

Certificate of Research Option Approval

School of Computer Science
College of Computing
Georgia Institute of Technology

This form will not be signed by the undergraduate coordinator until the two faculty readers have given their signature approval.

Date: 5/6/2021 | 2:00 AM EDT

Student Name	Dhruva Bansal
gtID#	903437552
Major	Computer Science
Faculty Mentor Name	Thad Starner
Thesis Title	Leveraging 2D pose estimators for American Sign Language Recognition

A. Required Coursework

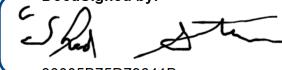
Please *individually* list all completed research courses (e.g., 2698, 2699, 4698, 4699). Must contain a minimum of nine hours of undergraduate research.

Course	Credit Hours	Semester/Year Completed
LMC 4701, Undergraduate Research Proposal Writing	1	Fall 2020
LMC 4702, Undergraduate Research Thesis Writing	1	Spring 2021
CS 4699	3	Fall 2019
CS 4699	3	Spring 2020
CS 4698	3	Spring 2021

B. Other Requirements

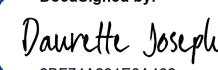
C. Completion of Thesis

We, the below signed, hereby state our full approval of the thesis submitted by the above student in partial fulfillment of the requirements for the Research Option

Thad Starner	 DocuSigned by: 30005B75BD79041B...	5/6/2021 9:09 AM EDT
Faculty Mentor (print name)	Signature	Date
Thomas Ploetz	 DocuSigned by: F8EC3BF6F376413...	5/6/2021 9:23 AM EDT
Faculty 2 nd Thesis Reader (print name)	Signature	Date

D. School Approval

Approval to Grant Research Option Designation

Daurette Joseph	 DocuSigned by: 2BF74A291E8A402...	5/6/2021 5:13 PM EDT
Undergraduate Coordinator (print name)	Signature	Date

E. Thesis Electronic Archiving Permissions

Certificate of Authenticity

I certify that the content of this electronic thesis is the same content approved by my committee.

 ^{ds}
DB I Agree

Copyright Statement

I hereby certify that, if appropriate, I have obtained and attached hereto a written permission statement from the owner(s) of each third party copyrighted matter to be included in my thesis, dissertation, or project report, allowing distribution as specified below. I certify that the version I submitted is the same as that approved by my advisory committee.

I hereby grant to Georgia Tech or its agents the non-exclusive license to archive and make accessible, under the conditions specified below, my thesis, dissertation, or project report in whole or in part in all forms of media, now or hereafter known. I retain all other ownership rights to the copyright of the thesis, dissertation or project report. I also retain the right to use in future works (such as articles or books) all or part of this thesis, dissertation, or project report.

 ^{ds}
DB I Agree

Availability

My advisory committee and I agree that the above-mentioned document be placed in the electronic research option thesis archive with the following status: (Please select one.)

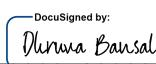
1. Release the entire work immediately for access worldwide after my graduation.
 2. Secure the entire work for one year. Access to all portions of the research option thesis will be restricted for a period of one year to all, including the Georgia Tech community. This option addresses situations such as when a patent application is planned, or when proprietary interests are at stake. NOTE: THIS OPTION MAY ONLY BE EXERCISED WITH THE WRITTEN CONSENT OF THE MAJOR PROFESSOR. YOU MAY CHOOSE THIS OPTION, BUT IF NO LETTER IS RECEIVED FROM THE FACULTY MEMBER ACCESS WILL BECOME WORLDWIDE. (This option may be extended if the major professor petitions the library in writing. If no written request for an extension is requested and granted, the thesis will be released after one year without further notice.)

Dhruva Bansal

College of Computing

Printed Name of Student

School Granting Research Option


DocuSigned by:
Dhruva Bansal
A79A133201B5C9465...

5/6/2021 | 2:00 AM EDT

Student Signature

Date

Mentor Approval

I certify that this Research Option Thesis can be available publicly through Georgia Tech's repository, SMARTech (smartech.gatech.edu), and I know of no contractual, proprietary, U.S. government security-related, or other reasons it should be restricted.

Thad Starner

5/6/2021 | 9:09 AM EDT

Printed Name of Mentor

Signature


DocuSigned by:
T Starner
300050730796418...

Date

Thesis Archiving Submission Instructions

1. Students must hand deliver a **copy** of this form between 8:00am and 4:00pm with all required signatures **no later than the last day of finals** to:

Undergraduate Research Opportunities Program (UROP)
Georgia Institute of Technology
Clough Undergraduate Learning Commons
266 Fourth Street NW, Suite 205
Atlanta, GA 30332-0940

Alternatively, students may scan and email their completed form to urop@gatech.edu.

The original form will remain with the school.

2. Students will create an account on the electronic Research Option thesis website <https://thesis.gatech.edu>. After all signatures have been obtained, the student will: (a) paste the thesis abstract into the appropriate box (350-word limit) on this website, (b) enter keywords and phrases (separated by commas), and (c) upload their thesis.
3. Upon receipt of the copy of the signed certification form in the UROP office and student graduation (confirmed by the Registrar's Office), the thesis will be posted electronically with the library via SMARTech. (Please allow at least two to three weeks for posting following graduation.)

Bachelor Thesis Attachment



Leveraging 2D pose estimators for American Sign

Language Recognition

Dhruva Bansal

College of Computing

Georgia Institute of Technology

Supervisor

Prof. Thad Starner and Prof. Thomas Ploetz

In partial fulfillment of the requirements for the degree of

Bachelor of Science in Computer Science

May 8, 2021

Abstract

Most deaf children born to hearing parents do not have continuous access to language, leading to weaker short-term memory compared to deaf children born to deaf parents. This lack of short-term memory has serious consequences on their mental health and employment rate. To this end, prior work has explored CopyCat, a game where children interact with virtual actors using sign language. While CopyCat has been shown to improve language generation, reception, and repetition, it uses expensive hardware for sign language recognition. This thesis explores the feasibility of using 2D off-the-shelf camera-based pose estimators such as MediaPipe for complementing sign language recognition and moving towards a ubiquitous recognition framework. We compare MediaPipe with 3D pose estimators such as Azure Kinect to determine the feasibility of using off-the-shelf cameras. Furthermore, we develop and compare Hidden Markov Models (HMMs) with state-of-the-art recognition models like Transformers to determine which model is best suited for American Sign Language Recognition in a constrained environment. We find that MediaPipe marginally outperforms Kinect in various experimental settings. Additionally, HMMs outperform Transformers by on average 17.0% on recognition accuracy. Given these results, we believe that a widely deployable game using only a 2D camera is feasible.

Contents

1	Introduction	1
1.1	Motivation	2
1.2	Contributions	3
2	Literature Review	4
2.1	Language Acquisition	5
2.2	Pose Estimation	5
2.3	Multimodal Recognition	6
2.4	State-of-the-art Recognition	7
3	Methods	9
3.1	Feature Extraction	9
3.2	Recognition Models	10
3.3	Experimental Setup	13
4	Results and Discussions	14
4.1	User Dependent and Adaptive	14
4.2	User Independent	15
5	Future Work	16
5.1	ASL Verification	16
5.2	Segmentally Boosted HMMs	18
5.3	User Studies	18
6	Conclusion	19
References		20

List of Figures

1.1	CopyCat game screen. Children tell Iris the cat where the animal is hiding (e.g., LION ON GREY WALL)	1
1.2	Six students playing CopyCat improved language skills much more quickly than six students attending standard classroom instruction. All improve- ments are statistically significant at $p < 0.05$	2
2.1	Originally, CopyCat used specially-designed kiosk (left) and custom sensor gloves (center) with embedded accelerometers (right) to achieve sufficient recognition accuracy for gameplay.	4
3.1	Pose estimation with Azure Kinect (left, sign "in"), and MediaPipe (right, sign "alligator") showing difficult signs	9
3.2	Transformer model implemented for performing ASL Recognition	12

List of Tables

4.1	User adaptive and user dependent word (sentence) percent accuracy . . .	14
4.2	User independent word (sentence) percent accuracy using HMMs	15
4.3	User independent word (sentence) percent accuracy using Transformers .	15

Chapter 1

Introduction

95% of deaf children have hearing parents that do not learn enough American Sign Language (ASL) to teach their infants [1]. Due to deficient short-term language memory skills, which are learned while acquiring a language, many prelingually deafened children can only repeat 1-2 signs compared to 4-6 signs for children with deaf parents. This deficiency of short-term language memory skills can lead to Language Deprivation Syndrome (LDS). LDS can cause lifelong challenges such as a 2-7x increase in mental health problems, a 50% unemployment rate, and a 3-60x increase in suicide [2]–[5]. The CopyCat project aims to augment early classroom teaching by presenting a game in which a virtual teacher focuses on repetition as a method for building short-term language memory skills.



Figure 1.1 CopyCat game screen. Children tell Iris the cat where the animal is hiding (e.g., LION ON GREY WALL)

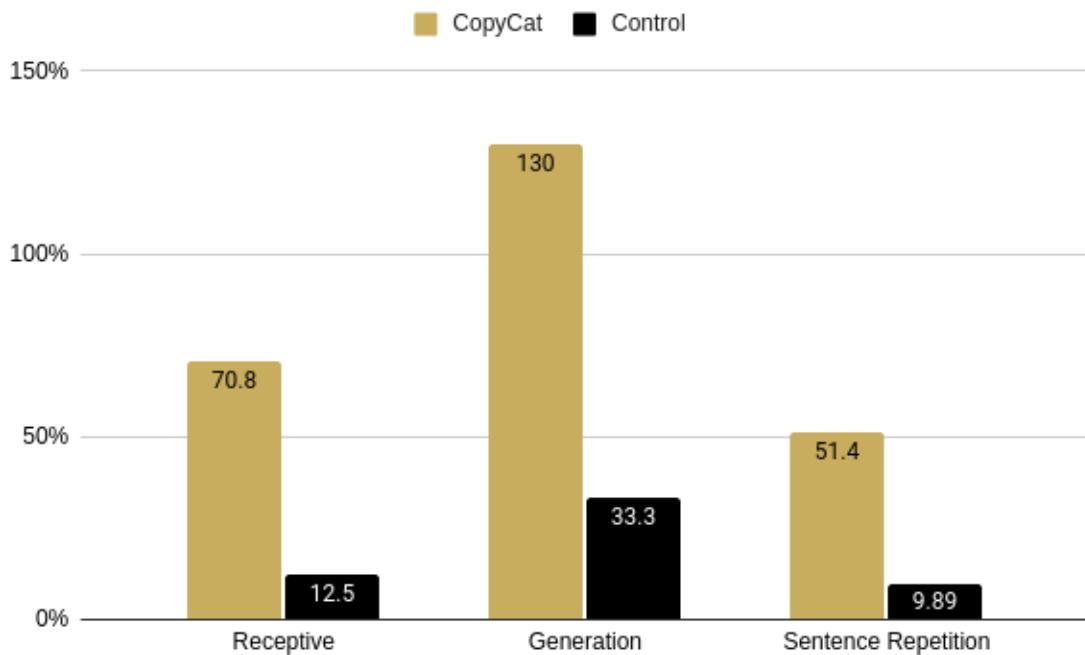


Figure 1.2 Six students playing CopyCat improved language skills much more quickly than six students attending standard classroom instruction. All improvements are statistically significant at $p < 0.05$

1.1 Motivation

Past research has shown that interactive games made to help deaf children acquire language can significantly bolster their short-term language memory skills. Weaver et al. first introduced CopyCat (shown in Fig 1.1) as an entertaining game that provides additional language exposure for these children to support short-term language memory acquisition [6]. Their Wizard of Oz experiment revealed that students who played CopyCat improved their scores on various metrics significantly more than the learners who did not (shown in Fig 1.2). The game's effectiveness naturally depends on the accuracy with which it classifies signs as correct or incorrect. Past research has relied on collecting data using expensive custom hardware, equipped with accelerometers along with Hidden Markov Models (HMM) for performing this classification. Brashears et al. initiated these efforts by reporting 90.48% sentence accuracy using an HMM trained on a multimodal dataset consisting of both videos and accelerometer data [7]. Unfortunately, this dependence on accelerometer data limits scale and ubiquity. Researchers progressively started moving towards commercially available 3D cameras like the Kinect in an attempt to move closer towards absolute ubiquity. Zafrulla et al. further explore this branch and show

how a verification-based approach on the Kinect data by leveraging the Viterbi algorithm can achieve accuracies similar to accelerometers [8].

1.2 Contributions

In an attempt to create a game that is more widely deployable, this study explores if data collected from 4K HD cameras along with state-of-the-art pose detection algorithms can be used to train models that achieve high accuracies on ASL Recognition. We build upon previous HMM models by incorporating different input modalities such as features from RGB and 3D videos. We also explore and quantify data sufficiency in this space. In order to do so, we compare results from HMMs with state-of-the-art translation models like Transformers and found that HMMs outperform Transformers by over 13.1% on word accuracy. Transformers are topologically complex and often require large datasets for achieving high accuracies. The relative accuracy levels of HMMs and Transformers gives us unique insights into data sufficiency in this space. Leveraging results and insight from these experiments, we hope to develop a robust recognition engine, capable of augmenting CopyCat’s efficacy and helping deaf children develop short-term memory skills.

Chapter 2

Literature Review

Due to the lack of exposure to language acquisition, deaf children born to hearing parents often develop Language Deprivation Syndrome, or LDS, leading to lifelong challenges such as educational concerns, social isolation, and mental health complications [2]–[5]. Past research has shown that American Sign Language (ASL) based gaming interfaces have helped deaf children develop short-term memory skills and thus avoid LDS [9]. Furthermore, recent developments in computer vision for pose detection and natural language translation have made real-time ASL recognition possible [10]. However, due to the lack of large datasets, algorithmic choices and reliance on custom hardware (shown in Fig 2.1), past work has been unsuccessful at developing a portable gaming interface which deaf children can use to augment language interaction.



Figure 2.1 Originally, CopyCat used specially-designed kiosk (left) and custom sensor gloves (center) with embedded accelerometers (right) to achieve sufficient recognition accuracy for gameplay.

2.1 Language Acquisition

In an attempt to help deaf children acquire short-term language memory skills, Weaver et al. introduced CopyCat, an interactive PC game made to help deaf children acquire short-term memory skills and language abilities [6]. The game entails a quest by the main character to collect items to remedy a problem, such as rescuing kittens from a villain. Children tell the main character what to do via sign language. To quantify the impact of the game, they conducted a study at a local school of the deaf and reported significant improvements in receptive, expressive, and sentence repetition abilities. The study was composed of 12 participants, aged between 6 and 11. Participants were asked to configure plastic toys based on signed instruction (receptive); express an event depicted in stop-motion animation (expressive); and repeat a signed phrase (sentence repetition), at both the beginning and end of the study. The students who played CopyCat improved their scores on all three measures - receptive, expressive and sentence repetition - significantly more than the learners who did not, thereby demonstrating CopyCat's effectiveness. This initial prototype employed a human to manually identify whether the signing was correct or not rather than using a computational recognition system.

2.2 Pose Estimation

In this work, we primarily focus on Microsoft's Azure Kinect and Google's MediaPipe for pose estimation [11] [12].

MediaPipe is a cross-platform framework that enables developers to integrate various machine learning solutions such as face, pose, motion, object detection into their applications. We specifically focus on their pose estimation solution for detecting human skeleton and hand pose from RGB videos. The pose detection algorithm first detects the location of the human in the video using a lightweight convolutional neural net (CNN) trained to predict human midpoint and incline [13]. Using this information, they crop the frame and use a secondary CNN to predict the human skeleton [14]. For detecting hand pose, they use a similar framework where a lightweight CNN is trained to predict the location and a secondary CNN is trained to predict pose from the cropped images of the hands.

Kinect leverages depth data alongside RGB videos to extract accurate 3D joint positions

of humans in videos. Similar to MediaPipe, the Azure Kinect also uses a 2-step architecture for predicting 3D joints. Using a large backbone CNN architecture (ResNet101) appended to a feedforward neural network for extracting keypoints, they predict 2D joints for each human in the video directly. Then, they perform model fitting between the 2D joints and the depth data to produce accurate 3D joints. Rather than taking a deep learning based approach for aligning 2D joints to depth data, they leverage Kinematic Models of humans to approximate joint angles, scaling factor, and rigid transform. Morphing kinematic models according to 2D joints and depth data outputs the 3D joint information.

2.3 Multimodal Recognition

Since then, several efforts have focused on using multimodal datasets - consisting of accelerometer output, depth data, and videos - for translating and understanding American Sign Language in order to build a robust recognition system. Brashear et al. initiated these efforts by not only introducing two datasets - a vision-based dataset and an accelerometer-based dataset - but also demonstrated how Hidden Markov Models (HMM) can be used to understand American Sign Language [7]. Their vision dataset consisted of videos from a camera mounted on a cap, giving them a direct view of the signing. On the other hand, their accelerometer dataset consisted of time series data from two accelerometers embedded into wearable gloves. On the vision and accelerometer datasets alone, they achieve sentence accuracies of 52.38% and 65.87% respectively. However, when combined, their sentence accuracy increases to 90.48% on the test dataset. Unfortunately, this dependence on accelerometer data limits their scalability. These custom gloves are expensive to manufacture, deeming them unfit for ubiquitous use.

In an attempt to move towards less expensive equipment, Zafrulla et al. demonstrate ASL recognition on data collected using 3D cameras like Kinect [8]. Additionally, Zafrulla et al. also contribute a dataset containing over 1,000 American Sign Language phrases collected using Kinect. In order to understand the effects of topology better, they also compared accuracies on data collected while sitting with accuracies on data collected while standing. Building upon the work done by Brashear et al., they used HMMs along with features like hand shape, hand velocity, and acceleration for understanding ASL. On data

collected while standing, they found 36.2% sentence accuracy while on the data collected while sitting, they found 36.3% sentence accuracy. However, in order to compensate for low recognition accuracies, they introduce the idea of verifying ASL by leveraging the Viterbi algorithm. Using this verification pipeline, they were able to achieve impressive accuracies - 82.18% sentence accuracy while sitting and 80.82% while standing. However, this dependence on 3D cameras like the Kinect further limits scalability as it makes it impossible to incorporate games like CopyCat into a smartphone application.

2.4 State-of-the-art Recognition

With the advent of deep learning in pose estimation via convolutional neural networks and natural language translation using Transformers, American Sign Language Recognition using solely 2D RGB data has been made possible.

Camgöz et al. propose the use of Transformers along with Convolutional Neural Networks for embedding images to train an end-to-end pipeline for understanding German Sign Language [10]. They also explore Connectionist Temporal Classification (CTC) loss to bind the recognition and translation problems into a single unified architecture. Using this novel architecture, they report state-of-the-art results on the RWTH-PHOENIX-Weather-2014T (PHOENIX14T) dataset. They reported a 27.62% word error rate on the test split of the above dataset. Note that they do not report sentence error rates which are not only likely to be lower but are also more relevant to games like CopyCat since the primary task of the recognition engine is to tell the user if they signed the whole sentence correctly. These results are promising since the PHOENIX14T dataset is relatively large and also contains several complex signs. However, German Sign Language isn't directly transferable to American Sign Language, rendering their dataset unusable for games like CopyCat. Furthermore, all signers in the PHOENIX14T dataset are standing in front of a solid black wall, which further helps recognition results. This setup is unrealistic to assume for games like CopyCat since they expect users to sit in cluttered environments like their homes and schools.

In an attempt to strike a fine balance between lack of data and innumerable model pa-

rameters, Yin et al. introduced Segmentally Boosted Hidden Markov Models (SBHMM) for time series data recognition [15]. They train SBHMMs in 3 phases. The first phase involves training a baseline HMM, much in the same manner as Brashear et al. and Zafrulla et al. In the second phase they use the Viterbi decoding algorithm to find the optimal state transition path for each video and label each frame with a hidden state in the HMM. Using this labeling they train AdaBoost classifier ensembles and transform each frame into a new feature space using class probabilities from AdaBoost. In the third phase, they train a new HMM using this new transformed dataset and use it for predicting time series data in real-time. They report a 36.4% increase in accuracy for American Sign Language Recognition. However, their data collection process actively uses custom gloves, which are impractical for games like CopyCat as outlined above.

Chapter 3

Methods

We leverage state-of-the-art pose estimators such as MediaPipe and Kinect along with Machine Learning models like hidden Markov models (HMMs) and Transformers to perform American Sign Language Recognition [11], [12]. Along with comparing HMMs with Transformers, we also compare features extracted from MediaPipe with features extracted from Kinect to determine if Kinect’s added depth data is necessary for recognition.



Figure 3.1 Pose estimation with Azure Kinect (left, sign “in”), and MediaPipe (right, sign “alligator”) showing difficult signs

3.1 Feature Extraction

The biggest difference between MediaPipe and Kinect (shown in Fig 3.1) is the dimensionality of their features. While Kinect uses RGBD (Red, Green, Blue, Depth) data to extract 3-dimensional features, MediaPipe only uses RGB data to extract 2-dimensional features. Both models return features (location) of 32 joints, but since Kinect also returns depth, it outputs a total of 96 raw features (x, y, z for each joint) compared to

MediaPipe’s 64 (x and y only for each joint). In addition to these raw features, we post-process pose recognition to generate three additional groups of features:

Delta Features: For each raw feature, we also calculate a corresponding delta feature that describes how much it has changed in one frame.

Relative Features: We perform nose detection (both Kinect and MediaPipe facilitate nose detection) and calculate the position of each raw frame with respect to the nose. This technique is particularly helpful in generalizing over users since it eliminates the variability due to the change in absolute position.

Relative Features: We perform nose detection (both Kinect and MediaPipe facilitate nose detection) and normalize all detected features with respect to the position of the nose. This is particularly helpful in generalizing over users since it eliminates the variability due to the change in absolute position.

3.2 Recognition Models

Algorithm 1 Baum-Welch Re-estimation

```

1:  $X = \text{Features}, Y = \text{Label}, T = \text{TotalTimesteps}$ 
2: Initialize HMM Parameters  $\theta = (A, B, \pi)$  randomly
3:  $\alpha(X_0) = P[Y_0, X_0] = P[Y_0|X_0]P[X_0]$ 
4:  $\beta(X_T) = 1$ 
5: while  $i \leq \text{iterations}$  do
6:   for  $k = 0 \rightarrow T$  do
7:      $\alpha(X_k) = \sum_{X_{k-1}} \alpha(X_{k-1})P(X_k|X_{k-1})P(Y_k|X_k)$ 
8:   end for
9:   for  $k = N \rightarrow 0$  do
10:     $\beta(X_k) = \sum_{X_{k+1}} \beta(X_{k+1})P(X_{k+1}|X_k)P(Y_{k+1}|X_{k+1})$ 
11:   end for
12:    $\eta(X_k) = \frac{\alpha(X_k)\beta(X_k)}{\sum_{X_k} \alpha(X_k)\beta(X_k)}$ 
13:    $\epsilon(X_k, X_{k+1}) = \frac{\alpha(X_k)\beta(X_{k+1})P[X_{k+1}|X_k]P[Y_{k+1}|X_{k+1}]}{\sum_{X_k} \alpha(X_k)\beta(X_{k+1})P[X_{k+1}|X_k]P[Y_{k+1}|X_{k+1}]}$ 
14:    $\pi_0^* = \eta(X_0)$ 
15:    $A_{ij}^* = \frac{\sum_k \epsilon(X_k=j, X_{k-1}=i)}{\sum_k \eta(X_{k-1}=i)}$ 
16:    $B_{ij}^* = \frac{\sum_k \eta(X_k=i)1_{Y_k=j}}{\sum_k \eta(X_k=i)}$ 
17: end while

```

We compare Transformers (current state-of-the-art) with Hidden Markov Models (HMMs) for American Sign Language Recognition.

Hidden Markov Models: HMMs are best suited for ASL Recognition in this setting due to their high performance in time series pattern recognition and low training data requirements. HMMs are probabilistic models that attempt to understand Markov processes - events where the next state depends only upon the previous state. HMMs model such processes by estimating transition probabilities between hidden (unobservable) states and emission probabilities from terminal states using the observations [16]. In our application, the unobservable states correspond to the states in sign language while the observations are features extracted from pose estimation frameworks like MediaPipe and Kinect. Our dataset consists of only 3,914 videos, which we believe may be too few for training a Transformer to generalize to. We implement HMMs using the HTK Speech Recognition Toolkit [17]. We first initialize a two-mixture, 18-state HMM model for each word in our dataset by performing a flat start initialization and a deterministic grammar describing the phrases in our dataset. Then, we train each HMM for 200 epochs using the Baum-Welch algorithm (shown in algorithm 1). Every 20-25 epochs, we increase the number of mixtures in the HMM by 1. Finally, Viterbi Decoding (shown in algorithm 2) is used to find the sequence of HMM models which yields the highest observation likelihood. A left-to-right HMM topology with no skip transitions was ultimately found to produce the most effective model based on extensive experimentation.

Transformers: We compare HMMs with the transformer model proposed by Camgöz

Algorithm 2 Viterbi Decoding

```

1: create path matrix  $viterbi[N, T]$ 
2: for  $s = 1 \rightarrow N$  do
3:    $viterbi[s, 1] = \pi_s * b_s(\emptyset_1)$ 
4:    $backp[s, 1] = 0$ 
5: end for
6: for  $t = 2 \rightarrow T$  do
7:   for  $s = 1 \rightarrow N$  do
8:      $viterbi[s, t] = \max viterbi[s', t - 1]a_{s', s}b_s(\emptyset_t)$ 
9:      $backp[s, t] = \operatorname{argmax} viterbi[s', t - 1]a_{s', s}b_s(o_t)$ 
10:   end for
11: end for
12:  $bestpathprop = \max viterbi[s, T]$ 
13:  $bestpathpointer = \operatorname{argmax} viterbi[s, T]$ 
14:  $bestpath = \text{path starting at } bestpathpointer, \text{ follows } backp[] \text{ to states back in time}$ 

```

et al. for German Sign Language Recognition [10]. The Transformer is made up of

two components - an encoder which understands the input sequence and a decoder which predicts the translation [18]. Each component is made up of an embedding layer and multiple iterations of multi-head attention and feedforward layer modules. While Camgöz et al. use convolutional neural networks to embed their input videos, we directly use MediaPipe/Kinect features as embeddings for our model. Due to our small dataset, we believe that directly using a pose estimator for embedding inputs speeds up learning. The multi-head attention layer is made up of multiple scaled dot-product attention layers, each of which attend to a different combination of input sequence position. The multi-head attention layer allows the transformer to simultaneously understand how different input sequence positions interact with each other, thus enabling the transformer to learn unusual patterns in the sequences. Finally, the feed-forward network builds upon the output from the multi-head attention layers and is jointly trained over all input positions. Encoder and decoder layers are often stacked together to further expand upon the transformer's flexibility. We used PyTorch to implement the model (shown in Fig 3.2) and Google Colab to train it on the same feature files the HMMs were trained on. The model consisted of two encoder and decoder layers, four multi-head attention layers, and a 2,048 dimensional feed-forward layer. We initialize all layers using Xavier initialization and train it using the Adam optimizer [19].

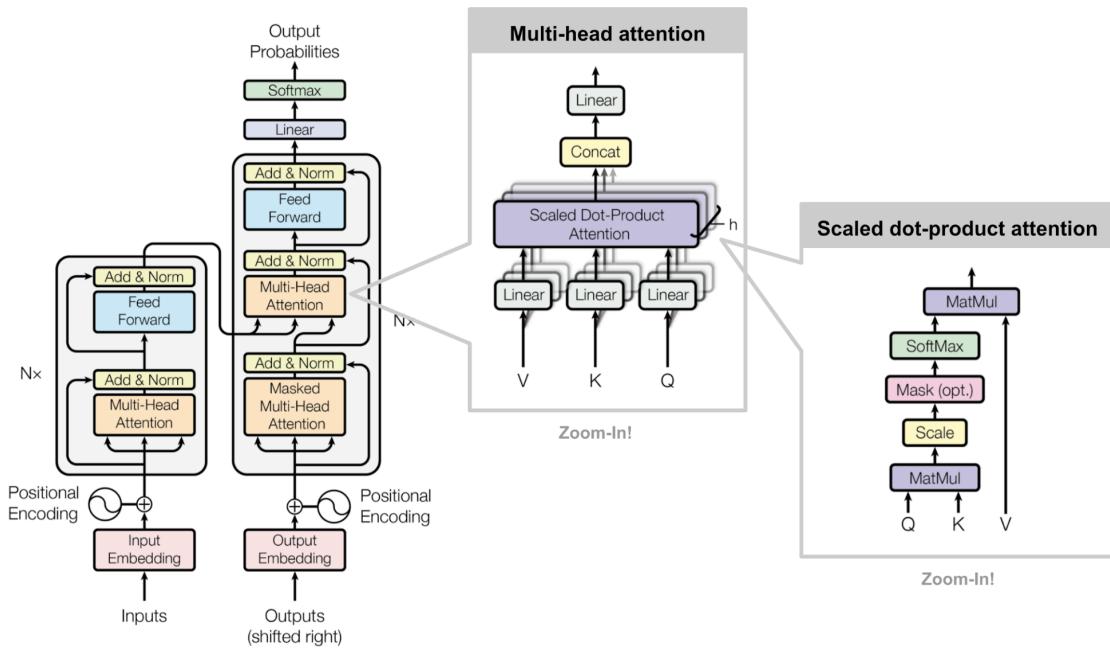


Figure 3.2 Transformer model implemented for performing ASL Recognition

3.3 Experimental Setup

We evaluate HMMs and Transformers on features from Kinect and MediaPipe in three different settings - User Dependent, User Adaptive and User Independent:

User Dependent: User Dependent refers to performing 10-fold cross-validation on each users' data. That is, for each user, we first split their data into 10 disjoint sets. Then, for each split, we take the remaining 9 splits, train each model on these splits, and evaluate on the remaining split.

User Independent: User Independent refers to performing cross-validation across users. That is, for each user A in the dataset, we train our models on the remaining users and evaluate on user A's data. This experimental setup usually results in the lowest accuracies when there is not enough data, as seen in the results section. **User Adaptive:** User Adaptive refers to adapting a User Independent model to a specific user by progressively training on the given user's videos. In doing so, the User Independent model asymptotically approaches the accuracy of User Dependent models. To simulate this, we split our complete dataset (3914 videos) into 10 splits and perform 10-fold cross-validation. Since we train and test on all users simultaneously, user-adaptive recognition rates are usually lower than User Dependent rates when we have enough data. However, the opposite is true here (as shown in the results section), further justifying our reasoning for choosing an HMM.

Chapter 4

Results and Discussions

In this section, we discuss and interpret our ASL recognition results using the methods developed above.

4.1 User Dependent and Adaptive

Models	HMMs	Transformers
Kinect	98.1 (94.9)	94.8 (91.4)
MediaPipe	98.8 (96.8)	97.4 (95.3)

(a) User adaptive

Models	HMMs	Transformers
Kinect	97.2 (91.6)	81.5 (68.9)
MediaPipe	98.1 (94.5)	91.1 (84.1)

(b) User dependent

Table 4.1: User adaptive and user dependent word (sentence) percent accuracy

Table 4.1a compares HMMs and Transformers with features from MediaPipe and Kinect in a user adaptive Setting. Note that although Kinect has access to depth features, both HMMs and Transformers achieve higher accuracies with MediaPipe. Furthermore, HMMs outperform Transformers regardless of the pose estimator. Table 4.1b further shows similar patterns in the User Dependent setting as well. However, note that HMMs and Transformers achieve higher accuracies in user adaptive setting rather than user dependent setting, indicating the lack of data since generalization is helping models perform better on native users.

4.2 User Independent

Participant	P1	P2	P3	P4	P5
Kinect	86.5 (64.4)	94.8 (82.0)	81.3 (49.7)	92.1 (77.0)	91.2 (71.4)
MediaPipe	79.6 (67.9)	94.7 (82.4)	80.6 (48.3)	98.0 (93.0)	93.6 (78.0)
	P6	P7	P8	Average	
	92.6 (77.3)	92.4 (71.4)	93.0 (79.4)	90.5 (71.6)	
	94.2 (78.3)	73.9 (29.8)	98.9 (96.6)	90.4 (71.7)	

Table 4.2: User independent word (sentence) percent accuracy using HMMs

Participant	P1	P2	P3	P4	P5
Kinect	57.0 (38.4)	88.3 (78.8)	56.6 (40.8)	77.3 (65.2)	80.0 (70.3)
MediaPipe	71.4 (60.5)	87.8 (79.4)	51.8 (32.8)	85.5 (78.8)	85.0 (75.9)
	P6	P7	P8	Average	
	79.8 (65.5)	80.7 (71.1)	85.4 (73.3)	75.7 (62.9)	
	84.7 (73.9)	65.0 (44.0)	88.9 (81.8)	77.5 (65.9)	

Table 4.3: User independent word (sentence) percent accuracy using Transformers

Table 4.2 and 4.1 show User Independent results for HMMs and Transformers respectively. HMMs were found to again outperform Transformers - by 14.8% and 12.9% for Kinect and MediaPipe respectively. We think that HMMs consistently outperform Transformers for two reasons:

Lack of Data: As shown by our earlier results, 3914 videos is not enough to learn ASL recognition even with a small vocabulary. We think this contributes significantly towards lowering recognition accuracies for Transformers since they have a large number of hyperparameters.

Grammar: HMMs can directly leverage the deterministic grammar that our dataset is built upon. However, Transformers must learn this grammar using the supplied dataset. Despite extensive training, Transformers tend to predict phrases which cannot be produced using the grammar used by our dataset.

Chapter 5

Future Work

In this chapter, we discuss opportunities for building upon the current work.

5.1 ASL Verification

Algorithm 3 Zafrulla's Verification Algorithm

```
1: for  $p = 1 \rightarrow P$  participants do
2:    $V_p$  =  $p$ 's data for validation
3:   for  $r = 1 \rightarrow R$  remaining participants do
4:      $V_r$  =  $r$ 's data for validation
5:     Train on  $N-2$  remaining participants
6:     For each instance in  $V_r$ , note average log-likelihood obtained via Viterbi alignment
7:   end for
8:   For each phrase, calculate  $\mu$  and  $\sigma$  of log-likelihood values.
9:   Calculate acceptance boundary as  $\mu - \kappa * \sigma$  where  $\kappa$  is a tunable parameter
10:  Train on remaining  $N - 1$  participants and test on  $V_p$  using computed thresholds.
11: end for
```

Past work has shown how ASL verification can be used to confirm whether signed phrases are correct or not [8]. ASL verification refers to deciding whether given a video and a phrase, the signer in the video signed the given phrase correctly. Specifically, they show that using verification achieves an improvement of over 15% correctness over ASL recognition. Building upon this idea (shown in algorithm 3), we have developed 2 algorithms for performing verification on our dataset. Zafrulla's verification algorithm uses a hard boundary for each sign over all users to decide whether a given phrase corresponds to the given label. To allow for a more dynamic decision making, we incorporate logistic

Algorithm 4 Logistic Regression for Verification

```

1: for  $p = 1 \rightarrow P$  participants do
2:    $V_p = p'$ s data for validation
3:   for  $r = 1 \rightarrow R$  remaining participants do
4:      $V_r = r'$ s data for validation
5:     Train on N-2 remaining participants
6:     For each instance in  $V_r$ , note average log-likelihood obtained via Viterbi alignment
7:     Note average log-likelihood obtained for one-off labels of each instance in  $V_r$ .
8:   end for
9:   For each phrase, train logistic classifier to classify signs.
10:  Train on remaining  $N - 1$  participants and test on  $V_p$  using trained logistic classifier.
11: end for

```

regression classifiers as shown in Algorithm 4. Specifically, using positive and negative examples of video and label pairings, we train a logistic regression classifier for each phrase. The logistic regression based verification algorithm is also limited in that it only

Algorithm 5 Neural Nets for Verification

```

1: for  $p = 1 \rightarrow P$  participants do
2:    $V_p = p'$ s data for validation
3:   for  $r = 1 \rightarrow R$  remaining participants do
4:      $V_r = r'$ s data for validation
5:     Train on N-2 remaining participants
6:     For each instance in  $V_r$ , note per sign log-likelihood obtained via Viterbi alignment
7:     Note per sign log-likelihood obtained for one-off labels of each instance in  $V_r$ .
8:   end for
9:   For each phrase, train neural net classifier to classify signs based on per sign likelihood.
10:  Train on remaining  $N - 1$  participants and test on  $V_p$  using trained neural nets.
11: end for

```

uses the average likelihood of all signs in a phrase to decide. To overcome this limitation, we propose Algorithm 5 which is trained on the log-likelihood of each sign in the given phrase. In the near future, we will be performing rigorous experimentation and analysis to determine the effectiveness of each algorithm and compare them with our recognition algorithms.

5.2 Segmentally Boosted HMMs

Segmentally Boosted Hidden Markov Models (SBHMMs) were first introduced by Yin et al. and were a breakthrough in ASL recognition as they achieved an error reduction of 36.4% on their vision dataset compared to HMMs [15]. We propose several improvements

Algorithm 6 Original Segmentally Boosted HMMs

- 1: Train HMMs using Baum Welch
 - 2: Find optimal transition path using Viterbi decoding
 - 3: Label every timestep in video with its most likely hidden state
 - 4: Train Adaboost ensembles for this labeling
 - 5: Project original data into a new feature space using the ensembles
 - 6: Train HMMs using Baum Welch in the new space
 - 7: Evaluate on test set
-

upon their algorithm (shown in Algorithm 6) and apply it to our current dataset in the User Adaptive setting. Rather than using Adaboost ensemble for learning relationship between timesteps and hidden states, we propose the use of K-Nearest Neighbors. Due to their simple structure, we propose that they would perform better in the current data poor setting. Furthermore, rather than training just one model for classifying over all hidden states, we train a KNN model for each hidden state. Lastly, we expand upon hidden states to differentiate between different positions a sign may occur at. We found that in the user-adaptive experimental setting, this model decreased sentence error by 20% compared to HMMs presented in the methods section above. However, further testing needs to be done to determine the effectiveness of this model in user-dependent and user-independent settings.

5.3 User Studies

While our current dataset is limited to 8 adults, we are hoping to extend it to 12 adults in the near future. We will use this dataset of 12 users to future improve our machine learning models and test both recognition and verification algorithms on them. Using insights from these results, we will move on to our target population - deaf children. For conducting effectiveness studies on deaf children, we will integrate our final machine learning model with our full-fledged PC game in a setting similar to the one used by Weaver et al. [6].

Chapter 6

Conclusion

The results we present above suggest that a game based on sign language recognition using only a 2D camera is feasible. We show that features extracted from MediaPipe tend to outperform features extracted from Kinect in most scenarios. Furthermore, HMMs can achieve accuracies high enough for a smooth gameplay despite the small dataset. We also show that despite high accuracies on German Sign Language, Transformers are not well suited for this task. In the near future, we hope to evaluate ASL verification and Segmental Boosting algorithms, expand our dataset to include the target population (deaf children), and perform user studies evaluating the effectiveness of the complete game.

References

- [1] R. E. Mitchell and M. Karchmer, “Chasing the mythical ten percent: Parental hearing status of deaf and hard of hearing students in the united states”, *Sign language studies*, vol. 4, no. 2, pp. 138–163, 2004. DOI: <https://doi.org/10.1353/sls.2004.0005>.
- [2] P. M. Brown and A. Cornes, “Mental health of deaf and hard-of-hearing adolescents: What the students say”, *Journal of deaf studies and deaf education*, vol. 20, no. 1, pp. 75–81, 2015. DOI: <https://doi.org/10.1093/deafed/enu031>.
- [3] R. E. Perkins-Dock, T. R. Battle, J. M. Edgerton, and J. N. McNeill, “A survey of barriers to employment for individuals who are deaf”, *JADARA*, vol. 49, no. 2, p. 3, 2015.
- [4] P. M. Sullivan and J. F. Knutson, “Maltreatment and disabilities: A population-based epidemiological study”, *Child abuse & neglect*, vol. 24, no. 10, pp. 1257–1273, 2000. DOI: [https://doi.org/10.1016/s0145-2134\(00\)00190-3](https://doi.org/10.1016/s0145-2134(00)00190-3).
- [5] O. Turner, K. Windfuhr, and N. Kapur, “Suicide in deaf populations: A literature review”, *Annals of General Psychiatry*, vol. 6, no. 1, p. 26, 2007. DOI: <https://doi.org/10.1186/1744-859X-6-26>.
- [6] K. A. Weaver, H. Hamilton, Z. Zafrulla, H. Brashear, T. Starner, P. Presti, and A. Bruckman, “Improving the language ability of deaf signing children through an interactive american sign language-based video game”, in *Proceedings of the 9th International Conference of the Learning Sciences - Volume 2*, ser. ICLS ’10, Chicago, Illinois: International Society of the Learning Sciences, 2010, pp. 306–307.
- [7] H. Brashear, T. Starner, P. Lukowicz, and H. Junker, “Using multiple sensors for mobile sign language recognition”, in *Proceedings of the 8th international ACM SIGACCESS conference on Computers and accessibility*, 2006, pp. 79–86. DOI: <https://doi.org/10.1109/iswc.2003.1241392>.
- [8] Z. Zafrulla, H. Brashear, P. Yin, P. Presti, T. Starner, and H. Hamilton, “American sign language phrase verification in an educational game for deaf children”, in *2010 20th International Conference on Pattern Recognition*, IEEE, 2010, pp. 3846–3849. DOI: <https://doi.org/10.1109/icpr.2010.937>.
- [9] J. M. Bebko, “Learning, Language, Memory, and Reading: The Role of Language Automatization and Its Impact on Complex Cognitive Activities”, *The Journal of Deaf Studies and Deaf Education*, vol. 3, no. 1, pp. 4–14, January 1998, ISSN: 1081-4159. DOI: [10.1093/oxfordjournals.deafed.a014339](https://doi.org/10.1093/oxfordjournals.deafed.a014339). eprint: <https://academic.oup.com/jdsde/article-pdf/3/1/4/1000762/3-1-4.pdf>. [Online]. Available: <https://doi.org/10.1093/oxfordjournals.deafed.a014339>.

- [10] N. C. Camgoz, O. Koller, S. Hadfield, and R. Bowden, “Sign language transformers: Joint end-to-end sign language recognition and translation”, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020. doi: 10.1109/cvpr42600.2020.01004.
- [11] David Coulter, Yijie Wang, and Phil Meadows, *Azure kinect body tracking joints*, <https://docs.microsoft.com/en-us/azure/kinect-dk/body-joints>, Accessed: 2020-01-07.
- [12] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee, W.-T. Chang, W. Hua, M. Georg, and M. Grundmann, “Mediapipe: A framework for building perception pipelines”, *CoRR*, vol. abs/1906.08172, 2019. arXiv: 1906.08172. [Online]. Available: <http://arxiv.org/abs/1906.08172>.
- [13] V. Bazarevsky, Y. Kartynnik, A. Vakunov, K. Raveendran, and M. Grundmann, *Blazeface: Sub-millisecond neural face detection on mobile gpus*, 2019. arXiv: 1907.05047 [cs.CV].
- [14] V. Bazarevsky, I. Grishchenko, K. Raveendran, T. Zhu, F. Zhang, and M. Grundmann, *Blazepose: On-device real-time body pose tracking*, 2020. arXiv: 2006.10204 [cs.CV].
- [15] P. Yin, I. Essa, T. Starner, and J. M. Rehg, “Discriminative feature selection for hidden markov models using segmental boosting”, in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008, pp. 2001–2004. doi: 10.1109/ICASSP.2008.4518031.
- [16] L. Rabiner and B. Juang, “An introduction to hidden markov models”, *IEEE ASSP Magazine*, vol. 3, no. 1, pp. 4–16, 1986. doi: 10.1109/MASSP.1986.1165342.
- [17] F. Aiman, Z. Saquib, and S. Nema, “Hidden markov model system training using htk”, in *2016 International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)*, 2016, pp. 806–809. doi: 10.1109/ICACCCT.2016.7831750.
- [18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, *Attention is all you need*, 2017. arXiv: 1706.03762 [cs.CL].
- [19] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, cite arxiv: 1412.6980 Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015, 2014. [Online]. Available: <http://arxiv.org/abs/1412.6980>.

Certificate Of Completion

Envelope Id: 60E1412EAB19416AB7B38ADDE24CF1AB

Status: Completed

Subject: Please DocuSign: Dhruva's Bachelor Thesis

Source Envelope:

Document Pages: 30

Signatures: 5

Envelope Originator:

Certificate Pages: 5

Initials: 2

Dhruva Bansal

AutoNav: Enabled

North Avenue

EnvelopeD Stamping: Disabled

Atlanta, GA 30332

Time Zone: (UTC-05:00) Eastern Time (US & Canada)

dbansal36@gatech.edu

IP Address: 128.61.55.201

Record Tracking

Status: Original

5/6/2021 1:42:40 AM

Holder: Dhruva Bansal

dbansal36@gatech.edu

Location: DocuSign

Signer Events

Dhruva Bansal

dbansal36@gatech.edu

Georgia Institute of Technology

Security Level: Email, Account Authentication
(Optional)

Signature



DocuSigned by:

Dhruva Bansal

414A1332D5C9465...

Timestamp

Sent: 5/6/2021 1:58:34 AM

Viewed: 5/6/2021 1:58:54 AM

Signed: 5/6/2021 2:00:20 AM

Electronic Record and Signature Disclosure:

Not Offered via DocuSign

Daurette Joseph

dj103@gatech.edu

Georgia Institute of Technology

Security Level: Email, Account Authentication
(Optional)



DocuSigned by:

Daurette Joseph

2BF74A231E8A462...

Sent: 5/6/2021 1:56:57 AM

Resent: 5/6/2021 2:00:23 AM

Viewed: 5/6/2021 11:53:45 AM

Signed: 5/6/2021 5:13:24 PM

Electronic Record and Signature Disclosure:

Not Offered via DocuSign

Thad Starner

ts133@gatech.edu

Professor

Georgia Institute of Technology

Security Level: Email, Account Authentication
(Optional)



DocuSigned by:

Thad Starner

30005B75D79641B...

Sent: 5/6/2021 2:00:23 AM

Viewed: 5/6/2021 9:08:39 AM

Signed: 5/6/2021 9:09:06 AM

Electronic Record and Signature Disclosure:

Not Offered via DocuSign

Thomas Ploetz

tplotz6@gatech.edu

Security Level: Email, Account Authentication
(Optional)



DocuSigned by:

Thomas Ploetz

F8EC3BF6F376413...

Sent: 5/6/2021 2:00:23 AM

Viewed: 5/6/2021 9:23:43 AM

Signed: 5/6/2021 9:23:57 AM

Electronic Record and Signature Disclosure:

Accepted: 5/6/2021 9:23:43 AM

ID: bd4282bd-8192-433c-9b4e-b8661d9d85e1

In Person Signer Events

Signature

Timestamp

Editor Delivery Events	Status	Timestamp
Agent Delivery Events	Status	Timestamp
Intermediary Delivery Events	Status	Timestamp
Certified Delivery Events	Status	Timestamp
Carbon Copy Events	Status	Timestamp
Witness Events	Signature	Timestamp
Notary Events	Signature	Timestamp
Envelope Summary Events	Status	Timestamps
Envelope Sent	Hashed/Encrypted	5/6/2021 1:56:57 AM
Certified Delivered	Security Checked	5/6/2021 9:23:43 AM
Signing Complete	Security Checked	5/6/2021 9:23:57 AM
Completed	Security Checked	5/6/2021 5:13:24 PM
Payment Events	Status	Timestamps
Electronic Record and Signature Disclosure		

ELECTRONIC RECORD AND SIGNATURE DISCLOSURE

From time to time, Georgia Institute of Technology (we, us or Company) may be required by law to provide to you certain written notices or disclosures. Described below are the terms and conditions for providing to you such notices and disclosures electronically through the DocuSign system. Please read the information below carefully and thoroughly, and if you can access this information electronically to your satisfaction and agree to this Electronic Record and Signature Disclosure (ERSD), please confirm your agreement by selecting the check-box next to 'I agree to use electronic records and signatures' before clicking 'CONTINUE' within the DocuSign system.

Getting paper copies

At any time, you may request from us a paper copy of any record provided or made available electronically to you by us. You will have the ability to download and print documents we send to you through the DocuSign system during and immediately after the signing session and, if you elect to create a DocuSign account, you may access the documents for a limited period of time (usually 30 days) after such documents are first sent to you. After such time, if you wish for us to send you paper copies of any such documents from our office to you, you will be charged a \$0.00 per-page fee. You may request delivery of such paper copies from us by following the procedure described below.

Withdrawing your consent

If you decide to receive notices and disclosures from us electronically, you may at any time change your mind and tell us that thereafter you want to receive required notices and disclosures only in paper format. How you must inform us of your decision to receive future notices and disclosure in paper format and withdraw your consent to receive notices and disclosures electronically is described below.

Consequences of changing your mind

If you elect to receive required notices and disclosures only in paper format, it will slow the speed at which we can complete certain steps in transactions with you and delivering services to you because we will need first to send the required notices or disclosures to you in paper format, and then wait until we receive back from you your acknowledgment of your receipt of such paper notices or disclosures. Further, you will no longer be able to use the DocuSign system to receive required notices and consents electronically from us or to sign electronically documents from us.

All notices and disclosures will be sent to you electronically

Unless you tell us otherwise in accordance with the procedures described herein, we will provide electronically to you through the DocuSign system all required notices, disclosures, authorizations, acknowledgements, and other documents that are required to be provided or made available to you during the course of our relationship with you. To reduce the chance of you inadvertently not receiving any notice or disclosure, we prefer to provide all of the required notices and disclosures to you by the same method and to the same address that you have given us. Thus, you can receive all the disclosures and notices electronically or in paper format through the paper mail delivery system. If you do not agree with this process, please let us know as described below. Please also see the paragraph immediately above that describes the consequences of your electing not to receive delivery of the notices and disclosures electronically from us.

How to contact Georgia Institute of Technology:

You may contact us to let us know of your changes as to how we may contact you electronically, to request paper copies of certain information from us, and to withdraw your prior consent to receive notices and disclosures electronically as follows:

To contact us by email send messages to: eis-distributed-applications@oit.gatech.edu

To advise Georgia Institute of Technology of your new email address

To let us know of a change in your email address where we should send notices and disclosures electronically to you, you must send an email message to us at docusign_admins@lists.gatech.edu and in the body of such request you must state: your previous email address, your new email address. We do not require any other information from you to change your email address.

If you created a DocuSign account, you may update it with your new email address through your account preferences.

To request paper copies from Georgia Institute of Technology

To request delivery from us of paper copies of the notices and disclosures previously provided by us to you electronically, you must send us an email to eis-distributed-applications@oit.gatech.edu and in the body of such request you must state your email address, full name, mailing address, and telephone number. We will bill you for any fees at that time, if any.

To withdraw your consent with Georgia Institute of Technology

To inform us that you no longer wish to receive future notices and disclosures in electronic format you may:

- i. decline to sign a document from within your signing session, and on the subsequent page, select the check-box indicating you wish to withdraw your consent, or you may;
- ii. send us an email to eis-distributed-applications@oit.gatech.edu and in the body of such request you must state your email, full name, mailing address, and telephone number. We do not need any other information from you to withdraw consent.. The consequences of your withdrawing consent for online documents will be that transactions may take a longer time to process..

Required hardware and software

The minimum system requirements for using the DocuSign system may change over time. The current system requirements are found here: <https://support.docusign.com/guides/signer-guide-signing-system-requirements>.

Acknowledging your access and consent to receive and sign documents electronically

To confirm to us that you can access this information electronically, which will be similar to other electronic notices and disclosures that we will provide to you, please confirm that you have read this ERSD, and (i) that you are able to print on paper or electronically save this ERSD for your future reference and access; or (ii) that you are able to email this ERSD to an email address where you will be able to print on paper or save it for your future reference and access. Further, if you consent to receiving notices and disclosures exclusively in electronic format as described herein, then select the check-box next to ‘I agree to use electronic records and signatures’ before clicking ‘CONTINUE’ within the DocuSign system.

By selecting the check-box next to ‘I agree to use electronic records and signatures’, you confirm that:

- You can access and read this Electronic Record and Signature Disclosure; and
- You can print on paper this Electronic Record and Signature Disclosure, or save or send this Electronic Record and Disclosure to a location where you can print it, for future reference and access; and
- Until or unless you notify Georgia Institute of Technology as described above, you consent to receive exclusively through electronic means all notices, disclosures, authorizations, acknowledgements, and other documents that are required to be provided or made available to you by Georgia Institute of Technology during the course of your relationship with Georgia Institute of Technology.