi Yang and Deva Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1385–1392. IEEE, 2011.

[134] Angela Yao, Juergen Gall, Gabriele Fanelli, and Luc J Van Gool. Does human action recognition benefit from pose estimation?. In *Proceedings of the British Machine Vision Conference*, volume 3, page 6, 2011.

[135] Angela Yao, Luc Van Gool, and Pushmeet Kohli. Gesture recognition portfolios for personalization. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1923–1930. IEEE, 2014.

[136] Gang Yu, Norberto A Goussies, Junsong Yuan, and Zicheng Liu. Fast action detection via discriminative random forest voting and top-k subvolume search. *IEEE Transactions on Multimedia*, 13(3):507–517, 2011.

[137] Tsz-Ho Yu, Tae-Kyun Kim, and Roberto Cipolla. Real-time action recognition by spatiotemporal semantic and structural forests. In *Proceedings of the British Machine Vision Conference*, volume 2, page 6, 2010.

[138] Konstantin Zakharov. Affect recognition and support in intelligent tutoring systems. Master's thesis, University of Canterbury, Computer Science and Software Engineering, 2007.

[139] Ramón Zatarain-Cabada, María Lucía Barrón-Estrada, José Luis Olivares Camacho, and Carlos A Reyes-García. Affective tutoring system for Android mobiles. In *International Conference on Intelligent Computing*, pages 1–10. Springer, 2014.

Ajjen Joshi Curriculum Vitae

Website: http://cs-people.bu.edu/ajjendj
Email: ajjendj@bu.edu
Phone: 860-501-8468

EDUCATION

Boston University | Boston, MA

Ph.D., Computer Science *expected* 2018

- Thesis: *Personalized Face and Gesture Analysis Using Hierarchical Neural Networks*
- Advisors: Dr. Margrit Betke and Dr. Stan Sclaroff

Boston University | Boston, MA

M.S., Computer Science 2014

- Thesis: *A Random Forest Approach to Segmenting and Classifying Gestures*
- Advisors: Dr. Margrit Betke and Dr. Stan Sclaroff
- GPA: 3.9/4.0

Connecticut College | New London, CT

B.A., Computer Science and Architectural Studies (Double Major) 2012

- Thesis: *Real-time Facial Animation by Gesture Imitation*
- Advisor: Dr. Ozgur Izmirli
- GPA: 3.96/4.0 *Summa Cum Laude*

WORK EXPERIENCE

Adobe Research | Cambridge, MA
Research Intern Summer 2016

- Explored a deep learning approach to automatically generate inbetween frames in 2D handdrawn animations. Advised by Masha Shugrina

Disney Research | Cambridge, MA
Research Intern Summer 2015

- Implemented prototype system for performing gesture recognition from glove sensor data and explored development of subject-specific hierarchical Bayesian classifiers. Advised by Dr. Hanspeter Pfister, Dr. Soumya Ghosh

Brown University | Providence, RI
Research Intern Summer 2011

- Created interactive multimedia installations in Max/MSP/Jitter using the Microsoft Kinect. Advised by Dr. Todd Winkler.

PUBLICATIONS

[1] Andrew Kurauchi, Wenxin Feng, Ajjen Joshi, Carlos Morimoto, Margrit Betke. **EyeSwipe: Towards Fast and Comfortable Text Entry Using Gaze Paths**. *In Preparation.*

[2] Rohit Agrawal, Ajjen Joshi, Margrit Betke. **Enabling Early Gesture Recognition by Motion Augmentation**. ACM International Conference on Pervasive Technologies Related to Assistive Environments (PETRA), 2018. *Oral.*

[3] Ajjen Joshi, Soumya Ghosh, Sarah Gunnery, Linda Tickle-Degnen, Margrit Betke, Stan Sclaroff. **Context-Sensitive Prediction of Facial Expressivity Using Multimodal Hierarchical Bayesian Neural Networks**. IEEE International Conference on Automatic Face and Gesture Recognition (AFGR), 2018. *Poster.*

[4] Ajjen Joshi, Soumya Ghosh, Margrit Betke, Stan Sclaroff, Hanspeter Pfister. **Personalizing Gesture Recognition Using Hierarchical Bayesian Neural Networks**. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017. *Poster.*

[5] Elham Saraee, Saurabh Singh, Kathryn Hendron, Mingxin Zheng, Ajjen Joshi, Terry Ellis, Margrit Betke. **ExerciseCheck: Remote Monitoring and Evaluation Platform for Home Based Physical Therapy**. ACM International Conference on Pervasive Technologies Related to Assistive Environments (PETRA), 2017. *Oral.*

[6] Elham Saraee, Ajjen Joshi, Margrit Betke. **A Therapeutic Robotic System for the Upper Body based on the Proficio Robotic Arm**. IEEE International Conference on Virtual Rehabilitation (ICVR), 2017. *Poster.*

[7] Elham Saraee, Saurabh Singh, Ajjen Joshi, Margrit Betke. **PostureCheck: Posture Modeling for Exercise Assessment using the Microsoft Kinect**. IEEE International Conference on Virtual Rehabilitation (ICVR), 2017. *Poster.*

[8] Ajjen Joshi, Soumya Ghosh, Margrit Betke, Hanspeter Pfister. **Hierarchical Bayesian Neural Networks for Personalized Classification**. Neural Information Processing Systems (NIPS) Workshop on Bayesian Deep Learning, 2016. *Poster*.

[9] Ajjen Joshi, Linda Tickle-Degnen, Sarah Gunnery, Terry Ellis, Margrit Betke. **Predicting Active Facial Expressivity in People with Parkinson's Disease**. ACM International Conference on Pervasive Technologies Related to Assistive Environments (PETRA), 2016. *Oral*.

[10] Ajjen Joshi, Camille Monnier, Margrit Betke, Stan Sclaroff. **Comparing Random Forest Approaches to Segmenting and Classifying Gestures**. Image and Vision Computing (IMAVIS), 2016.

[11] Andrew Kurauchi, Wenxin Feng, Ajjen Joshi, Carlos Morimoto, Margrit Betke. **EyeSwipe: Dwell-free Text Entry Using Gaze Paths**. ACM Conference on Human Factors in Computing Systems (CHI), 2016. *Oral*.

[12] Huy Le, Ajjen Joshi, Margrit Betke. **b3.js: A Library for Interactive Virtual Reality Web 3D Graphs**. IEEE Conference on Virtual Reality and 3D User Interfaces (VR), 2016. *Research Demo*.

[13] Ajjen Joshi, Camille Monnier, Margrit Betke, Stan Sclaroff. **A Random Forest Approach to Segmenting and Classifying Gestures**. IEEE International Conference on Automatic Face and Gesture Recognition (AFGR), 2015. *Oral*.

PRODUCTS

[1] Ajjen Joshi, Danielle Allessio, John Magee, Jacob Whitehill, Beverly Woolf, Stan Sclaroff, Margrit Betke. **Student Learning Outcome Prediction Dataset**, *to be released 2018. Dataset*

[2] Andrew Kurauchi, Wenxin Feng, Ajjen Joshi, Carlos Morimoto, Margrit Betke. **EyeSwipe**, 2018. *Software*.

[3] Ajjen Joshi, Soumya Ghosh, Margrit Betke, Stan Sclaroff, Hanspeter Pfister. **Hierarchical Bayesian Neural Networks**, 2017. *Software*.

TALKS

[1] **Analysis of Facial Expressivity in Parkinson's Disease Patients using Hierarchical Bayesian Neural Networks.** Tufts University Health Quality of Life Lab Seminar. Medford, MA. 2017.

[2] **Personalizing Gesture Recognition Using Hierarchical Bayesian Neural Networks.** New England Computer Vision Workshop. Boston, MA. 2016.

[3] **Deeptween: A Data-Driven Approach to Automatic Inbetweening in Hand-drawn Animations.** Adobe Research Intern Presentation. Cambridge, MA. 2016.

[4] **Predicting Active Facial Expressivity in People with Parkinson's Disease.** PETRA. Corfu, Greece. 2016.

[5] **Hierarchical Bayesian Models for Subject-specific Gesture Recognition.** Disney Research Intern Presentation. Cambridge, MA. 2015.

[6] **Victory Over the Sun: Panel Discussion (along with Harlow Robinson, Larissa Shmailo and Anna Winestein).** Boston, MA. 2015.

[7] **A Random Forest Approach to Segmenting and Classifying Gestures.** AFGR. Ljubljana, Slovenia. 2015.

TEACHING

- Artificial Intelligence (Senior undergraduate course in AI) Spring 2017
  Rating: 4.65/5 (rated by 32 students)
- Artificial Intelligence (Senior undergraduate course in AI) Spring 2016
  Rating: 4.68/5 (rated by 19 students)
- Image and Video Computing (Graduate course in computer vision) Fall 2014
  Rating: 4.82/5 (rated by 22 students)
- Application Programming (Introductory course in programming) Fall 2013
  Rating: 4.43/5 (rated by 44 students)

MENTORING

[1] Muhammad Zuhayr Raghib, Master's Project on **Using 3D-CNNs for Student Engagement Prediction in Intelligent Tutoring Systems.** Spring 2018.

[2] Yitian Lin, Master's Project on **Person Identification using Gaze Patterns.** Spring 2018.

[3] Pratikkumar Patel, Master's Project on **Using LSTMs To Improve Text Input Speed In Eye Typing Systems.** Fall 2017.

[4] Rohit Agrawal, Master's Project on **Enabling Early Gesture Recognition by Motion Augmentation.** Fall 2017. [Publication 2]

[5] Srivathsa Rajagopal, Master's Project on **Facial Expression Analysis of US Presidential Debates.** Fall 2016.

[6] Huy Le, Senior Undergraduate Research Project on **Building a Library for Data Visualization in Virtual Reality.** Fall 2015. [Publication 12]

AWARDS

[1] AFGR 2018 Best Reviewer Award (2018)

[2] AFGR 2018 Doctoral Consortium Award (2018)

[3] PETRA 2016 Doctoral Consortium Award (2016)

[4] One of best reviewed papers of Automatic Face and Gesture Recognition (AFGR 2015)

[5] Boston University Computer Science Teaching Excellence Award (2015)

[6] Phi Beta Kappa (2012)

[7] Architectural Studies Award for Outstanding Graduating Senior (2012)

[8] Winthrop Scholar, Connecticut College's highest academic honor (2011)

[9] Keck Research Grant (2010)

[10] Ranked 1st out of 108 students of high school graduating class (2007)

SERVICE

- Reviewer/Program Committee
  ECCV ‘18, CVPR ‘18, AFGR ‘18, AFGR ‘17, PSIVT ‘17, CVPRW ‘17, PETRA ‘17, PETRA ’16, Pattern Recognition, Journal of AI Research

- AI@BU Seminar Coordinator (Fall 2016-Spring 2018)