AUTOMATIC RECOGNITION OF CHILDREN'S TOUCHSCREEN STROKE GESTURES

By

ALEX SHAW

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

2020

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Lisa Anthony, for her continued support, encouragement, and advice throughout my PhD. I thank my co-advisor, Dr. Jaime Ruiz, as well as the rest of my committee, for their invaluable support through the process. Finally, I thank my labmates for being excellent coworkers and I thank my parents for their support through the long journey that obtaining a PhD has been.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

AUTOMATIC RECOGNITION OF CHILDREN'S TOUCHSCREEN STROKE GESTURES

By

Alex Shaw

April 2020

Chair: Lisa Anthony
Major: Computer Science

Children are increasingly using touchscreen applications in many different contexts. Smartphones, tablets, touchscreen computers, and other touchscreen devices have become commonplace, and many applications developed on these devices are designed for children. A common form of interaction in applications is gesture input, but little work has analyzed how children make gestures and how well automatic recognition algorithms are able to recognize them. In this dissertation, we begin by analyzing the ability of existing recognition algorithms to recognize children's touchscreen gestures.

Our findings show that recognition algorithms uniformly perform poorly on recognizing children's gestures. As a benchmark for future work analyzing recognition accuracy, we examined human ability to recognize children's gestures and found human accuracy was significantly better than machine accuracy, indicating potential for improvement in future recognition algorithms. To better understand why children's gestures were recognized more poorly than adults' gestures, we then analyzed the gestures through the lens of articulation features.

We first analyzed children's gestures based on a set of 22 articulation features that had been developed to characterize adults' gestures. We found a significant effect of age on the values of 18 of the 22 features. Our results showed that children were highly inconsistent in their gesturing patterns when compared to adults.

We noticed in our work on articulation features that the features were designed with well-formed gestures from adults in mind. However, children's gestures are often not well formed, so these features do not capture some of the common articulation mistakes we see in children. Thus, we developed a set of six new articulation features specifically designed to capture common patterns we observed in a set of children's gestures.

Based on our findings in our studies of recognition rates and articulation features, we offer a set of guidelines for developers creating gesture-based applications for children. Finally, we lay out several potential avenues of future work based on the findings of our recognition and feature studies.

CHAPTER 1
INTRODUCTION

With the increasing popularity of smartphones, tablets, and other touch-enabled devices, touchscreen stroke gesture[1] interaction has become an important modality for enabling natural use of touchscreen devices. Much existing work has focused on characterizing and recognizing touchscreen gestures, but most of this work has focused on adults [12, 35, 64, 84, 103, 105, 105, 110, 111]. However, children are using touchscreen devices now more than ever in a variety of contexts [29], including personal entertainment and educational purposes. With the increase in children's use of these devices, researchers have begun to study how children's interactions differ from those of adults, allowing designers to create applications specifically tailored toward children's interactions. However, while this prior work has established clear differences between children's and adults' physical touchscreen interactions, little work has focused specifically on children's gesture patterns, so designers have largely employed the same methods for recognizing children's input as have been used for recognizing adults' gestures. In this dissertation, we present a series of studies we conducted to better understand how children create touchscreen gestures, thereby allowing designers to improve children's experiences when using gesture-based applications.

In this dissertation, we show that both commonly used template-matching approaches and more advanced machine learning techniques perform worse on recognizing children's gestures than on recognizing adults' gestures. We describe a study wherein we evaluated a variety of different types of recognizers and show that in all cases, recognition accuracy for young children is poor.

After seeing the poor recognition accuracy in the results of our recognition experiments, we analyzed the gestures that were being misrecognized to better understand what issues

[1] In this work, use use the term *gesture* to a series of one or more single-finger input strokes made by a user to create a letter, number, shape, or symbol.

may be occurring. As we viewed these gestures, we noticed that many of the gestures which the recognizers misidentified were easily recognizable to us as human observers. Some gestures, however, were unrecognizable to us. If we as humans are unable to recognize these gestures, perhaps it is unreasonable to expect that a machine algorithm would be able to do so, especially given that adults have a lifetime of experience reading common handwritten letters, numbers, shapes, and symbols. After establishing that automated recognition of children's gestures was poor, we examined human ability to recognize children's gestures. We found that humans recognized the gestures significantly more accurately than machine algorithms, indicating a potential for improvement in recognition rates. After seeing the difference between machine and human recognition, we wondered what aspects of the children's gestures caused them to be recognized more poorly by machine algorithms. Specifically, we wanted to analyze the *behaviors* children exhibited which made it more difficult for machine algorithms to recognize their gestures. To help us in this analysis, we turned to articulation features to gain further insight into the ways children make their gestures.

Articulation features are designed to quantify some aspect of how a gesture is made, typically either geometrically or temporally. For example, the average speed with which the gesture is created is an articulation feature that additional information beyond what could be gleaned from simply looking at the gesture. By looking at a set of 24 previously developed articulation features from prior publications [10, 104] and calculating their values on children's gestures, we gained new insight into reasons for the poor performance of recognition algorithms. In particular, we found that children were much less consistent in the way they make their gestures in terms of several of the articulation features which directly impact recognition process, which we discuss in this dissertation.

Our work on articulation features showed clear differences between the ways children and adults make gestures. However, the articulation features we used had originally been designed with well-formed adults' gestures in mind. Children, however, often create poorly formed gestures with articulation patterns that are not captured by traditional adult-focused

articulation features. Futhermore, we found that many of the misrecognitions that occurred in our prior work could not be explained by these existing articulation features. To bridge these gaps, we developed a set of 6 new articulation features specifically designed to capture common patterns we observed in children's gestures. We calculated the values of these articulation features on children's gestures and showed how they can be leveraged to improve recognition of children's gestures as well as how they help quantify behaviors that improve our understanding of how children make touchscreen stroke gestures.

Finally, based on our work on analyzing recognition rates and developing new articulation features, we introduce a set of new design guidelines for creating gesture-based applications for children. It is our hope that application designers will adopt these suggestions to improve the experience of children using such applications. We also provide ideas for continued research in improving the state of touchscreen gesture interaction and recognition for children.

1.1 Contributions

The contributions of this dissertation include:

- Analysis of recognition rates when recognizing children's gestures with existing template-based stroke gesture recognition algorithms
- A comparison of various types of recognizers' ability to recognize children's touchscreen gestures, including template matchers, feature-based statistical classifiers, support vector machines (SVMs), hidden Markov models (HMMs), and neural networks
- A breakdown of recognition rates by age, gender, and grade level for $P, a template matching algorithm often used in the Human-Computer Interaction literature
- A comparison of human ability to recognize children's touchscreen gestures versus existing machine recognition rates
- A detailed characterization of children's touchscreen gestures through the lens of articulation features from prior work
- Six new articulation features specifically designed to capture common inconsistencies we observed in children's gestures, as well as an analysis of how these features are correlated with recognition rates

- Design implications for the creation of new gesture sets[2] and recognition algorithms for children

[2] We use the term *gesture set* to refer to the list of gesture types and the term *gesture corpus* to refer to a dataset of collected gestures.

CHAPTER 2
RELATED WORK

In this chapter, our goal is to describe the prior work that has been conducted to analyze children's touchscreen gestures to help set the stage for our work. Because much of the prior work has built on studies conducted using adults' gestures, we also discuss important research on adults' gesture interactions and how they relate to our research goals. We describe existing work on adults' gestures through the lens of how it may relate to children's gesture interactions.

It has been well documented that children's touchscreen input behaviors are not equivalent to those of adults [8, 15, 42, 102]. It has been shown that the differences in how children of different ages interact with touchscreens is not the same as the differences in the ways they use traditional mouse input [36], indicating the importance of studying touchscreen interactions in children of specific age groups for comparison as children develop their cognitive and motor skills. Most commercial hardware devices with touchscreens like iPads or Android tablets are generally designed for adults, but specifically investigating children's interaction patterns allows application designers to make smarter choices to better design their software for specific age groups of children. Prior work shows that younger children (e.g., ages 5 to 7 years old), for example, tend to be less consistent in creating gestures than older children (e.g., 8 years old and older) [24, 113]. Figure 2-1 illustrates the wide variety among gestures from children of different ages. Thus, the age group of intended users of touchscreen applications is an important factor for designers to consider. Prior studies have offered a number of guidelines for designing touchscreen applications for various ages of children [5, 8, 42, 65, 71, 95, 113]. For example, Anthony et al. [8] suggest training age-specific recognizers for recognizing children's gestures, and Woodward et al. [113] suggest using more training examples when training recognizers for younger children. We focus our discussion of existing work on touchscreen interactions and gesture recognition into several major categories: (A) developmentally appropriate prompts and feedback, (B) selection of gestures, (C) gesture elicitation, (D)

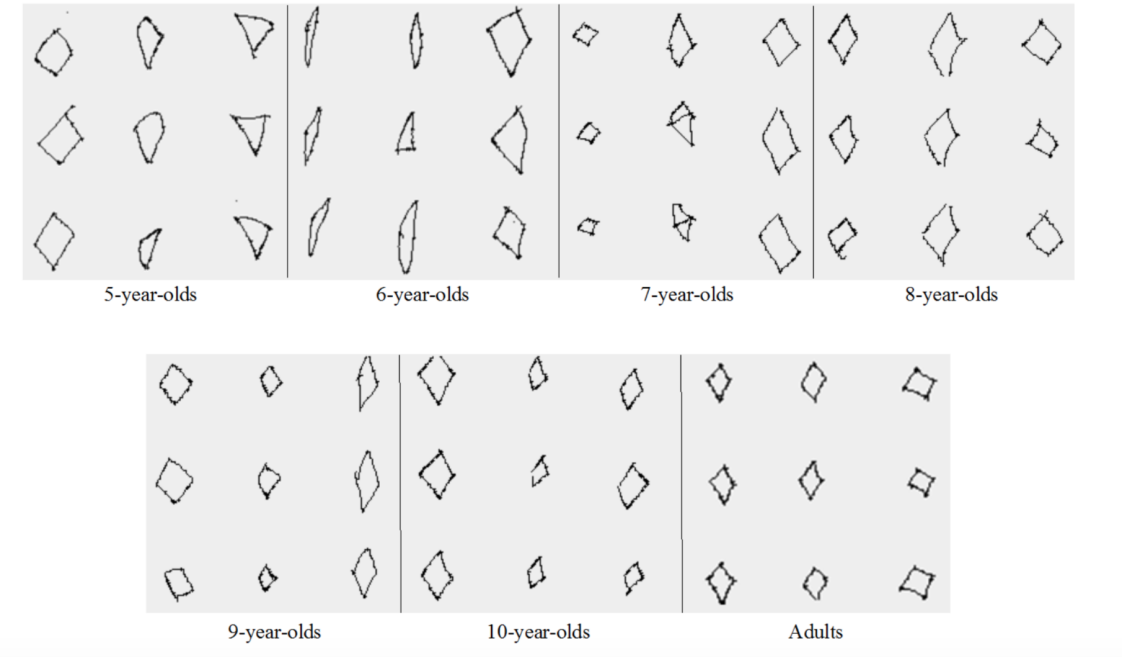

Figure 2-1: Diamond gestures produced by children ages 5 to 10 and adults from one of our studies [113]. Each column represents a different user.

recognition and classification of children's gestures, and we conclude by discussing the challenges associated with studying children's touchscreen gestures (E).

2.1 Developmentally Appropriate Prompts and Feedback

Understanding the differences in interaction patterns among children of different ages allows designers to create applications targeting specific age groups. Several studies have investigated ways in which designers can adapt their applications to make them more suitable for specific age groups of children. For example, Hiniker et al. [42] showed that designers should consider the age of children who are the target audience when prompting them to make gestures, since 2-year-olds respond better to visual cues than audio cues, but the opposite is true for 5-year-olds. McKnight and Fitton's [65] study of 6- to 7-year-olds found that they make different errors when prompted to provide different types of input, such as press and drag, select, and double click. The amount of time taken to respond to those prompts also varied significantly.

Prior work has also found that recognition of children's gestures can benefit from additional feedback when interacting with touchscreens as compared to adults. Anthony et al. [7] tested the effect of visual feedback on recognition accuracy of two different template matchers, $N-Protractor [12] and $P [103], two recognizers which are discussed in more detail later in this paper. The study found that children created gestures that were different enough to affect the accuracy of $P [103] in the presence or absence of visual feedback, but that $N-Protractor [12] did not display this sensitivity. Furthermore, a number of different features of the gestures, including the width and height of the gestures, were significantly more affected by whether there was visual feedback in the younger participants compared to the older participants. Both younger children and adults reported that they preferred gesture interactions with visual feedback to those without feedback. In the design of all of our gesture collection experiments described in this dissertation, we chose to provide visual feedback for the users creating the gestures.

2.2 Selection of Gestures

Another issue faced by application designers when creating gesture-based applications for children is that of selecting an appropriate gesture set that will be easily used by children without being constrained by its simplicity. A study by Nacher et al. [71] on multi-touch gestures in 2- to 3-year-old children suggests that children are able to perform complex gestures like rotation and scale-up. However, in contrast to Nacher's study, children ages 2 to 4 who participated in a study by Aziz et al. [1] had trouble executing free rotate, drag and drop, and pinch and spread gestures. Thus, it is unclear what level of gesture complexity is appropriate for children of varying ages.

Research with adults has also shown that users are better able to remember gesture sets that they themselves define rather than having them predefined [48, 70]. It is not clear whether this finding would extend to children, given the rapid changes in a child's memory during development, e.g., between the ages of 4 and 8 [38], and children's tendency to try novel

new gestures when interacting with new devices [85, 95]. Further work is needed to better understand the types of gestures children are best able to remember.

A number of different gesture sets have been used to test the accuracy of gesture recognition algorithms. Most have not been evaluated in recognizing children's gestures, but familiarity with the gesture sets is useful for understanding the overall state of gesture recognition. Table 2-1 shows some of these gesture sets. The Unistroke set [111] was designed specifically for testing general stroke gesture recognizers that were limited to single stroke gestures, and Anthony & Wobbrock [8] later developed a gesture set to test general multi-stroke recognizers. HHReco [45] and NicIcon [109] provide domain-specific gesture sets reflecting geometrical and safety symbols, respectively. Anthony et al.'s gesture set [12], the only set in this discussion designed specifically for kids, was created based on a survey of psychological and developmental literature as well as existing applications for children, and has been used in several studies on children's gestures including some of our own studies [23, 90, 106, 113]. In the work presented in this dissertation, we use Anthony et al.'s [8] gesture set.

Table 2-1: Examples of gesture sets employed in prior work on recognition.

Dataset	**Gesture Types**
Anthony et al. [8]	A, E, K, Q, X, 2, 4, 5, 7, 8, -, +, arch, arrowhead, checkmark, circle, rectangle, triangle, diamond, heart
Anthony & Wobbrock [11]	arrowhead, asterisk, D, exclamation point, five-pointed star, H, half-note, I, line, N, null symbol, P, pitchfork, six-pointed star, T, X
Algebra [13]	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, x, y, a, b, c, +, -, =, (,)
Unistoke [111]	triangle, X, rectangle, circle, check, caret, zig-zag, arrow, left square bracket, right square bracket, V, delete, left curly brace, right curly brace, star, pigtail
HHReco [45]	arch, callout, crescent, cube, cylinder, ellipse, heart, hexagon, parallelogram, pentagon, pentagon, square, trapezoid, triangle
NicIcon [109]	accident, bomb, car, casualty, electricity, fire, fire brigade, flood, gas, injury, paramedics, person, police,roadblock

2.3 Recognition and Classification of Children's Gestures

Early work in gesture recognition [31, 84, 111] focused on recognizing gestures produced by adults, largely overlooking children. In those studies that do examine recognition of children's gestures, the same recognition algorithms are used for the children as for the adults, even though they generally perform much worse for children's than adults' gestures [7, 113]. Because studies of adults' gestures have been so important in the development of the algorithms applied to children's gestures, this section discusses studies conducted with both children's and adults' gesture data.

Existing studies analyzing recognition rates of children's touchscreen gestures have been limited. Anthony et al.[7] examined recognition rates for children ages 10 to 17 using two popular template-based recognizers, $N and $P [12, 103]. Anthony et al. [7] found that children's gestures were recognized with the least accuracy for the youngest children, and that accuracy is higher for older children. Furthermore, Anthony et al. [7] reported a significant interaction between the user's age and the number of samples used to train recognizers with respect to the accuracy achieved. Based on this finding, the authors recommended using more training samples for younger children in order to achieve higher accuracy. However, this recommendation can present a challenge, since it can be time consuming to collect gestures from young children due to their tendency to become disinterested during laboratory studies [21]. Other methods of collecting gestures, such as longitudinal data collection outside of the lab [50], may provide an easier method of obtaining sufficient training data from children.

Another study that examined gesture recognition in children in a different context was that of Kim et al. [56] on KimCHI, a system designed to classify children's developmental skill and gender based on their execution of the digits 0 to 9 and the letters A to F. The system distinguished between preschool versus grade schoolers with 82.7% accuracy and classified the gender of the children in the study based on their sketches with 72.8% accuracy. The authors collected a total of 725 gestures from four adults, twelve 7- to 8-year-olds, and eight 3- to 4-year-olds. The system does not, however, attempt to recognize the gestures. The

distance between the age groups included in the study makes it difficult to develop a cohesive understanding to characterize children's touchscreen interactions across all ages. The gap in age groups led Kim et al. to conduct another study which built on their KimCHI framework, introducing EasySketch2 [55], an intelligent interface that combines KimCHI's developmental classifier with a gesture recognizer to help development of children's motor control. The system, which was tested on 70 children from ages 3 to 8, helped children improve their ability to draw by providing a developmentally appropriate interface with adaptive prompts and feedback; it used an algorithm developed by Valentine et al. [99] to perform recognition. The system provides feedback and a 'trace-the-dots' activity to help the child develop their skills. The system quantified children's improvement in drawing ability based on the similarity of each gesture to a predefined template. Kim et al.'s [55] system shows how recognition can be incorporated into an educational gesture-based application for children.

2.3.1 Challenges in Studying Children's Gestures

Recognition of children's gestures presents several challenges that are less problematic when dealing with adults' gestures. One such challenge is that recognition experiments generally require a large number of samples for training, which can require participants to take part in long studies. As mentioned, children are prone to lose interest and stop participating in empirical studies if they find them uninteresting [21, 80]. To deal with this issue, Brewer et al. [21] introduced a method of gamifying gesture collection in which participants were awarded points for completing individual components of the study. After completion of the study, the children had the opportunity to claim a prize, such as a small toy or stickers, based on the number of points they earned. The paper reported an increase from 73% completion without gamification up to 97% completion with gamification, in a study with children ages 5 to 7 years old, indicating a significant benefit in data fidelity can be gained by making an empirical study more engaging for children.

Beyond the challenge of collecting large numbers of samples of gestures from children, there are also challenges in dealing with the data produced by children. For example, several

prior studies have reported that younger children sometimes draw the wrong gesture or scribble randomly when prompted to draw some of the gestures [112, 113]. Papers on such studies with children have considered that giving the participants the ability to erase their gestures is not ideal, since it may encourage them to try to produce 'beautified' gestures rather than more natural ones, leading to less accurate representations of the gestures that would need to be recognized in a real application [6, 91, 113]. A method of preventing children from randomly drawing that has been employed in previous studies is to have them produce an example of each of the gestures in the corpus on a sheet of paper, which they can then use as a reference if they feel unsure [7, 21, 113], preventing the children from getting confused during gesture collection.

Children of different ages can produce very different gestures, due in part to both cognitive and motor development progress. Figure 2-1 shows an example of how drastic these variations in gestures can be, in data collected during one of our studies [113]. The diamond gestures are highly variable for the younger children, but become more consistent for older children and adults.

The developmental differences among age groups are also reflected by research on fine motor control, which shows that children rapidly develop gross motor skills during their first two years of life [73], and continue to develop fine motor skills for the next several years. From age two to seven, children reach maturity in several motor tasks, like walking and running, and they begin to exhibit more refined motor control [88] in their use of their hands, fingers, and feet. Development of these fine motor skills is not only affected by age, but a number of other factors, such as a child's experience [52]. There is a great deal of individual variation among children, indicating the importance of studying gesture patterns across ages.

The high level of variability indicates the value of studying children's gestures at the fine-grained level of individual age groups, but this is not possible in all cases due to sample size limitations. Thus, researchers may analyze gesture interactions using groupings in which children of similar developmental levels are analyzed together. Several past studies

[7, 30, 55, 91] on children's touchscreen interactions have grouped children based on Piaget's [77] theory of cognitive development, which posits that children undergo four stages of cognitive development. Stage 1 of Piaget's model is the sensorimotor stage, which begins at birth and lasts until age two. Stage 2 is the preoperational stage, which spans from age two until age seven, followed by Stage 3, the concrete operational stage, which begins at age seven and ends at age eleven. Stage 4, the final stage of the model, is the formal operational stage, spanning from age eleven to adulthood. These classifications allow for useful and interesting comparisons, but Figure 2-1 demonstrates that there are substantial differences within these groupings, indicating the importance of studying the interactions of individual ages of children.

2.3.2 Summary

In this chapter, we discussed existing work on children's and adults' touchscreen stroke gestures. We examined work on developmentally appropriate prompts and feedback, gesture set design, and recognition and classification and classification of children's gestures. In the next chapter, we continue our analysis of existing work on recognition by examining various types of recognition algorithms that have been employed in gesture recognition.

CHAPTER 3
RECOGNITION METHODS

In this chapter, we continue our examination of existing work by surveying the various types of recognizers that have been employed in prior work on classification of touchscreen gestures. Most of these algorithms from prior work can be grouped into the following major categories: (A) template matching approaches, (B) feature based statistical classifiers, (C) hidden Markov models (HMMs), (D) support vector machines (SVMs), (E) neural networks, and (F) combinations of these methods. A short description of each of these types of recognition algorithms is provided here, with some examples of some recognizers in each category, followed by a discussion of their advantages and disadvantages, and promise for recognizing children's gestures. Table 2 provides a summary of the different types of recognizers discussed in this paper.

3.1 Template Matchers

Template matching recognizers compare candidate gestures to preselected examples of the gestures and returning the closest match as the result. The members of the $-family (dollar family) of recognizers, which includes the $1, $N, and $P recognizers [11, 103, 111] are notable examples of this type of recognizer, and have been tested on children's gestures. These algorithms are popular due to being relatively easy to implement and highly accurate despite their simplicity [96]. The $1 recognizer [111], the first of the three aforementioned $-family recognizers to be developed, recognizes unistroke gestures by scaling and resampling the points of each gesture uniformly, then finding the template which minimizes the distance between each corresponding pair of points in the candidate gesture and the template gesture. The $N recognizer [11, 12], the second in the $-family, built on the limitations of the $1 recognizer, extending it to multistroke gestures. The $N recognizer works by matching stroke sequences and ordering between the candidate and template gesture. Thus, the algorithm must consider every possible ordering and direction of strokes, leading to relatively long runtimes for gestures with many strokes. For example, if a child drew an asterisk gesture with 8 strokes, the

algorithm would have to consider $8! = 40,320$ orderings of strokes and $2^8 = 256$ directions for a total of $40,320 * 256 = 10,321,920$ total combinations. The $P recognizer [103] treats each multistroke gesture as a cloud of points without regard to individual strokes. After scaling and rotating, the $P recognizer finds the best possible match between a candidate and template gesture, returning the template with the highest score as the result. $P is reported to deliver up to 99% accuracy on a corpus of 10 examples of each of 16 gesture types from 20 adults at a much lower computational cost than $N [103]. However, $P does not consider the ordering of the strokes or the direction, which may be needed to recognize some types of gestures with high accuracy (such as, for example, a line drawn left to right versus a line drawn right to left). In such cases, $N may achieve higher accuracy. Vatavu et al. [103] provide a summary of the advantages and disadvantages of the $-family recognizers in the form of a $-family cheat sheet. $1 can only recognize unistroke gestures, whereas $N and $P can recognize both unistroke and multistroke gestures. All three of the recognizers have high accuracy (>98%). The algorithmic complexity, and thus the time taken to perform the recognition, is lowest for $1, followed by $P and then $N with the highest. Thus, $1 is ideal for unistroke gestures, and $P is the best choice for multistroke gestures. However, $P is not rotationally invariant, so it cannot distinguish between A and $\forall$, for example. The lack of rotational invariance in $P implies that it can only be used in contexts with predefined gesture sets that are guaranteed not to have rotational collisions between the gestures. In these cases, $N must be preferred over $P. Rotational invariance is important in recognizing children's gestures, particularly due to their tendency to engage in mirror writing [32], a common phenomenon in which children draw the intended gesture backwards or upside down.

The popularity of the $-family has led to a number of adaptations and improvements in other work. Herold and Stahovich [41] built on the $1 recognizer to build their 1 cent recognizer, which improves runtime by providing a one-dimensional representation of the gestures. Taranta and LaViola [96] introduced a $-family inspired multistroke recognizer called Penny Pincher that achieves high accuracy even in constrained timeframes. Penny Pincher

operates by breaking gestures down into a series of two-dimensional vectors between pairs of adjacent points. The angles between corresponding vectors in test and template gestures are then used to determine the result of the recognition. In their $3 recognizer, Kratz and Rohls [57] built on the $1 recognizer to create a 3-dimensional recognizer by representing 3-dimensional gestures as continuous strokes and using a similar matching algorithm based on Euclidean distance. Another template matching algorithm was presented by Connell and Jain [31]. The algorithm operates by first reducing the gesture to a string based on the coordinates of the points in the strokes of the gesture, then performing a string-matching algorithm using a decision tree by calculating the distance between each pair of strings. The authors report a recognition rate of 86.9% accuracy on a corpus of approximately 18,000 gestures. In total, there were 36 classes of gestures collected from at least 21 users. Gestimator [115] is a template matching algorithm that focuses on recognizing more complex gestures than other template matching approaches by first segmenting the gesture into its constituent strokes, then comparing combinations of individual strokes using a more traditional template matching approach. The authors report high accuracy: 99% with 5 training examples of 6 different gesture types from each of 13 participants, but the computational overhead is increased in adding segmentation. Lee et al. [62] presented a graph-based template matching approach to symbol recognition that examined four different matching techniques. The first technique, stochastic matching, tests random combinations of matches for a fixed number of iterations. The second technique, error-driven matching, makes matches until a predetermined number of comparisons do not result in a better match than the current match. In the third technique, greedy matching, one element is successively chosen at random and the best match is found and returned. Finally, in the fourth technique, sort matching, elements are sorted then adjacent elements are paired to determine the match. The study compared the four methods of matching, finding that the sort matching approach was significantly faster but less accurate than the other three, all of which obtained over 97% matching accuracy on a corpus of 15 examples of each of 23 gesture types collected from each of 9 participants. Template

matching algorithms are usually quite simple in their implementation, making them ideal for novice programmers who wish to quickly add gesture recognition to their user interface prototypes [103, 111]. Despite their simplicity, template matching algorithms can achieve very high accuracy on adults' gestures, up to 99% in some cases, with sufficient training examples. For example, $P can reach 99% overall with 5 training examples [103]. As prior work has shown [12], template matchers are generally not well suited for recognizing children's gestures due to low recognition rates. Furthermore, template matching approaches are intended to be quick, easy substitutes for more complex recognizers when prototyping. They are not intended to be state of the art recognizers, so in this sense we expect recognition rates for other types of recognizers will be higher for children. We suggest the poor performance seen in template matchers is primarily due to the fact that template matchers rely on consistency among gestures of the same type.

3.2 Feature-based Statistical Classifiers

Feature based statistical classifiers employ a vector of features (that is, a group of metrics or measurements that are calculated on the gesture) to quantify the gesture and to use in classification. These features can take on any kind of value, but are usually numeric since they are calculations based on geometric properties. These calculated values are then compared to a predefined threshold, and the recognizer returns the best-suited match as the result. These recognizers are relatively easy to implement, but the complexity depends on the features used. One of the earliest and most well-known feature-based recognition algorithms was described by Rubine in 1991 [84]. The recognizer computes 13 geometric features, such as the sine and cosine of the initial angle and the total length of the gesture and stores them as a vector. This vector is used to compare the candidate gestures using a linear discriminator. The candidate gesture with the most similar features to the test gesture is chosen as the result. Rubine's algorithm has been used as the basis for a number of gesture-based interfaces, such as Garnet [68], Amulet [69], and gdt [64]. Cho [28] used a technique similar to Rubine in which 9 different features are used to classify hand-drawn Korean characters with over 99% accuracy.

Apte et al's [14] GDE recognizer used various features as filters to recognize multistroke geometric shapes, but suffered from the inability to recognize those same shapes when drawn in a single stroke. Smithies [94] used a feature vector of approximately 50 dimensions to recognize handwritten characters in a math equation editor. Blagojevic et al. [20] developed a set of over 100 geometric features for use in classifying gestures and sketches, then used machine learning techniques to determine which combination of features resulted in the highest accuracy rate. The authors then altered Rubine's algorithm to use the selected features rather than Rubine's original features. The majority of the most effective features selected by the machine learning algorithm were related to either the curvature of the gesture or its size. The preference for these features indicates they may have a major impact on the recognition process, which could prove useful in future work on designing new recognition algorithms.

In Olsen et al.'s [75] feature-based classifier, the extracted feature vector is a representation of the angle at which each substroke of a unistroke gesture is drawn. The feature vector is divided into six components, with the first representing the number of substrokes whose angle is between 0 and 30 relative to the horizontal, the second the number of substrokes between 30 and 60, and so on. Euclidean distance matching is then used to recognize results by comparing feature vectors of the gestures. The authors report 100% recognition accuracy with a set of 9 different gesture types over a total of 204 gestures collected from 3 users.

Feature-based classifiers benefit from their relative simplicity compared to other forms of recognizers, allowing them to achieve very fast runtimes when the number of features computed is low. However, feature-based statistical classifiers are limited in that they make the assumption that gestures can be described by a mathematical formula derived from the features chosen for recognition. In cases where the gestures fit the model very nicely, high recognition rates are likely, but when they do not fit the model the recognition rates will likely be much lower. In our work analyzing gesture articulation features of children's gestures, we found high levels of variance [91]. This high level of variance indicates that the models used by these feature-based statistical classifiers will probably not fit children's gestures well, so we

suggest that feature-based models will likely not be able to recognize children's gestures with high accuracy.

3.3 Hidden Markov Models (HMMs)

While both template matchers and feature-based statistical classifiers are relatively simple for developers to implement, we hypothesize that machine learning approaches can achieve much higher accuracy for children's gestures. However, these machine learning approaches can be prohibitive since they require much larger datasets than the previously discussed methods. Two of the most common machine learning approaches are Hidden Markov Model recognizers and Neural Networks. Hidden Markov Models (HMMs) can be used to recognize gestures by breaking the gesture into a series of points or strokes that are input to the HMM, then used to determine a recognition result based on prior training examples. HMMs describe probabilistic processes in which there are unseen (hidden) states. The HMM uses these states to compute the most likely sequence of inputs [17]. As with general Markov models, the transition from one state to the next depends on only the current state. The HMM uses a dynamic programming [19] algorithm to determine the recognition result by matching the observed sequence of inputs to the best fit among training data. This method of sequential states is useful in gesture and handwriting recognition, wherein each stroke (or letter in the case of handwriting) can be treated as a separate state. HMMs are particularly well-suited for gesture recognition because they represent a statistical model of spatio-temporal data that can handle variations in the articulation of the gesture. Li and Leung [114] presented an HMM-based recognizer that classifies constituent strokes of a gesture based on their position on the canvas, which serve as the states for the model. They report a recognition accuracy of 91% over a gesture corpus including 62 classes of letters and numbers produced by 21 people. Sezgin and Davis [89] introduced an HMM-based recognition algorithm for sketches in which, as with the previous recognizer, each stroke of the sketch is treated as a state. In Sezgin and Davis's [89] approach, however, a separate HMM represents each of the gesture types in the set. The algorithm then matches the observed sequence of states to the HMM that it most

closely resembles. The authors report 96.5% accuracy after training HMMs on 10 different gesture types with a total of 6 examples each, from a total of 10 users. Jiang and Sun's [49] stroke-based HMM approach resulted in 95% recognition after being trained on 14,611 gestures across 9 gesture types from 2 users. In contrast to other HMM-based recognizers, Anderson et al. [4] used individual points of gestures as the states in their HMM-based recognizer. The authors report an overall accuracy of 94.18% over 10 samples of 11 different gesture categories from a total of 3 participants. HMMs have also been used in handwriting at the level of recognizing words rather than characters. An example of an early system of this type is that of Chen [27], who used an HMM-based approach to recognize handwriting by first segmenting the word into individual letters then using each letter as a state in the model. The author reports 43.6% accuracy over 1,563 words from an unspecified number of users. The ability of HMM-based algorithms to recognize gestures from large corpuses has made them a popular choice for recognizing handwriting in many languages [2, 33, 46, 47, 67, 72, 83, 83]. Further information on the use of HMMs for handwriting recognition can be found in Plotz and Fink's [78] survey of the topic.

HMMs have the advantage that they can achieve high accuracy even for very large gesture corpuses and sets. However, they can incur high overhead due to the segmentation into states required to create and train the model, and training can require a large corpus of data, making their use impractical in some cases. In contrast to feature-based statistical classifiers and template matchers, HMM-based recognizers generally require that a developer have some knowledge of machine learning, thus making them less accessible to novice developers. However, we hypothesize that future work using HMM-based recognition of children's gestures may be more accurate than that of template matchers or feature-based classifiers, especially since HMMs can account for some variations in the way the gesture is articulated.

3.4 Support Vector Machines (SVMs)

In its simplest form, a support vector machine (SVM) is a binary classifier that labels data as belonging to one of two classes by finding the optimal hyperplane to separate training data [66]. A new data point can then be classified based on which side of the hyperplane it falls on. Multiple SVMs can be used in concert to perform multi-class recognition. For example, a system attempting to label input data as belonging to Class A, B, or C could operate by having an initial SVM to classify the data as belonging to A or not belonging to A, then data not classified as A could be classified using an SVM which classifies it as B or C. The number of SVMs can be extended to perform recognition for an arbitrary number of classes.

An example of a classifier that uses the multiple-SVM model is Camastra's [25] cursive character recognition system, introduced in 2007. Camastra reports higher recognition rates on a set of 57,293 characters than several popular neural networks from the time period. Bahlmann et al. [16] used dynamic time warping as part of their SVM character recognizer, reporting similar accuracy measurse as an HMM-based system when run on the UNIPEN dataset [79] of 1,364 handwritten characters from 11 adult writers. Many other recognition approaches combine SVMs with other types of recognizers, so they fall under the category of mixed methods.

Our hypothesis is that SVMs will perform about as well as feature-based recognizers for children's gestures. Ultimately, the ability to classify the input depends on the quality of the split provided by the hyperplane, which in turn is fully dependent on the feature space used.

3.5 Neural Networks

Another common machine learning based approach to recognizing gestures is through the use of neural networks. Neural networks consist of a number of nodes (or neurons) arranged in layers such that the output of one layer is received as input in the next layer, and so on until the final layer outputs a classification decision [66]. While being trained, the network compares its output to the correct output and back-propagates findings to the previous levels, allowing them to adjust the weights of the inputs used in their calculations, thereby improving

their accuracy. Each node accepts numerical input values and then produces numerical output values. This can be applied to gesture recognition by feeding features calculated on the gesture as input to the first layer of nodes. For example, a method used in recognizing handwritten digits is to convert each digit to an image, then treat it as an array of pixels where the RGB value of each pixel is sent to a different input layer. The first layer of nodes then performs computations on these values to feed to the next layer, and so on until a recognition result is reached [74]. Neural networks are well suited for achieving high accuracy rates in identifying gestures from very large sets with over 100 gesture types [118]. Neural networks can be used to recognize gestures by supplying the network with a large body of correctly identified gestures on which to train, as well as the features that should be used to determine the result. The neural network can then create a model based on the input and selected features that will classify further examples of the gestures. An early neural network-based recognizer was that of LeCun et al. [60], which showed that back-propagation could be used to recognize handwritten digits from a large number of users. The authors report 90% recognition accuracy on a set of over 10,000 digits from multiple users. Singh and Amin [93] used a neural network algorithm to recognize hand-printed characters, achieving 86% accuracy in recognizing characters from a set of 52 different characters from 21 writers by extracting primitive features such as straight lines, curves, and loops. Lee et al. [61] described a neural network recognizer that recognizes single numerals with up to 99% accuracy on a set of 22,000 gestures of 22 different types from 9 participants. Another neural network recognition algorithm, though not used for handwriting/gesture recognition, was presented by Shrivastava and Sharma [92], which classified 360 computer-generated characters from 20 different fonts with high accuracy (up to 97%). As with Singh and Amin's approach, Shrivastava and Sharma's algorithm used an image-based method that extracts features of the gestures such as vertical and horizontal symmetry. Neural networks are beneficial in that, when they are applied to large gesture corpora, they can obtain high accuracy. However, one of the main drawbacks to using neural networks, beyond the technical knowledge required to implement them, is the large amount of

data required to train the recognizer. This presents a challenge in the domain of recognizing children's gestures due to the lack of publicly available data and the difficulty involved in collecting new data from children [80].

3.6 Mixed Methods

Several recognition algorithms employ a combination of the above techniques, allowing them to capitalize on the strong points of each, but also suffering from the overhead incurred by combining them. A notable example of a combined method is Kristensson and Zhai's SHARK^2 recognizer [58, 117] for shorthand writing in pen-based computers, which uses a recognition pipeline including template pruning, that is, removing unneeded candidate gestures from the set of templates, and geometric analyses based on the shape of gestures produced when interacting with a digital keyboard. While the authors do not perform a recognition study, they do report a study showing that users are able to achieve fast word-entry rates using their system. Hong and Landay's SATIN [43] also employs this mixed methods paradigm, using ten different interpreters in concert to produce a recognition result. The authors illustrate the use of their system in a sketch-based application for drawing circuits. Because the focus of the work is on the design of the system, the authors do not perform a recognition study. Yin and Sun [116] created a novel multi-stroke recognition algorithm using a template matching method based on minimum fitting supported by optimization via dynamic programming. The authors report 98% accuracy on a set of 100 examples of each of 4 gesture types produced by 4 people. Alimoglu and Alpaydin [3] described a number of methods for incorporating multiple classifiers into a single recognizer, reporting improved rates for dynamic (mixed) recognizers over static, consistent with findings in other similar work [37, 107]. The overall accuracy rate reported is 99.3% after training on 3,748 gestures from a total of 10 users. The advantages and disadvantages of using mixed methods for gesture recognition accuracy are largely determined by the types of recognizers included. Mixed methods allow an algorithm to capitalize on the parts of an algorithm that perform best depending on the type of input. However, the need to combine various recognizers adds additional overhead. The format of

the input and output for each recognizer may also be different, so the developer may have the additional burden of converting the data between formats.

3.6.1 Summary

In this chapter, we described several categories of recognizers, including traditional template matching and feature-based approaches as well as more complex machine learning algorithms. We discussed their mechanism of operation as well as examples of existing recognizers that fall into each category. In the next chapter, we discuss our experiments using these various types of recognizers to establish recognition accuracy rates for children's touchscreen gestures.

CHAPTER 4
ESTABLISHING RECOGNITION RATES

In this chapter, we discuss our work on establishing recognition rates for children's touchscreen stroke gestures. We begin by analyzing recognition rates using a template matching algorithm, $P [103], which had been used to recognize adults' gestures. We then compare recognition rates across a variety of recognizers from the various categories discussed in Chapter 3. We then discuss another study in which we compared human ability to recognize children's touchscreen gestures with machine recognition rates to establish a target accuracy.

Prior to our work, there had been minimal work on recognition of children's touchscreen gestures. Anthony et al. [8] examined recognition of gestures from children ages 7 to 11 and adults and found significant differences in recognition when using the $N recognizer in a user-dependent fashion. Another study by Anthony et al. [12] examined recognition rates of children ages 7 to 16 years old and adults using the $N-Protractor recognizer in a user-dependent fashion. The authors found a significant difference in recognition rates, with an average rate of 90% for adults and 81% for children. These studies left questions about how well younger children's gestures would be recognized. Furthermore, no prior study had investigated between individual age groups, instead grouping children together in similar age groups (e.g. 7- to 11-year-olds versus 12- to 16-year-olds). Because understanding these differences could be important for designers of gesture-based applications targeting specific ages, we conducted a study to examine recognition rates.

4.1 Recognition with $P

To better understand how existing recognition techniques would perform on children's gestures, we conducted a study in which we collected gestures from children ages 5 to 10 years old. We chose this age group because of the rapid psychological and developmental changes children undergo in this time frame [77, 88] and because children typically begin school at age 5 in the United States, and to help fill in the gap in understanding of younger children's

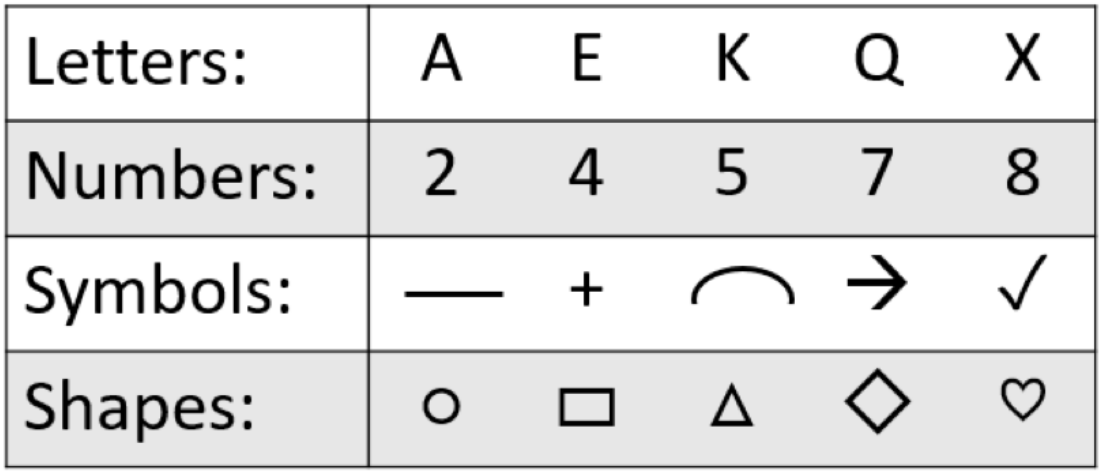

Figure 4-1: The gesture set we use in our studies.

gestures from prior studies. We also collected data from adults ages 18 and older to compare recognition rates.

Before our work, past work on touchscreen gesture recognition had primarily employed template matching algorithms [12, 98, 103, 117]. Thus, a logical next step was for us to apply these recognition algorithms to children's gestures. We found that children's gestures were not recognized as well as those of adults. We used the $P recognizer [103] since it had become a popular choice in the HCI community.

4.1.1 Gesture Set

In our experiment, we had each user produce 6 examples of each of 20 gestures. The gesture set included letters, numbers, shapes, and symbols and was designed in prior work based on a survey of developmental and psychological literature as well as the types of gestures employed in real applications for children [8, 9]. Figure 4-1 illustrates the gesture set, which we use in this study as well as the other studies presented in this dissertation.

4.1.2 Participants

The participants in our study included 30 adults and 30 children. One child was removed from the data due to incomplete data, leaving 29 children. The participants included three 5-year-olds, six 6-year-olds, four 7-year-olds, seven 8-year-olds, four 9-year-olds, and five

10-year-olds. Three children were removed due to incomplete data, leaving 26 children. In total, we had 2,600 gestures from children (26 children x 20 gesture types x 5 repetitions per gesture type).

4.1.3 Equipment

The gesture collection applications for this study were run on Samsung Google Nexus S Smartphones with the Android 4.0.4 operating system. The display resolution was 480 x 800 pixels, the pixel density was approximately 234 pixels per inch, and the phones were 4.88 x 2.48 x 0.43 inches, with a 4-inch screen.

4.1.4 Applications

The gestures in this study were collected using two different applications: one simple abstract application and one more complex, game-like application. The study was designed to investigate whether children's touchscreen interactions differ when complexity is added. Figure 4-2 shows screenshots of the two gesture collection applications. A repeated-measures ANOVA on recognition accuracy with a between-subjects factor of *age* and a within-subjects factor of *application (abstract vs. complex)* found no significant main effect of application on recognition accuracy ($F_{1,50} = 0.31$, n.s.) [113]. Because we found no significant statistical difference, we combined the gestures collected with both apps into a single dataset, which we use throughout the work presented in this dissertation.

4.1.5 Setup

Prior work on recognizing children's gestures has generally reported results in both user-dependent and user-independent scenarios. In the *user-dependent* scenario, the training and testing data come from the same user, giving an idea of how well a recognizer can perform when trained for a specific user. In the *user-independent* scenario, the gestures used to train the recognizer come from different users than the user who produces the gesture being recognized, providing a measure of how good 'out-of-the-box' accuracy can be without training on a specific user's gestures.

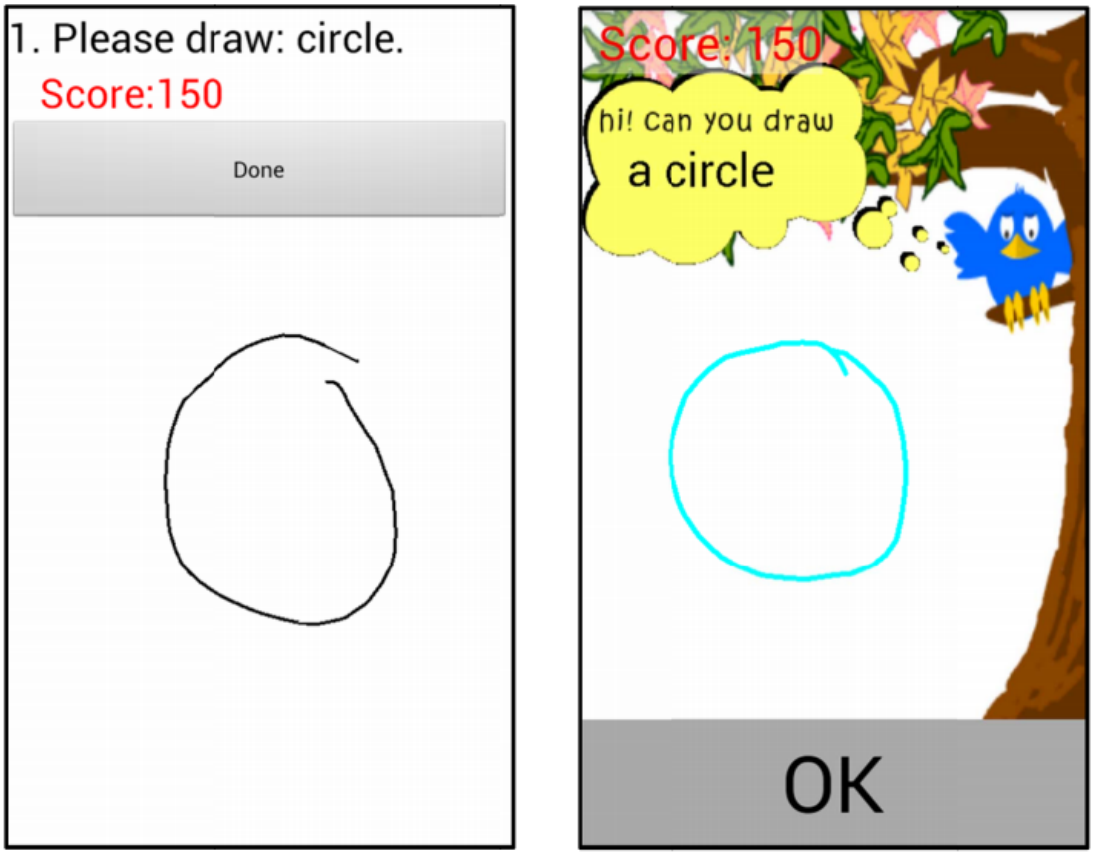

Figure 4-2: The applications used to collect gestures in our study: an abstract application (left) and a more complex, game-like application (right).

4.1.6 Results

Figure 4-3 shows the recognition rates we found in our study. Recognition rates are lowest for the youngest children and increase for older children. Recognition rates were particularly poor for 5-year-olds, at only 64.05% [SD = 29.27%], and most of the children's recognition rates were well below the 91% that children report as acceptable in handwriting recognition [81].

4.1.7 Effect of Grade Level and Gender on Recognition

When collecting the gesture data, we also asked for demographic information on the children's grade level, gender, and handedness. We analyzed the effect of grade level and gender on recognition, which we report here since a better understanding of how these factors affect recognition may be of interest to application designers and researchers.

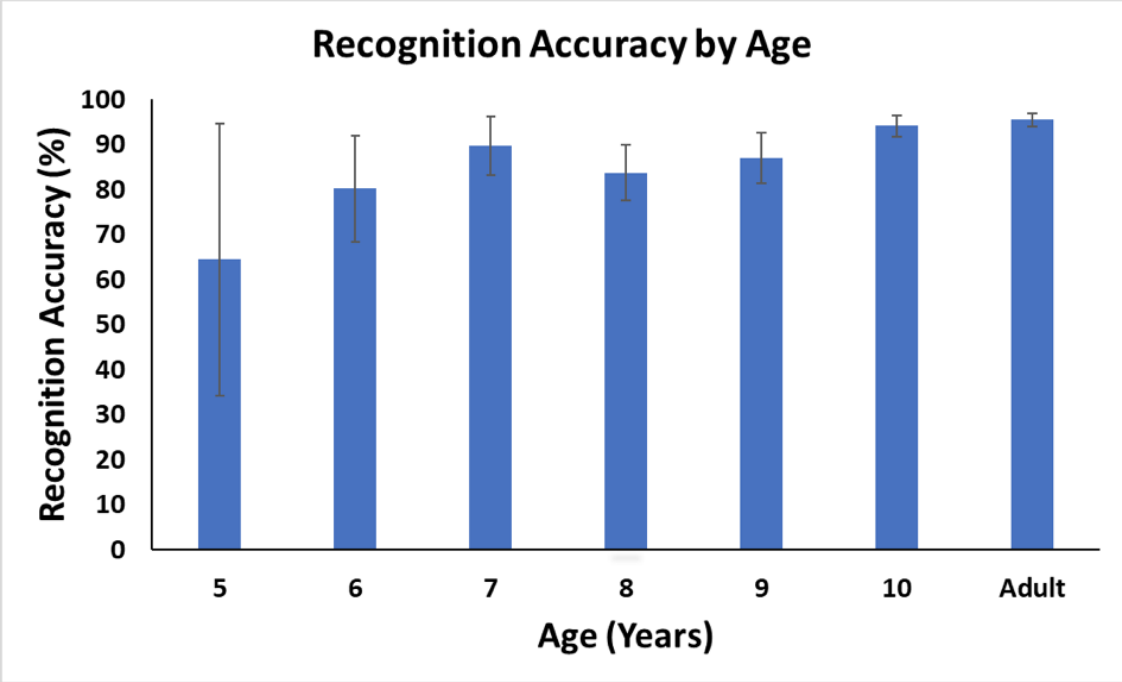

Figure 4-3: Effect of age on $P [103] recognition rates. Error bars represent the 95% confidence interval.

Grade Level. Most prior work has used age rather than grade level to group children when analyzing recognition rates. Children in the same grade are generally approximately the same age, but there are exceptions which could cause the trend to be different. A one-way ANOVA with a between-subjects factor of *grade level* on *recognition accuracy* found a marginal main effect of grade level on accuracy ($F_{4,20} = 2.685$, p $= 0.0611$). Recognition accuracy was the lowest for Pre-K, the lowest of the grade levels, and it was highest for third grade, the highest of the grade levels. Figure 4-4 shows the effect of grade level on $P recognition accuracy. We believe the reason that the test shows only marginal accuracy is due to the lower number of groupings compared to analysis of the different ages of children.

Gender and Age. Another factor we examined was gender. We were interested to see if there were differences in recognition accuracy between males and females. A two-way ANOVA with between-subjects factors of *gender* and *age* showed no significant main effect of *gender* ($F_{1,5} = 353.2$, p $= 0.160$) and no significant interaction effect between *gender* and *age* ($F_{5,13}$

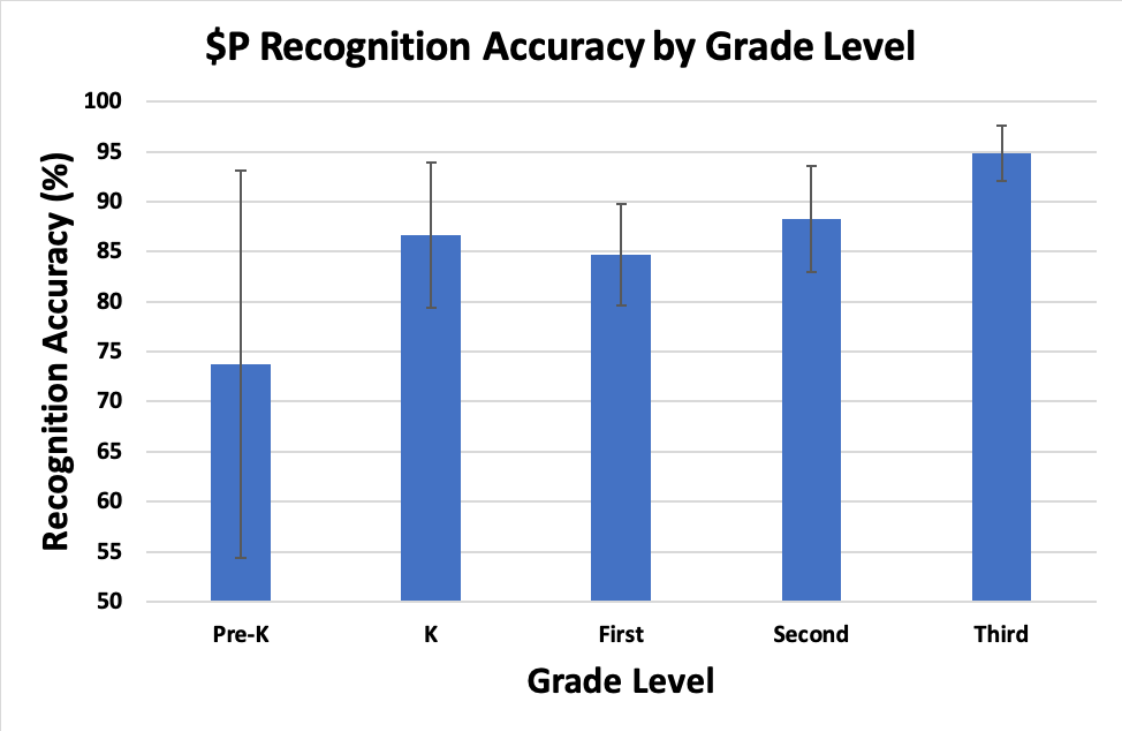

Figure 4-4: Effect of grade level on $P [103] recognition rates. Error bars represent the 95% confidence interval.

= 0.944). Figure 4-5 shows the how recognition rates vary with gender and age. Recognition rates for females are typically slightly higher regardless of age, but not significantly so.

Thus, we established that recognition rates were rather poor for children when using a popular template-matching approach. However, this is only one recognizer from one category, and it is possible that more complex machine learning methods like those described in Chapter 3 might be able to obtain higher accuracy. The additional complexity of these algorithms comes from their basis in statistical models that generally require some background in machine learning to understand. Furthermore, some of these algorithms are adaptive, meaning the parameters of the model are updated as the algorithm receives new training data. This higher level of complexity may allow for more adaptive and more accurate recognition, so we next conducted an experiment comparing different types of recognizers across these categories.

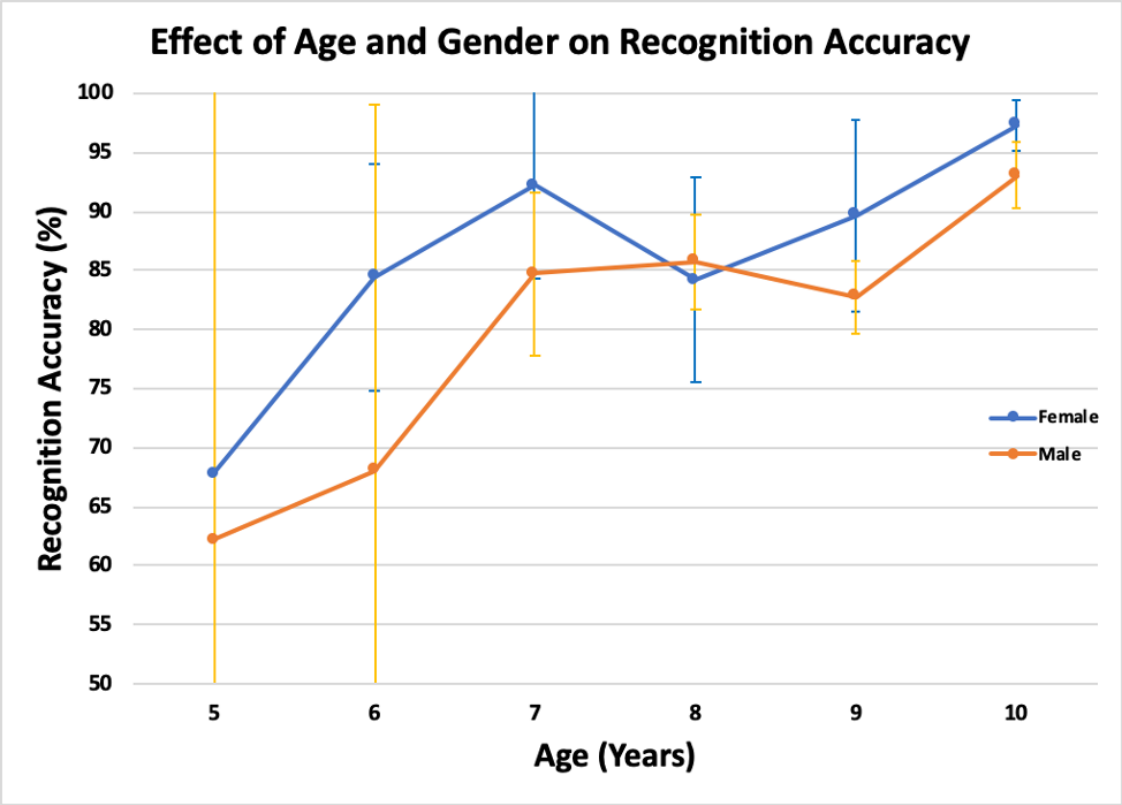

Figure 4-5: Effect of age and gender on $P [103] recognition rates. Error bars represent the 95% confidence interval.

4.2 Comparing Recognizers

To test our hypotheses regarding different categories of recognizers, we selected recognizers from the each of the categories presented in Chapter 3. We made the selections based on popularity and applicability to the problem of recognizing the gestures in our set. More specifically, we conducted a thorough literature search using the Association for Computing Machinery (ACM) Digital Library and Google Scholar on methods of recognizing touchscreen gestures. We selected one to two recognizers from each category to allow us to compare between groups. Table 4-1 shows the recognizers we selected as well as a brief description of each recognizer.

Table 4-1: The recognizers compared in our study.

A) Template Matchers	
$N-Protractor [12]	Closed-form multistroke template matching recognizer that matches constituent strokes.
$P [103]	Point-cloud multistroke template-based approach for recognizing touchscreen gestures. Unlike $N-Protractor, $P does not use stroke information, but treats the gesture as a series of points.
$P+ [101]	Point-cloud multistroke template-based approach similar to $P, but with looser point-matching allowing for many-to one matching between gestures. Designed for users with low vision.
B) Feature-Based Statistical Classifiers	
GRANDMA [84]	One of the earliest gesture recognition algorithms. Uses a set of 13 geometric and temporal features to classify candidate gestures.
GDE [14]	Multistroke recognizer using feature-based filters' to perform recognition.
Feature-based Rubine [20]	Builds on Rubine's [84] recognizer by introducing a set of 114 features and using machine learning to select the best subset to use for recognition.
C) Support Vector Machines (SVMs)	
Kato et al. [54]	SVM character recognizer.
D) Hidden Markov Models (HMMs)	
Sezgin and Davis [89]	HMM approach to segmentation and recognition of symbols.
Anderson HMM [4]	HMM approach to recognizing touchscreen gestures.
E) Neural Networks	
LeCun et al. [60]	Neural network approach for recognizing handwritten digits.
Srivastama and Sharma [92]	Neural network approach for recognizing characters.
F) Mixed Methods	
Yin and Sun [116]	Multi-stroke template matcher based on minimal fitting error, supported by optimization via dynamic programming.
Alimoglu and Alpaydin [3]	Handwritten digit recognizer that uses a neural network to select an optimal combination of four recognizers which are run simultaneously.

4.3 Results of Recognition Experiments

4.3.1 Setup of Experiments

For template matchers, we report the accuracy rates using both setups. However, because machine learning approaches require more data to obtain a reasonable accuracy rate [66], we conduct only user-independent experiments for the other recognizers. In the user-independent scenario, we use leave-one-out cross-validation (LOOCV), in which one user of the target age is chosen as the test subject and the remaining users are chosen as training participants. This process is repeated until all users have been chosen as the test participant. We then average over the accuracies for all users of each age group to get the final recognition accuracy for that age. Before running the experiments with children's gestures, we verified the correctness of our implementations of each of the recognizers by reproducing the experiments described in the papers presenting each of these recognizers using adults' gesture data. The recognition rates for children are summarized in Figure 4-6, and recognition rates for adults are discussed in Appendix A.

4.3.2 Template Matchers

$N-Protractor.

User-Dependent The average user-dependent recognition rate for $N-Protractor [12] was 56.68% [SD = 21.68%] for 5-year-olds, 71.70% [SD = 16.57%] for 6-year-olds, 83.37% [SD = 9.05%] for 7-year-olds, 79.13% [SD = 7.69%] for 8-year-olds, 84.27% [SD = 6.08%] for 9-year-olds, and 89.09% [SD = 3.30%] for 10-year-olds. A one-way ANOVA on recognition accuracy with a between-subjects factor of *age* found a significant main effect of age on recognition accuracy ($F_{5,28} = 3.667$, $p < 0.05$). A Tukey HSD post-hoc test found a significant difference in recognition accuracy between 5-year-olds and 10-year-olds. Recognition accuracy is lowest for the youngest children and increases for older children. This echoes findings in prior work that recognition rates of template matching algorithms are significantly impacted by age. Figure 4-7 shows the effect of age on user-dependent accuracy rates.

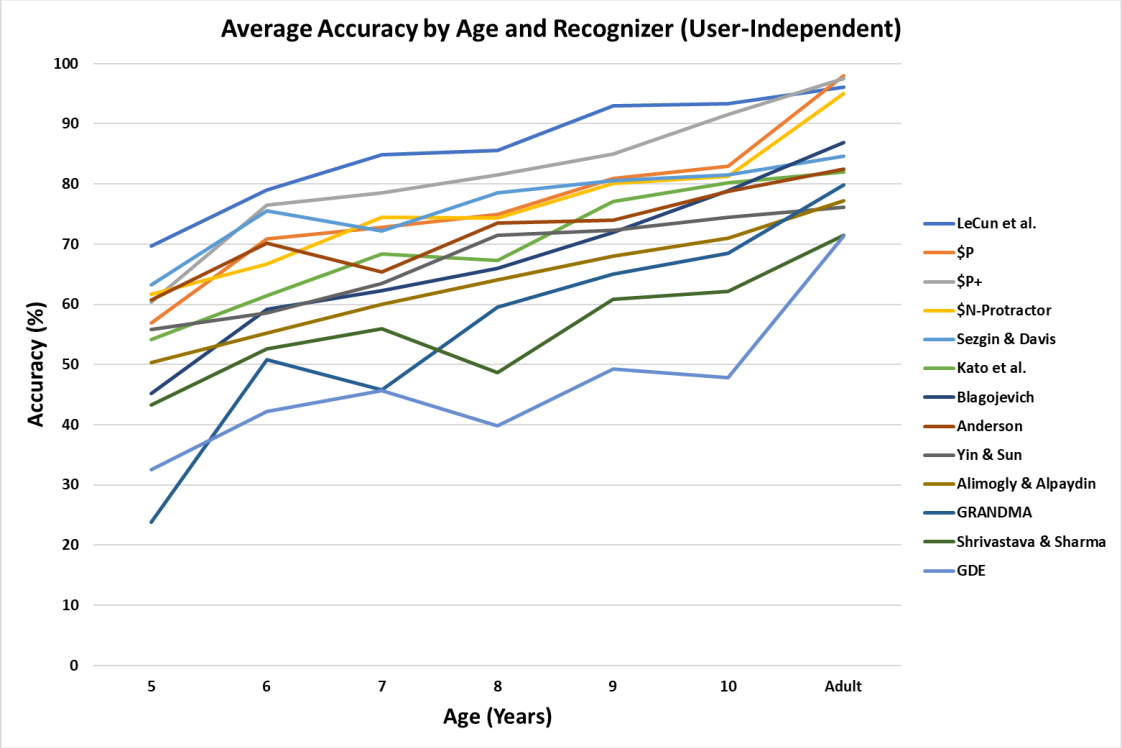

Figure 4-6: Effect of age on user-independent recognition rates for each of the recognizers in our study. Recognizers are listed roughly in order of performance, from highest accuracy to lowest.

User-Independent The average user-independent recognition rate for $N-Protractor [12] was 61.67% [SD = 18.74%] for 5-year-olds, 66.67% [SD = 14.62%] for 6-year-olds, 74.41% [SD = 12.97%] for 7-year-olds, 74.30% [SD = 8.75%] for 8-year-olds, 80.14% [SD = 4.95%] for 9-year-olds, and 81.30% [SD = 2.37%] for 10-year-olds. A one-way ANOVA on user-independent recognition accuracy with a between-subjects factor of *age* found a significant main effect of age on user-independent recognition with $N-Protractor ($F_{5,28} = 2.661$, $p < 0.05$). As in prior work, recognition rates for $N-Protractor are the lowest for the youngest children and increase for older children. Figure 4-8 shows the effect of age on recognition accuracy.

$P.

User-Dependent The average user-dependent recognition rate for $P [103] was 65.05% [SD = 29.28%] for 5-year-olds, 78.96% [SD = 12.88%] for 6-year-olds, 89.70% [SD = 6.66%]

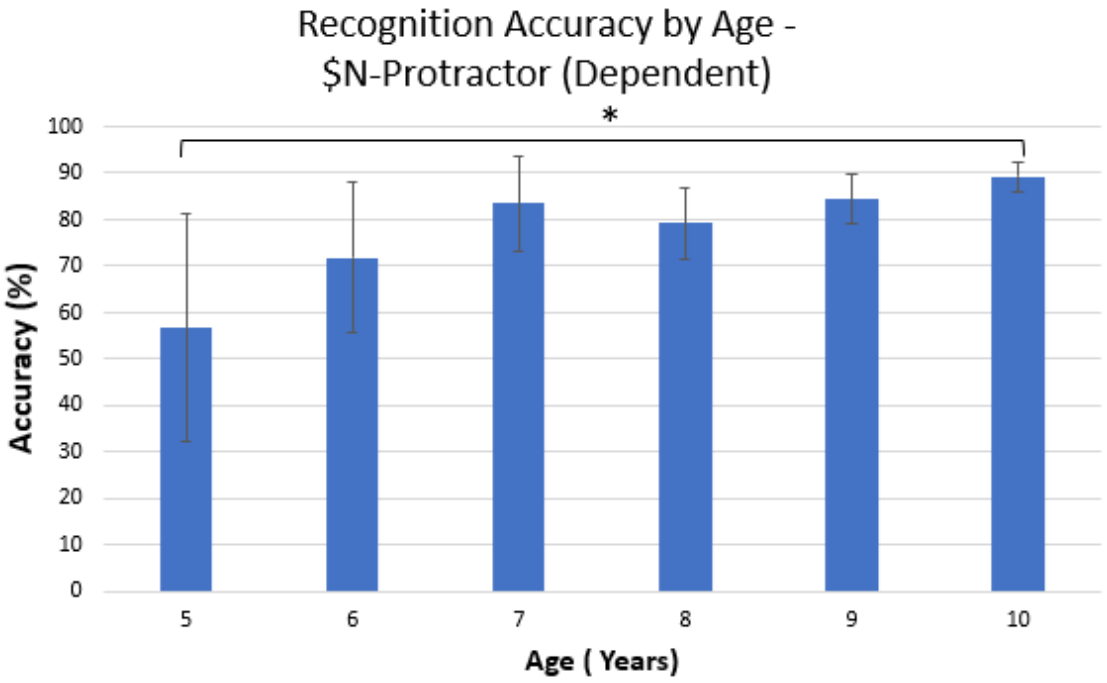

Figure 4-7: Effect of age on user-dependent recognition rates using $N-Protractor [12]. Error bars represent the 95% confidence interval.

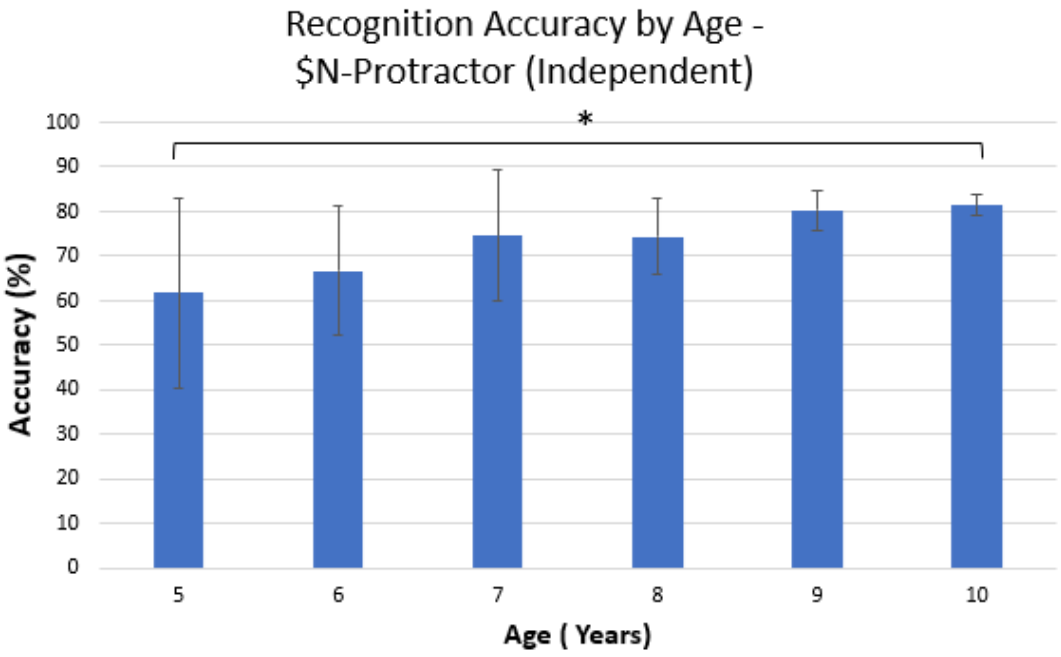

Figure 4-8: Effect of age on user-independent recognition rates using $N-Protractor [12]. Error bars represent the 95% confidence interval.

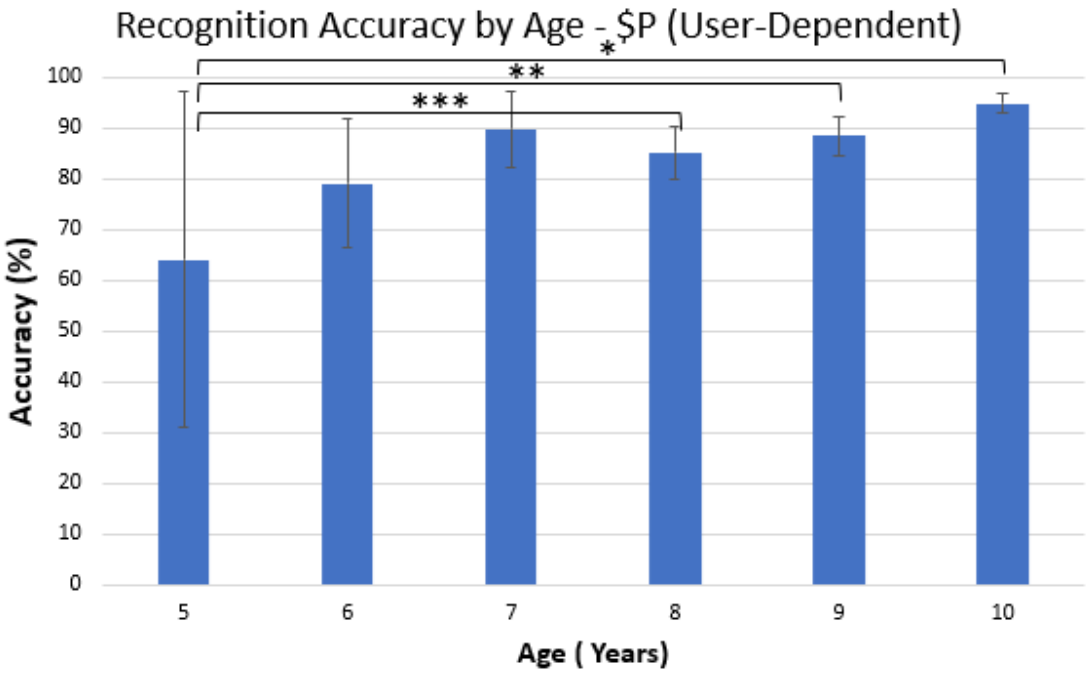

Figure 4-9: Effect of age on $P [103] on user-dependent recognition rates. Error bars represent the 95% confidence interval.

for 7-year-olds, 85.07% [SD = 5.40%] for 8-year-olds, 88.31% [SD = 4.35%] for 9-year-olds, and 94.77% [SD = 1.89%] for 10-year-olds. A one-way ANOVA on user-dependent recognition accuracy with a between-subjects factor of *age* found a significant main effect of age on recognition accuracy ($F_{5,24} = 3.105$, $p < 0.05$). A Tukey HSD post-hoc test found a significant difference between recognition rates for 5-year-olds and 8-year-olds ($p < 0.05$), between 5-year-olds and 9-year-olds ($p < 0.05$), and between 5-year-olds and 10-year-olds ($p < 0.05$). Recognition is lowest for the youngest children and increases for older children. This confirms the established pattern in prior work that recognition rates are affected by age. Figure 4-9 shows the recognition rates for $P in the user-dependent case.

User-Independent The average user-independent recognition rate for $P [103] was 56.90% [SD = 19.78%] for 5-year-olds, 70.92% [SD = 14.73%] for 6-year-olds, 72.82% [SD = 7.59%] for 7-year-olds, 74.89% [SD = 4.42%] for 8-year-olds, 80.97% [SD = 3.71%] for 9-year-olds, and 82.96% [SD = 2.01%] for 10-year-olds. A one-way ANOVA on user-independent recognition accuracy for $P with a between-subjects factor of *age* found a significant main

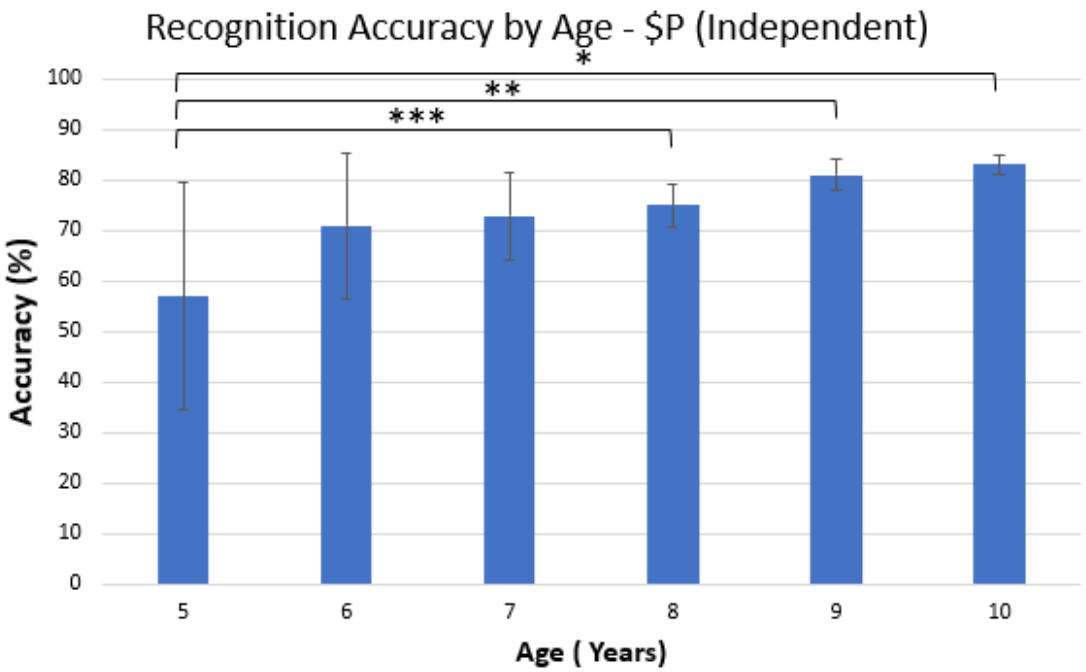

Figure 4-10: Effect of age on $P [103] for user-independent recognition rates. Error bars represent the 95% confidence interval.

effect of age on recognition accuracy ($F_{5,24} = 2.98$, $p < 0.05$). Recognition is lowest for the youngest children and increases for older children, as shown in prior work. A Tukey HSD post-hoc test found a significant difference between recognition rates for 5-year-olds and 8-year-olds ($p < 0.05$), between 5-year-olds and 9-year-olds ($p < 0.05$), and between 5-year-olds and 10-year-olds ($p < 0.05$). Figure 4-10 shows the effect of age on recognition accuracy.

$P+. *User-Dependent* The average user-dependent recognition rate for $P+ [101] was 69.03% [SD = 22.36%] for 5-year-olds, 85.45% [SD = 11.83%] for 6-year-olds, 94.37% [SD = 7.58%] for 7-year-olds, 92.54% [SD = 8.27%] for 8-year-olds, 94.07% [SD = 6.08%] for 9-year-olds, and 97.24% [SD = 2.10%] for 10-year-olds. A one-way ANOVA on user-dependent recognition accuracy with a between-subjects factor of *age* found a significant main effect of age on recognition accuracy ($F_{5,23} = 3.07$, $p < 0.05$). Recognition rates are lowest for the youngest children and highest for the oldest children, as with the other template matchers. A Tukey HSD post-hoc test found a significant difference between

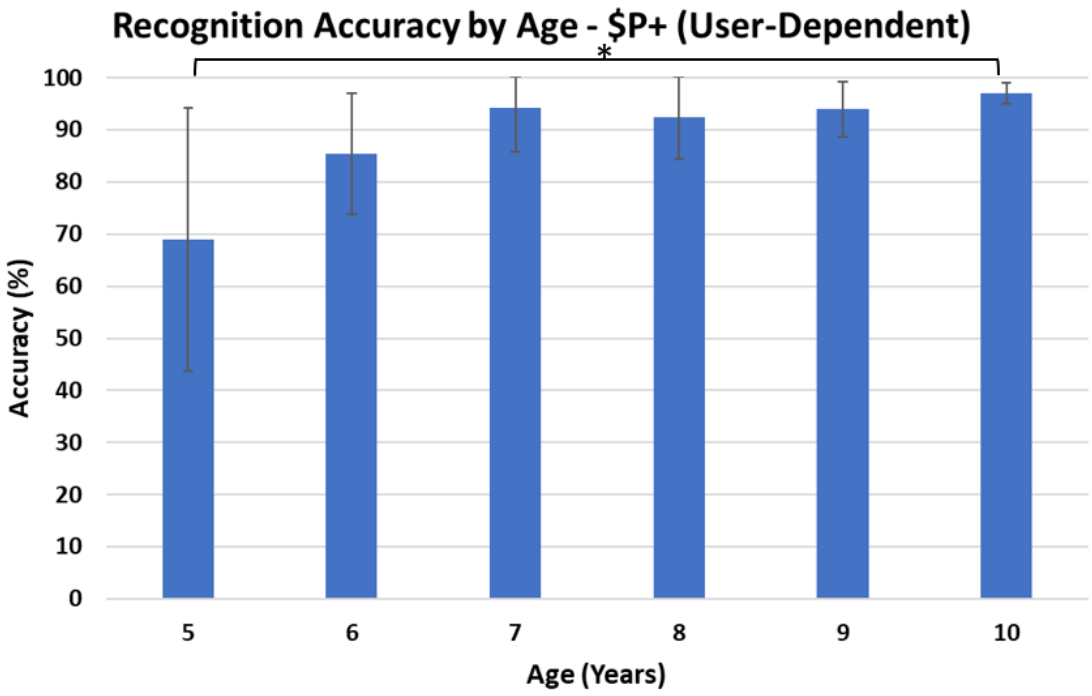

Figure 4-11: Effect of age on $P+ [101] for user-independent recognition rates. Error bars represent the 95% confidence interval.

recognition rates between 5-year-olds and 10-year-olds ($p < 0.05$). Figure 4-11 shows the effect of age on recognition accuracy for the user-dependent case with $P+.

User-Independent The average user-independent recognition rate for $P+ [101] was 60.38% [SD = 23.46%] for 5-year-olds, 76.52% [SD = 12.89%] for 6-year-olds, 78.52% [SD = 7.28%] for 7-year-olds, 81.54% [SD = 8.50%] for 8-year-olds, 85.04% [SD = 7.40%] for 9-year-olds, and 91.53% [SD = 3.32%] for 10-year-olds. A one-way ANOVA on user-independent recognition accuracy with a between-subjects factor of *age* found a significant main effect of age on recognition accuracy ($F_{5,23} = 3.14$, $p < 0.05$). Recognition rates are lowest for the youngest children and increase for older children as with the user-dependent case. A Tukey HSD post-hoc test found a significant difference between recognition rates for 5-year-olds and 10-year olds ($p < 0.05$). Figure 4-12 shows the effect of age on user-independent recognition accuracy.

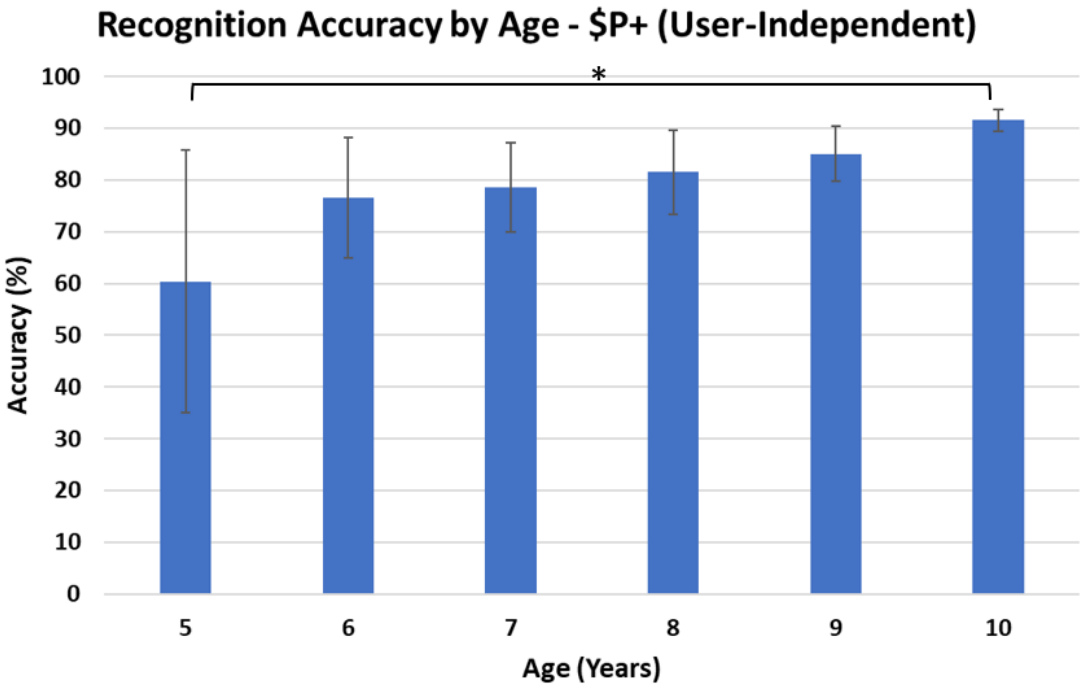

Figure 4-12: Effect of age on $P+ [101] for user-independent recognition rates. Error bars represent the 95% confidence interval.

4.3.3 Feature-Based Statistical Classifiers

GDE. The average recognition rate for GDE [14] was 32.50% [SD = 19.92%] for 5-year-olds, 42.26% [SD = 17.59%] for 6-year-olds, 45.67% [SD = 11.70%] for 7-year-olds, 39.83% [SD = 12.47%] for 8-year-olds, 49.20% [SD = 10.67%] for 9-year-olds, and 47.79% [SD = 6.63%] for 10-year-olds. A one-way ANOVA on recognition accuracy with a between-subjects factor of *age* found no significant effect of age on accuracy ($F_{5,28} = 1.071$, *n.s.*). While the recognition accuracy exhibits a similar trend as with other recognizers, with the youngest children having the lowest accuracy and the oldest children having the highest accuracy, the differences are not statistically significant. Note that because GDE was designed to recognize a limited set of gestures, we only used shapes and symbols in the experiments with GDE, removing all letters and numbers. Figure 4-13 shows the effect of age on recognition accuracy for GDE.

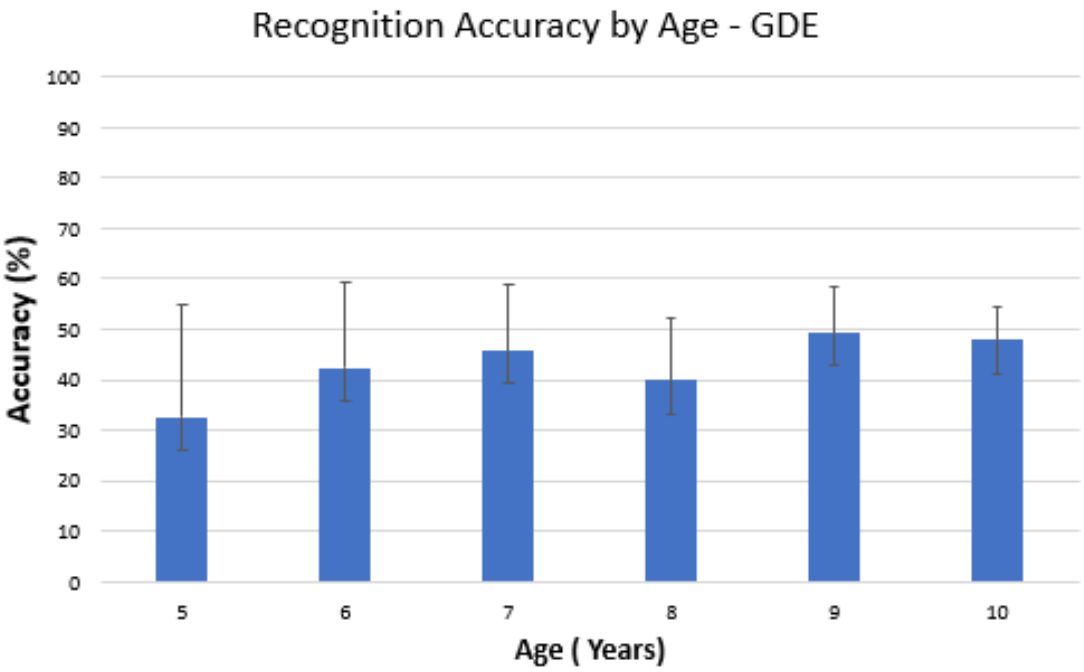

Figure 4-13: Effect of age on GDE [14] recognition rates. Error bars represent the 95% confidence interval.

GRANDMA. The average recognition rate for GRANDMA [84] was 23.80% [SD = 19.53%] for 5-year-olds, 50.87% [SD = 15.71%] for 6-year-olds, 45.77% [SD = 12.55%] for 7-year-olds, 59.53% [SD = 15.70%] for 8-year-olds, 65.00% [SD = 15.01%] for 9-year-olds, and 68.44% [SD = 10.34%] for 10-year-olds. A one-way ANOVA on recognition accuracy with a between-subjects factor of *age* found a significant effect of age on accuracy ($F_{5,28}$ = 7.061, p $<$ 0.05). Recognition accuracy for GRANDMA is lowest for the youngest children and increases for older children. Figure 4-14 shows the recognition rates by age for the GRANDMA recognizer.

Blagojevic. The average recognition rate for Blagojevic [20] was 32.50% [SD = 19.92%] for 5-year-olds, 42.26% [SD = 17.59%] for 6-year-olds, 45.67% [SD = 11.70%] for 7-year-olds, 39.83% [SD = 12.47%] for 8-year-olds, 49.20% [SD = 10.67%] for 9-year-olds, and 47.79% [SD = 6.63%] for 10-year-olds. A one-way ANOVA on recognition accuracy with a between-subjects factor of *age* found a significant effect of age on recognition accuracy ($F_5, 28$ = 3.105, p $<$ 0.05). A Tukey HSD post-hoc test found a significant difference in

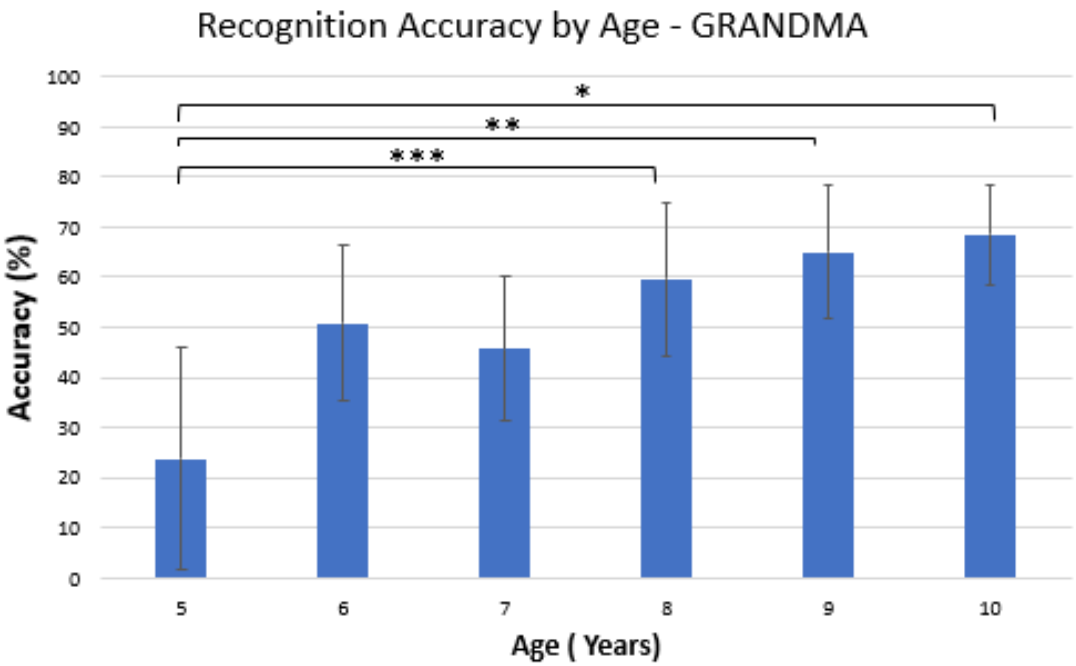

Figure 4-14: Effect of age on Rubine's GRANDMA [84] recognition rates. Error bars represent the 95% confidence interval.

recognition rates between 5-year-olds and 9-year-olds and between 5-year-olds and 10-year-olds. As with most of the other recognizers in our study, recognition accuracy is lowest for the youngest children and increases for older children. Figure 4-15 shows the effect of age on recognition accuracy for Blagojevic's recognizer.

4.3.4 Support Vector Machines (SVMs)

Kato et al. The average recognition rate for Kato et al. [54] was 54.17% [SD = 15.79%] for 5-year-olds, 61.45% [SD = 14.62%] for 6-year-olds, 68.33% [SD = 9.29%] for 7-year-olds, 67.26% [SD = 11.44%] for 8-year-olds, 66.10% [SD = 9.40%] for 9-year-olds, and 80.20% [SD = 4.63%] for 10-year-olds. A one-way ANOVA on recognition accuracy with a between-subjects factor of *age* found a significant main effect of age on recognition accuracy ($F_{5,28} = 2.895$, $p < 0.05$). A Tukey HSD post-hoc test found a significant difference in accuracy rates between 5-year-olds and 10-year-olds ($p < 0.05$). Recognition accuracy is lowest for the youngest children and increases for older children. The trend in SVM recognition

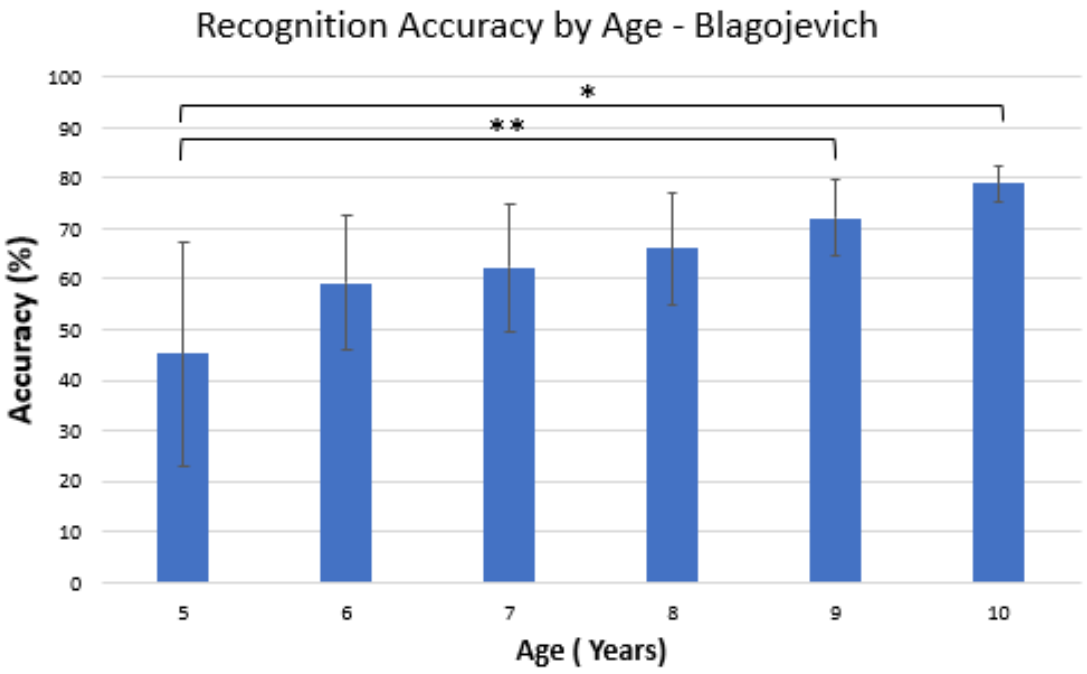

Figure 4-15: Effect of age on Blagojevic [20] recognition rates. Error bars represent the 95% confidence interval.

is similar to that observed previously in template matchers. Figure 4-16 shows the effect of age on recognition accuracy.

4.3.5 Hidden Markov Models (HMMs)

Sezgin and Davis. The average recognition rate for Sezgin and Davis [89] was 63.26% [SD = 20.61%] for 5-year-olds, 75.58% [SD = 15.71%] for 6-year-olds, 72.24% [SD = 11.12%] for 7-year-olds, 78.53% [SD = 9.78%] for 8-year-olds, 80.57% [SD = 7.34%] for 9-year-olds, and 81.53% [SD = 4.32%] for 10-year-olds. A one-way ANOVA on recognition accuracy with a between-subjects factor of *age* found a significant main effect of age on recognition accuracy ($F_{5,28} = 2.972$, $p < 0.05$). A Tukey HSD post-hoc test found a significant difference between 5-year-olds and 10-year-olds. Recognition is lowest for the youngest children and increases for older children. Thus the recognizer follows the the established trend of recognition increasing with age group. Figure 4-17 shows the effect of age on recognition accuracy.

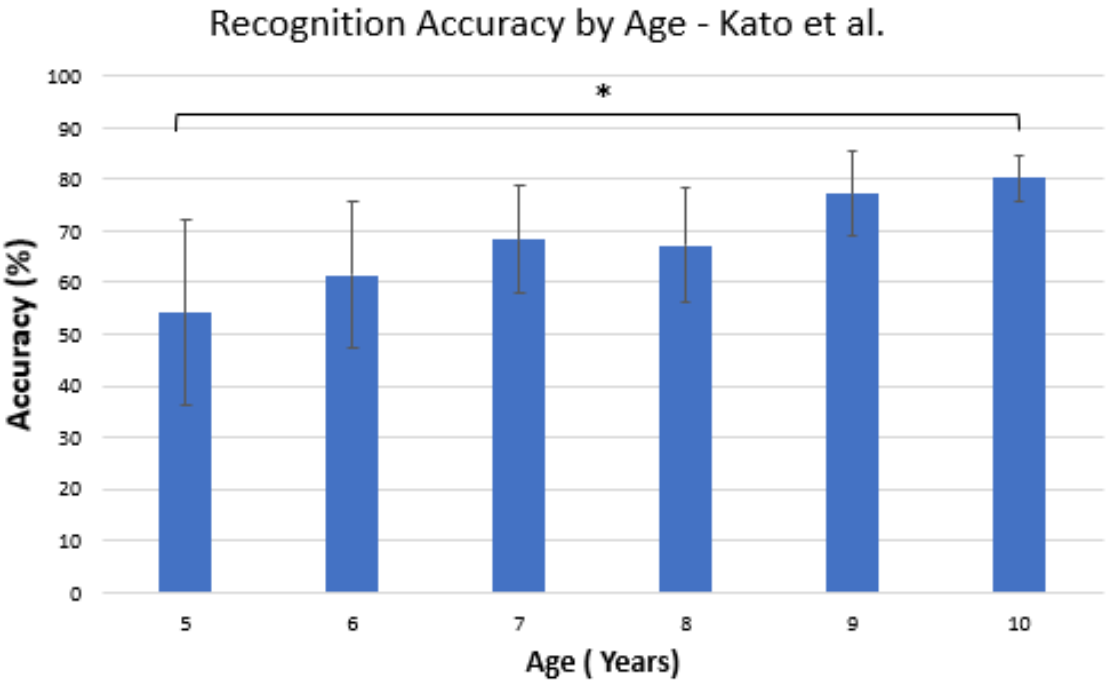

Figure 4-16: Effect of age on Kato [54] recognition rates. Error bars represent the 95% confidence interval.

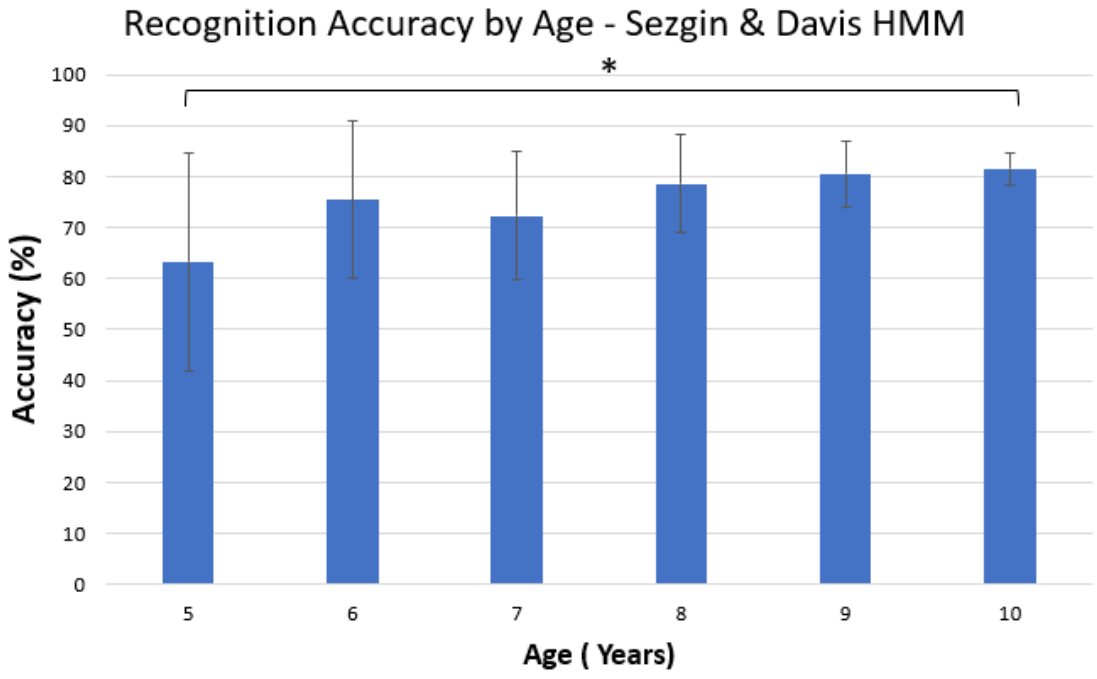

Figure 4-17: Effect of age on Sezgin & Davis HMM [89] recognition rates. Error bars represent the 95% confidence interval.

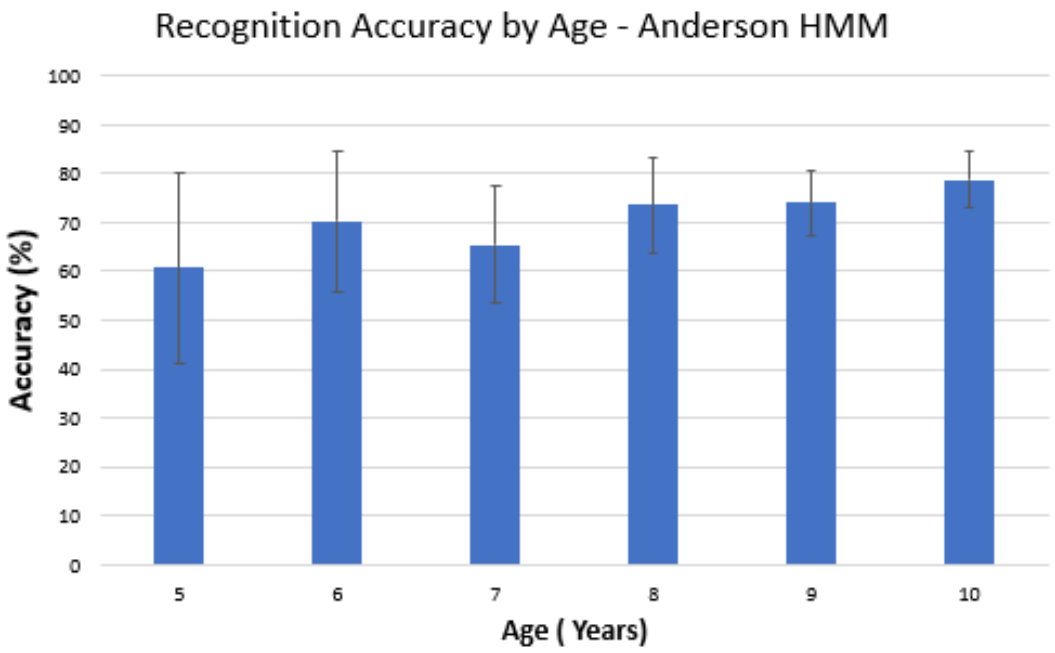

Figure 4-18: Effect of age on Anderson HMM [4] recognition rates. Error bars represent the 95% confidence interval.

Anderson. The average recognition rate for Anderson [4] was 60.67% [SD = 17.26%] for 5-year-olds, 70.19% [SD = 14.67%] for 6-year-olds, 65.33% [SD = 10.60%] for 7-year-olds, 73.54% [SD = 10.11%] for 8-year-olds, 74.01% [SD = 7.41%] for 9-year-olds, and 78.80% [SD = 5.74%] for 10-year-olds. A one-way ANOVA on recognition accuracy with a between-subjects factor of *age* found no significant effect of age on recognition accuracy ($F_{5,28} = 1.175$, *n.s.*). However, the absolute accuracy is lowest for the youngest children as with the other recognizers. Figure 4-18 shows the effect of age on recognition for the recognizer.

4.3.6 Neural Networks

LeCun et al. The average recognition rate for LeCun et al. [60] was 69.64% [SD = 22.53%] for 5-year-olds, 78.98% [SD = 18.73%] for 6-year-olds, 84.86% [SD = 6.03%] for 7-year-olds, 85.53% [SD = 4.94%] for 8-year-olds, 93.00% [SD = 3.72%] for 9-year-olds, and 93.29% [SD = 2.31%] for 10-year-olds. A one-way ANOVA on recognition accuracy with a between-subjects factor of *age* found a significant main effect of age on recognition

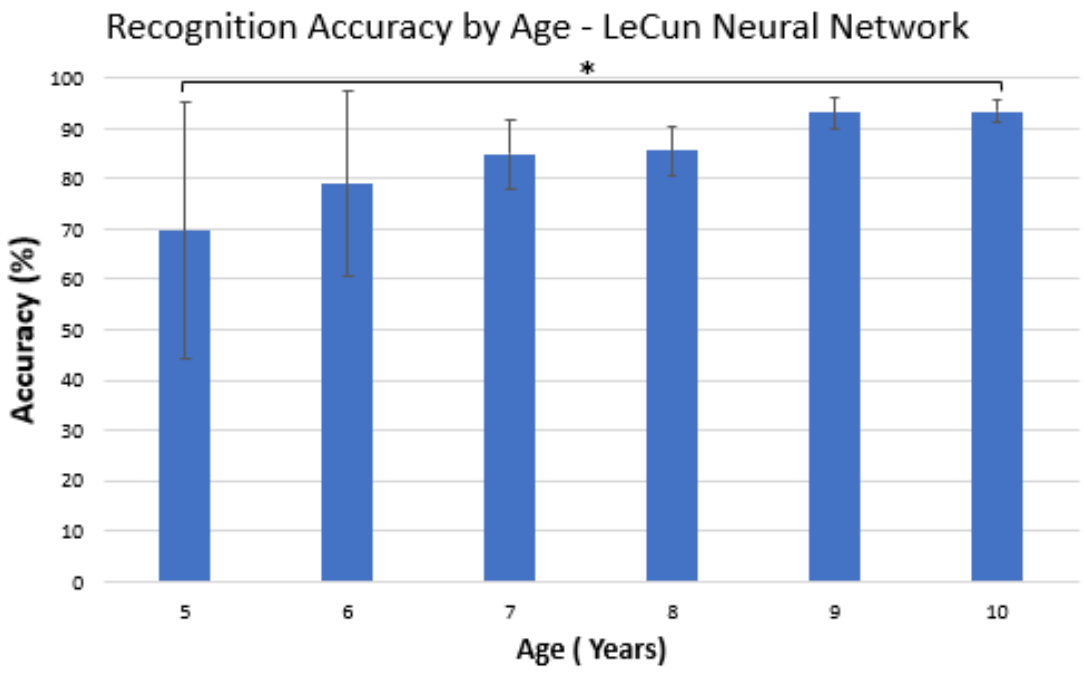

Figure 4-19: Effect of age on Lecun et al. [60] recognition rates. Error bars represent the 95% confidence interval.

accuracy ($F_{5,28} = 2.814$, $p < 0.05$). Recognition accuracy is the lowest for youngest children and increases for older children. A Tukey HSD post-hoc test showed a significant difference between accuracy rates for 5-year-olds and 10-year-olds ($p < 0.05$). LeCun et al.'s neural network architecture achieves the highest level of accuracy for 5-year-olds at just under 70% [SD = 22.53%]. Figure 4-19 shows the effect of age on recognition for the recognizer.

Srivastama and Sharma. The average recognition rate for Srivastava and Sharma [92] was 43.33% [SD = 22.32%] for 5-year-olds, 52.61% [SD = 13.53%] for 6-year-olds, 56.05% [SD = 10.81%] for 7-year-olds, 48.62% [SD = 11.46%] for 8-year-olds, 66.10% [SD = 9.40%] for 9-year-olds, and 62.25% [SD = 7.13%] for 10-year-olds. A one-way ANOVA on recognition accuracy with a between-subjects factor of *age* found no significant main effect of age on recognition accuracy ($F_{5,28} = 1.671$, *n.s.*). The lowest accuracy is the lowest for youngest children, at just under 45%. Figure 4-20 illustrates the recognition rates by age.

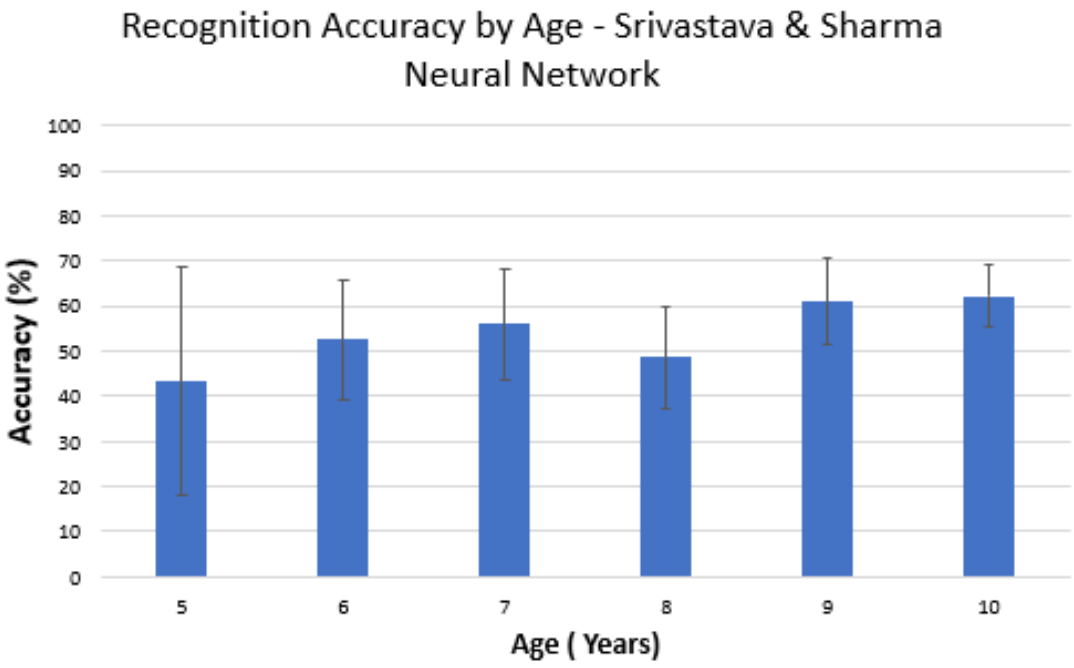

Figure 4-20: Effect of age on Srivastama and Sharma [92] recognition rates. Error bars represent the 95% confidence interval.

4.3.7 Mixed Methods

Yin and Sun. The average recognition rate for Yin and Sun [116] was 55.83% [SD = 24.18%] for 5-year-olds, 58.59% [SD = 20.04%] for 6-year-olds, 63.52% [SD = 14.70%] for 7-year-olds, 71.52% [SD = 11.56%] for 8-year-olds, 72.26% [SD = 11.01%] for 9-year-olds, and 74.52% [SD = 3.31%] for 10-year-olds. A one-way ANOVA on recognition accuracy with a between-subjects factor of *age* found no significant main effect of age on recognition accuracy ($F_{5,28} = 2.047$, *n.s.*). Recognition accuracy is lowest for the youngest children. Figure 4-21 shows the effect of age on recognition accuracy.

Alimoglu and Alpaydin. The average recognition rate for Alimoglu and Alpaydin [3] was 50.32% [SD = 22.43%] for 5-year-olds, 55.27% [SD = 17.61%] for 6-year-olds, 60.04% [SD = 9.05%] for 7-year-olds, 64.03% [SD = 8.43%] for 8-year-olds, 68.07% [SD = 7.51%] for 9-year-olds, and 71.02% [SD = 4.62%] for 10-year-olds. A one-way ANOVA on recognition accuracy with a between-subjects factor of *age* found no significant main effect of age on

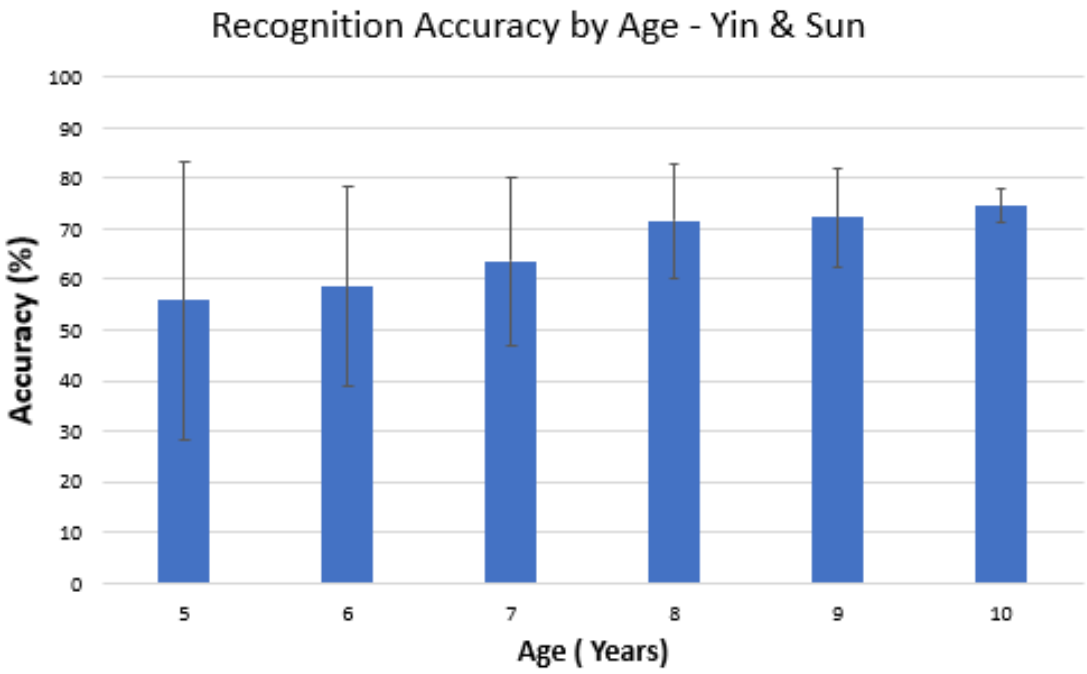

Figure 4-21: Effect of age on Yin and Sun [116] recognition rates. Error bars represent the 95% confidence interval.

recognition accuracy ($F_{5,28} = 2.814$, *n.s.*). As with the other recognizers, recognition accuracy is lowest for the youngest children. Figure 4-22 shows the effect of age on recognition.

4.3.8 Statistical Differences Between Recognizers

To dig into differences between recognizers, we ran a two-factor ANOVA on recognition accuracy with a between-subjects factor of *recognizer* and a within-subjects factor of *age*. We found a significant main effect of recognizer ($F_{13,15} = 4.013$, $p < 0.05$) on recognition accuracy. Figure 4-23 shows the results of a Tukey HSD posthoc tests. We see that the primary differences lie between the most accurate and least accurate recognizers. In particular, none of the top recognizers are significantly different from one another.

4.3.9 Discussion

As we see in our analyses, none of the recognizers perform particularly well for young children. Even the best case accuracy is around 70% [SD = 25.5%] for 5-year-olds with LeCun et al.'s neural network recognizer [60], but this rate is well below the 91% children report as

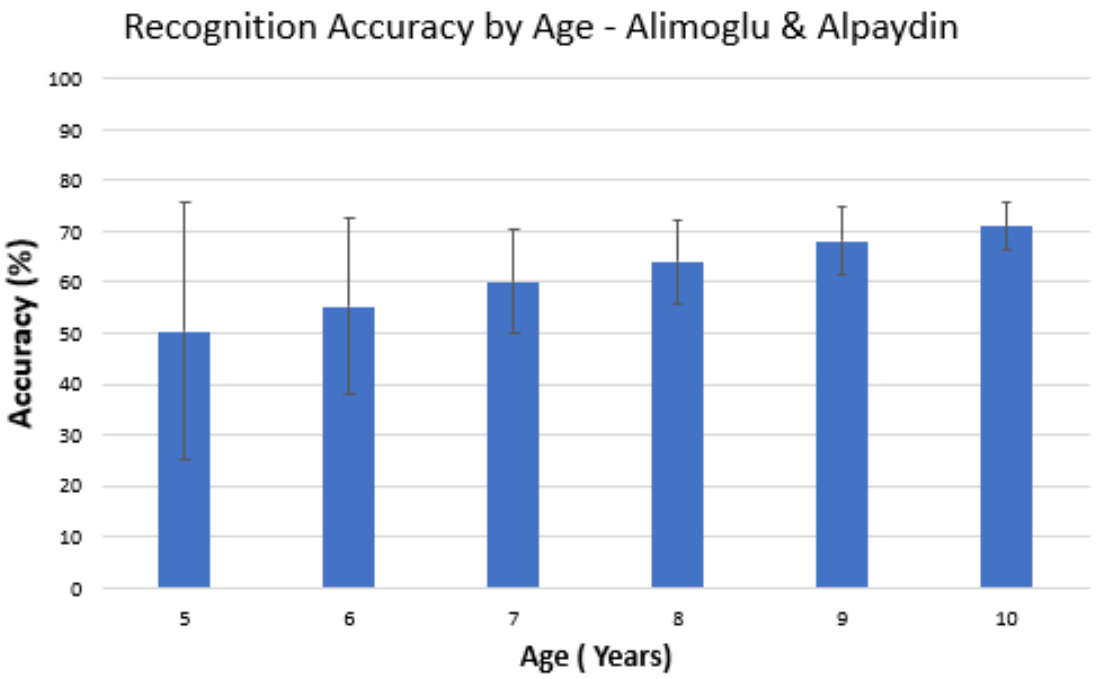

Figure 4-22: Effect of age on Alimoglu and Alpaydin [3] recognition rates. Error bars represent the 95% confidence interval.

acceptable according to Read et al.'s [81] study on handwriting recognition. Lecun et al.'s recognizer does, however, achieve the highest accuracy for all age groups. Neural networks may provide the best case results for accuracy, but they can be difficult to implement and often require large amounts of data to produce a good result. Thus, in cases where ease of implementation is not an issue and where large amounts of data can be easily obtained, a neural network may be the best option. Realistically, application developers are unlikely to have access to such a dataset, so we generally recommend against this approach.

Despite a fairly wide range of recognition accuracies, we can see a clear trend across all the recognizers in terms of difference among age groups: the recognition rate for 5-year-olds is the lowest in all cases, and the rate is highest for 10-year-old in all cases. The rate typically increases slightly for each increase in age. The only category where all recognizers did not show a significant effect of age on recognition rate was mixed methods, while in other categories some showed a significant effect while others did not. This makes it difficult

	GDE	Shrivastava & Sharma	GRANDMA	Alimoglu & Alpaydin	Yin & Sun	Anderson	Blagojevich	Kato et al.	Sezgin & Davis	$N-Protractor	$P+	$P
LeCun et al.	X	X	X	X	X							
$P	X	X	X	X	X							
$P+	X	X	X	X	X							
$N-Protractor	X	X	X									
Sezgin & Davis												
Kato et al.	X											
Blagojevich												
Anderson												
Yin & Sun												
Alimoglu & Alpaydin												
GRANDMA												
Shrivastava & Sharma												

Figure 4-23: Posthoc test results for our recognizer comparison. An X in a cell represents a significant difference between the two recognizers ($p < 0.05$). The horizontal text lists the recognizers roughly in order of performance, from highest accuracy to lowest.

to make generalizations about the categories of recognizers -- they must be considered on an individual basis. The class of recognizers that performed most poorly on average was feature-based statistical classifiers. Since these recognizers are primarily characterized by the features they employ, this indicates the features that were used in these recognizers were not adequate for distinguishing across children's gestures. We dive more deeply into articluation features in Chapter 5.

Clearly, there is no one recognizer that can provide the best performance in all cases. Based on the results of our experiments, we generally recommend against machine learning approaches in most practical cases as they require specialized knowledge on the part of the programmer and additional training samples. Feature-based statistical classifiers also seem to perform poorly in comparison to template matchers, so until further work is conducted we

recommend against the use of feature-based statistical classifiers. However, continued work on recognition should consider how the analyses in this work could be used to work towards better recognition across the different types of recognizers, as discussed in Chapter 6. In general, we recommend designers of gesture based applications use template matching approaches unless they have a specific need for a more advanced recognition method.

In the remainder of the work presented in this dissertation, we use accuracy rates from $P [103] as a point of comparison. The choice of $P was initially due to its popularity in the Human-Computer Interaction, ease of implementation, and reported high recognition rates. That said, the recognition rates are only a point of comparison, and any recognizer could have been used in its place. Furthermore, our analysis earlier in this chapter showed no significant difference between $P and the most accurate recognizer (LeCun et al. [60]). In the remainder of our work, we are more concerned with the trend across age groups and how it can help us understand user behaviors that affect recognition rates.

4.4 Comparison of Recognition Rates across Devices

All of the work described to this point used gestures collected on touchscreen phones. However, children are increasingly using other touch-enabled devices, such as tablets, touchscreen computers, and larger tabletop devices [29]. Furthermore, children also use touchscreen devices with styluses and pens, which may be different than finger-based gestures. Thus, it would be valuable to understand how recognition rates are impacted by screen size and input modality to allow for improved gesture-based applications. We designed two studies to address this need: one study in which children created gestures using a tablet device and another in which children created gestures using a tabletop touchscreen devices [112]. In the tablet study, children created gestures with a pen and with their fingers to allow for comparison.

4.4.1 Tablet Study

In this study, we investigated pen and touch input by 13 children on a tablet device. The application in our experiment (Figure 4-24) was run on a Wacom Cintiq Companion Hybrid

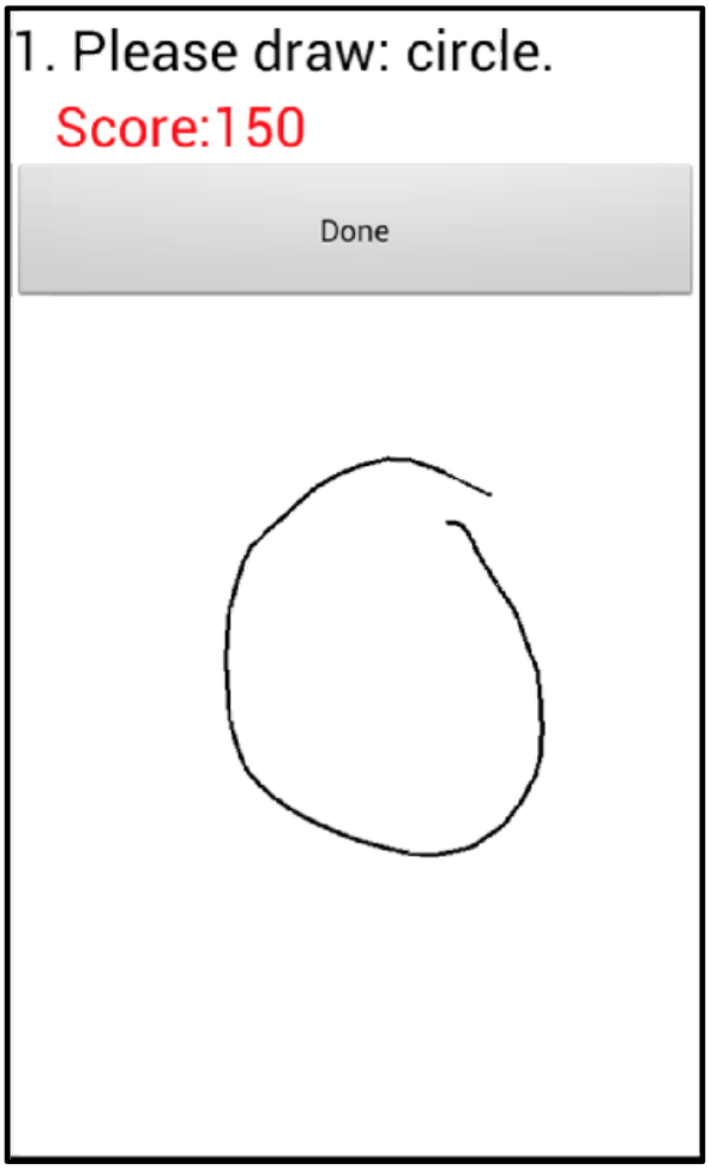

Figure 4-24: Screenshot of the gesture application we used in our study.

tablet with 8 GB of DDR3 RAM. The resolution was 1080 x 1920 (166 DPI), and the display size was 13.3 inches, measured diagonally. We used the Wacom Cintiq Companion Hybrid tablet because of its ability to measure detailed touch information that a standard tablet device, such as an iPad, typically does not [108].

4.4.1.1 Participants

The participants in our tablet study included 13 children ages 5 to 10 (M = 7.31, [SD = 1.6]). There were 2 five-year-olds, 2 six-year-olds, 4 seven-year-olds, 1 eight-year-old, 3 nine-year-olds, and 1 ten-year-old. Six participants were female. One participant was left handed. A total of 3,120 gestures were created in our study (2 conditions x 20 gestures x 6 repetitions x 13 participants). Each participant completed the gesture task twice: once using touch and once using a pen. The order of input modality was counterbalanced across participants, so that 7 children completed the pen task first while the other 6 completed the touch task first.

4.4.1.2 Results

Based on our review of the motor development literature [34, 39, 82, 86, 87], we divide the participants into age groups: 5- to 6-year-olds (4 participants), 7- to 8-year-olds (5 participants), and 9- to 10-year-olds (4 participants). We use these groupings in our analysis of the tablet results.

User-Dependent Recognition Results. As in previous studies, we used the testing procedure introduced by Wobbrock et al. [111] and used in many studies [11, 12, 103]. We systematically increased the number of training examples from T = 1 to 4 (1 must be chosen for testing, leaving a maximum of 4 for the training set). There were about 19,690 user-dependent recognition tests (2 input modalities x 10 trials x 13 participants x 4 values of T x 20 gestures; actual value is lower because some users were missing gestures).

A repeated measures ANOVA on accuracy with a between- subjects factor of *age* (5, 6, 7, 8, 9, 10) and a within-subjects factor of *input modality* (touch vs. pen) showed a significant effect of age on accuracy ($F_{2,12} = 4.43$, $p < .05$). Accuracy was worst for the 5- year-olds (66.22% [SD = 5.4%]) and improved for older children (e.g., 10- year-olds: 81.81% [SD = 1.33%]), which is consistent with prior work that has shown that accuracy increases with age [8, 113]. The same ANOVA found no significant difference ($F_{1,8} = 3.83$, n.s.) between pen

(81.81% [SD = 13.49%]) and touch (81.04% [SD = 11.86%]). Thus, application designers can expect similar gesturing performance for children in both pen and touch modalities.

User-Independent Recognition Experiments For our user-independent tests, we used the procedure explained by Vatavu et al. [103], with 10 trials. Because the number of participants is different in each age group, the maximum value of training participants varies by age. To make our results comparable, we used 3 randomly chosen training participants for each age group (since some groups had only 4 participants and one must be the test participant) and 5 training examples. We used these results to analyze the differences by age group and input modality. For age, a one-way ANOVA on accuracy with a between-subjects factor of *age group* (5-6, 7-8, 9-10) showed a significant effect of age on recognition accuracy ($F_{2,9} = 16.34$, $p < 0.001$). As with the user-dependent scenario, accuracy was directly proportional to age: worst for the youngest children and best for the oldest children (5-6: 46.78% [SD = 4.93%]; 7-8: 57.71% [SD = 5.12%]; 9-10: 78.36% [SD = 4.26%]). As is typical with previous children's gesture recognition experiments, user-independent accuracy rates are lower than those of user-dependent rates [6, 8, 9, 113]. A repeated measures ANOVA on accuracy with input modality (pen vs. touch) as a within-subjects factor found no significant difference ($F_{1,21} = 0.411$, n.s.) between pen (66.23% [SD = 10.76%]) and touch (63.60% [SD = 10.45%]). There was no effect of input modality on recognition rates in both the user-independent and user-dependent case. Therefore, designers can expect similar accuracy for touch and pen even when the recognizer is trained on different children's gestures.

4.4.2 Tabletop

Our tabletop study was run in a similar manner as the tablet study. The applications in our tabletop experiment were run on a Samsung SUR40 with 4GB RAM. The resolution was 1920 x 1080 (55 DPI), and the display size was 40 inches, measured diagonally.

4.4.2.1 Participants

The participants in our tabletop study included 18 children, ages 6 to 10 (M = 7.83, [SD = 1.38]): 4 six-year-olds, 4 seven-year-olds, 3 eight-year-olds, 5 nine-year-olds, and 2

ten-year-olds. Ten participants were female, and three participants were left handed. For this study, we collected a total of 2,160 gestures (20 gestures x 6 repetitions x 18 participants).

4.4.2.2 Results

User-Dependent Recognition Rates. A one-way ANOVA on accuracy with a between-subjects factor of age (6, 7, 8, 9, 10) showed no significant effect of age on recognition accuracy (F4,13 = 1.35, n.s.). As in previous experiments, accuracy was lowest for the youngest children (in this case, 6 year-olds; 63.65% [SD = 19.09%]) and increased for older children, with the highest accuracy for 10 year-olds (93.74% [SD = 5.09%]), as shown in Figure 4-25. The high variance in the 6- and 7-year-olds' data could explain why the result is not significant. Additionally, our tabletop study did not include 5-year-olds.ANOVA showed no significant effect of age on accuracy ($F_{1,16} = 2.411$, n.s.). However, we see the same trend that the youngest children's accuracy had high variance.

User-Independent Recognition Rates. Because the ages of participatns in the tabletop study ranged from 6 to 10, we used different age groupings in the user-independent case here than in the tablet study: 6- to 7-year-olds (8 participants) and 8- to 10-yearolds (10 participants). In the case of user-independent recognition, accuracy was lower for the 6- to 7-year-olds (67.63% [13.05%]) than the 8- to 10-year olds (76.45% [11.08%]).

4.5 Human Recognition

After establishing that recognition rates for 5- to 10-year-old children were poor (as low as 64% for 5-year-olds [113]), we aimed to continue working toward improved recognition. However, we wondered if it might be overly optimistic to think large gains in recognition accuracy would be possible. Perhaps children's gestures are simply so messy that not even humans would be able to recognize them. To answer this question, we developed a study in which we had human participants classify gestures created by children. We then compared the recognition rates achieved by humans with the recognition rates obtained by the $P recognition algorithm in our prior work.

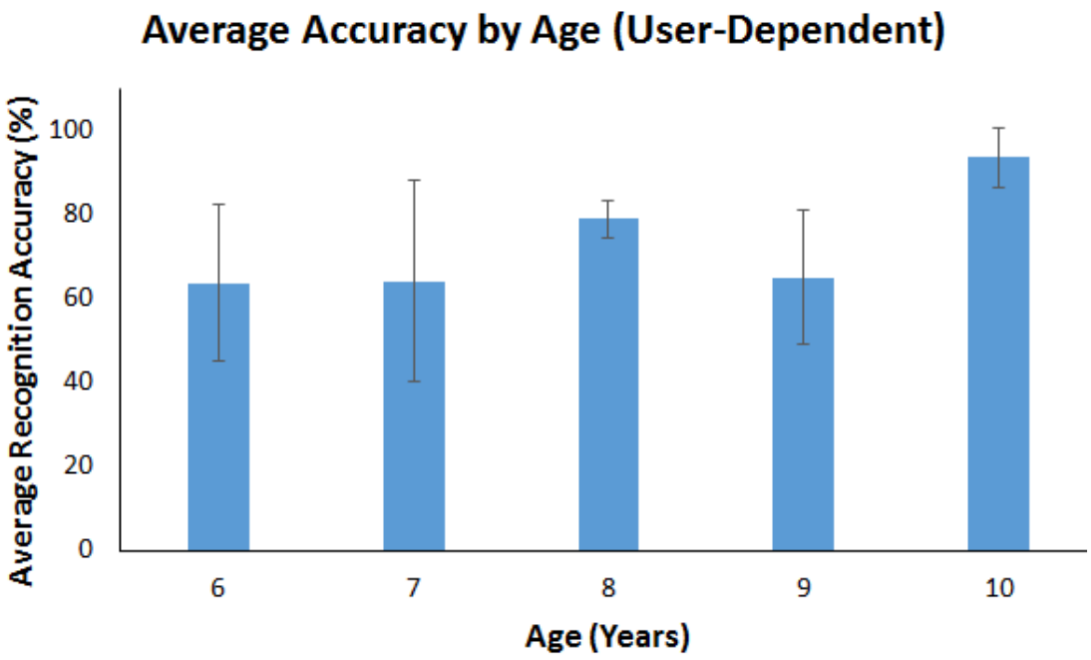

Figure 4-25: User-dependent recognition rates for the tabletop experiment. Error bars represent the 95% confidence interval.

4.5.1 Procedure

In our experiment, we made image captures of each of the 2,600 gestures in the corpus from our touchscreen phone study [113], scaling them such that they had the same physical onscreen size as when they were produced. For each gesture, we created a single online survey question showing the image of the gesture and asking which of the 20 possible gesture types the gesture most resembled. Participants were required to select one of the options, and could guess if they had no idea. Because it is not practical for each participant to answer 2,600 survey questions, the questions viewed by each participant were randomly selected such that each participant saw exactly 20 gestures from each age group, and of those 20 gestures, each represented exactly one of the distinct gesture types. Thus, each participant was asked a question about each gesture type for 5-, 6-, 7-, 8-, 9-, and 10-year-olds, a total of 6 age groups x 20 gesture types = 120 questions, in a randomized order. Participants were not told the age of the creators of the gestures, and they did not know that they would see an equal number of each gesture type. In fact, participants did not even know the gestures were produced by

Table 4-2: Human Accuracy of gestures seen by a small number of participants versus a larger number. The accuracy is not significantly different when a large number of participants sees the gestures.

	% Human Accuracy (SD)	
Age (Years)	**3-5 Participants**	**6+ Participants**
5	74.99 (39.56)	75.30 (34.89)
6	84.50 (29.80)	82.09 (31.08)
7	90.52 (19.44)	92.31 (20.41)
8	90.99 (21.19)	91.74 (17.52)
9	94.28 (16.89)	87.05 (21.43)
10	97.41 (9.47)	94.08 (12.08)

children. The random selection of gestures for the survey was such that each of the gestures was seen approximately the same number of times as other gestures from the same age group across the study. Each gesture in our corpus was evaluated by at least 3 participants. To ensure this, we initially recruited a small sample of approximately 50 people, then looked at which gestures still had fewer than 3 responses. We redeployed the survey with only those questions that still needed to be evaluated to reach at least 3 participants. We repeated this process until all gestures had been evaluated by at least 3 participants (max: 32 due to imbalance in the number of gestures per age group). No participant was allowed to take more than one version of the survey. Table 4-2 shows that similar accuracy rates were found for the gestures evaluated by a smaller number of participants (3-5) and a larger number (6+), and a paired t-test found no significant difference in accuracy between the two categories ($t(5) = 1.3$, *n.s.*).

4.5.2 Recognition Accuracy by Recognizer Type

To understand how well humans are able to recognize children's gestures, we first determined whether each individual gesture was recognized correctly. To do this, we looked at the survey takers' categorization of each individual gesture from each writer. If at least half of the survey takers who saw that gesture identified it correctly, it was counted as a correct recognition, otherwise it was incorrect. Ties were broken by having an additional human participant classify the gesture as described above. We thus represented each gesture in our

dataset of gestures as either correctly recognized (labeled 1) or incorrectly recognized (labeled 0). For each child in the dataset of gestures, we computed the average accuracy for each type of gesture produced by that child, then the overall accuracy of recognition of that child's gestures. We compared the per-child average recognition rate by humans to the per-child average recognition rate by machine. The majority-based method we employ has been used in previous crowdsourcing studies [22, 53].

All of the factors in our study were analyzed using the same three-way repeated-measures ANOVA on recognition accuracy with a between-subjects factor of *age* and within-subjects factors of *recognizer type* (human vs. machine) and *gesture category* (the general type of each gesture). All references to our ANOVA refer to this single test.

For overall accuracy including all age groups, the human recognition rate was 90.60% [SD = 10.4%], compared to 84.14% [SD = 9.9%] for machines. Our ANOVA found a significant main effect of recognizer type (human vs. machine) on recognition accuracy ($F_{1,20} = 42.197$, $p < 0.001$). The same test found a marginal effect of age ($F_{5,20} = 2.485$, $p = 0.066$) and no significant interaction between age and recognizer type ($F_{1,48} = 0.761$, *n.s.*). Figure 4-26 illustrates the overall recognition rates for each of the age groups in our study, showing a higher accuracy for each group in the case of human recognition. The significant gap between human and machine recognition is illustrative of the potential for improvement of machine-based recognizers.

A Tukey post-hoc test on the interaction between age and recognizer type (human vs. machine) found a significant difference in recognizer type for 5-year-olds, 8-year-olds, and 9-year-olds ($p < 0.05$). There was also a marginal difference for 6-year-olds ($p < 0.1$). The mean difference in recognition accuracy between human and machine is greatest for the youngest group, the 5-year-old writers. For the gestures produced by writers in this age group, the machine recognizer achieved only 65.30% accuracy, compared to 74.63% accuracy for humans, a difference of nearly 10%. While the human recognition accuracy is higher than the machine accuracy, even the human accuracy for 5-year-olds is far below what children

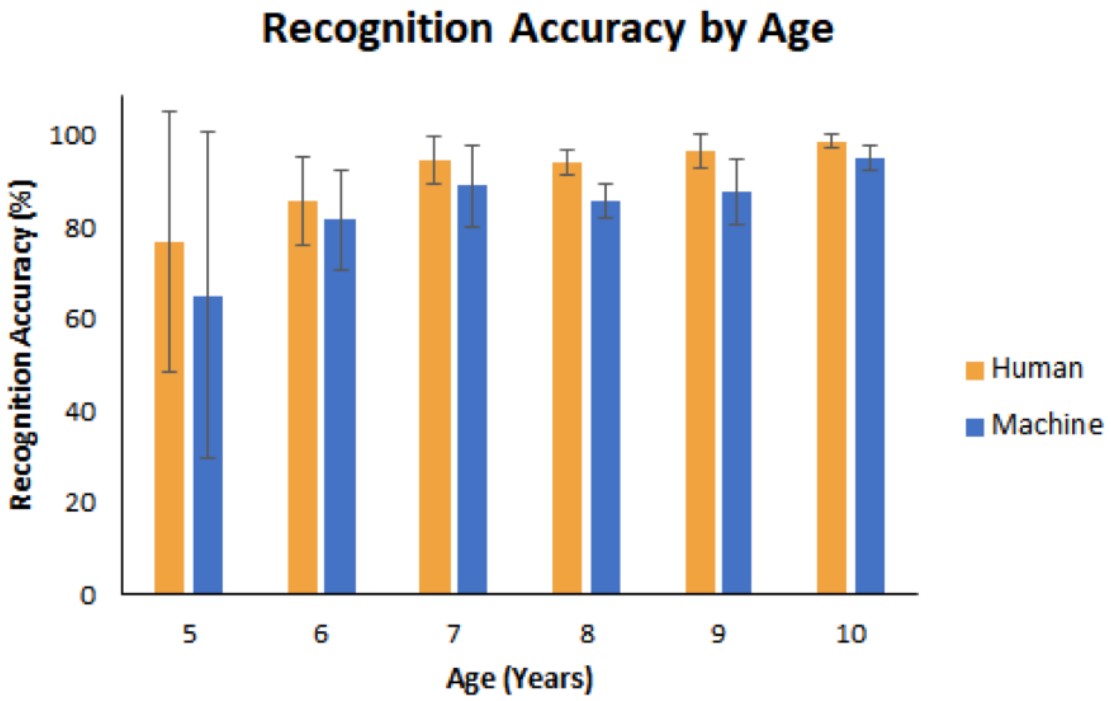

Figure 4-26: Accuracy rates for human versus machine recognition. Error bars represent the 95% confidence interval.

report as an acceptable error level, 91% [81]. However, our work provides a realistic target for future work in machine recognition. Because age was only marginally significant, we focus our discussion on gesture category and recognizer type (human vs. machine).

4.5.3 Recognition Accuracy by Gesture Category

To better understand the types of gestures humans are best able to recognize compared to machines, we also examined recognition rates by category. We grouped each of the 20 gesture types in our analysis into one of four general categories of gestures: letters (A, E, K, Q, X), numbers (2, 4, 5, 7, 8), shapes (circle, square, triangle, diamond, heart), and symbols (line, plus, arch, arrowhead, checkmark). These categories are consistent with those described by the gesture set's creators [12].

We hypothesized that children would be more familiar with letters and numbers than shapes and symbols, and that therefore they may be less skilled at creating the latter two categories of gestures. The younger children in our study, in particular, may have still been

developing their ability to produce shapes and symbols [18]. Humans are presented with different variations of letters, numbers, symbols, and shapes throughout their daily lives. Therefore, we conducted our experiment under the assumption that humans should have superior ability at recognizing our gesture set than machines, which are often trained on a limited number of examples for each gesture (e.g., up to 4 examples in our study). Thus, we hypothesized that machines would be more affected by deviations than humans (e.g., because of children's articulation issues [91]).

Our ANOVA found a significant main effect of gesture category on accuracy ($F_{3,60} = 3.093$, $p < 0.05$). We also found a significant interaction between recognizer type (human vs. machine) and gesture category ($F_{3,72} = 5.745$, $p < 0.05$). Figure 4-27 shows the average recognition rates for each gesture category by recognizer type. Humans were even more accurate in classifying letter and number gestures than the other gesture categories as compared to the machine. We believe there are two reasons for this: first, the children tend to learn to draw letters and numbers before learning shapes and symbols, meaning they have more practice in creating them. Secondly, our initial assumption was that the ubiquity of letters and numbers means that humans will inherently have more practice interpreting letters and numbers drawn by others than shapes and symbols, which may have played a role in the result. Prior work has shown significant differences in the articulation features of gestures based on whether the user was familiar or unfamiliar with the gesture [10, 100, 104]. For example, Vatavu et al. [104] found that users had a lower level of variability in speed of execution of gestures that they were familiar with compared to unfamiliar gestures. Thus, it may be easier for humans to recognize gestures that are more familiar to the writer due to the writer's increased fluency in creating those gestures.

The wide gap between human and machine recognition rates, particularly for letters and numbers, shows the importance of improving the accuracy of recognition of children's gestures. As previously mentioned, Read et al.'s [81] prior work shows that children are not satisfied with rates below 91% accuracy, a target attained by humans in our study but not machines.

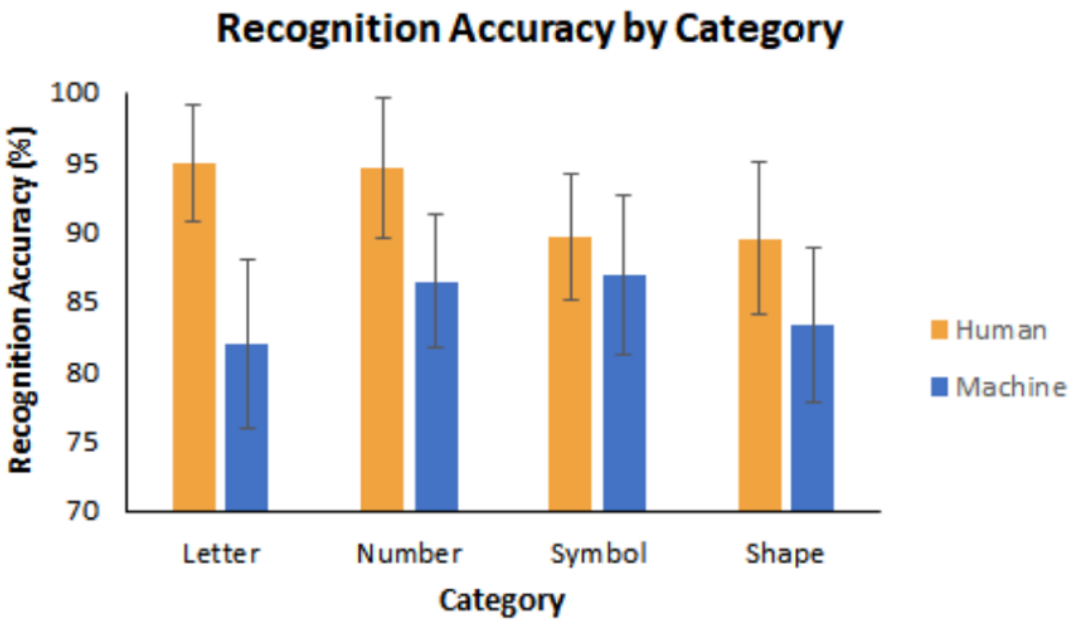

Figure 4-27: Effect of recognizer (human vs. machine) and category on recognition accuracy. Error bars represent the 95% confidence interval.

By showing that humans can achieve this accuracy, we show that it is a worthwhile goal to work toward in machine recognition. If humans were not able to meet this mark, we would be unsure if it is a reasonable target for accuracy. Thus, while recognition rates are low for children, we establish that current recognition rates should not be accepted as optimal, but instead that future work should push toward achieving higher accuracy.

4.5.4 Confusion Matrices

To dig into specific examples of the issues that humans and machines had in classifying the children's gestures in our study, we created and examined confusion matrices. A confusion matrix [97] (p. 209) is used to visualize the frequency with which each pair of gestures is recognized as one another. In our study, we had 20 gestures, so our confusion matrix has 20 rows and 20 columns. The rows of the matrix represent the correct answer, and the columns represent the recognition response. The value in each cell represents the percentage of times that all gestures of that corresponding row label were identified as the corresponding column

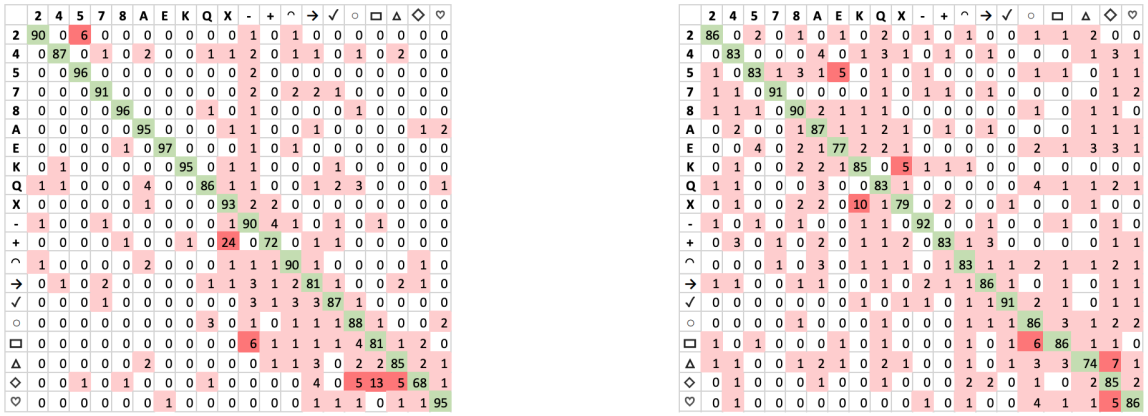

Figure 4-28: Confusion matrix for human recognition (left) and machine recognition (right) of the gestures in the dataset. Each cell represents the percentage of times the gesture in the row label was recognized as the gesture in the column label. Values are rounded to the nearest integer.

label. Thus, each row's values should sum to 100%; we round to the nearest integer in our matrix for simplicity, so the sum is $\pm 4\%$ for some rows.

Figure 4-28 shows the confusion matrices for the human (left) and machine (right) recognition accuracy in our experiment. The cells in the main diagonal (highlighted in green) represent correct recognitions. These cells generally contain high values, while other cells generally contain low values since incorrect recognitions are not as common as correct recognitions. We provide the machine confusion matrix for reference, but we focus our discussion on the human confusion matrix.

In the human confusion matrix, only six cells outside of the main diagonal have at least 5% confusion (darker red shading). Each of these provides interesting insight into recognition mistakes made by humans, so we discuss each of these commonly confused types. For the rest of this section, we use the term gesture to refer to one of the 20 types shown in Figure 4-1, whereas we use the term instance to refer to a specific example of one of these gestures produced by a writer.

"Plus" Confused for "X". The "plus" gesture was confused for the "X" gesture 23% of the time. In total, this misrecognition occurred a total of 148 times over 72 different instances. The confusion is likely due to the resemblance between the two gestures (the

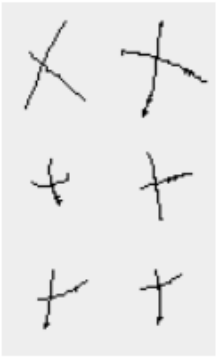

Figure 4-29: "Plus" instances commonly confused for "X".

same instance could be seen as an "X" or a "plus" depending on the viewer's orientation). Conversely, the "X" gesture was only confused for the "plus" gesture 1% of the time. One possible explanation for this discrepancy could be that children have difficulty drawing the lines perfectly vertically or horizontally on a touchscreen, causing them to appear slanted, thus resulting in many participants classifying them as "X" gestures. In some cases, shown in Figure 4-34, the reason for the misrecognition is quite apparent. The "plus" gestures are slanted and appear much more like "X" gestures. Other cases, however, are much less clear cut, with mostly vertical and horizontal lines, yet they were still often misclassified. Another reason for this could be that the "X" option was the last choice in our survey, so it may have "jumped out" at users more readily than the "plus" option, which was in the middle of the choices.

"2" Confused for "5". The "2" gesture was confused for the "5" gesture 6% of the time. In total, this misrecognition occurred 41 times over 6 different instances, and 37 of those 41 misrecognitions were of the same writer's instances. Examining these instances, indicated in Figure 4-30, it becomes quite clear why this was such a common mistake. Although these are meant to be "2" gestures, it appears to a human that the writer drew "5" gestures. We believe that this mistake was either due to the child drawing the "2" gestures backwards, causing it to resemble a "5", or by the child simply drawing the wrong gesture. Interestingly, this child's "5" gestures resembled standard gestures of the same type and were well recognized.

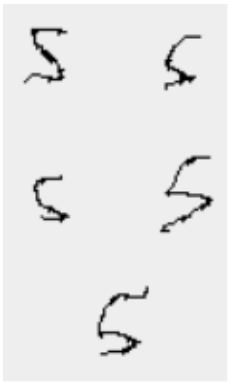

Figure 4-30: "2" instances commonly confused for "5".

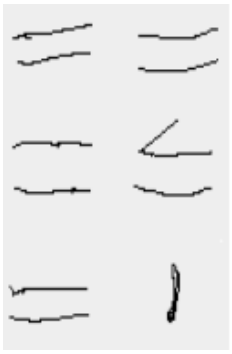

Figure 4-31: "Rectangle" instances commonly confused for "line".

"Rectangle" Confused for "Line". The "rectangle" gesture was confused for the "line" gesture 6% of the time. In total, this misrecognition occurred 39 times over 6 different instances. 4-31 shows the "rectangle" instances that were most confused for "line" gestures. In most of these cases, the child drew the rectangle without vertical lines, causing the instance to resemble an equals sign. Since this was not an option in the survey, the most logical next choice for many of the survey participants was to select the line option, since an equals sign is composed of two lines. The distance between the two lines may have caused participants to perceive the instance as two pieces rather than a single unit.

"Diamond" Confused for "Circle". The "diamond" gesture was confused for the "circle" gesture 5% of the time. This misrecognition occurred a total of 41 times over 6 instances, most of which were produced by 5- and 6-year-olds. These younger children may have been unfamiliar with the diamond shape compared to most of the other gestures in the

Figure 4-32: "Diamond" instances commonly confused for "circle."

set. Children typicallly learn how to draw a circle, rectangle, and triangle before a diamond, so children are likely to have more experience drawing these shapes. Supporting this idea, previous work using the same gesture set identified the diamond gesture as being difficult for children to draw [6]. Figure 4-32 shows examples of some of the "diamond" instances that were most commonly confused for "circle" gestures. While none of these look very similar to a circle, they do have more "rounded" sides than other gestures, which may have led some participants to classify them as "circle" gestures.

"Diamond" Confused for "Rectangle". The "diamond" gesture was confused for the "rectangle" gesture 12% of the time. This misrecognition occurred a total of 81 times over 48 different instances, spanning all age groups. Again, the high level of confusion may be due to the relative unfamiliarity of the "diamond" gesture compared to the other shapes in our corpus. Geometrically, a diamond and a rectangle are similar in that they are both quadrilaterals, which could partially account for some of the confusion we see in this case. Figure 4-33 shows examples of some of the most commonly confused instances in this set. While most of these instances are quadrilaterals that do resemble the traditional definition of a diamond, it is not unreasonable to think that they could be interpreted as slanted rectangles. A participant rushing through the survey may very well see the "rectangle" option before the "diamond" option, and decide to select that option without carefully considering the other options. In the case of the instance in the middle of the left column of the figure, the intent of the user is less

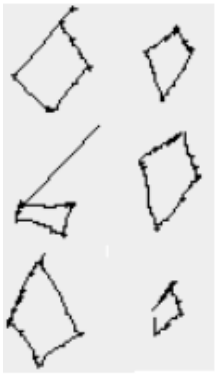

Figure 4-33: "Diamond" instances commonly confused for "rectangle".

clear. The long, straight line is likely to be disregarded by the participant as an error, leaving a rectangle.

"Diamond" Confused for "Triangle". The "diamond" gesture was confused for the "triangle" gesture 5% of the time. This misrecognition occurred a total of 81 times over 19 different instances. As in the prior two cases, the confusion of this pair may be due to the relative shape complexity compared to the other shapes in our gesture set. Figure 4-34 shows examples of "diamond" instances that were commonly confused for "triangle" gestures. Most of these instances look very similar to triangles, indicating that the child either drew the wrong gesture or did not know how to properly draw the diamond. In the last case, the child drew a diamond, but added a line through the middle. The extra line likely led many participants to interpret the instance as two triangles rather than a single diamond, leading to confusion of the gestures.

4.5.5 Discussion

We now briefly discuss the results of our human recognition study, including findings and their implications as well as the limitations and conclusions of our work. We conclude with a short discussion of our future work.

4.5.5.1 Human vs. Machine Recognition

Our work shows a significant difference between machine and human recognition of children's gestures. We also found a significant effect of gesture category on recognition, and

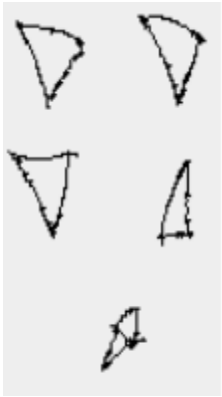

Figure 4-34: "Diamond" instances commonly confused for "triangle".

a significant interaction between gesture category and recognizer type. Our study provides an empirically established threshold by which future recognition algorithms can be judged: 90.6% [SD = 10.4%] for 5- to 10-year-olds by humans, compared to 84.14% [SD = 9.9%] by machines. We also show that letters and numbers are the gesture categories with the biggest gaps between human and machine recognition, indicating the potential for larger increases in recognition of these categories of gestures. Our experiment provides a good idea of the range of accuracy we can expect for various age groups from 5 to 10 and the accuracy that new recognizers should aim for. That said, as Read et al. [81] and LaLomia's [59] work shows, users are willing to tolerate a certain level of error in recognition. Interestingly, the 90.60% accuracy achieved by humans in our study is very similar to the 91% reported as acceptable by children in Read et al.'s study [81]. However, while we found an overall accuracy rate of 90.60% for children's gestures, human accuracy was much lower for the younger children's gestures than older children's. Human accuracy was 80.97% for 5-year-olds [SD = 38.2%], 83.20% for 6-year-olds [SD = 29.1%], 95.67% for 7-year-olds [SD = 19.7%], 91.52% for 8-year-olds [SD = 18.9%], 93.80% for 9-year-olds [SD = 17.6%], and 98.24% for 10-year-olds [SD = 9.6%]. While humans are able to recognize gestures by children of all ages better than machines, human accuracy for 5- and 6-year-olds' gestures is under the 91% tolerance level reported by Read et al. [81]. The low recognition rates of the youngest children's gestures by

humans can be at least partially attributed to the way the children create their gestures, which affects both human and machine recognition.

4.5.5.2 Commonly Confused Pairs

Since the types of gestures that were most confused varied, it is difficult to draw general conclusions about what particular habits of children cause these recognition errors. For example, confusion of "rectangle" gestures for "line" gestures appears to be mainly due to one child's drawing the rectangles without vertical lines, causing viewers to interpret them as two horizontal lines. In the case of "plus" being confused for "X", a greater number of writers' gestures were involved. Overall, humans recognized "plus" gestures with only 71% accuracy, compared to 83% for machine (this gesture is an example of one that was recognized more accurately by machine than human, although on average over all gestures human recognition was higher). The machine algorithm we used, $P [103], can detect differences between rotationally similar gestures like "plus" and "X", but not all recognizers can do so [11, 111]. We also found that humans had more difficulty distinguishing between pairs of rotated gestures. This high level of confusion in humans may represent a bias toward the "X" gesture over "plus", but the machine recognizer has no such bias. We saw a similar pattern for the "diamond" gesture: depending on rotation, it may resemble a rectangle, and it was recognized with only 67% accuracy by humans versus 85% by machine. These types of confusions may represent a bias in humans toward one gesture over another when they are rotationally similar (e.g., the more common gesture is chosen first). Based on this finding, we recommend that gesture set designers carefully consider whether using gestures that are rotationally similar in an application is necessary: if the children will likely exhibit inconsistent rotation behavior, the recognition algorithm chosen may or may not be able to detect the difference.

The fact that the "diamond" gesture was so widely confused points to another interesting issue. No other gestures were commonly (more than 5% of the time) confused with more than one other gesture, yet diamond was confused with three different shapes at least 5% of the time each. We speculate that many children know that a diamond is a shape, but are

not familiar with exactly how to draw it, leading their gesture to resemble other shapes in the survey. Another potential explanation is that survey takers may have expected diamond shapes to be more like "gem" or "jewel" shapes. In any case, the confused pairs provide interesting insight into the differences between human and machine recognition that result from children's gesture articulation patterns. Based on these findings, we recommend that gesture set designers carefully consider how familiar children will be with certain gestures, as children of a young age will not yet have developed fluency with basic shapes.

4.6 Summary

In this chapter, we discussed several studies we conducted with regard to establishing how well recognition algorithms are able to recognize children's touchscreen gestures. In our first study, we used an existing template matching algorithm called $P [103] to establish a baseline. We chose to use a template matcher because prior work on adults' gestures had shown template matchers were able to effectively recognize gestures with high accuracy. However, we found recognition rates for children was poorer than for adults (65.05% [SD = 29.28%] for 5-year-olds compared to 95.46% for adults [SD = 5.49%]). We found a significant effect of age on recognition accuracy, with low rates for the younger children in our study. After establishing low recognition rates for children, we conducted another study in which we compared the performance of different types of recognizers. In particular, we included several machine learning algorithms that add additional complexity beyond that of template matchers. We found that even with these more advanced methods, recognition was poor for young children's gestures, with even the best recognizer obtaining only 69.64% [SD = 22.53%] accuracy in recognizing 5-year-olds' gestures. Because recognition rates were consistently poor across all categories of algorithms, we wondered if the goal of improving accuracy might not be possible due to certain gestures being unrecognizable. We designed an experiment where naive viewers were asked to classify the gestures, allowing us to compare human performance with machine accuracy. We found that humans achieved significantly higher accuracy (90.6% [SD = 10.4%]) compared to a machine algorithm (84.1% [SD = 9.9%]) indicating potential for

improvement. We also found the difference between human and machine accuracy was greater for younger children than for older children. In the next chapter, we use articulation features to better understand the behaviors that lead to poor accuracy in recognizing children's gestures.

CHAPTER 5
ARTICULATION FEATURES

In this chapter, we discuss our work on using articulation features to better understand the differences between children's and adults' gestures. We begin by examining the values of existing articulation features on children's gestures and show that children's gestures are much less consistent as measured by these articulation features. We then describe a set of six new articulation features that we developed to better quantify common patterns in children's gestures that were not captured by existing features. Finally we examine the correlation of recognition rates with each of the existing and new features from our analysis to better understand how these features could be leveraged to improve recognition rates.

To help better understand the ways in which users make gestures, several researchers have developed articulation features [10, 64, 84, 104], which are measures designed to quantify some aspect of the way a gesture is created. These features are typically either geometric or temporal in nature. Examining these features helps designers and researchers to develop a better understanding of the nuances of users' gesturing behavior to help motivate improved design of gesture-based applications for children. Articulation features have been used to help motivate the design of improved recognition algorithms for specific populations, as in Vatavu's [101] work on improving recognition for users with low vision. In this chapter, we discuss the work we have carried out using existing and new articulation features to better understand the ways in which children make gestures with an eye toward improving recognition rates and providing a better experience for children using gesture-based touchscreen applications.

5.1 Existing Articulation Features

Prior to our work, there had been several analyses of touchscreen articulation features carried out on adults' gestures [10, 64, 104]. These features allow researchers and designers to gain new insights into how users make gestures, which can help them to design better applications for the users. Thus, we reasoned that because these features had been used in the past to improve adults' experiences, they may provide new insight on how to improve

children's experiences as well. However, these features had not yet been applied to children's gestures even though children are commonly using touchscreen applications involving gesture interactions. We hypothesized that even though these articulation features were not created for children, the information gleaned from these features would present an opportunity for us to better understand children's gestures and how we may be able to improve recognition rates. Table 5-2 shows the features we used in our study along with a brief description of each of them.

Our study included gestures from 24 children ages 5 to 10 years old and 27 adults. The gestures were collected in our prior work on interface complexity in which we established recognition rates using $P, as described in chapter 4 [113]. For the purpose of these analyses we grouped the users into four age groups: 5- to 6-year-olds, 7- to 8-year-olds, 9- to 10-year-olds, and adults. We broke our ananalyses down into two categories of features: simple features and relative accuracy. The simple features, which were taken from prior work by Anthony et al. [10], include geometric and temporal measures that are calculated on a single instance of a gesture. The relative accuracy features, which were introduced by Vatavu et al. [104], quantify the level of consistency between two gestures of the same type by the same user. We examined the effect of age on the 10 simple features and 12 relative accuracy features listed in Table 5-2.

5.1.1 Results

For each feature in our study, we ran a one-way ANOVA on the value of the feature with a between-subjects factor of *age group*. We found that 6 of the 10 simple features and all 12 of the relative accuracy features were significantly affected by age group. The only features that were not significantly impacted by age group were number of strokes, area of bounding box, curviness, and speed.

5.1.1.1 Simple Features

Number of Strokes. Average number of strokes is defined as a user's average number of pen or finger down-up events per gesture. The average number of strokes was 1.72 [SD

Table 5-1: Articulation features we examined in our study.

Simple Features	
Number of Strokes	The total number of strokes in a gesture.
Path Length	The total amount of ink used to create the gesture.
Area of Bounding Box	The area of the smallest box completely enclosing the gesture.
Line Similarity	A measure of how similar a gesture is to a line.
Global Orientation	The angle of a gesture's bounding box.
Total Turning Angle	The sum of the absolute values of the angle at each point on a gesture.
Sharpness	The sum of the squares of the angle at each point on a gesture.
Curviness	Total turning angle (6) divided by path length (2).
Production Time	Total amount of time taken to produce the gesture, including time between strokes.
Average Speed	Path length (2) divided by production time (9).
Relative Accuracy Features	
Shape Error	The average Euclidean distance between corresponding points of two gestures.
Shape Variability	The standard deviation of the distance between corresponding points of two gestures.
Length Error	A measure of the inconsistency in length between two gestures.
Size Error	A measure of the inconsistency in bounding box area between two gestures.
Bending Error	The average difference between corresponding turning angles of two gestures.
Bending Variability	The standard deviation of differences between turning angles of two gestures.
Time Error	The difference in the amount of time taken to articulate two gestures.
Time Variability	The standard deviation of the differences of timestamps between corresponding points of two gestures.
Speed Error	The difference in the speed of production of two gestures.
Speed Variability	The standard deviation of differences in the speed of production of two gestures.
Stroke Count Error	The difference in number of strokes between two gestures.
Stroke Ordering Error	A measure of the difference in stroke articulation order between two gestures.

Table 5-2: Formulae for calculating the features we examined in our study, where N is the number of points in the gesture, S is the number of strokes, x_i is the x coordinate of the ith point, y_i is the y coordinate of the ith point, and t_i is the time (in milliseconds) of the ith point. See Figure 5-1 for reference.

Simple Feature Formulae	
Number of Strokes	$f_1 = \sum_{n=1}^{S} 1$
Path Length	$f_2 = \sum_{n=2}^{N} \sqrt{(x_n - x_{n-1})^2 + (y_n - y_{n-1})^2}$
Area of Bounding Box	$f_3 = (x_{max} - x_{min})(y_{max} - y_{min})$
Line Similarity	$f_4 = \frac{\sqrt{(x_N - x_1)^2 - (y_N - y_1)^2}}{f_2}$
Global Orientation	$f_5 = \arctan \frac{y_{max} - x_{min}}{x_{max} - x_{min}}$
Total Turning Angle	$f_6 = \sum_{n=2}^{N-1} \lvert \arctan \frac{(x_{n+1} - x_n)(y_n - y_{n-1}) - (x_n - x_{n-1})(y_{n+1} - y_n)}{(x_{n+1} - x_n)(x_n - x_{n-1}) - (y_n - x_{y-1})(y_{n+1} - y_n)} \rvert$
Sharpness	$f_7 = \sum_{n=2}^{N-1} (\arctan \frac{(x_{n+1} - x_n)(y_n - y_{n-1}) - (x_n - x_{n-1})(y_{n+1} - y_n)}{(x_{n+1} - x_n)(x_n - x_{n-1}) - (y_n - x_{y-1})(y_{n+1} - y_n)})^2$
Curviness	$f_8 = \frac{f_6}{f_{12}}$
Production Time	$f_9 = t_N - t_1$
Average Speed	$f_{10} = \frac{f_2}{f_9}$

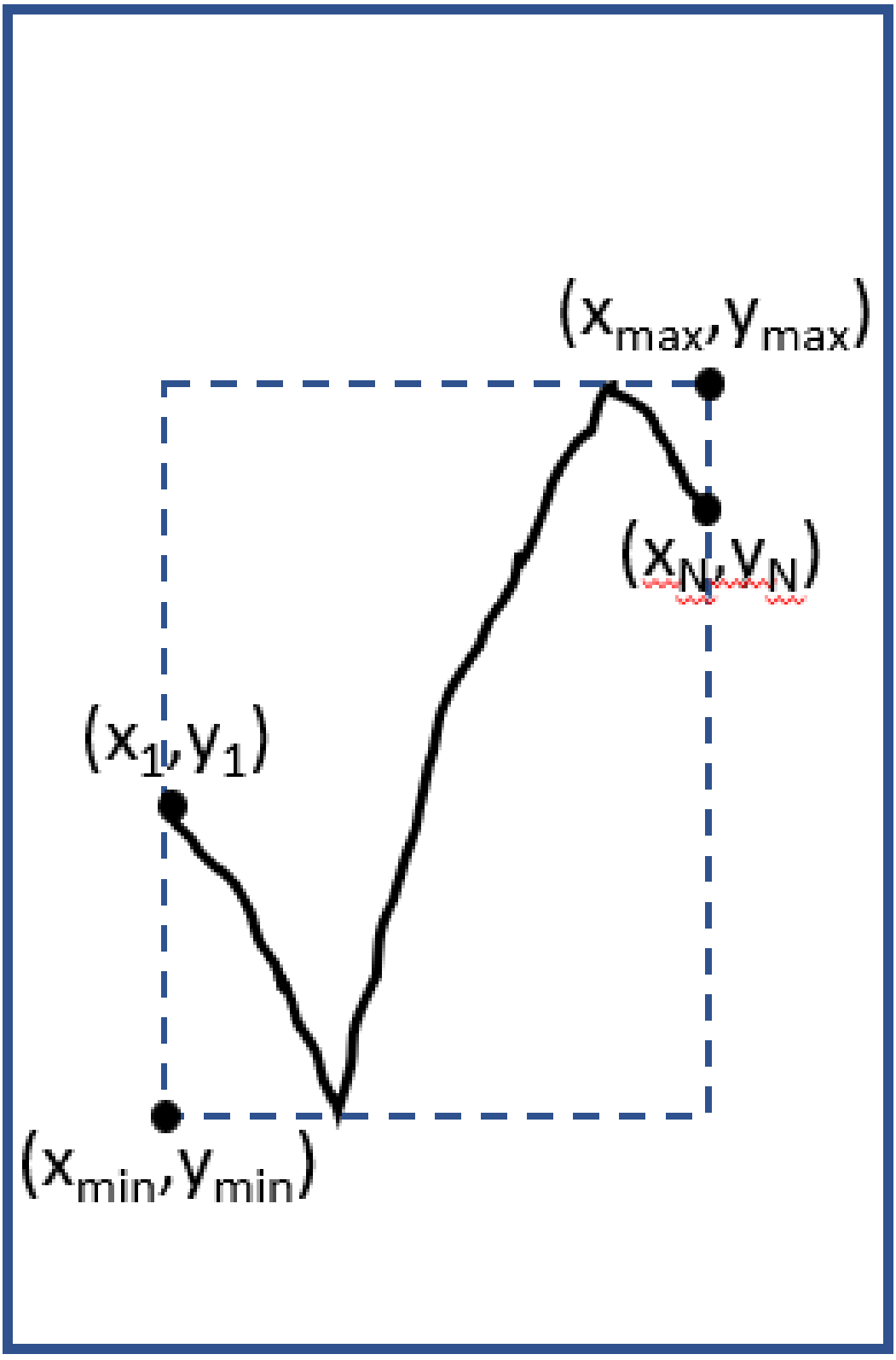

Figure 5-1: Example gesture for reference when consulting formulae. The solid line blue rectangle represents the canvas on which the user draws and the dashed line represents the gesture's bounding box.

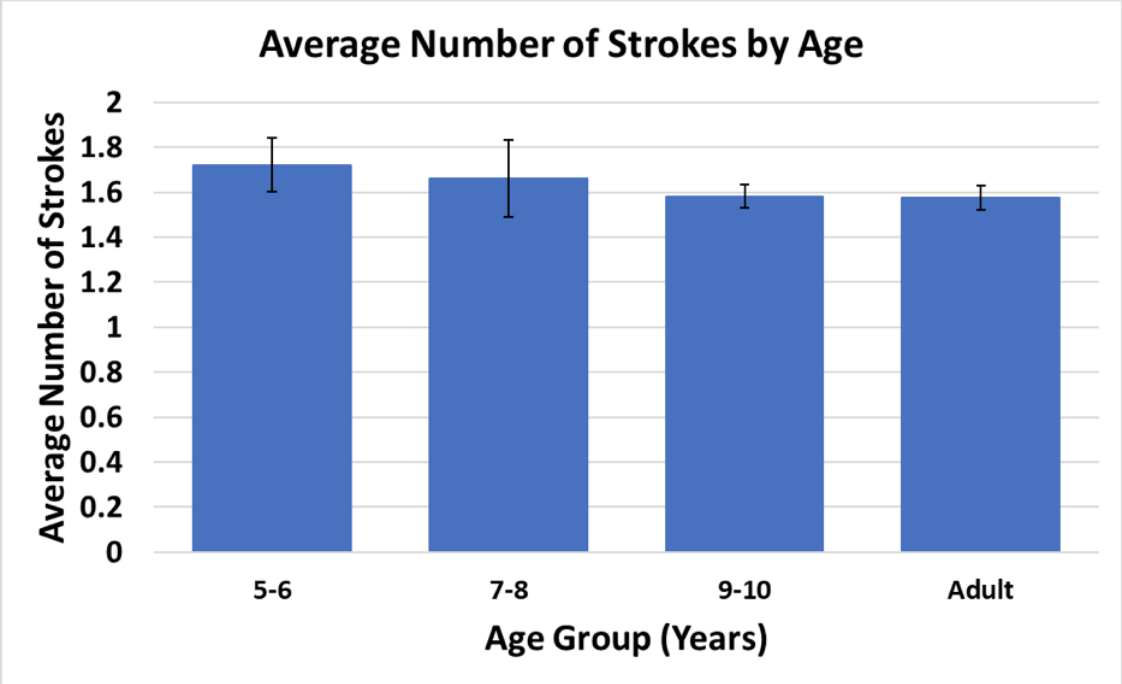

Figure 5-2: Effect of age group on average number of strokes. Error bars represent the 95% confidence interval.

= 0.16] for 5- to 6-year-olds, 1.66 [SD = 0.23] for 7- to 8-year-olds, 1.58 [SD = 0.08] for 9- to 10-year-olds, and 1.58 [SD = 0.14] for adults. A one-way ANOVA showed a marginal effect of age group on average number of strokes ($F_{3,47} = 2.208$, p = 0.0996). The average number of strokes is highest for the youngest children, and decreases for older participants. In general, younger children tend to use more strokes when creating gestures than adults. This may be due to children being less familiar with the gestures or with the ways in which they are commonly articulated. Figure 5-2 shows the effect of age group on average number of strokes.

Path Length. Path length refers to the sum of the distances between each adjacent pair of points in a gesture. In other words, it is a measure of the amount of ink used in creating a gesture. The average path length was 756.15 px [SD = 152.64 px] for 5- to 6-year-olds, 697.07 px [SD = 108.62 px] for 7- to 8-year-olds, 677.02 px [SD = 89.71 px] for 9- to 10-year-olds, and 596.35 px [SD = 123.67 px] for adults. A one-way ANOVA showed a significant main effect of age group on average path length ($F_{3,47} = 4.124$, p < 0.05). Post-hoc tests found a significant difference between 5-6 year-olds and adults (p <

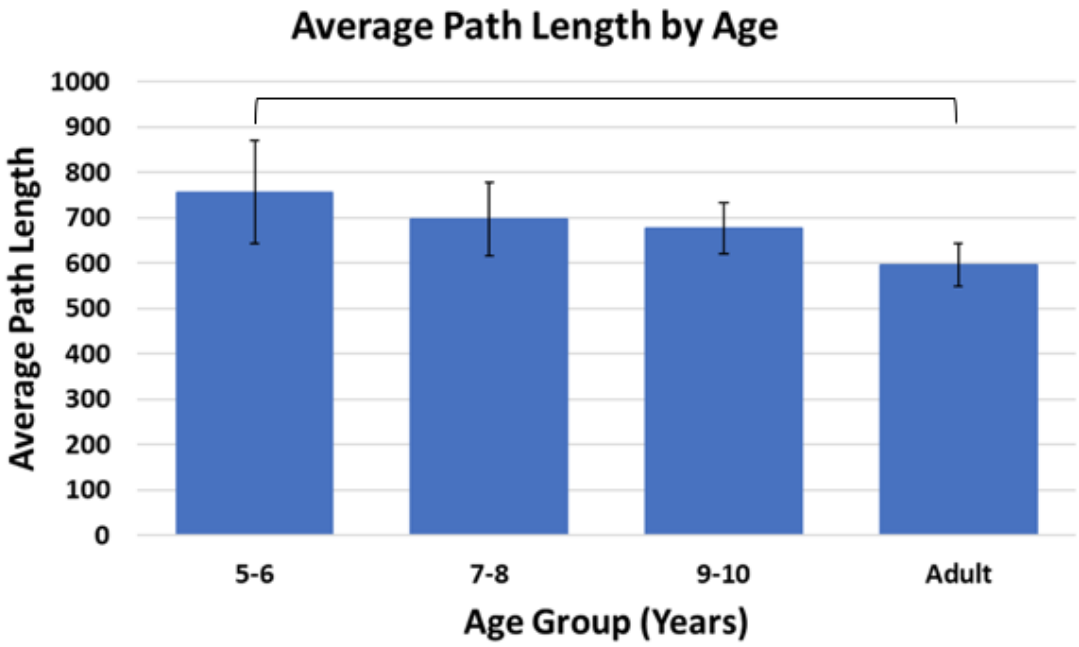

Figure 5-3: Effect of age group on average path length. Error bars represent the 95% confidence interval.

0.05). Average path length is highest for the youngest participants, and decreases for older participants. Thus, younger children tend to use less more ink when drawing gestures. This behavior may be related to children's tendency to overtrace gestures or due to their lack of experience in creating the gestures. Figure 5-3 shows the effect of age group on average path length.

Area of Bounding Box The area of the bounding box of a gesture refers to the area of the smallest box that fully encloses all points of a gesture. The average area of bounding box was 61,531.00 px^2 [SD = 17,362.40 px^2] for 5- to 6-year-olds, 55,790.94 px^2 [SD = 19,034.66 px^2] for 7- to 8-year-olds, 54,941.53 px^2 [SD = 12,748.45 px^2] for 9- to 10-year-olds, and 16,941.25 px^2 [SD = 16,941.25 px^2] for adults. A one-way ANOVA showed a marginal main effect of age group on average area of bounding box ($F_{3,47} = 2.566$, p = 0.0657). Average area of bounding box is highest for the youngest participants, and decreases for older children. Thus, younger children tend to use a wider area of the canvas when creating gestures. This may be due to the fact that children are still developing their motor control, and it may be

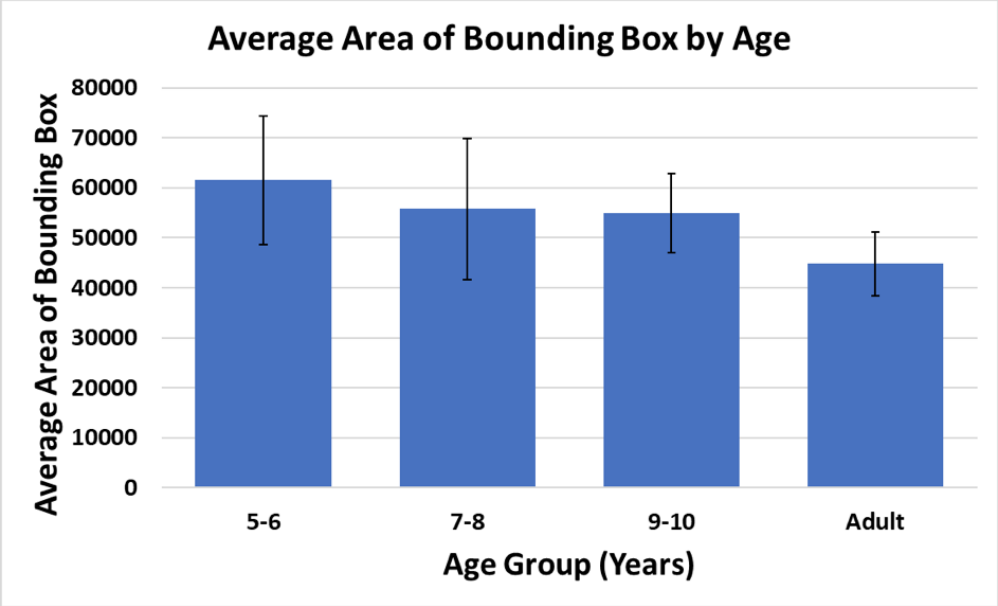

Figure 5-4: Effect of age group on average area of bounding box. Error bars represent the 95% confidence interval.

related to children's tendency to use more ink in their gestures. Figure 5-4 shows the effect of age group on average area of bounding box.

Line Similarity. Line similarity is a measure of how similar the strokes in a user's gestures are to a straight line. The maximum possible value, 1, indicates a perfectly straight line. The average line similarity was 0.29 [SD = 0.02] for 5- to 6-year-olds, 0.32 [SD = 0.03] for 7- to 8-year-olds, 0.33 [SD = 0.02] for 9- to 10-year-olds, and 0.34 [SD = 0.03] for adults. A one-way ANOVA showed a significant main effect of age group on average line similarity ($F_{3,47} = 5.622$, $p < 0.001$). Post-hoc tests found a significant difference between 5-6 year-olds and adults ($p < 0.05$). Average line similarity is lowest for the youngest children, and increases for older participants. Thus, younger participants tended to draw strokes that were less similar to straight lines than older participants. It is important to note, however, that this value is highly influenced by the gesture set. We see this pattern because the gesture set that was used contains many gesture types made primarily of straight lines (e.g., line, plus, X, E), but the finding may not hold for another gesture set consisting of more curved gestures. Taking this feature to the extreme and analyzing only the line gesture in our set, for example, shows

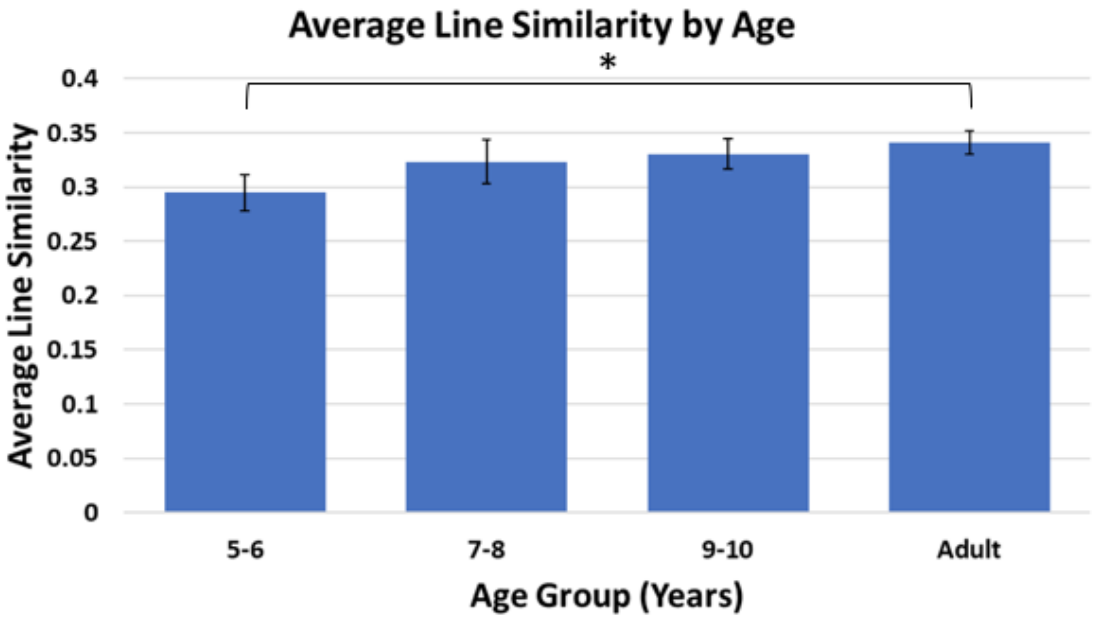

Figure 5-5: Effect of age group on average line similarity. Error bars represent the 95% confidence interval.

the same pattern, with the youngest children having the lowest values for line similarity. To test the significance of whether the gesture type influences line similarity, we ran a two-way ANOVA on *line similarity* with a within-subjects factor of *gesture type* and a between-subjects factor of *age*. We found a significant effect of both age ($F_{6,88} = 3.882$, $p < 0.05$) and gesture type ($F_{1,88} = 414.752$, $p < 0.05$). Figure 5-5 shows the effect of age group on average line similarity.

Global Orientation A gesture's global orientation is equal to the angle of the diagonal of the gesture's bounding box. A very tall and thin gesture would have a high global orientation, while a short and wide gesture would have a low one. The average global orientation was 0.92 [SD = 0.04] for 5- to 6-year-olds, 0.89 [SD = 0.05] for 7- to 8-year-olds, 0.90 [SD = 0.05] for 9- to 10-year-olds, and 0.83 [SD = 0.05] for adults. A one-way ANOVA showed a significant main effect of age group on average global orientation ($F_{3,47} = 10.14$, $p < 0.001$). Post-hoc tests found a significant difference between 5-6 year-olds and adults ($p < 0.001$), between 7-8 year-olds and adults ($p < 0.05$), and between 9-10 year-olds and adults ($p < 0.05$). Average global orientation is greatest for the youngest participants, and generally decreases for older

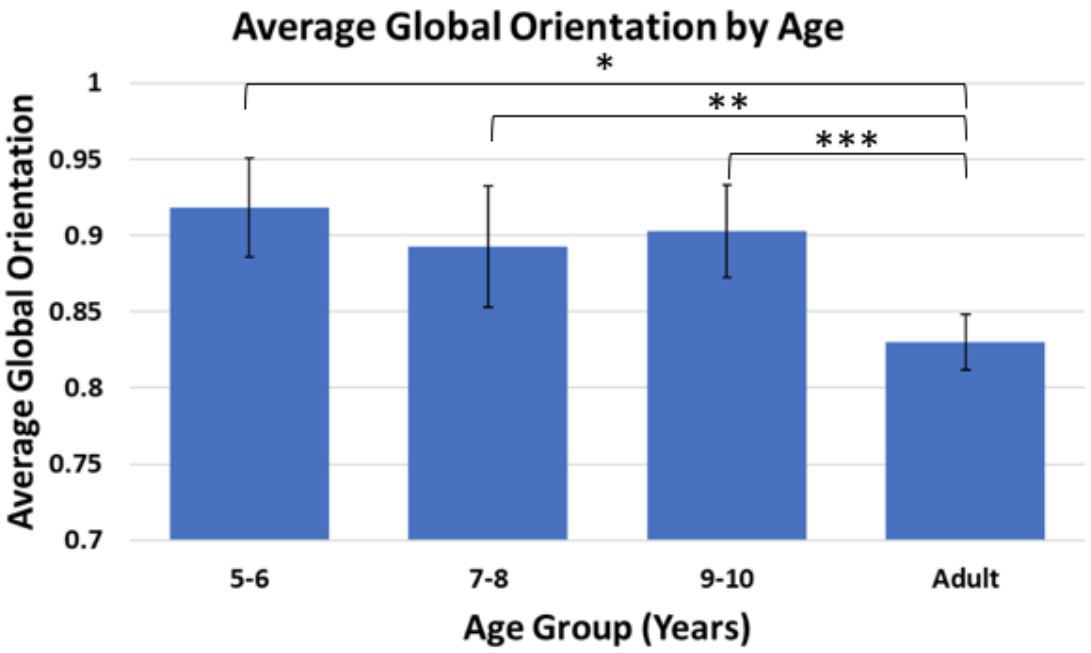

Figure 5-6: Effect of age group on average global orientation. Error bars represent the 95% confidence interval.

participants. The trend illustrates that children are more likely to skew the proportions of their gestures in such a way that they are taller than they are wide. It should be noted that the values of average global orientation for the three groups of children in our analysis are very similar, but that adults have a much lower average value. This feature does not appear to be highly correlated with recognition rates. However, since previous work has shown that recognition rates after age 13 are similar to those of adults [6, 7], future research may consider examining this feature in children ages 11 to 12 to see where this behavior begins to resemble that of adults. Figure 5-6 shows the effect of age group on average global orientation.

Total Turning Angle. The total turning angle of a gesture refers to the sum of the absolute value of the angle made at each point on each stroke of a gesture. The average global orientation was 11.31 deg [SD = 1.57 deg] for 5- to 6-year-olds, 10.54 deg [SD = 1.62 deg] for 7- to 8-year-olds, 9.58 deg [SD = 0.65 deg] for 9- to 10-year-olds, and 8.95 deg [SD = 0.67 deg] for adults. A one-way ANOVA showed a significant main effect of age group on average global orientation ($F_{3,47} = 12.89$, $p < 0.001$). Post-hoc tests found a significant

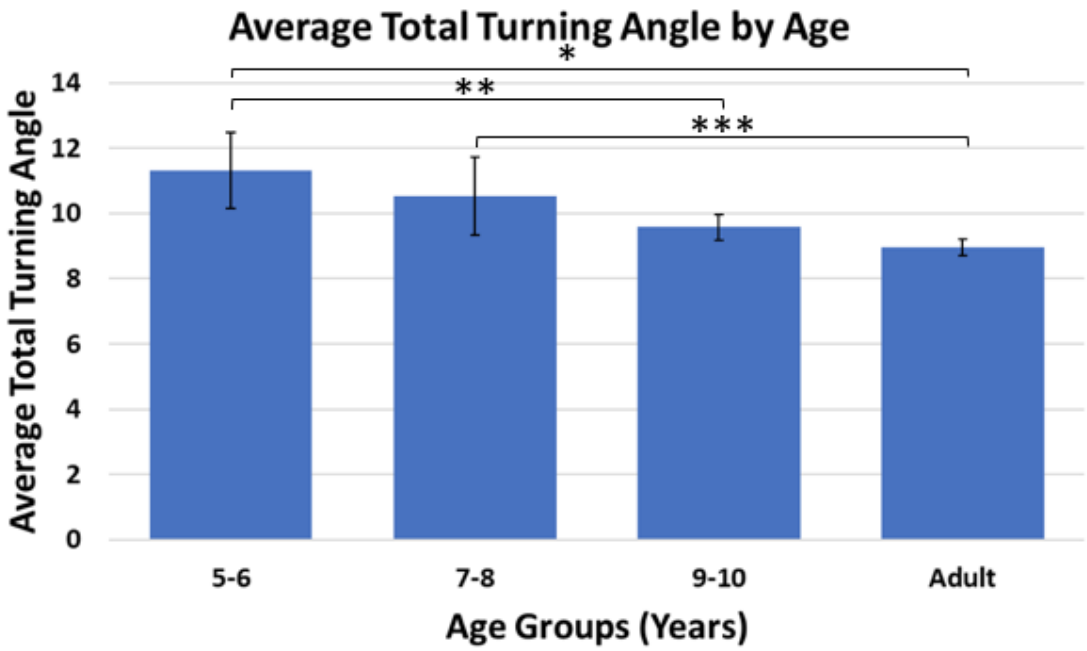

Figure 5-7: Effect of age group on average total turning angle. Error bars represent the 95% confidence interval.

difference between 5-6 year-olds and 9-10 year-olds ($p < 0.05$), between 5-6 year-olds and adults ($p < 0.001$), and between 7-8 year-olds and adults ($p < 0.05$). Average total turning angle is highest for the youngest participants, and decreases for older participants. Thus, younger children tend to have more overall variation in the angles of the points along the paths of their gestures. This corroborates the finding that older participants tend to draw straighter lines as seen in the line similarity feature. A two-way ANOVA on *total turning angle* with a between-subjects factor of *age* a within-subjects factor of *gesture type* (composed of straight lines or not composed of straight lines) found a significant effect of both age ($F6, 88 = 112.11$, $p < 0.05$) and gesture type ($F_{1,88} = 7.94$, $p < 0.05$). Thus, the results would likely be different with a gesture set consisting of more curved gestures. Overall, younger children tend to have more (and sharper) angles in their gestures. Figure 5-7 shows the effect of age group on average total turning angle.

Sharpness. The sharpness of a gesture is equal to the sum of the squares of the angles at each point of the gesture. The average sharpness was 13.55 deg^2 [SD = 2.56 deg^2] for

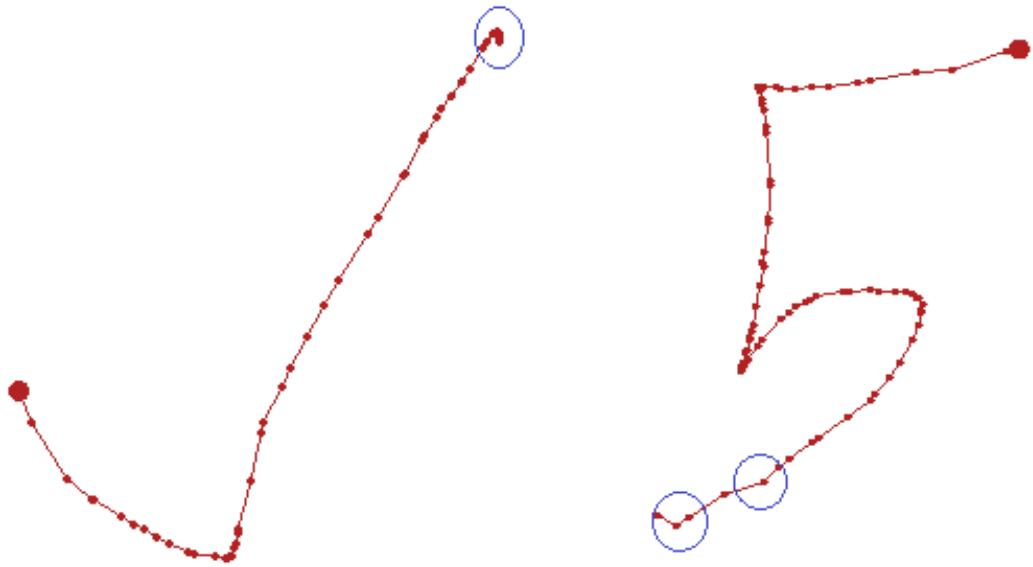

Figure 5-8: Examples of gestures from the corpus we used that exhibit behaviors that lead to high sharpness (circled).

5- to 6-year-olds, 12.72 deg^2 [SD = 3.29 deg^2] for 7- to 8-year-olds, 10.43 deg^2 [SD = 0.97 deg^2] for 9- to 10-year-olds, and 9.77 deg^2 [SD = 1.34 deg^2] for adults. A one-way ANOVA showed a significant main effect of age group on average sharpness ($F_{3,47} = 10.66$, p < 0.001). Post-hoc tests found a significant difference between 5-6 year-olds and 9-10 year-olds (p < 0.05), between 5-6 year-olds and adults (p < 0.001), and between 7-8 year-olds and adults (p < 0.001). The youngest children had the highest average value for sharpness, and it decreases for older participants. As with the previous feature, this behavior is likely due to younger children's tendency to have more frequent, more pronounced angles in their gestures, as in the examples in Figure 5-8. This finding further shows how children's motor development stage impacts their gestures. Figure 5-9 shows the effect of age group on average sharpness.

Curviness. The curviness of a gesture is equal to the total turning angle of the gesture divided by the path length. The average curviness was 0.016 [SD = 0.003] for 5- to 6-year-olds, 0.016 [SD = 0.004] for 7- to 8-year-olds, 0.015 [SD = 0.002] for 9- to 10-year-olds, and 0.015 [SD = 0.003] for adults. A one-way ANOVA showed no significant

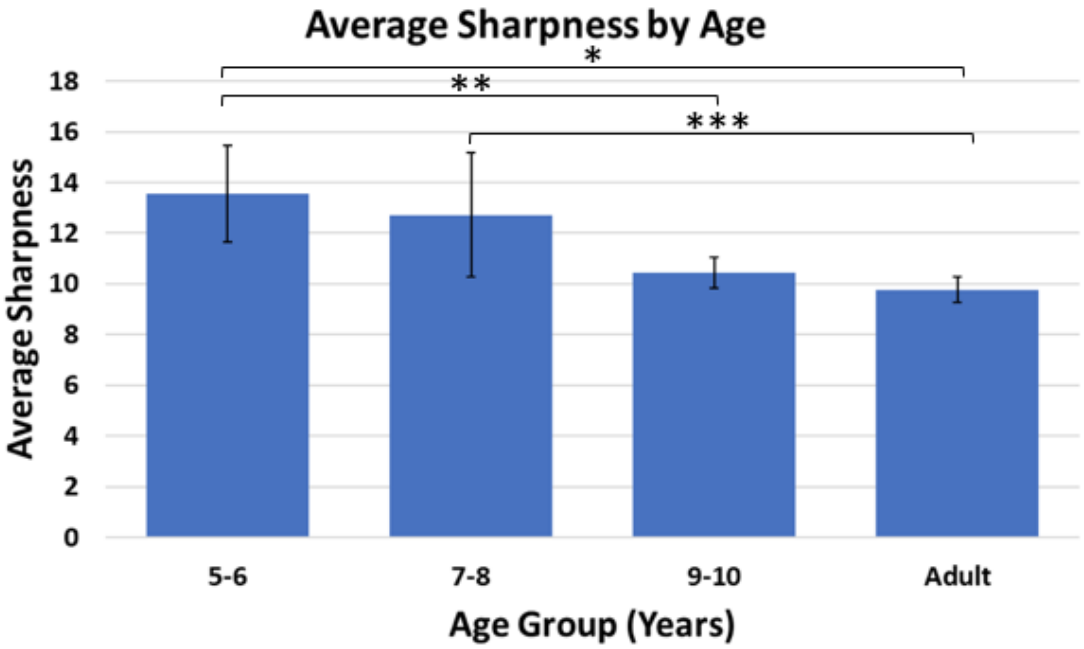

Figure 5-9: Effect of age group on average sharpness. Error bars represent the 95% confidence interval.

main effect of age group on average curviness ($F_{3,47} = 0.233$, *n.s.*). The value of average curviness is roughly the same across all age groups in our analysis, so it does not appear to be a good feature for discriminating across different ages. It may seem strange that curviness does not exhibit the same pattern as total turning angle and path length, but it shows that the ratio of the two features stays roughly the same across ages, so they both exhibit approximately the same variation with age. This may indicate that the two features are correlated, which is logical since they both rely on the angles of each point along the gesture. Figure 5-10 shows the effect of age group on average curviness.

Production Time. Production time of a gesture is equal to the total time taken to create it, including the time between strokes. The average procution time was 1,880.32 ms [SD = 628.35 ms] for 5- to 6-year-olds, 1,202.00 ms [SD = 326.02 ms] for 7- to 8-year-olds, 1,166.86 ms [SD = 264.26 ms] for 9- to 10-year-olds, and 894.89 ms [SD = 285.15 ms] for adults. A one-way ANOVA showed a significant main effect of age group on average production time ($F_{3,47} = 14.94$, $p < 0.001$). Post-hoc tests found a significant difference

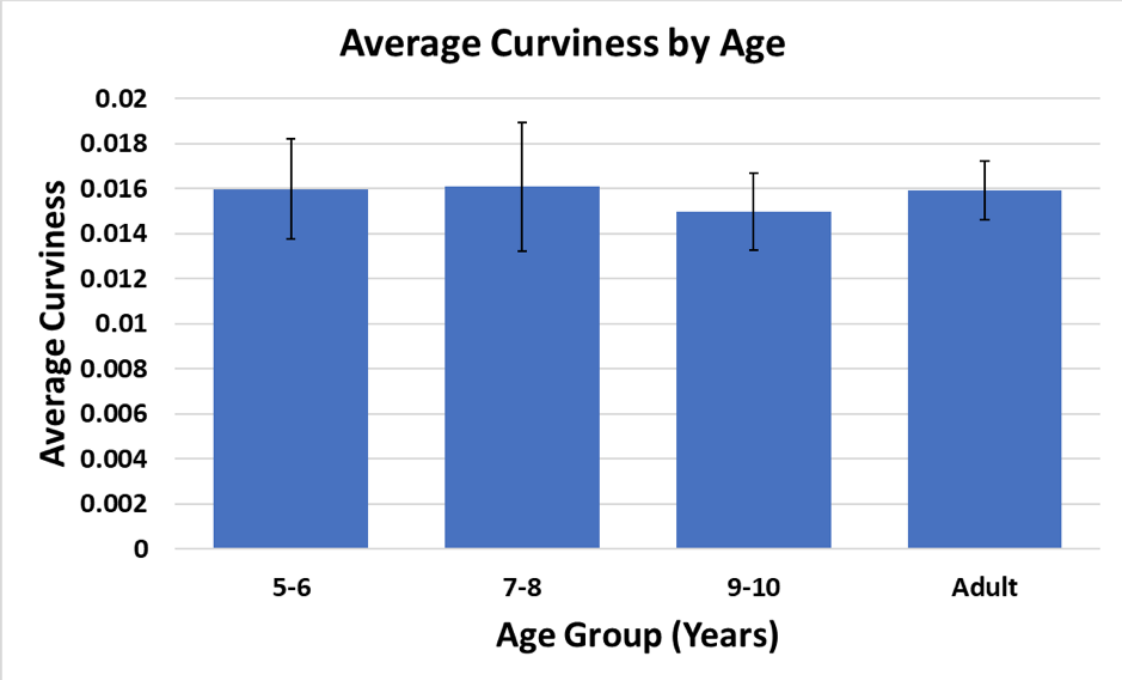

Figure 5-10: Effect of age group on average curviness. Error bars represent the 95% confidence interval.

Table 5-3: The average time between strokes of gestures (in milliseconds) for each of the age groups in our corpus.

5 to 6	7 to 8	9 to 10	Adults
1025.13	482.21	437.25	247.63

between 5-6 year-olds and 7-8 year-olds ($p < 0.05$), between 5-6 year-olds and 9-10 year-olds ($p < 0.001$), and between 5-6 year-olds and adults ($p < 0.001$). Production time is highest for the youngest age group, and decreases for older participants. We hypothesize that this is likely due to younger children's tendency to take breaks between strokes, as evidenced by the timestamps of the strokes in their gestures. This behavior causes them to take longer to produce a gesture. Table 5-3 shows the average amount of time between strokes for each age group, which confirms this pattern clearly. Figure 5-11 shows the effect of age group on average production time.

Average Speed. Average speed of a gesture is defined as path length divided by production time. The average speed was 0.59 px/ms [SD = 0.11 px/ms] for 5- to 6-year-olds, 0.76 px/ms [SD = 0.13 px/ms] for 7- to 8-year-olds, 0.80 px/ms [SD = 0.12 px/ms] for 9- to

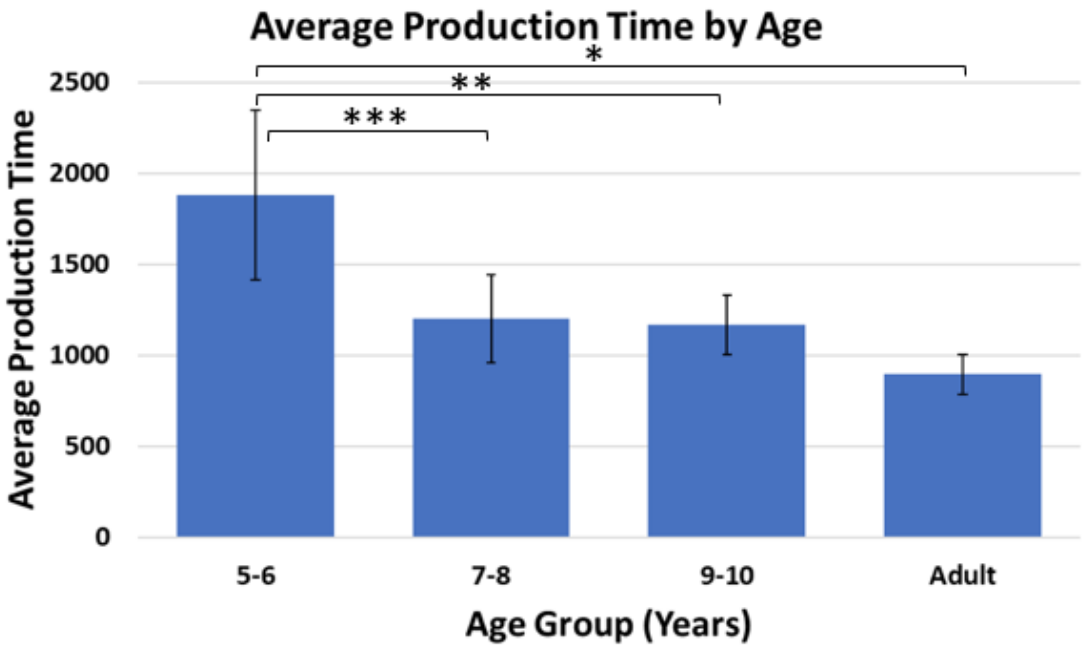

Figure 5-11: Effect of age group on average production time. Error bars represent the 95% confidence interval.

10-year-olds, and 0.015 px/ms [SD = 0.10 px/ms] for adults. A one-way ANOVA showed no significant main effect of age group on average speed ($F_{3,47} = 1.75$, *n.s.*). Thus, articulation speed does not vary significantly across the age groups. However, the average speed for the youngest age group is much lower than that of the other age groups, likely due to the fact that their average gesture production time is so much higher. Recognition algorithms typically account for differences in speed by resampling over the path of the gesture, so these differences do not affect the recognition process. Figure 5-12 shows the effect of age group on average speed.

5.1.1.2 Relative Accuracy Features

Relative accuracy features are computed between two gestures of the same type from the same user. The features quantify the level of consistency among a user's gestures. In prior work, researchers have calculated the values of the relative accuracy features by first calculating the task axis, which is an average over all of a user's gestures of the same type [10]. Then, the relative accuracy features are computed on each gesture using the task axis as the

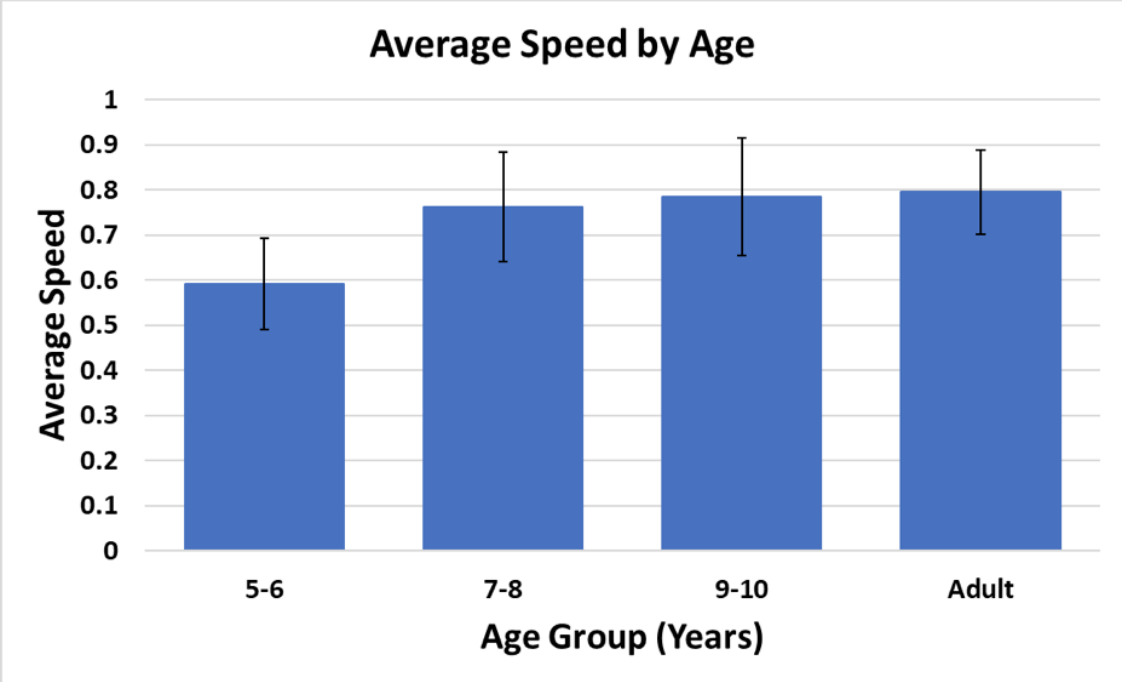

Figure 5-12: Effect of age group on average speed. Error bars represent the 95% confidence interval.

other input. In our work, rather than computing a task axis, we chose to compute the value of each relative accuracy between every pair of gestures of the same type, helping to maintain individual articulation differences that may be "washed out" by averaging.

Shape Error. Shape error refers to the average deviation between two gestures based on Euclidean distance. The average shape error was 37.82 px [SD = 12.17 px] for 5- to 6-year-olds, 29.05 px [SD = 5.29 px] for 7- to 8-year-olds, 25.71 px [SD = 5.03 px] for 9- to 10-year-olds, and 18.91 px [SD = 4.30 px] for adults. A one-way ANOVA showed a significant main effect of age group on shape error ($F_{3,55} = 22.92$, $p < 0.001$). Post-hoc tests found a significant difference between 5-6 year-olds and 7-8 year-olds ($p < 0.05$), between 5-6 year-olds and 9-10 year-olds ($p < 0.001$), between 5-6 year-olds and adults ($p < 0.001$), between 7-8 year-olds and adults ($p < 0.001$), and between 9-10 year-olds and adults ($p < 0.05$). Shape error is highest for the youngest age group, and decreases for older participants. Thus, younger children are less consistent in the way they draw the same gesture multiple times. Since template matching approaches rely on consistency between the shape of the

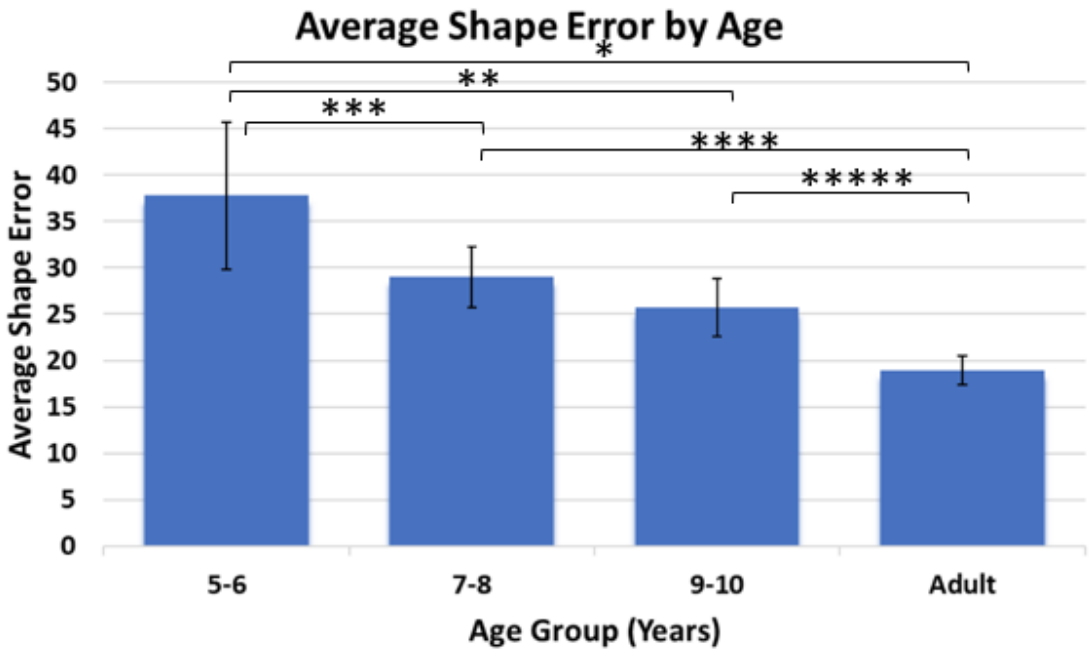

Figure 5-13: Effect of age group on average shape error. Error bars represent the 95% confidence interval.

gestures, we speculate that this feature is highly correlated with recognition rates. Figure 5-13 shows the effect of age group on average shape error.

Shape Variability. Shape variability is equal to the standard deviation of the distances between the points of two gestures. The average shape variability was 19.95 px [SD = 6.35 px] for 5- to 6-year-olds, 14.76 px [SD = 2.59 px] for 7- to 8-year-olds, 13.05 px [SD = 2.67 px] for 9- to 10-year-olds, and 9.55 px [SD = 2.30 px] for adults. A one-way ANOVA showed a significant main effect of age group on average shape variability ($F_{3,55} = 24.81$, $p < 0.001$). Post-hoc tests found a significant difference between 5-6 year-olds and 7-8 year-olds ($p < 0.05$), between 5-6 year-olds and 9-10 year-olds ($p < 0.001$), between 5-6 year-olds and adults ($p < 0.001$), between 7-8 year-olds and adults ($p < 0.001$), and between 9-10 year-olds and adults ($p < 0.05$). Shape variability is highest for the youngest age group, and decreases for older participants. This behavior shows that not only do younger children have a higher level of shape error, they have a larger range of shape errors. Thus, not only are children inconsistent in the way they make shapes, they are even inconsistent about the ways they are

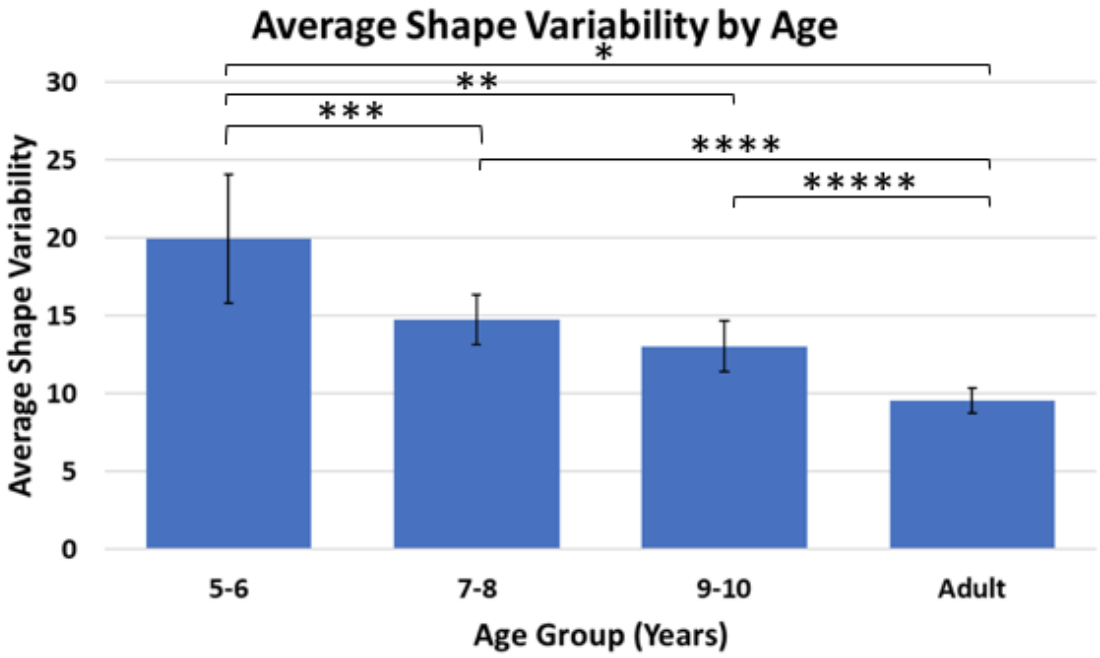

Figure 5-14: Effect of age group on average shape variability. Error bars represent the 95% confidence interval.

inconsistent. As with shape error, this has a direct impact on template-based recognizers since those recognizers rely on consistency between gestures. Figure 5-14 shows the effect of age group on average shape variability.

Length Error. Length error is a measure of the inconsistency of lengths of strokes between two gestures. The higher the value of length error, the more inconsistent the lengths are. The average length error was 188.97 px [SD = 92.63 px] for 5- to 6-year-olds, 153.09 px [SD = 91.80 px] for 7- to 8-year-olds, 99.76 px [SD = 18.56 px] for 9- to 10-year-olds, and 73.91 px [SD = 23.17 px] for adults. A one-way ANOVA showed a significant main effect of age group on length error ($F_{3,55} = 12.9$, $p < 0.001$). Post-hoc tests showed a significant difference between 5-6 year-olds and 9-10 year-olds ($p < 0.05$), between 5-6 year-olds and adults ($p < 0.001$), and between 7-8 year-olds and adults ($p < 0.05$). Length error is highest for the youngest participants, and decreases for older participants. Younger kids, therefore, have more inconsistency in the amount of ink used in their gestures. This behavior could be due to children's tendency to vary their gesture by, for example, writing in block letters,

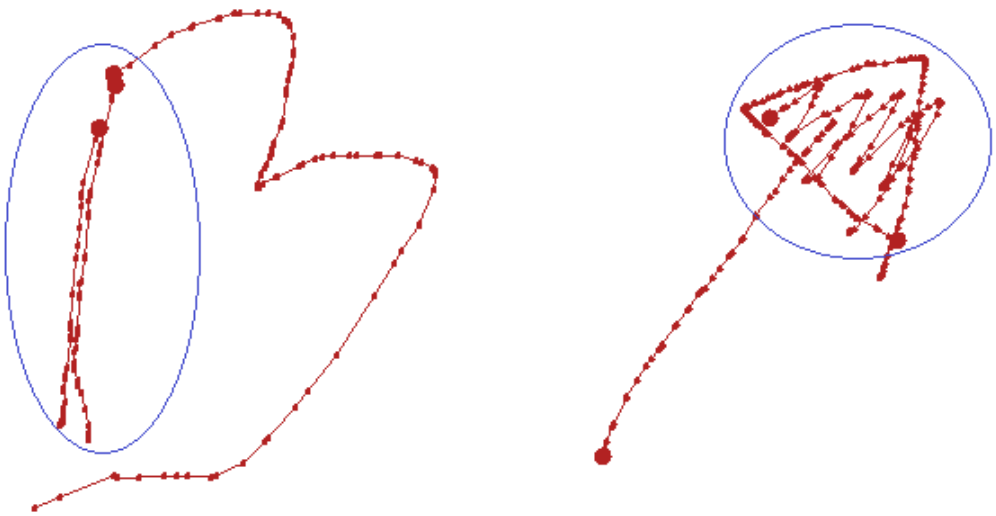

Figure 5-15: Examples of gestures from the corpus we used that exhibit behaviors that lead to high length error (circled).

embellishing their gestures with curly tails, or tracing over their strokes. Figure 5-15 shows examples of gestures exhibiting behaviors that may affect length error, and Figure 5-16 shows the effect of age group on average length error.

Size Error. Size error is a measure of the inconsistency between the areas of the bounding boxes of two gestures. The higher the value of size error, the more inconsistent the areas of the bounding boxes. The average size error was 27,731.25 px^2 [SD = 12,646.73 px^2] for 5- to 6-year-olds, 18,018.36 px^2 [SD = 6,923.18 px^2] for 7- to 8-year-olds, 15,141.78 px^2 [SD = 3,556.44 px^2] for 9- to 10-year-olds, and 10,272.31 px^2 [SD = 3,565.78 px^2] for adults. A one-way ANOVA showed a significant main effect of age group on size error ($F_{3,55}$ = 18.56, $p < 0.001$). Post-hoc tests showed a significant difference between 5-6 year-olds and 7-8 year-olds ($p < 0.05$), between 5-6 year-olds and 9-10 year-olds ($p < 0.001$), between 5-6 year-olds and adults ($p < 0.001$), and between 7-8 year-olds and adults ($p < 0.05$). Size error is the highest for the youngest participants, and decreases for older participants. Thus, there is a greater discrepancy between the sizes of gestures of the same type elicited from younger participants than from older participants. Figure 5-17 shows the effect of age group on average path length.

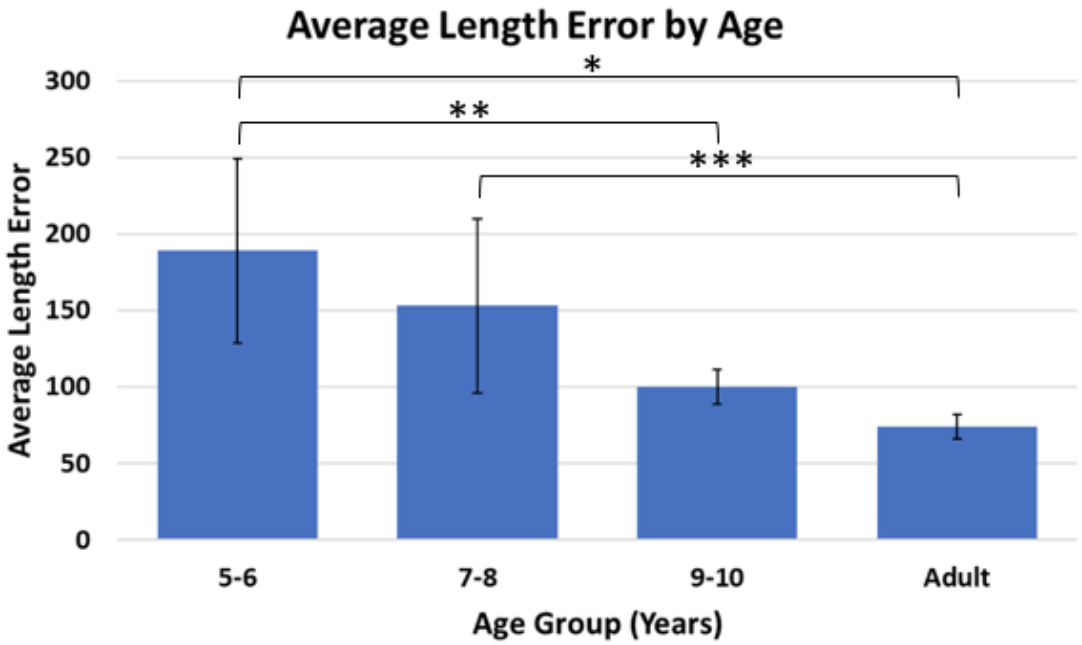

Figure 5-16: Effect of age group on average length error. Error bars represent the 95% confidence interval.

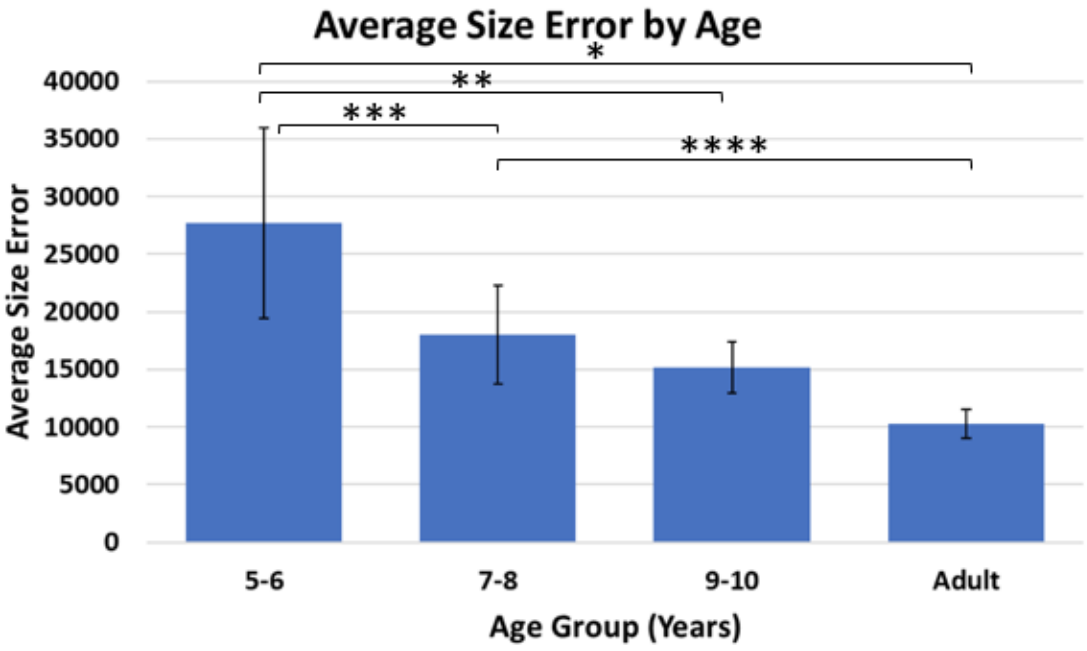

Figure 5-17: Effect of age group on average size error. Error bars represent the 95% confidence interval.

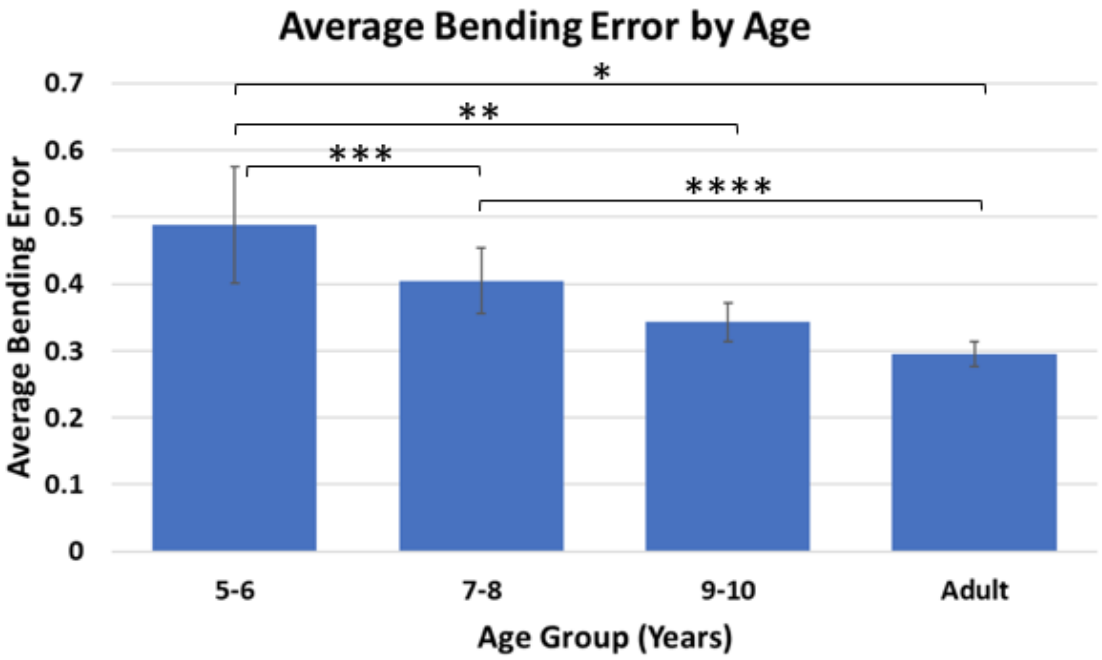

Figure 5-18: Effect of age group on average bending error. Error bars represent the 95% confidence interval.

Bending Error. Bending error refers to the average of differences between corresponding turning angles of two gestures of the same type. The average bending error was 0.49 deg [SD = 0.13 deg] for 5- to 6-year-olds, 0.40 deg [SD = 0.08 deg] for 7- to 8-year-olds, 0.34 deg [SD = 0.05 deg] for 9- to 10-year-olds, and 0.30 deg [SD = 0.05 deg] for adults. A one-way ANOVA showed a significant main effect of age group on bending error ($F_{3,55} = 18.17$, $p < 0.001$). Post-hoc tests showed a significant difference between 5-6 year-olds and 9-10 year-olds ($p < 0.001$), between 5-6 year-olds and adults ($p < 0.001$), and between 7-8 year-olds and adults ($p < 0.001$). Bending error is highest for the youngest age group, and decreases for older participants. Young children are less consistent in the angles made during the articulation of their gestures, indicating that they often do not take the same path when drawing a gesture multiple times. Figure 5-18 shows the effect of age group on average bending error.

Bending Variability. Bending variability refers to the standard deviation of differences between corresponding turning angles of two gestures of the same type. The average bending variability was 0.73 deg [SD = 0.10 deg] for 5- to 6-year-olds, 0.67 deg [SD = 0.08 deg] for

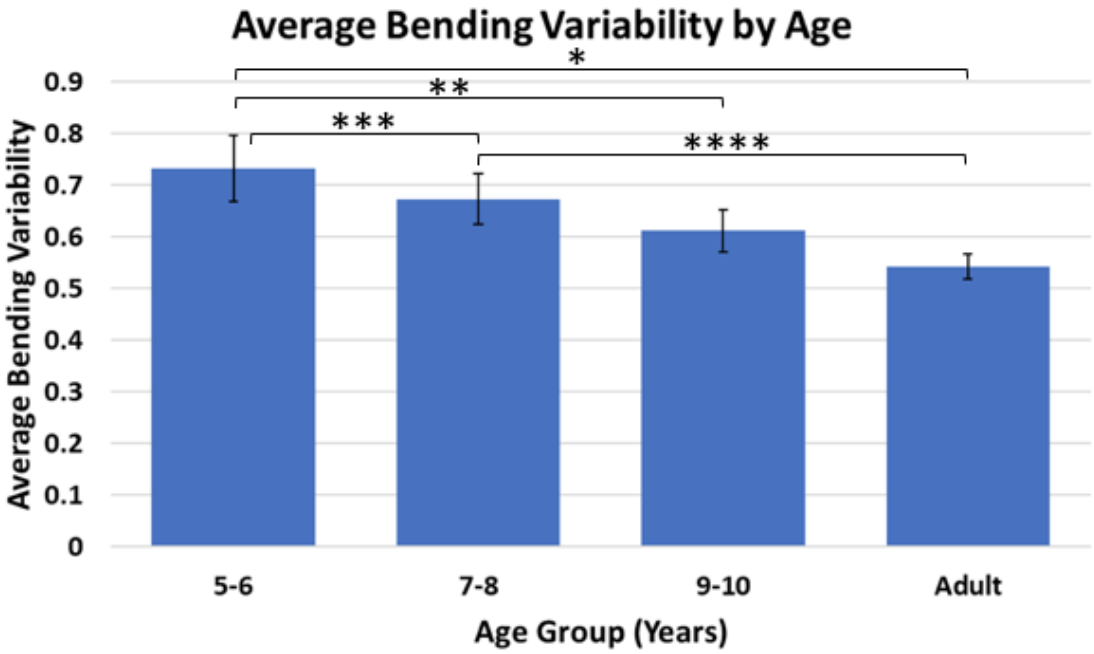

Figure 5-19: Effect of age group on average bending variability. Error bars represent the 95% confidence interval.

7- to 8-year-olds, 0.61 deg [SD = 0.66 deg] for 9- to 10-year-olds, and 0.54 deg [SD = 0.07 deg] for adults. A one-way ANOVA showed a significant main effect of age group on bending variability ($F_{3,55} = 18.78$, $p < 0.001$). Post-hoc tests showed a significant difference between 5-6 year-olds and 7-8 year-olds ($p < 0.001$), between 5-6 year-olds and 9-10 year-olds ($p < 0.001$), between 5-6 year-olds and adults ($p < 0.001$), and between 7-8 year-olds and adults ($p < 0.001$). Bending variability is highest for the youngest participants and decreases for older participants. Thus, younger children tend to not only have the highest average difference in corresponding angles on their gesture articulation path, but they also tend to have a wider variety of values. Figure 5-19 shows the effect of age group on average bending variability.

Time Error Time error refers to the difference in the amount of time taken to articulate two gestures. The average time error was 1,076.90 ms [SD = 503.11 ms] for 5- to 6-year-olds, 563.65 ms [SD = 327.76 ms] for 7- to 8-year-olds, 387.60 ms [SD = 161.30 ms] for 9- to 10-year-olds, and 190.66 ms [SD = 103.43 ms] for adults. A one-way ANOVA showed a significant main effect of age group on time error [$F_{3,55} = 29.44$, $p < 0.001$]. Post-hoc tests

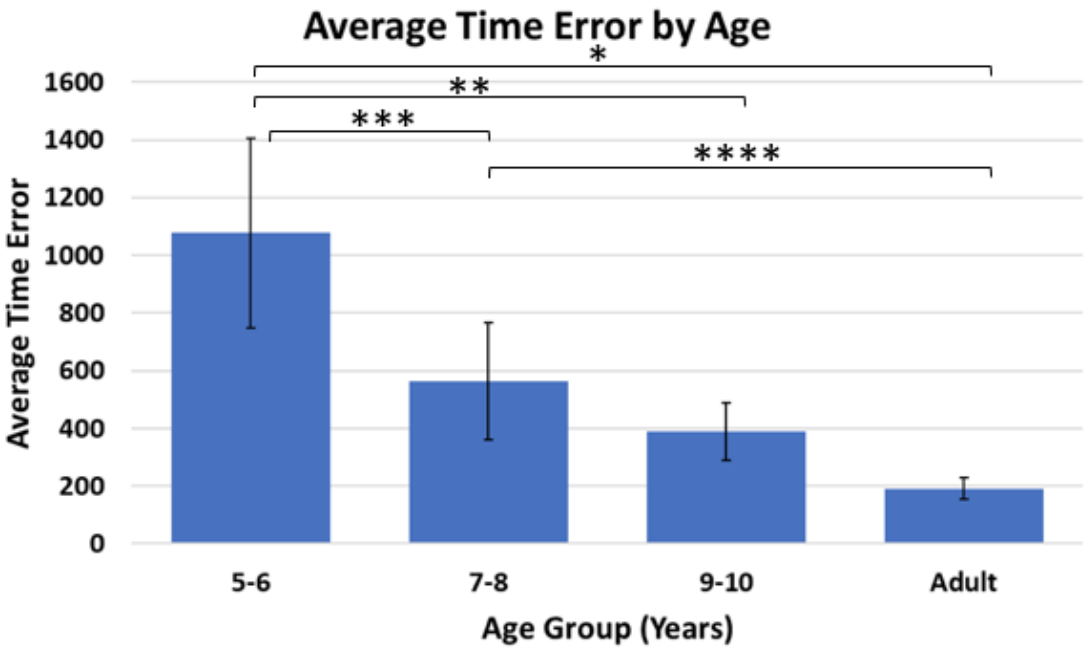

Figure 5-20: Effect of age group on average time error. Error bars represent the 95% confidence interval.

showed a significant difference between 5-6 year-olds and 7-8 year-olds ($p < 0.001$), between 5-6 year-olds and 9-10 year-olds ($p < 0.001$), between 5-6 year-olds and adults ($p < 0.001$), and between 7-8 year-olds and adults ($p < 0.001$). Time error is the highest for the youngest group of participants and decreases for older participants. Younger children have a much larger average discrepancy in the time taken to produce different samples of the same type of gesture. Figure 5-20 shows the effect of age on time error.

Time Variability. Time variability refers to the standard deviation of the differences of the timestamps of each individual point in a gesture. The average time variability was 645.73 ms [SD = 259.83 ms] for 5- to 6-year-olds, 354.90 ms [SD = 153.73 ms] for 7- to 8-year-olds, 266.90 ms [SD = 75.01 ms] for 9- to 10-year-olds, and 171.86 ms [SD = 54.70 ms] for adults. A one-way ANOVA showed a significant main effect of age group on time variability ($F_{3,55} = 33.05$, $p < 0.001$). Post-hoc tests showed a significant difference between 5-6 year-olds and 7-8 year-olds ($p < 0.001$), between 5-6 year-olds and 9-10 year-olds ($p < 0.001$), between 5-6 year-olds and adults ($p < 0.001$), and between 7-8 year-olds and adults ($p < 0.001$).

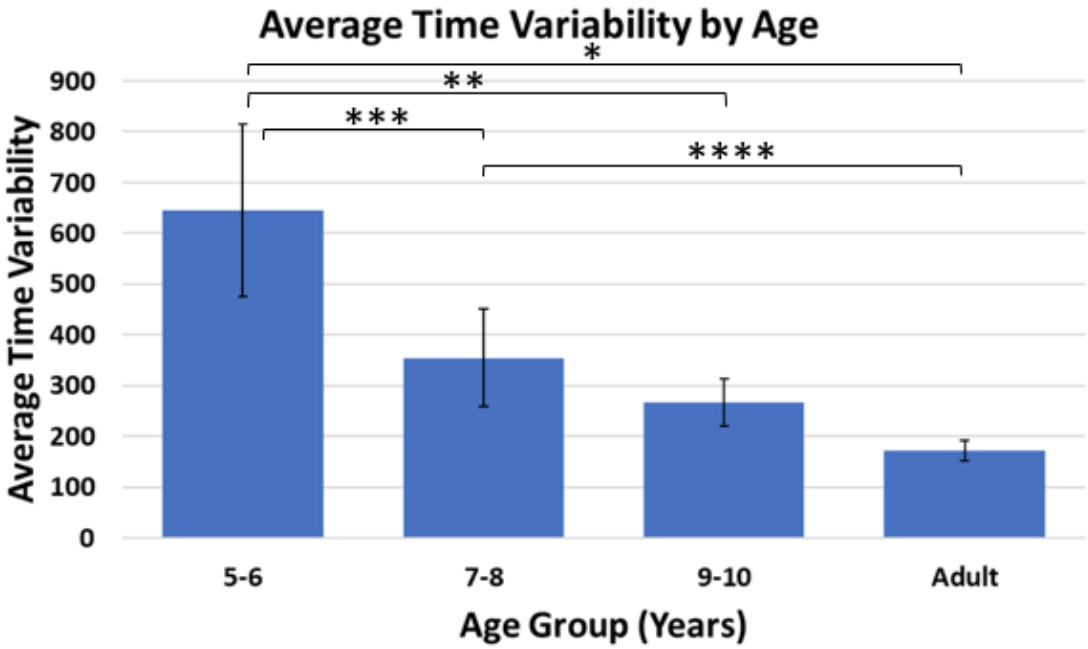

Figure 5-21: Effect of age group on average time error. Error bars represent the 95% confidence interval.

Time variability is the highest for the youngest group of participants and decreases for older participants. Thus, not only do younger children tend to have the most inconsistency in the overall amount of time taken to produce a gesture, they also show the most inconsistency in the amount of time it takes them to articulate each individual point along the path of the gesture. Figure 5-21 shows the effect of age group on average time variability.

Speed Error. Speed error refers to the difference in speed of production of two gestures. The average speed error was 1.16 px/ms [SD = 0.43 px/ms] for 5- to 6-year-olds, 0.94 px/ms [SD = 0.25 px/ms] for 7- to 8-year-olds, 0.85 px/ms [SD = 0.27 px/ms] for 9- to 10-year-olds, and 0.70 px/ms [SD = 0.21 px/ms] for adults. A one-way ANOVA showed a significant main effect of age group on speed error ($F_{3,55} = 7.289$, $p < 0.001$). Post-hoc tests showed a significant difference between 5-6 year-olds and adults ($p < 0.001$). Speed error is highest for the youngest age group, and decreases for older participants. There is more variation in the speeds of children's gestures than adults'. Figure 5-22 shows the effect of age group on average speed error.

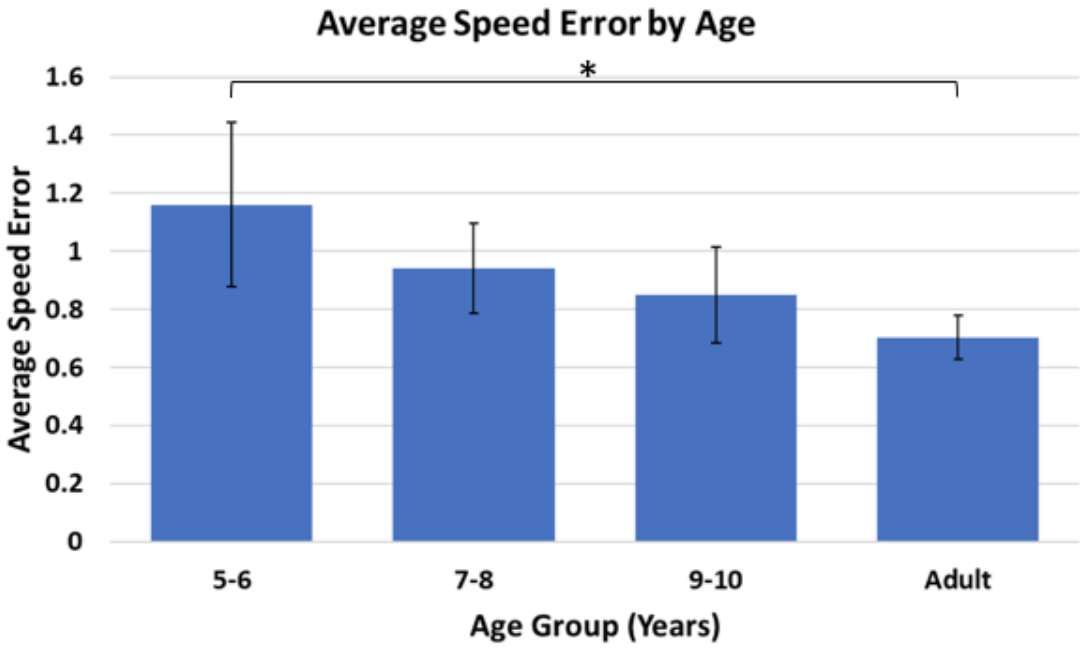

Figure 5-22: Effect of age group on average speed error. Error bars represent the 95% confidence interval.

Speed Variability. Speed variability refers to the standard deviation of differences in the speed of production of two gestures. The average speed variability was 1.43 px/ms [SD = 0.66 px/ms] for 5- to 6-year-olds, 1.19 px/ms [SD = 0.35 px/ms] for 7- to 8-year-olds, 1.09 px/ms [SD = 0.39 px/ms] for 9- to 10-year-olds, and 0.86 px/ms [SD = 0.28 px/ms] for adults. A one-way ANOVA showed a significant main effect of age group on speed variability ($F_{3,55} = 6.02$, $p < 0.001$). Post-hoc tests showed a significant difference between 5-6 year-olds and adults ($p < 0.05$). Speed error is highest for the youngest age group, and decreases for older participants. This is consistent with our findings for time variability, and it should also be noted that the calculation of speed variability depends on speed error. Figure 5-23 shows the effect of age group on average speed variability.

Stroke Count Error. Stroke count error refers to the difference in number of strokes of two gestures of the same type. The average speed error was 0.35 [SD = 0.20] for 5- to 6-year-olds, 0.25 [SD = 0.17] for 7- to 8-year-olds, 0.12 [SD = 0.07] for 9- to 10-year-olds, and 0.10 [SD = 0.10] for adults. A one-way ANOVA showed a significant main effect of age

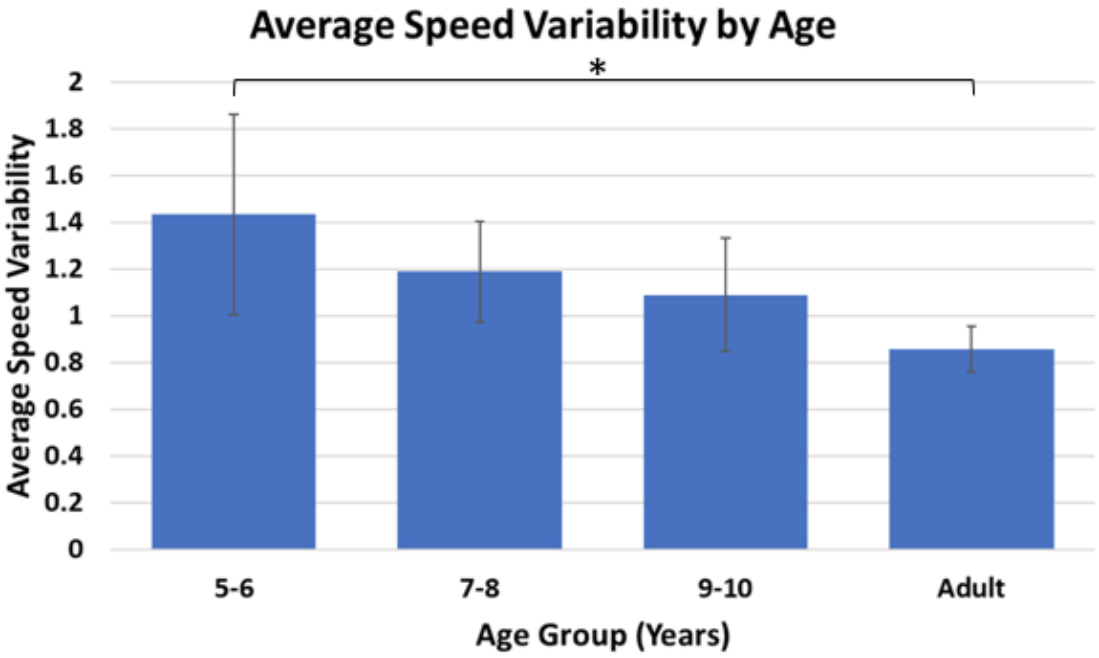

Figure 5-23: Effect of age group on average speed variability. Error bars represent the 95% confidence interval.

group on stroke count error ($F_{3,55} = 9.94$, $p < 0.001$). Post-hoc tests showed a significant difference between 5-6 year-olds and 9-10 year-olds ($p < 0.05$), between 5-6 year-olds and adults ($p < 0.001$), and between 7-8 year-olds and adults ($p < 0.05$). Stroke count error is highest for the youngest children, and decreases for older users. Younger children tend to have higher variation in the number of strokes used. Figure 5-24 shows the effect of age group on average stroke count error.

Stroke Ordering Error. Stroke ordering error is a measure of the inconsistency in the order that different strokes of a gesture are drawn between two samples of that gesture. The stroke ordering error was 2,292.88 [SD = 907.72] for 5- to 6-year-olds, 1,359.76 [SD = 529.12] for 7- to 8-year-olds, 1,049.64 [SD = 387.52] for 9- to 10-year-olds, and 804.29 [SD = 284.31] for adults. A one-way ANOVA showed a significant main effect of age group on stroke ordering error ($F_{3,55} = 22.71$, $p < 0.001$). Post-hoc tests showed a significant difference between 5-6 year-olds and 7-8 year-olds ($p < 0.001$), between 5-6 year-olds and 9-10 year-olds ($p < 0.001$), between 5-6 year-olds and adults ($p < 0.001$), and between 7-8 year-olds and

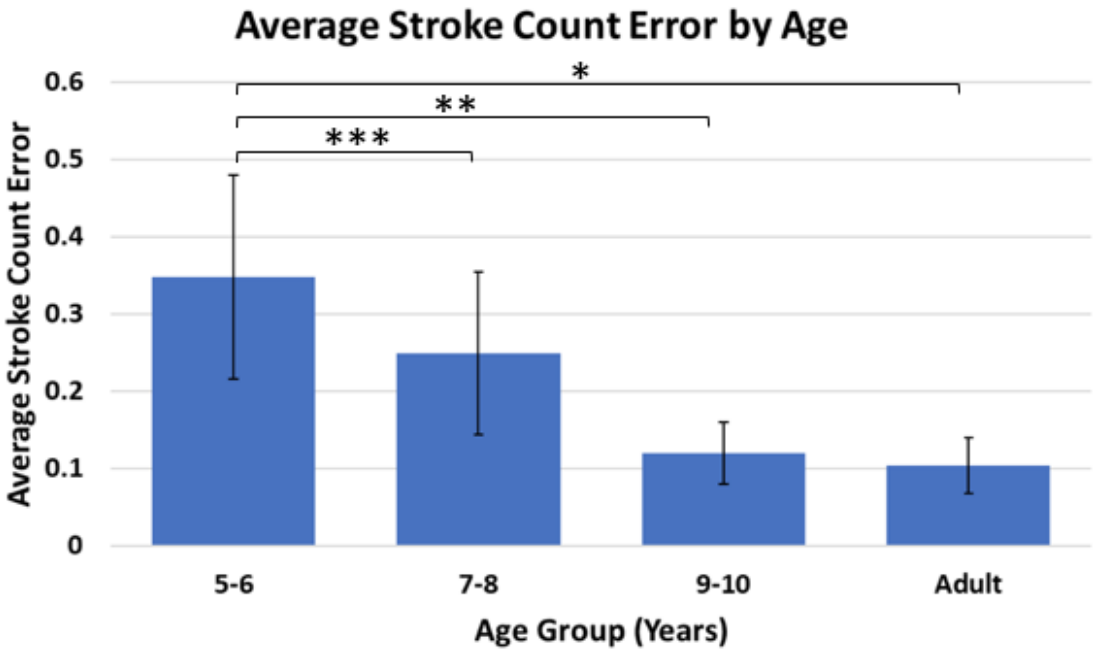

Figure 5-24: Effect of age group on average speed variability. Error bars represent the 95% confidence interval.

adults ($p < 0.05$). Stroke ordering error is highest for the youngest age group, and decreases for older participants. Younger children tend to have more variation in the ordering of their strokes. Figure 5-25 shows the effect of age group on average stroke ordering error.

5.1.1.3 Discussion

In general, we found that relative accuracy features appear to be much more discriminative among different age groups than simple geometric or kinematic features. In our analysis, all 12 of the relative accuracy features showed a significant main effect of age group on the value of the features, but only 6 out of 10 of the simple features did. Relative features also showed more post-hoc differences by age. This finding indicates that the discrepancy in recognition rates between children's and adults' gestures is not a function of individual gestures in isolation, but rather of the way different age groups show different levels of consistency. All of the relative features show the same general pattern as recognition rates for those age groups [113], indicating they may be strongly associated with recognition accuracy.

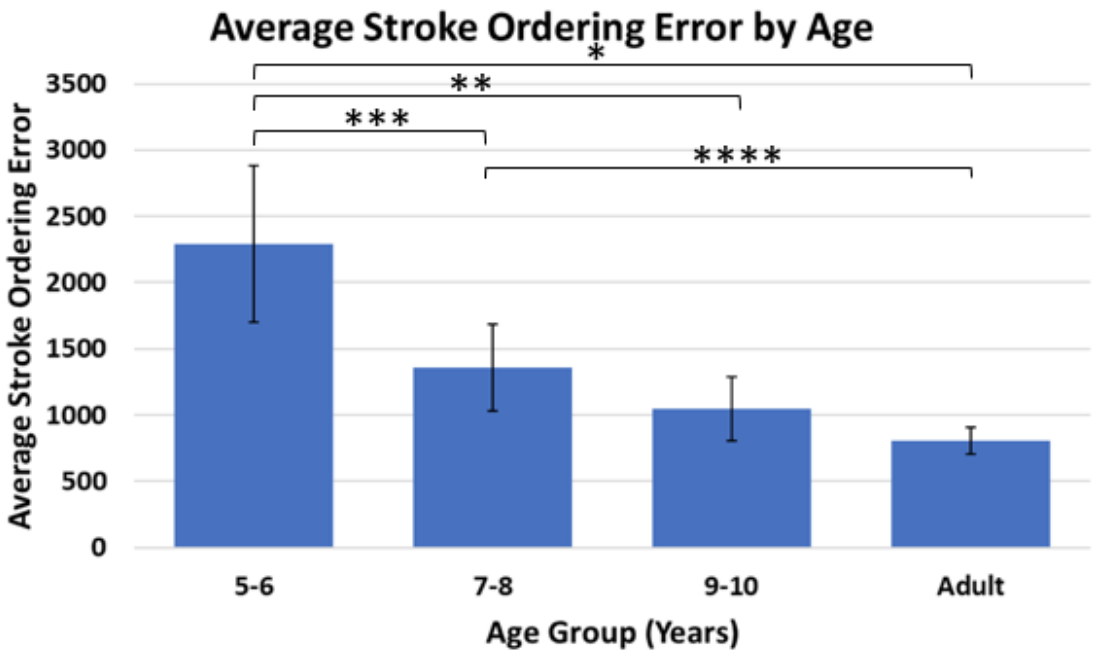

Figure 5-25: Effect of age group on average speed variability. Error bars represent the 95% confidence interval.

5.2 Child-Specific Articulation Features

While our work on existing features helped gain a deeper understanding of children's articulation patterns, we still felt there were aspects of children's gestures that were not being captured by existing features. The features used in our study were originally designed with well-formed adults' gestures in mind, but children often produce much noisier gestures that deviate from the canonical forms of the gestures. Because of this, we decided to develop a set of new touchscreen articulation features that can quantify these behaviors. We systematically analyzed the touchscreen gestures from our prior work (presented in Chapter 4 [113] to look for common patterns in children's gestures. We developed a total of six new articulation features: joining error, number of enclosed areas, rotation error, proportion of gesture in "tails", average percentage of stray ink, and disconnectedness.

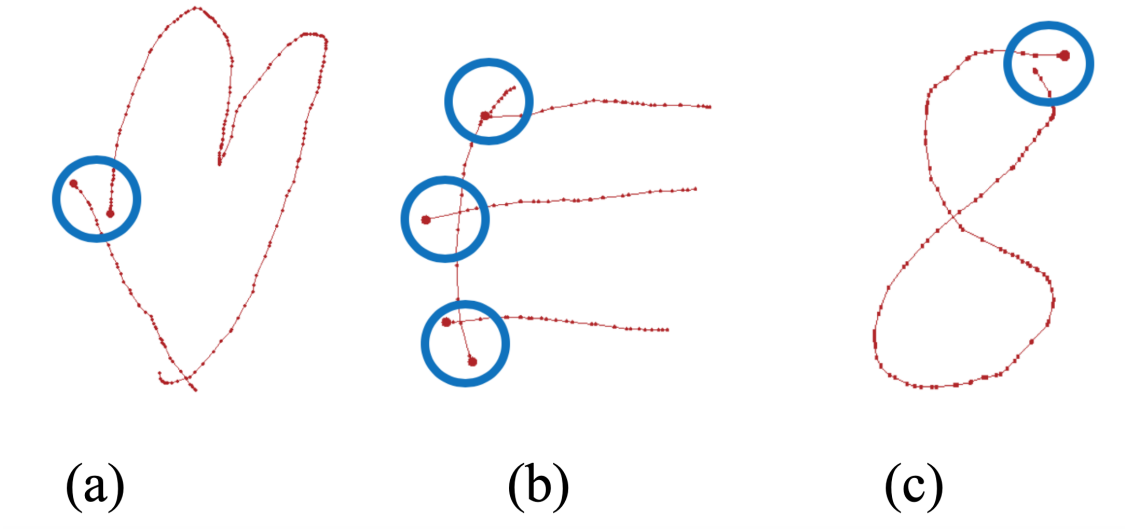

Figure 5-26: An example of (a) a heart, (b) an E, and (c) an 8 gesture exhibiting joining error. The strokes that should be joined are circled.

5.2.1 Description of the Features

5.2.1.1 Joining Error

Based on our observations from Woodward et al.'s dataset [113], children often have trouble correctly joining the ends of their strokes when they should meet (such as, for example, the top of an A gesture). This is partially because children are still developing their motor skills, including proprioception [73]. To capture this behavior, we introduce *joining error*, which is defined as the sum of the distances between all points on a gesture that should be connected. Figure 5-26 shows three examples of joining error.

5.2.1.2 Number of Enclosed Areas

When gesturing, children sometimes have trouble maintaining the orientation of the ink they are creating compared to what they have already drawn. As a result, children often create ink that overlaps, leading to an increased *number of enclosed* areas of blank canvas completely surrounded by ink, indicating a lack of proper stroke alignment. For example, we would expect an 8 gesture to have exactly two enclosed areas, but often children's gestures have more enclosed areas. Figure 5-27 shows how the number of enclosed areas is counted.

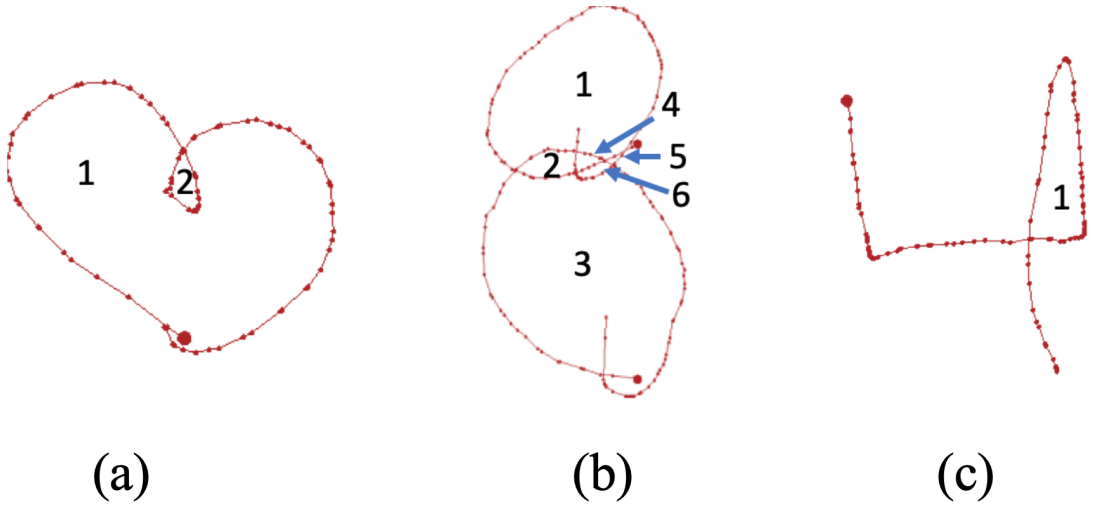

Figure 5-27: An example of (a) a heart, (b) an 8, and (c) a four gesture with the enclosed areas labeled.

5.2.1.3 Rotation Error

In our analysis of Woodward et al.'s [113] dataset, we noticed that children's gestures are often slightly rotated from the standard, canonical form of the gesture (e.g., the canonical arrowhead would point upward in Figure 5-28), and there is often variation in the rotational alignment between children's gestures of the same type. Some point matching approaches attempt to find an optimal rotational alignment between candidate and template gestures [63, 111], while others do not [103], so this behavior could affect the success of recognition of children's gestures. To help better understand the extent to which children exhibit the behavior of variably rotating their gestures, we introduced rotation error, a feature computed between two gestures of the same type, similar to Vatavu et al.'s [104] relative accuracy measures like shape error and length error. To compute the rotation error between two gestures, we first find the line between each gesture's initial point and its centroid. The rotation error is equal to the angle between these two lines. We use Li's [63] closed form solution to angular alignment to find the angle between the lines. Figure 5-28 shows how rotation error is found.

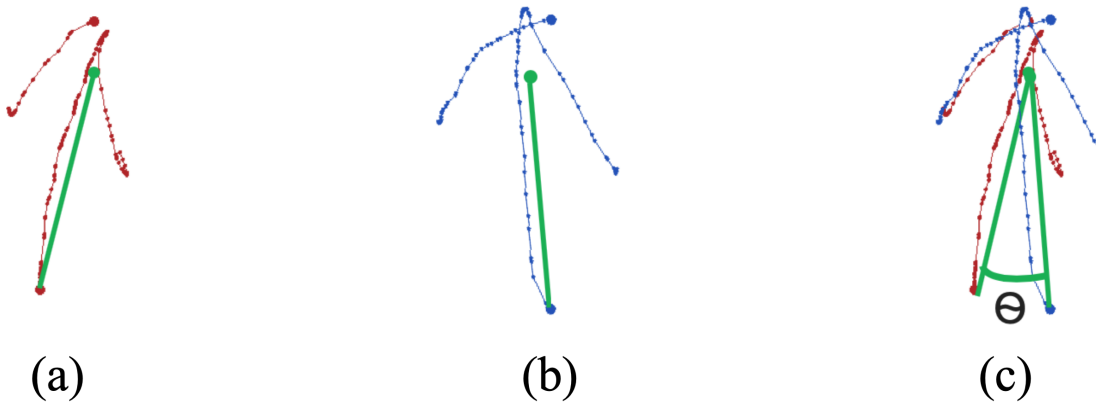

Figure 5-28: An example of how rotation error is found, from two raw input arrowhead gestures, shown in (a) and (b). The line between each of the gestures' first point and centroid is found (shown in green), and the rotation error is the angle between these two lines, indicated by the thick green line (c).

5.2.1.4 Proportion of Extra Gesture in "Tails"

In our investigation of Woodward et al.'s [113] dataset, we noted that children sometimes exhibit a behavior in which the ends of their gestures have small sections where the path sharply turns, which we refer to as tails. We define a tail as a part of a gesture that quickly changes from its prior trajectory near the end of a stroke. Tails can be created intentionally or unintentionally, and may partially be caused by children's pen or fingers slipping when gesturing, which has been observed in children's gestures in previous work in touchscreens and mouse-based interaction [44, 106]. In order to quantify the magnitude of this behavior, we computed the average amount of ink per gesture that was part of a tail and divided it by the total amount of ink in the gesture. Figure 5-29 shows examples of tails in gestures from the dataset we use.

5.2.1.5 Average Percentage of Stray Ink

When creating gestures, children often make strokes that are clearly not intended to be part of the gesture, which we refer to here as a stray stroke. We refer to the ink that comprises a stray stroke as stray ink. We computed the total length of stray ink divided by the amount of ink in the whole gesture. Figure 5-30 shows three examples of gestures with a high amount of stray ink compared to the total ink in the gesture.

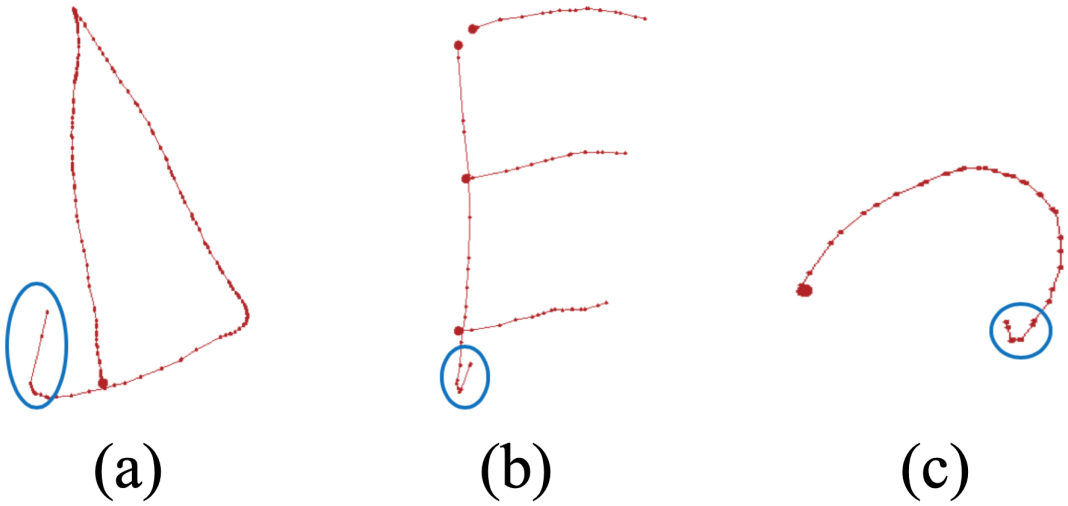

Figure 5-29: An example of (a) a triangle, (b) an E, and (c) an arch gesture, each with a tail circled. The tail is defined as the part of the gesture that sharply turns from its prior trajectory. We measured the length of each tail to calculate the proportion of ink.

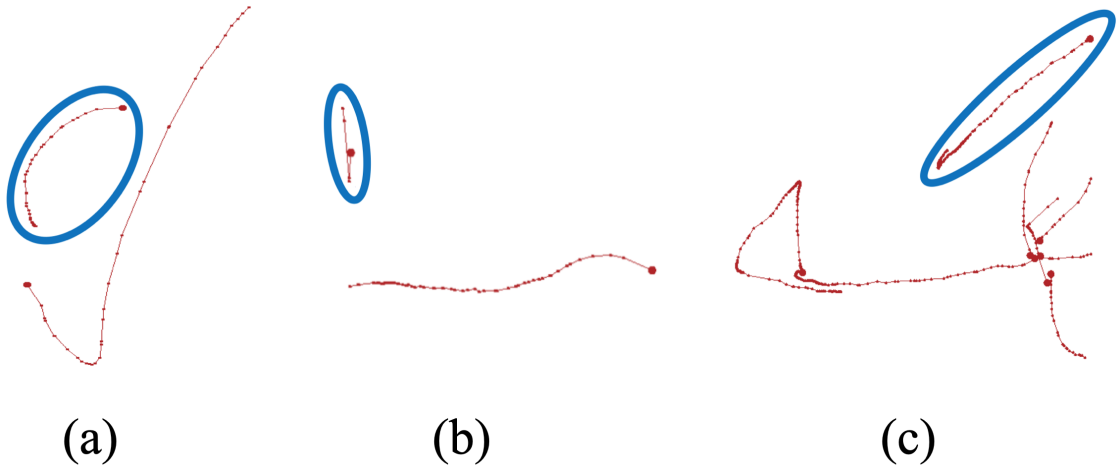

Figure 5-30: Examples of gestures from the corpus we use exhibiting high amounts of stray ink.

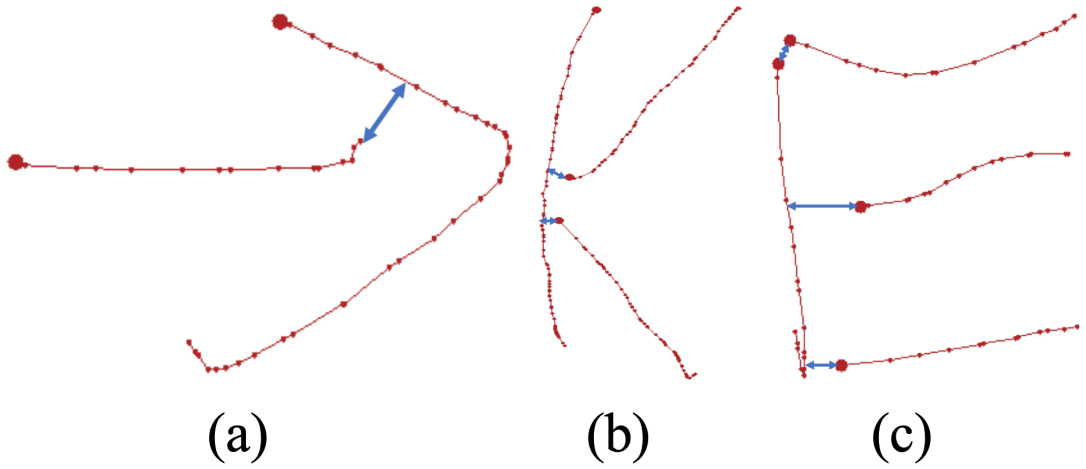

Figure 5-31: Examples of gestures from the corpus we use exhibiting joining error.

5.2.1.6 Disconnectedness

Another behavior we noticed in our examination of our gesture corpus [113] was that children sometimes exhibit a behavior wherein the non-stray strokes are quite far apart. To quantify this, we developed a feature called disconnectedness, which is defined as the average distance from each stroke in a gesture to the closest point on another stroke. In other words, disconnectedness measures the minimum amount of ink that would be required to connect all strokes, normalized over the number of strokes. Figure 5-31 shows examples of how disconnectedness is computed.

5.2.2 Annotating the Features

Our initial plan for computing these features was to develop algorithms that could automatically calculate them given raw gesture input. We were able to design simple algorithms to calculate the values of number of enclosed areas, rotation error, and disconnectedness. However, automatically calculating the remaining features would be very difficult to perform with high precision. For example, the stray strokes could be calculated by examining the distance from each stroke to the nearest stroke and marking it as stray if it was less than a certain threshold. However, because this method relies on a threshold, it is possible that some strokes would be marked as stray when they were not and vice versa.

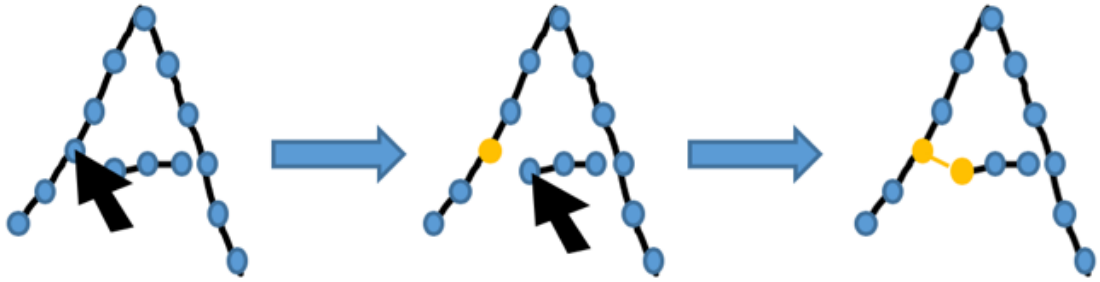

Figure 5-32: An example of how our annotation tool works. The user marks joining error by clicking two points that should be connected, then the system marks them.

Thus, this method is error prone and not guaranteed to obtain perfect accuracy. Since this dissertation is the first introduction of these features and how they may be useful, we wanted to ensure we determined the values exactly. Thus, we decided to manually mark the stray strokes. Joining error and length of tails are similarly easy to identify manually but difficult to compute exactly. We developed a tool for manually annotating the gestures to find joining error, tails, and stray strokes (features 1, 4, and 5 respectively). Figure 5-32 illustrates the process of annotating joining error using our tool. The size of each point along the gesture is amplified to allow easily clicking any point. For annotating tails, we visually examined the end of each stroke to determine whether there was a sharp change in trajectory, marking its beginning and endpoints. After marking the points, our annotation tool automatically found the distance between the points. To annotate the stray strokes, we simply clicked on the stroke of interest to mark it as stray, then our tool logged relevant information about the stroke, including its length. To annotate joining error, we configured the tool to allow us to click on two points in a gesture to mark them as intended to be joined. Our tool then indicated the selection and logged the distance between the points. During the annotation process, we did not display the age of the participants who created the gestures being shown.

We used manual annotation to exactly determine joining error, proportion of tails, and stray ink percentage. However, these features would need to be calculated automatically to be used in live recognition. We offer a few suggestions on how this could be implemented in future work. For joining error, strokes that should be connected could be detected by assuming

Table 5-4: The percentage of gestures with nonzero values for each feature by age group.

	Joining Error	Enclosed Areas	Tails	Stray Ink	Disconnectedness
5-6yo	47.85%	31.40%	19.77%	5.85%	49.41%
7-8yo	49.82%	27.04%	18.73%	3.30%	45.72%
9-10yo	45.66%	26.68%	12.58%	2.03%	47.75%
Adult	47.25%	27.55%	17.15%	0.53%	47.05%
AVERAGE	47.62%	27.55%	17.06%	2.93%	46.08%

they should be connected when the distance between them is below a certain threshold. Tails could be detected by examining the velocity profile of the gestures and looking for sharp changes near the end of strokes. As mentioned earlier, stray ink could be identified by looking for very short strokes or by looking for strokes far away from the rest of the gesture.

5.2.3 Results and Analysis

5.2.3.1 Occurrence of Features

Most of the features we have introduced quantify behaviors that are not present in all gestures. For example, joining error captures failure to properly join two strokes, but this behavior is not possible in simple unistroke gestures like line and checkmark. We thus assign a value of zero for the feature when the behavior is not present or when the behavior is not possible in the given gesture. Table 5-4 shows the frequency of nonzero values in each age group, which gives an idea of what percentage of users in each age group exhibited the behavior for gesture types in which that behavior is possible. Rotation error is omitted from the table since all users exhibited this behavior. While the percentage of gestures that exhibit the behaviors is relatively similar for most of the features, the magnitudes of these values shows more variation with age group, as we will show in the next section.

5.2.3.2 Effect of Age

We begin our analysis by examining the effect of age on each of our new features, as we did with the simple and relative accuracy features in section 5.1.1. To help facilitate comparison between the new feature work and our previous work on children's articulation features, we use the same age groupings in our analysis: 5- to 6-year-olds, 7- to 8-year-olds,

9- to 10-year-olds, and adults [90]. We also use the same gesture dataset from our prior work [113]. We examine the effect of age group on each of our six new features using a one-way repeated measures ANOVA on the magnitude of the value of the feature with age group as a between-subjects factor. Note that each ANOVA is calculated on the value of each feature, not the percentage of gestures in which it occurs. For each feature, we also examine its correlation with recognition accuracy to help show which features are associated with poor recognition.

Joining Error. The average rotation error was 16.60 px [SD = 3.46] for 5- to 6-year-olds, 16.45 px [SD = 6.41 px] for 7- to 8-year-olds, 14.30 [SD = 4.64 px] for 9- to 10-year-olds, and 12.08 px [SD = 4.25 px] for adults. A one-way ANOVA on joining error with a between-subjects factor of age group found a significant main effect of age group on joining error ($F_{3,56} = 3.64$, $p < 0.05$). Joining error was approximately the same for 5- to-6-year-olds and 7- to 8-year-olds, with adults having a lower value, as shown in Figure 10a. A Tukey post-hoc test found a significant difference between 7- to 8-year-olds and adults ($p < 0.05$) and a marginally significant difference between 5- to 6-year-olds and adults ($p < 0.1$). Thus, we confirm our hypothesis the magnitude of joining error exhibited by children's gestures is greater than that of adults. Figure 5-33 shows the effect of age group on average joining error.

Number of Enclosed Areas. The average number of enclosed areas was 0.59 [SD = 0.40] for 5- to 6-year-olds, 0.58 [SD = 0.46] for 7- to 8-year-olds, 0.35 [SD = 0.13] for 9- to 10-year-olds, and 0.30 [SD = 0.10] for adults. A one-way ANOVA on number of enclosed areas with a between-subjects factor of age group found a significant main effect of age group on number of enclosed areas ($F_{3,55} = 4.89$, $p < 0.05$). Number of enclosed areas is highest for the youngest children and increases for older children. The number of enclosed areas was similar for 5- to 6-year-olds and 7- to 8-year-olds, both of which had much higher values than adults, as shown in Figure 5-34. A Tukey post-hoc test found significant differences between 5- and 6-year-olds and adults, and 7- to 8-year-olds and adults ($p < 0.05$). Younger children tend to have a higher number of enclosed areas, which may be because children have more trouble

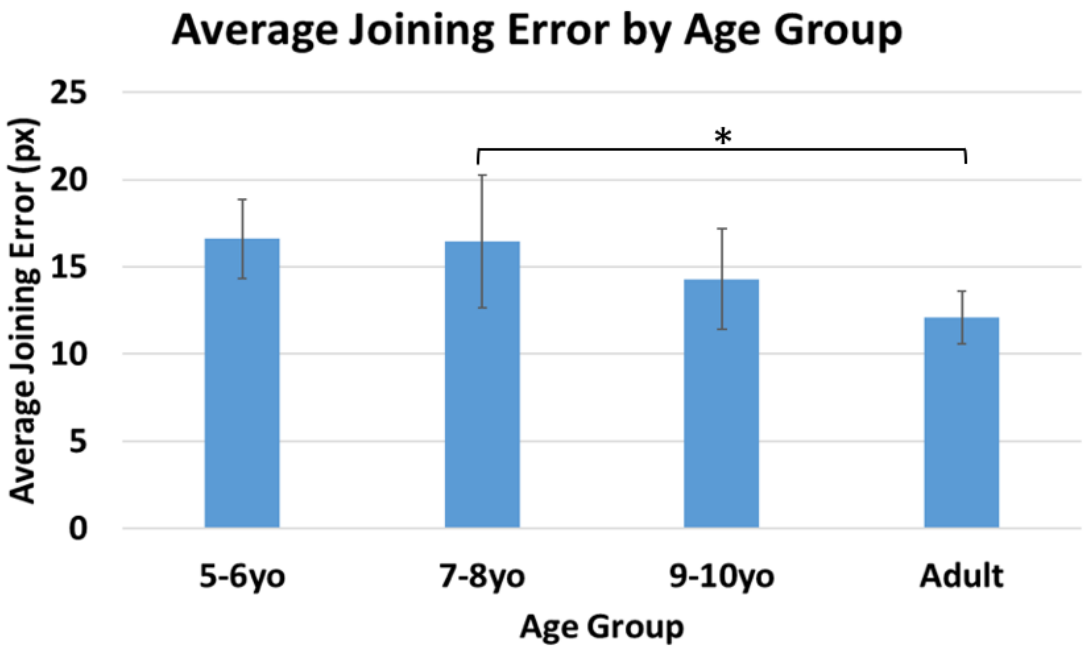

Figure 5-33: Effect of age group on average joining error. Error bars represent the 95% confidence interval.

maintaining spatial awareness of the ink they have created due to their developing motor skills [90]. Figure 5-34 shows the effect of age group on average number of enclosed areas.

Rotation Error. The average rotation error was 11.70 [SD = 3.90] for 5- to 6-year-olds, 8.72 [SD = 1.98] for 7- to 8-year-olds, 6.79 [SD = 1.61] for 9- to 10-year-olds, and 5.31 [SD = 1.27] for adults. A one-way ANOVA on rotation error with a between-subjects factor of age group found a significant main effect of age group on rotation error ($F_{3,55} = 25.71$, $p < 0.05$). Rotation error was highest for 5- to 6-year-olds and decreased steadily for older participants, with adults having the lowest value, as shown in Figure 10c. A Tukey post-hoc test found significant differences in all pairs except between 7- to 8-year-olds and 9- to 10-year-olds and between 9- to 10-year-olds and adults. Thus, as age increases, children tend to have less variation in how they orient their gestures, which is in line with our prior findings [90] that older children tend to be more consistent than younger children. Figure 5-35 shows the effect of age group on average rotation error.

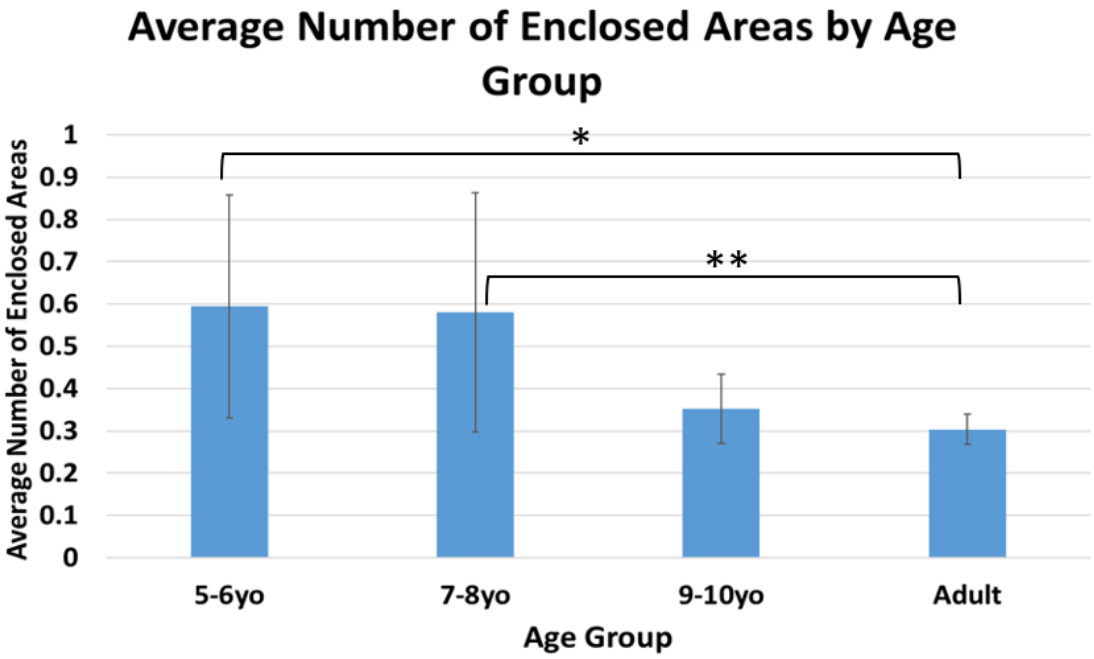

Figure 5-34: Effect of age group on number of enclosed areas. Error bars represent the 95% confidence interval.

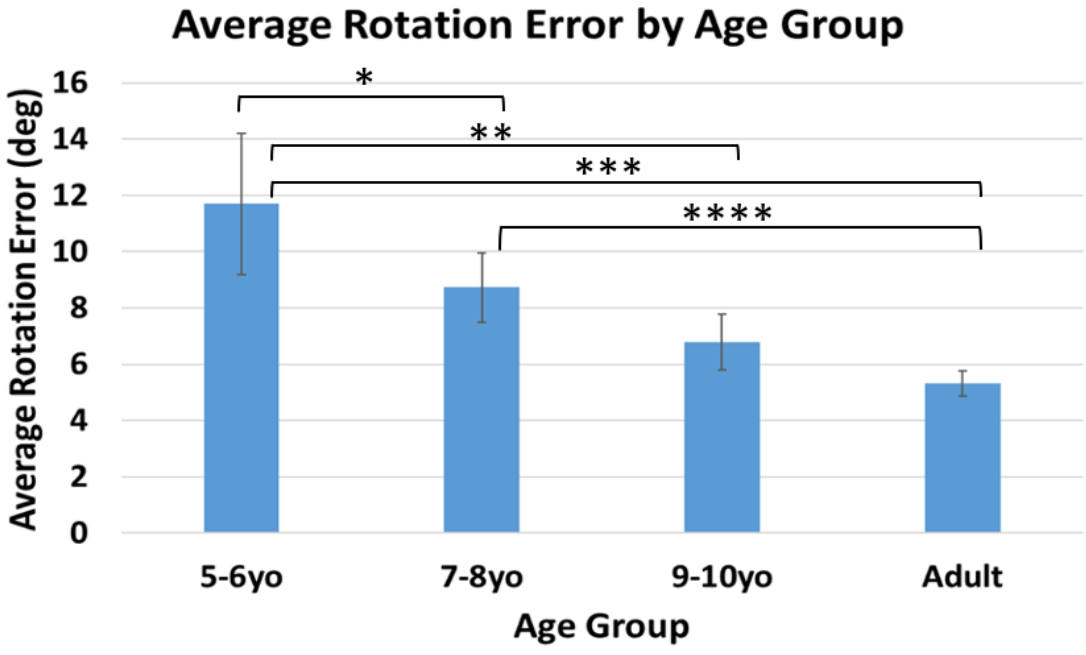

Figure 5-35: Effect of age group on average rotation error. Error bars represent the 95% confidence interval.

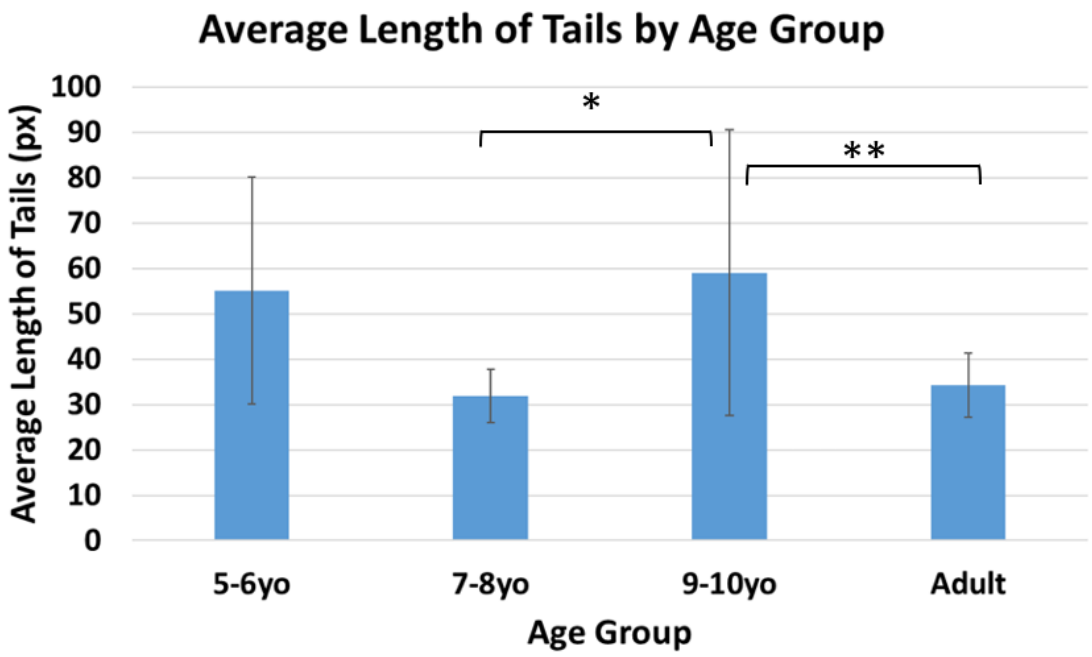

Figure 5-36: Effect of age group on average proportion of gesture in tails. Error bars represent the 95% confidence interval.

Proportion of Extra Gesture in Tails. The average length of gesture in tails was 55.10 px [SD = 38.31 px] for 5- to 6-year-olds, 31.85 px [SD = 9.44 px] for 7- to 8-year-olds, 59.00 px [SD = 48.09 px] for 9- to 10-year-olds, and 34.19 px [SD = 19.73 px] for adults. A one-way ANOVA on proportion of gesture in tails with a between-subjects factor of age group found a significant main effect of age group on the average proportion of tails ($F_{3,54} = 2.95$; $p < 0.05$). Unlike the previous features, the highest value for proportion of tails was for 9- to 10-year-olds, with a similar value for 5- to 6-year-olds. The values for the other two age groups were much lower. A Tukey post hoc test found a significant difference between 7- to 8-year-olds and 9- to 10-year-olds and between 9- to 10-year-olds and adults. We believe this trend is a result of the fact that several individuals in the 9- to 10-year-old age group had very high values for length of tails due to their tendency to embellish the gestures with intentional tails [90], as demonstrated in Figure 5-36. Not all tails are the result of intentional behavior, but in this case we believe the increase for 9- to 10-year-olds is due to intentional tails. Figure 5-36 shows the effect of age group on proportion of tails.

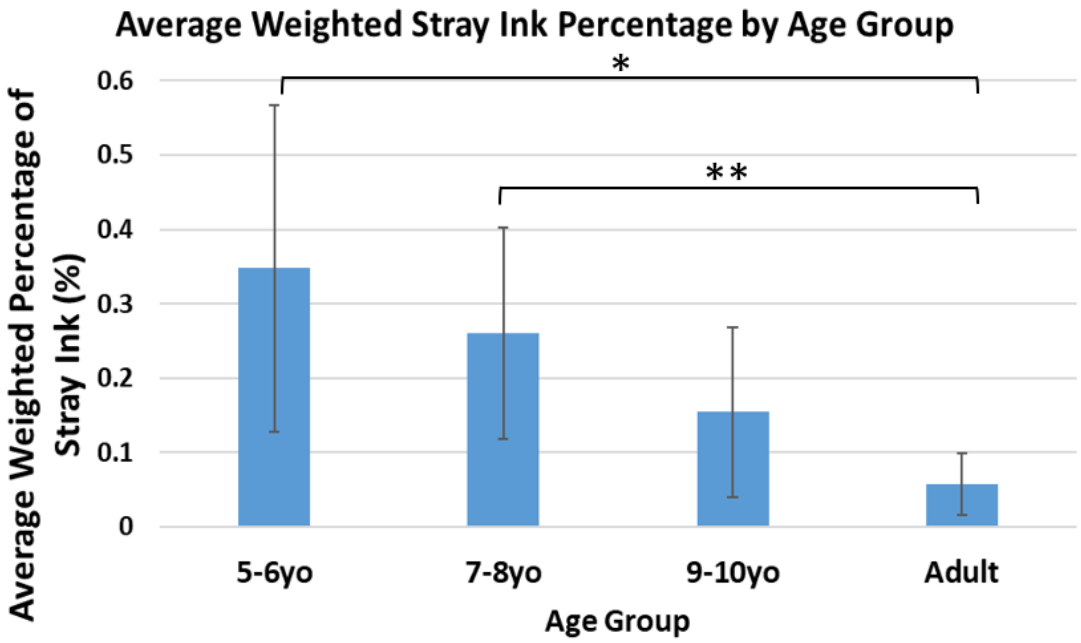

Figure 5-37: Effect of age group on average weighted amount of stray ink. Error bars represent the 95% confidence interval.

Average Weighted Stray Ink Percentage. The average weighted amount of stray ink was 0.35% [SD = 0.34%] for 5- to 6-year-olds, 0.26% [SD = 0.23%] for 7- to 8-year-olds, 0.15% [SD = 0.18%] for 9- to 10-year-olds, and 0.06% [SD = 0.12%] for adults. A one-way ANOVA on average weighted stray ink percentage with a between-subjects factor of age group found a significant main effect of age group on the average percentage of stray ink ($F_{3,54} = 6.443$, $p < 0.05$). The average weighted percentage of stray ink is highest for the youngest children and increases for older participants. A Tukey post hoc test found a significant difference between 5- to 6-year-olds and adults, and between 7- to 8-year-olds and adults ($p < 0.05$). Children tend to have more stray ink than adults, and a greater percentage of children's ink is stray than that of adults. Figure 5-37 shows the effect of age group on the average weighted amount of stray ink. We found this feature had a particularly high standard deviation within each age group, leading the large error bars seen in the graph.

Disconnectedness. The average disconnectedness was 14.40 [SD = 3.37] for 5- to 6-year-olds, 13.83 [SD = 2.73] for 7- to 8-year-olds, 12.77 [SD = 2.06] for 9- to

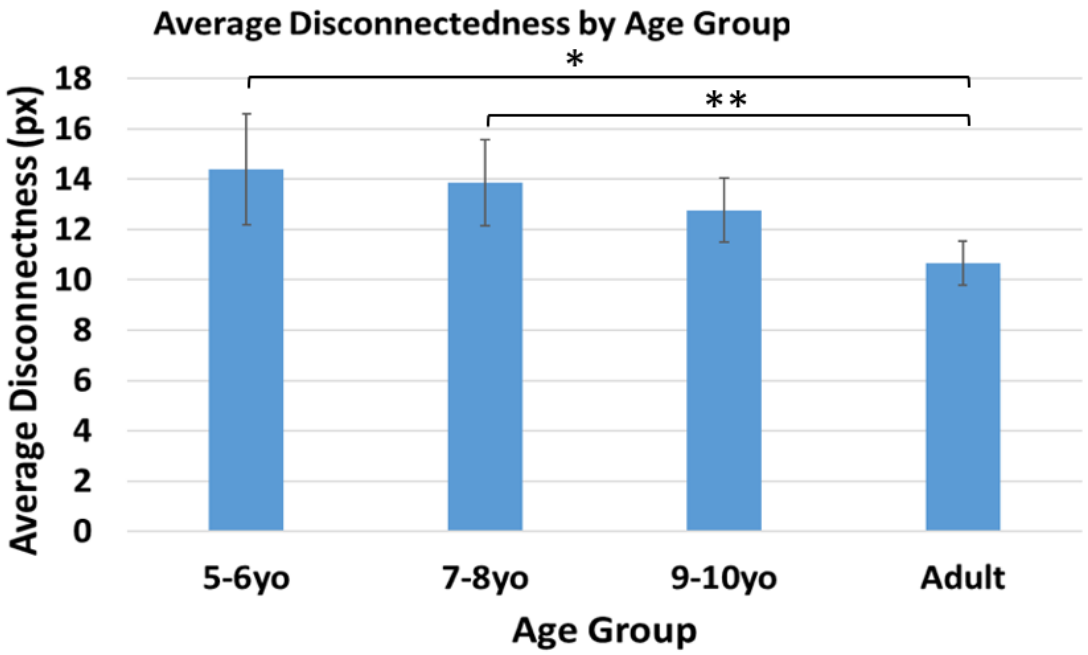

Figure 5-38: Effect of age group on average disconnectedness. Error bars represent the 95% confidence interval.

10-year-olds, and 10.67 [SD = 2.44] for adults. A one-way ANOVA on disconnectedness with a between-subjects factor of age group found a significant main effect of age group on the average disconnectedness ($F_{3,55} = 7.13$, $p < 0.05$). Disconnectedness was highest for the youngest children, and increased for older children, as shown in Figure 5-38. A Tukey posthoc test found a significant difference between 5- to 6-year-olds and adults, and between 7- to 8-year-olds and adults ($p < 0.05$). On average, younger children have more distance between strokes than older users. Figure 5-38 shows the effect of age group on average rotation error.

5.2.3.3 Correlation with Recognition

Because one of our primary goals in analyzing articulation features is to examine how they may affect recognition rates, we examined the correlation between each feature and recognition rates for both the existing features we used as well as our new features. Table 5-5 shows the values of the correlation coefficients for each of the features we discussed in this dissertation. We see the strongest correlation for features that quantify the inconsistency (i.e. relative accuracy features), indicating the importance of consistency in recognition. In fact, all

Table 5-5: Correlation coefficients for the articulation features in our studies. Features with a significant correlation are colored based on the magnitude of the r value: green for $0.25 \leq |r| < 0.50$, blue for $0.50 \leq |r| < 0.75$, and red for $0.75 \leq |r| \leq 1.00$.

1) Number of Strokes (r = -0.42, $p < 0.01$)	15) Bending Error (r = -0.87, $p < 0.01$)
2) Path Length (r = -0.30, $p < 0.01$)	16) Bending Variability (r = -0.80, $p < 0.01$)
3) Area of Bounding Box (r = -0.23, *n.s.*)	17) Time Error (r = -0.80, $p < 0.01$)
4) Line Similarity (r = 0.29, $p < 0.05$)	18) Time Variability (r = -0.80, $p < 0.01$)
5) Global Orientation (r = -0.35, $p < 0.05$)	19) Speed Error (r = -0.74, $p < 0.01$)
6) Total Turning Angle (r = -0.52, $p < 0.01$)	20) Speed Variability (r = -0.61, $p < 0.01$)
7) Sharpness (r = -0.56, $p < 0.01$)	21) Stroke Count Error (r = -0.77, $p < 0.01$)
8) Curviness (r = -0.56, *n.s.*)	22) Stroke Ordering Error (r = -0.80, $p < 0.01$)
9) Production Time (r = -0.46, $p < 0.01$)	23) Joining Error (r = -0.50, $p < 0.01$)
10) Speed (r = -0.08, *n.s.*)	24) Num. Enclosed Areas (r = -0.57, $p < 0.01$)
11) Shape Error (r = -0.82, $p < 0.01$)	25) Rotation Error (r = -0.82, $p < 0.01$)
12) Shape Variability (r = -0.83, $p < 0.01$)	26) Percentage of Tails (r = 0.06, *n.s.*)
13) Length Error (r = -0.80, $p < 0.01$)	27) Pct. Stray Ink (r = -0.51, $p < 0.01$)
14) Size Error (r = -0.72, $p < 0.01$)	28) Disconnectedness (r = -0.36, $p < 0.01$)

of the features with a correlation coefficient with magnitude above 0.75 are relative accuracy features that are calculated between gestures of the same type. These findings help point toward specific types of inconsistencies that could potentially be accounted for in further work on recognition algorithms. The correlations also show moderate correlations between both speed error and speed variability, indicating that even though resampling is done to account for variation in speed, it could be related to another factor that affects recognition. For example, perhaps users are less careful when drawing more quickly. Joining error, number of enclosed areas, and amount of stray ink showed moderate correlations with recognition. These new features quantify how well formed the gesture is and point to specific behavior that may be impacting the recognition process. Number of enclosed areas and stray ink percentage could both affect the resampling process used by many recognizers, thereby affecting recognition rates.

5.3 Discussion

In this chapter, we first presented an analysis of 22 articulation features calculated on a set of children's touchscreen gestures. We found a significant effect of age on the values of 6 simple features and 12 relative accuracy features. Our findings help characterize the

ways in which children make gestures, particularly ways in which they are inconsistent. Later, we described six new articulation features that provide valuable new information about the way children make touchscreen stroke gestures. In particular, we characterized some of the common inconsistencies in children's gestures when creating gestures that add noise not typically seen in adults' gestures, which could have an impact on the recognition process. In our study, we found that there was a significant effect of age on the value of all our features except proportion of tails. We found strong negative correlations with recognition accuracy for rotation error, moderate negative correlations with recognition accuracy for joining error, number of enclosed areas, and disconnectedness, and a weak negative correlation with recognition accuracy for percentage of stray ink. Feature-based analyses have the potential to provide new insights that lead to significant improvements in touchscreen gesture interaction. Gesture sets could be designed to use gestures that are naturally less likely to lend themselves to users using these behaviors, which may improve the legibility of children's gestures. For example, designers might choose to use gestures whose canonical forms have less ink to reduce the chance of joining error, stray ink, and enclosed areas. Designing better gesture sets could lead to improved recognition, and recognition algorithms themselves could also be improved using our features. One way of leveraging these features to improve recognition would be to design a formal preprocessing step to automatically transform the gestures slightly before recognition. For example, gestures with high disconnectedness could automatically have their strokes moved to be closer together, reducing potential error in the point matching and recognition process. Joining error could be used to identify endpoints that should match and automatically move them to be connected, thus reducing error in the point matching process. After the gesture is produced, the system could detect the distance between endpoints and join them if their distance is less than a predetermined threshold. This would lead to more uniform gestures, which would improve the point-matching process since the gestures of the same type would have less variation. We now provide two examples of how our features can be used to improve recognition rates.

Stray Ink. In our study, we found a significant main effect of age on the weighted amount of stray ink in a gesture. We also found a strong negative correlation between stray ink and recognition. Though we used manual annotation to identify stray strokes, we noticed during our analysis that stray strokes tend to be much shorter than non-stray strokes. A paired t-test between average per-user length of non-stray strokes ($M = 419.54$, $SD = 92.93$) and the average per-user length of stray strokes ($M = 40.73$, $SD = 50.20$) showed a significant difference ($t(59) = 31.73$, $p < 0.01$). To help illustrate the potential effect of stray strokes on recognition accuracy, we examined recognition rates for children's gestures in our corpus with and without stray strokes removed. When using our annotations to identify stray strokes, we found that the average user-dependent recognition rate of the gestures which had one or more stray strokes was 61.19% before removing the stray strokes, compared to 73.37% when the stray strokes were removed, an increase of over 12%. Figure 5-39 shows a breakdown of the recognition results by age. However, when gestures are recognized in real-time as they would be in production apps, manual annotation is not feasible. As a first step toward exploring how automatic removal of stray strokes may perform, we also ran experiments where a stroke was considered stray if its length was less than 5% of the total path length of the gesture containing it. While the gains in recognition accuracy were smaller, Figure 5-39 shows that there was some improvement for several age groups, indicating this is a good start towards dealing with stray input. Clearly, removing stray strokes is a step that can be applied to achieve noticeable improvements in recognition accuracy for gestures exhibiting the behavior. While we use a 5% threshold as a preliminary analysis, further work may investigate optimizing the threshold and potentially combining it with other strategies. A system might also take into account the distance of the strokes from other strokes, for example. Designers could benefit from using a simple algorithm to classify strokes as stray based on their length as a preprocessing step to minimize input error.

Joining Error and Disconnectedness. In our analysis, we show a significant effect of age on joining error and disconnectedness, both of which could affect the process of matching

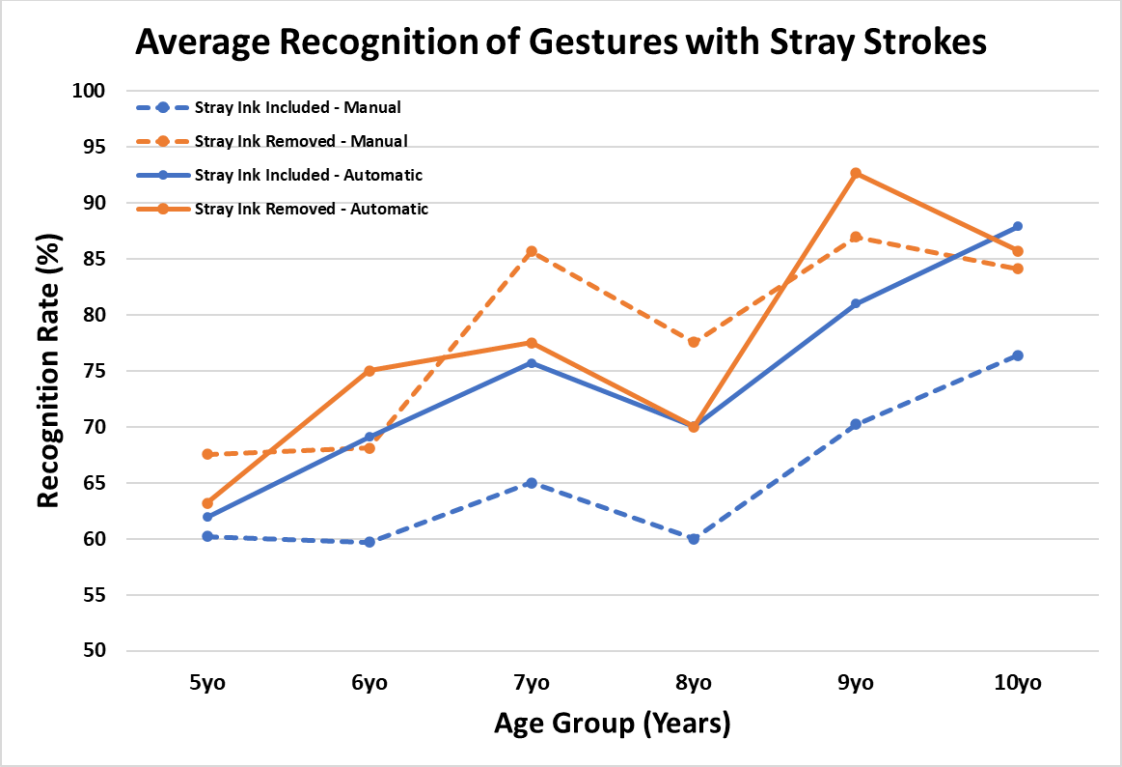

Figure 5-39: Average recognition rates of gestures with and without stray strokes.

the points between candidates and templates. In order to improve this, we suggest that further work on recognition using point matching approaches consider finding less stringent methods of matching the corresponding points. An example of an algorithm that has already been designed to improve the point matching process is Vatavu's $P+ [101], which was developed for recognizing gestures produced by users with low vision. $P+ was designed to improve recognition by modifying the point-matching process to be less affected by several articulation features, including shape error and bending error. As a point of comparison, we examined the difference between recognition accuracy for $P and $P+ in the gestures we used in this study. A paired t-test showed a significant difference in recognition accuracy between the two recognizers ($t(23) = -13.19$, $p < 0.01$). $P+ performs better due to the relaxed point matching approach used, as shown in Figure 5-40. However, recognition for young children remains poor, increasing only from 64% to 68% for 5-year-old children. Our analysis shows while $P+ improves compared to $P, the changes reflected in $P+ are only part of the picture

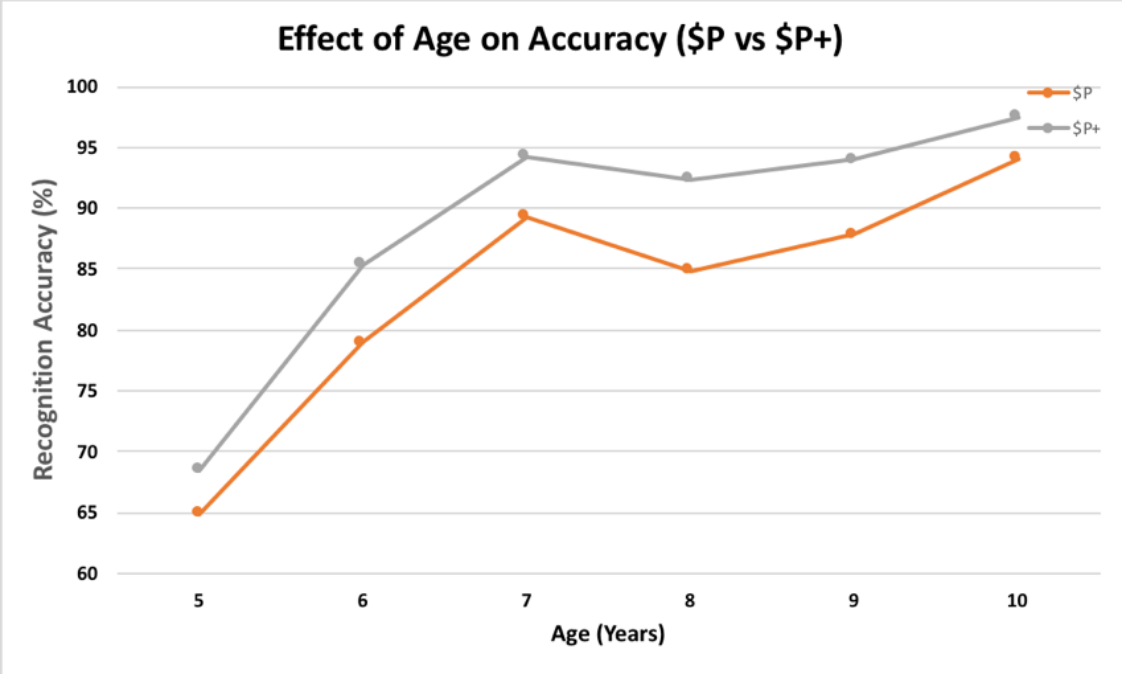

Figure 5-40: Recognition rates when using $P+.

for better recognizing children's gestures, so future work could take a similar approach with more child-specific features like rotation error or disconnectedness. We provide new features that show correlations to recognition. Continued work on improving recognition algorithms for children can build on our work to produce better gesture sets and point matching approaches based on children's articulation patterns.

5.3.1 Comparison between Studies

In our analysis of our new features, we found a similar relationship between age group and the value of the articulation features as in our initial study on the application of adult-focused features to children's gestures [90]. In general, the youngest children have the worst, or highest, values for the features, and the values of the features decrease as age increases. This trend makes sense because our features are designed to capture inconsistent or nonstandard gesture articulation behaviors, which we show are more prevalent in younger children's gestures. Table 5-5 shows correlation coefficients between each feature and recognition accuracy, including features analyzed in prior work [90] and our new features. We can conclude that

the features with the weakest correlation, including path length, area of bounding box, speed, and percentage of tails are not causes of poor recognition. Gesture set designers can use this information to create better gesture sets for children. For example, since area of the bounding box is not strongly correlated with recognition, designers need not focus on differences in size among gestures. Instead, designers can focus on the features that are strongly correlated to help motivate design of sets with fewer collisions. In terms of recognition algorithms themselves, many recognizers are already size and scale invariant, meaning size and speed should not affect recognition rates. The features with higher correlation coefficients give a better idea of what behaviors to account for in continued work on recognition. Because we see such a high correlation between rotation error and accuracy, for example, future work should explore ways to minimize the impact of rotation on the recognition process. We showed that preprocessing of gestures based on our new features can lead to improvements in recognition, and we hypothesize that a combination of gesture-set design and similar preprocessing based on our new features could lead to even greater improvements.

5.3.2 Summary

In this chapter, we discussed existing work on articulation features and how they are used to quantify touchscreen gestures. We applied existing touchscreen gesture articulation features to children's gestures to gain insight into their behavior patterns. We then developed six new articulation features designed specifically to capture common patterns in children's touchscreen gestures. We analyzed the correlation between the values of each of the existing and new features with recognition rates to gain insight into which features are most associated with recognition. In the next chapter, we synthesize our findings from our work on recognition and articulation features to offer a set of design guidelines for future researchers and designers.

CHAPTER 6
DESIGN IMPLICATIONS

Throughout the work described in this dissertation, we have described work we conducted to gain new insight that could help designers of applications to provide more engaging, personalized experiences for children using touchscreen gesture interaction. We now provide a list of all the design recommendations we have developed during all phases of our work. We break our design implications into the categories of gesture collection, gesture set design, and gesture recognition.

6.1 Gesture Collection

Collection of gestures from children can be difficult due to children's limited attention span. We offer some guidelines based on our work to help researchers and practitioners who wish to collect children's gesture data.

Minimize Repetition in Studies In all of our studies, we used Brewer et al.'s [21] method of gamification to make the studies more enjoyable for the children. Children were granted small prizes after each phase of the study to help maintain their engagement. We recommend this method, especially when the number of gestures collected per user is large, as this can be time consuming and risks the child losing interest. We also found children were more willing to complete our studies when we broke them down into smaller tasks that they can finish quickly. Introducing small changes, such as varying the input modality (e.g., pen to touch) can also help make the study more engaging for children.

Collect Gesture on Whichever Device is Most Convenient. Throughout our work, we saw consistent recognition rates for both children and adults regardless of the device used, including smartphones, tablets, and tabletop computers [112]. We even found that recognition rates were not changed when a recognizer was trained on gestures from one device and tested on gestures from a different device. Thus, when collecting gestures it may be easiest to collect the gestures on a portable device like a phone or tablet even if they are intended for use on a touchscreen computer or tabletop device.

6.1.1 Gesture Set Design.

When designing a gesture-based application for children, it is important to consider whether the gesture set used in the application will be suited to children's abilities. When using recognition, care should be taken to avoid collisions which can arise from the two gestures being very similar (such as X and +) or due to children's articulation habits (such as the tendency for children to exhibit mirror writing, which can cause a 2 to look like a 5). We offer guidelines for designers creating gesture sets for children.

Favor Unistroke Gestures. In our analysis of our new articulation features in Section 5.2.3.1, we show that children have a higher value of joining error than adults, indicating that children often have trouble joining the strokes of their gestures correctly. Thus, we suggest designers consider using unistroke gestures since they generally do not require connecting strokes. An example of an existing unistroke gesture set is Wobbrock et al.'s set [111] to evaluate the $1 recognizer.

Avoid Rotationally Similar Gestures. Some gesture sets include rotationally similar pairs of gestures, such as A and the symbol ∀, where rotating one of the gestures can make it appear to belong to another class. Because we see such a pronounced occurrence of rotation error in children in Section 5.2.3.1, we suggest designers avoid such pairs when choosing gesture sets for children. Furthermore, we also suggest that application designers attempt to find an optimal rotational match between template and candidate gestures when using template-based recognition approaches. This rotational matching process is included as part of some recognizers, such as the $1 recognizer [111], but it could be implemented as a preprocessing step for any recognizer by aligning the gestures along the line from their centroids and their starting points.

6.2 Gesture Recognition

Finally, we offer a set of guidelines for improving recognition based on our work on articulation features.

Expect Stray Ink in Children's Input. In section 5.3, we illustrated how removing stray ink in children's gesture input could result in improved accuracy. We recommend that application designers consider handling stray ink in children's input by using a distance or length-based threshold. In our future work, we plan to develop a concrete method of removing stray strokes from gesture input.

Consider Looser Point Matching Approaches In section 5.3, we showed how the use of a looser template matching approach called $P+ improved recognition rates compared to a standard template matcher. $P+ allows for many-to-one matching of points between candidate and test gestures rather than the required one-to-one matching of other template matchers. We recommend designers consider this and other less stringent point-matching approaches to account for the high level of joining error and disconnectedness.

6.3 Summary

In this chapter we introduced several guidelines for developers working with touchscreen-based applications for children. In particular, we provide recommendation for gesture collection, gesture set design, and improving gesture recognition. We encourage developers to consider adapting these guidelines to help build child-centered gesture-based applications.

CHAPTER 7
FUTURE WORK

In this chapter, we describe valuable possible future avenues of research building on the work presented in this dissertation.

7.1 Improving Recognition Rates

As we have shown throughout this dissertation, recognition of children's gestures has been and continues to be poor, especially for young children. Even the advanced machine learning techniques we tested provide only slightly better recognition results than traditional template matching approaches. We now present some methods that may be useful in further work on improving recognition.

7.1.1 Beautification

One of the methods of improving gesture recognition rates that we have explored is the use of beautification, in which users' natural gestures are automatically transformed into a more idealized, perfect version [51]. A beautification algorithm attempts to detect what the user meant to do based on the geometric properties of the gesture, such as curvature and angles of intersection. This technique is inspired by previous work that has employed beautification as part of sketch-based interfaces [43, 45, 76], particularly for domain-specific applications, such as circuit sketching in engineering courses [40]. Figure 7-1 shows the results of applying a simple beautification algorithm we have developed on some of the gestures from the corpus. To perform the beautification, our algorithm examines each component stroke and converts it to a line, an arc, or a circle, depending on its geometric properties. The gestures are thus represented in a model known as the Curve, Lines, and Corners (CLC) model [26]. After the gesture's strokes have been converted, our algorithm attempts to decide whether each pair of strokes should be connected based on their proximity to one another. As a simplification, each stroke can be joined to another stroke at its midpoint or one of its endpoints. While the algorithm we have developed achieves good results for the gestures shown in Figure 7-1, more complicated gestures can be problematic. In particular, because we

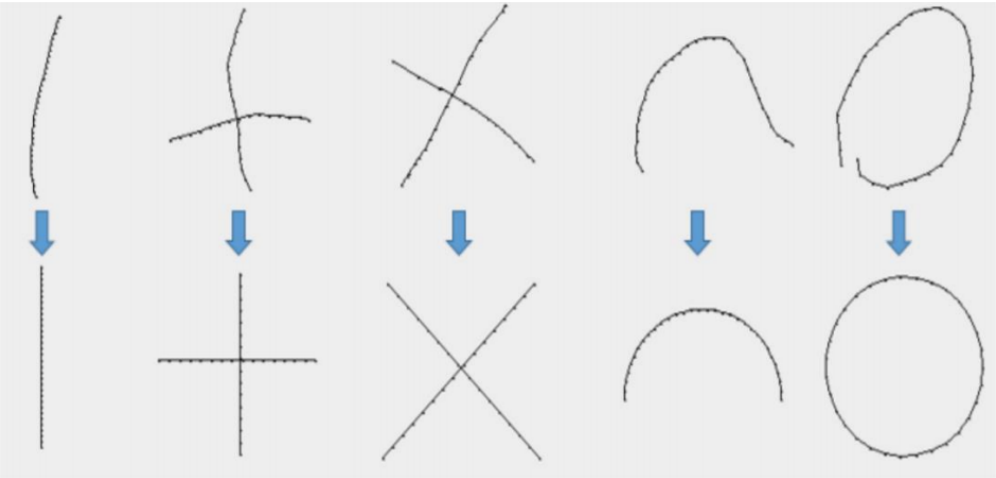

Figure 7-1: Examples of gestures for which the beautification process works well.

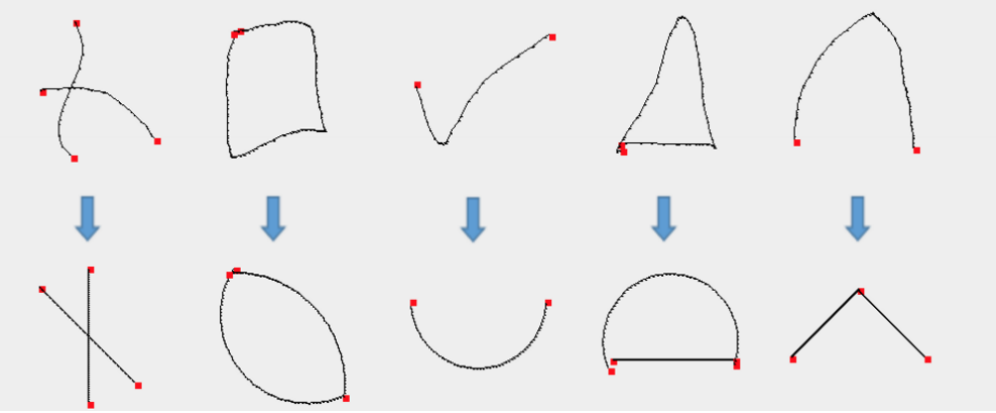

Figure 7-2: Examples of gestures for which the beautification process works poorly.

use a stroke-based approach, proper corner detection is crucial. However, the corner detection methods that are typically used for adults' gestures do not work well for children due to the high levels of inconsistency in their gestures. Figure 7-2 shows examples of gestures in our corpus for which the beautification algorithm performs poorly. Further work in improving the beautification technique would likely require an improved corner detection algorithm that could better handle children's input. With better beautification, gestures could be transformed into their canonical representation, which could improve the performance of recognizers.

7.1.2 Child-Specific Algorithms

In our work, we have shown that existing algorithms perform quite poorly in recognizing children's gestures. This is in part because these recognizers were designed with the goal of recognizing adults' gestures rather than children's gestures. There still exists the possibility of creating a recognizer specifically designed for children's gestures, which may achieve higher results. In particular, we believe that the new articulation features we developed may be useful in designing a child-centered recognizer since they capture gesturing patterns specific to children which we have shown can have an impact on the recognition process.

7.2 Articulation Features

While we believe our work forms a good foundation for understanding children's touchscreen gestures through articulation features, there are still opportunities to further our understanding of the ways children create these gestures. New articulation features may allow for further analysis that improve our understanding of children's gesturing behaviors, allowing designers to provide better experiences for children using touchscreen devices.

7.3 Other Analyses

7.3.1 Effect of Context and Motivation

In our studies, all gestures were collected while children used a touchscreen application in the context of an abstract application or a simple game. However, children's gesturing behavior may change when the context differs. For example, a child using a gesture-based application in the context of learning for school may be more focused or interested, causing a change in their interactions. In our controlled laboratory studies, we allowed children to hold the touchscreen devices in whatever manner felt most comfortable to them with the goal of collecting gestures as naturally as possible. However, it is possible that variations in how children held the devices could cause differences among their gestures. Further work may consider this and other interaction variables, such as whether the participants were sitting or standing, time of day, behavior over time, and other factors that may have some impact on the children's gesturing behavior.

7.3.2 Relationship with Cognitive Development

While most of our work has examined the relationship of recognition with age, future work may consider analyzing gesture behavior with respect to children's stage of cognitive development and motor skills. Even within the same grade level or age group, children typically have a wide range of different cognitive and motor skills [88]. Looking at the effect of children's cognitive development on their gesture interactions may enable designers to make further improvements of gesture-based applications for children.

CHAPTER 8
CONCLUSION

This dissertation focused on the challenges of understanding and recognizing children's touchscreen stroke gestures. Because children's interactions are different from adults, it is important that designers be aware of ways in which they are able to improve children's experiences when using gesture-based applications. We addressed the following research goals throughout this dissertation:

Research Goal 1: Determine How Well Existing Recognizers are Able to Recognize Children's Gestures

1.1 How well can popular template matchers recognize children's gestures?

1.2 How well can more advanced machine learning methods recognize children's gestures?

1.3 How does human ability to recognize children's gestures compare with machine algorithms?

Research Goal 2: Understanding Children's Gestures Through Articulation Features

2.1 What can we learn about children's gestures by studying existing articulation features?

2.2 Can new child-specific articulation features provide a fuller characterization of children's touchscreen gestures?

2.3 Can these new features be leveraged to improve recognition of children's touchscreen gestures?

This final chapter will outline the key findings for each research goal and highlight our contributions.

8.1 Research Goal 1

8.1.1 Establishing Recognition Rates

In Chapter 3, we analyzed gesture recognition rates across a variety of recognition algorithms. We began with $P [103], a template matcher widely used in the Human-Computer Interaction community for recognizing children's gestures. We found that recognition rates

were poor for children, especially younger children. Recognition rates were 84% on average for children ages 5 to 10, with an average of 65% for 5-year-old children.

Having established poor recognition rates, we performed an experiment in which we employed other types of recognizers, particularly more advanced machine learning methods. The categories of recognizers included template matchers, feature-based statistical classifiers, support vector machines (SVMs), hidden-Markov models (HMMs), and neural networks. While we found some variation in performance, none of these methods achieved significantly higher accuracy in recognizing children's touchscreen gestures. Overall, recognition was roughly the same across all categories of recognizers, indicating that further work is needed to improve the state of recognition.

8.1.2 Human Recognition

Having established that recognition rates are poor, we sought to determine how well humans would be able to recognize these gestures. We reasoned that humans would be good at this task due to our natural ability to recognize patterns and because we have a lifetime of experience seeing examples of the letters, numbers, shapes, and symbols in our gesture set. Thus, human ability to recognize these gestures would give us a good target by which further work in recognition can be assessed. We conducted a survey in which 131 adults attempted to classify the same gestures from 5- to 10-year-olds as in our previous study. We found a significant increase in accuracy, with human accuracy reaching 91% compared to 84% for machine accuracy.

8.2 Research Goal 2

8.2.1 Analyzing Existing Articulation Features

In Chapter 5, we set out to better understand the reasons for poor recognition by analyzing a set of articulation features that had previously been designed to provide insight regarding the ways adults create gestures. We selected a set of 22 articulation features from prior work, including 10 simple features and 12 relative accuracy features [104]. The simple features included geometric and temporal measures such as the amount of ink used and

the average velocity of the gesture. The relative accuracy features provide a measure of the consistency among the gestures of a single type created by a single user. An example is the shape error, which refers to the average difference in Euclidean distance between corresponding points in two gestures of the same type. We grouped the users into four groups: 5- to 6-year-olds, 7- to 8-year-olds, 9- to 10-year-olds, and adults. We found a significant effect of age group on 6 of the 10 simple features and all 12 of the relative accuracy features. The values of the relative accuracy features were highest for the younger children and decreased for older children and adults, indicating that younger children have a lower level of consistency in their gestures.

8.2.2 Developing New Articulation Features for Children

While we gleaned useful information from our analysis of using existing articulation features to characterize children's touchscreen gestures, we felt further work was needed for a full characterization. In particular, we noted that the existing articulation features were designed for quantifying clean, well-formed gestures. Children's gestures, however, are often messy and not well-formed, so we wanted to develop new features for characterizing child-specific behavior. To do this, we analyzed our corpus of over 26,000 gestures from children ages 5- to 13-years-old, carefully observing the common inconsistencies. Based on our observations, we developed a set of new articulation features specifically designed to capture children's gesturing behavior and calculated the values of these features on children's gestures.

8.2.3 Analyzing the Correlation between Articulation Features and Recognition Rates

To further understand the relationship between the behaviors represented by the articulation features we examined, we also analyzed the correlation between the articulation features and recognition rates with respect to age. We found that several of the relative accuracy features and our new child-specific features were correlated with recognition rates, indicating that they may provide some inside into why recognition rates are poor.

8.3 Contributions

The work presented in this dissertation offers several contributions to the fields of Human-Computer Interaction and gesture recognition. First, we establish recognition rates using a variety of recognition algorithms on touchscreen stroke gestures from children ages 5- to 10-year-olds. We show that even advanced machine learning techniques achieve poor recognition rates, especially for younger children. We show that humans are able to achieve significantly higher accuracy than machine algorithms in recognizing these gestures, indicating potential for improvement in the automatic recognition of children's touchscreen gestures.

8.4 Publications

- **Alex Shaw**, Jaime Ruiz, and Lisa Anthony. 2017. Comparing human and machine recognition of children's touchscreen stroke gestures. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction (ICMI '17)*. Association for Computing Machinery, New York, NY, USA, 3240. DOI:https://doi.org/10.1145/3136755.3136810

- Julia Woodward, **Alex Shaw**, Aishat Aloba, Ayushi Jain, Jaime Ruiz, and Lisa Anthony. 2017. Tablets, tabletops, and smartphones: cross-platform comparisons of children's touchscreen interactions. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction (ICMI '17)*. Association for Computing Machinery, New York, NY, USA, 514. DOI:https://doi.org/10.1145/3136755.3136762

- **Alex Shaw**. 2017. Human-centered recognition of childrens touchscreen gestures. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction (ICMI '17)*. Association for Computing Machinery, New York, NY, USA, 638642. DOI:https://doi.org/10.1145/3136755.3137033

- **Alex Shaw** and Lisa Anthony. 2016. Analyzing the articulation features of children's touchscreen gestures. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction (ICMI '16)*. Association for Computing Machinery, New York, NY, USA, 333340. DOI:https://doi.org/10.1145/2993148.2993179

- **Alex Shaw** and Lisa Anthony. 2016. Toward a Systematic Understanding of Children's Touchscreen Gestures. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '16)*. Association for Computing Machinery, New York, NY, USA, 17521759. DOI:https://doi.org/10.1145/2851581.2892425

- Julia Woodward, **Alex Shaw**, Annie Luc, Brittany Craig, Juthika Das, Phillip Hall, Akshay Holla, Germaine Irwin, Danielle Sikich, Quincy Brown, and Lisa Anthony. 2016. Characterizing How Interface Complexity Affects Children's Touchscreen Interactions. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*

(CHI '16). Association for Computing Machinery, New York, NY, USA, 19211933. DOI:https://doi.org/10.1145/2858036.2858200

APPENDIX A
ADULT RECOGNITION RATES

In order to verify the correctness of our recognition results for children's gestures in Chapter 4, we also computed recognition rates for gestures from adults. We then compared the rates we found to the rates reported in the papers introducing the recognizers. Table A-1 shows these recognition rates. We show that accuracy rates are similar between existing work and our research to add confidence in the correctness of our implementations. We note, however, that there are some exceptions due to differences in the gesture set between the original studies and our study. In particular, almost all of the previously reported rates are higher than the rates we saw in our study. There are several factors that affect most or all of these recognizers. For all of the recognizers except $P, the gesture set used was different, meaning the numbers are not comparable. Furthermore, most of the original recognition rates were calculated using a user-dependent approach, which typically results in higher recognition rates than the user-independent approach that we use. The following are other recognizer-specific factors that likely impact recognition:[1]

GRANDMA. GRANDMA [84] was designed to operate on unistroke gestures, so we had to treat all of our gestures as unistroke in order to obtain a recognition rate for our dataset. We did this by connecting the endpoints of consecutive strokes, which may have impacted recognition accuracy.

GDE. GDE [14] was designed for recognizing geometric shapes based on the relationship between the strokes of a gesture. The original accuracy rate was obtained using a set of six gestures, compared to the 20 in our set. With a smaller set, there are less opportunities for collisions, which could result in a higher recognition rate.

[1] Note that $N-Protractor, $P, and $P+ are excluded because our setup and accuracy rates are very similar to those reported in the original papers.

Feature-based Rubine. Blagojevic's [20] feature-based Rubine experiments were run using a set of six different gesture types, compared to the 20 different types in our set.

Kato et al. Kato et al.'s [54] recognizer operates on image data, so we converted all the gestures from XML logs to images, which could impact recognition rates.

Anderson Anderson's HMM [4] recognizer was evaluated on a set of 11 gesture types compared to the 20 in our gesture set.

Lecun et al. LeCun et al.'s [60] algorithm operates using image data, so we converted our data as with Kato's recognizer. This may have impacted recognition rates. The recognizer was also tested only on handwritten digits, so there were a total of 10 gesture types in the set compared to our 20.

Shrivistava and Sharma. Shrivistava and Sharma's [92] neural nework approach was designed for optical character recognition, meaning it is image based as with several other recognizers. Thus, we converted our data from XML to image files.

Yin and Sun. Yin and Sun's [116] recognizer was evaluated on a set of 11 gesture types compared to the 20 in our gesture set. Because the work was in the domain of sketch recognition, several of the gestures were more complex than those seen in our gesture set.

Sezgin and Davis. Sezgin and Davis's [89] recognizer performs sketch recognition by segmenting input into multiple strokes, but we skip the segmentation step since our data was stored by individual strokes. The authors report using a set of 88 different objects, which tended to be more complex than the gestures in our set, as is typical in sketch recognition literature.

Alimoglu and Alpaydin. Alimoglu and Alpaydin's [3] recognizer was designed for recognizing gestures created using pen input, but our data was primary collected with finger input, which may have impacted recognition rates.

Table A-1: Comparison of recognition rates of adults' gestures from previous work with rates from our work to verify correctness of implementation.

Recognizer	**Previous Recognition Rate**	**Our Recognition Rate**
$N-Protractor [12]	97%	95.5% [SD = 3.2%]
$P [103]	99%	98% [SD = 1.9%]
$P+ [103]	98.2%	97.0% [SD = 2.1%]
GRANDMA [84]	97.1%	79.9% [SD = 3.5%]
GDE [14]	97.5%	71.4% [SD = 3.6%]
Feature-based Rubine [20]	96.9%	86.9% [SD = 2.1%]
Kato et al. [54]	89.1%	82.0% [SD = 2.2%]
Anderson et al. [4]	94.2%	82.5% [SD = 1.9%]
LeCun et al. [60]	99.0%	96.1% [SD = 1.2%]
Shrivastama and Sharma [92]	85.3%	71.5% [SD = 2.4%]
Yin and Sun [116]	96.5%	76.1% [SD = 3.1%]
Sezgin and Davis [89]	96.5%	84.63% [SD = 3.9%]
Alimoglu and Alpaydin [3]	93.81%	77.2% [SD = 3.2%]

REFERENCES

[1] Abdul Aziz, N. A., Batmaz, F., Stone, R., and Chung, P. W. H. Selection of touch gestures for children's applications. In *Proceedings of the Science and Information Conference* (2013), 721--726.

[2] Ahmed, H., and Azeem, S. A. On-line Arabic handwriting recognition system based on HMM. In *Proceedings of the International Conference on Document Analysis and Recognition* (2011), 1324--1328.

[3] Alimoglu, F., and Alpaydin, E. Combining multiple representations and classifiers for pen-based handwritten digit recognition. In *Proceedings of the Fourth International Conference on Document Analysis and Recognition*, vol. 2, IEEE Computing Society (1997), 637--640.

[4] Anderson, D., Bailey, C., and Skubic, M. Hidden Markov Model Symbol Recognition for Sketch-Based Interfaces. In *AAAI Fall Symposium*, AAAI Press (Menlo Park, 2004), 15--21.

[5] Anthony, L. Physical dimensions of children's touchscreen interactions: Lessons from five years of study on the mtagic project. *International Journal of Human-Computer Studies 128* (2019), 1 -- 16.

[6] Anthony, L., Brown, Q., Nias, J., and Tate, B. Examining the need for visual feedback during gesture interaction on mobile touchscreen devices for kids. In *Proceedings of the International Conference on Interaction Design and Children (IDC '13)*, ACM Press (New York, New York, USA, June 2013), 157--164.

[7] Anthony, L., Brown, Q., Nias, J., and Tate, B. Children (and Adults) Benefit From Visual Feedback During Gesture Interaction on Mobile Touchscreen Devices. *International Journal of Child-Computer Interaction. 6* (2015), 17--27.

[8] Anthony, L., Brown, Q., Nias, J., Tate, B., and Mohan, S. Interaction and recognition challenges in interpreting children's touch and gesture input on mobile devices. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '12)*, ACM Press (New York, New York, USA, Nov. 2012), 225--234.

[9] Anthony, L., Brown, Q., Tate, B., Nias, J., Brewer, R., and Irwin, G. Designing smarter touch-based interfaces for educational contexts. *Personal and Ubiquitous Computing 18*, 6 (Nov. 2014), 1471--1483.

[10] Anthony, L., Vatavu, R.-D., and Wobbrock, J. O. Understanding the Consistency of Users Pen and Finger Stroke Gesture Articulation. In *Proceedings of Graphics Interface (GI '13)*, Canadian Information Processing Society (May 2013), 87--94.

[11] Anthony, L., and Wobbrock, J. O. A lightweight multistroke recognizer for user interface prototypes. In *Proceedings of Graphics Interface (GI '10)*, GI '10, Canadian Information Processing Society (Toronto, 2010), 245--252.

[12] Anthony, L., and Wobbrock, J. O. $N-Protractor : A Fast and Accurate Multistroke Recognizer. In *Proceedings of Graphics Interface (GI '12)*, Canadian Information Processing Society (2012), 117--120.

[13] Anthony, L., Yang, J., and Koedinger, K. R. A paradigm for handwriting-based intelligent tutors. *International Journal of Human Computer Studies 70*, 11 (2012), 866--887.

[14] Apte, A., Vo, V., and Kimura, T. D. *Recognizing multistroke geometric shapes*. 1993.

[15] Arif, A. S., and Sylla, C. A comparative evaluation of touch and pen gestures for adult and child users. In *Proceedings of the International Conference on Interaction Design and Children (IDC '13)*, ACM Press (New York, New York, USA, June 2013), 392--395.

[16] Bahlmann, C., Haasdonk, B., Burkhardt, H., and Freiburg, A.-l.-u. On-line Handwriting Recognition with Support Vector Machines A Kernel Approach.

[17] Baum, L. E., and Petrie, T. Statistical Inference for Probabilistic Functions of Finite State Markov Chains. *The Annals of Mathematical Statistics 37*, 6 (1966), 1554--1563.

[18] Beery, K. E., Buktenica, N. A., and Beery, N. A. *The Beery-Buktenica Developmental Test of Visual-Motor Integration, 5th Edition*, 5th editio ed. Modern Curriculum Press, New Jersey, 2004.

[19] Bellman, R. On the Theory of Dynamic Programming. *Proceedings of the National Academy of Sciences of the United States of America 38*, 8 (1952), 716--719.

[20] Blagojevic, R., Chang, S., and Plimmer, B. The power of automatic feature selection: rubine on steroids. *Joint Session of the Seventh Sketch-Based Interfaces and Modeling Workshop and Eighth Symposium on Non-Photorealistic Animation and Rendering* (2010), 79--86.

[21] Brewer, R., Anthony, L., Brown, Q., Irwin, G., Nias, J., and Tate, B. Using gamification to motivate children to complete empirical studies in lab environments. In *Proceedings of the International Conference on Interaction Design and Children (IDC '13)*, ACM Press (New York, New York, USA, June 2013), 388--391.

[22] Broekman, F., Piotrowski, J., Beentjees, H., and Valkenburg, P. A parental perspective on apps for young children. *Computers in Human Behavior 63* (2016), 142--151.

[23] Brown, Q., and Anthony, L. Toward comparing the touchscreen interaction patterns of kids and adults. In *Proceedings of the SIGCHI Workshop on Educational Software, Interfaces and Technology* (2012), 4pp.

[24] Brown, Q., Anthony, L., Nias, J., Tate, B., Brewer, R., and Irwin, G. Towards Designing Adaptive Touch-Based Interfaces. In *Proceedings of the ACM SIGCHI Mobile Accessibility Workshop* (2013), 4pp.

[25] Camastra, F. A SVM-based cursive character recognizer. 3721--3727.

[26] Cao, X., and Zhai, S. Modeling human performance of pen stroke gestures. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '07* (2007), 1495.

[27] Chen, M.-Y. Off-Line Handwritten Word Recognition Using a Hidden Markov Model Type Stochastic Network. *IEEE Transactions on Pattern Analysis and Machine Intelligence 16*, 5 (1994), 481--496.

[28] Cho, M. G. A new gesture recognition algorithm and segmentation method of Korean scripts for gesture-allowed ink editor. *Information Sciences 176*, 9 (May 2006), 1290--1303.

[29] Common Sense Media. Zero to Eight: Children's Media Use in America 2013, 2013.

[30] Connell, S., Kuo, P.-Y., Liu, L., and Piper, A. M. A Wizard-of-Oz elicitation study examining child-defined gestures with a whole-body interface. In *Proceedings of the International Conference on Interaction Design and Children* (2013), 277--280.

[31] Connell, S. D., and Jain, A. K. Template-based online character recognition. *Pattern Recognition 34*, 1 (Jan. 2001), 1--14.

[32] Cornell, J. M. Spontaneous Mirror-Writing in Children. *Canadian Journal of Psychology 39*, 1 (1985), 174--179.

[33] El-Hajj, R., Likforman-Sulem, L., and Mokbel, C. Arabic handwriting recognition using baseline dependant features and hidden markov modeling. *Eighth International Conference on Document Analysis and Recognition (ICDAR'05)* (2005), 893--897 Vol. 2.

[34] Erhardt, R. *Developmental Hand Dysfunction*. Therapy Skill Builders, 1994.

[35] Field, M., Gordon, S., Peterson, E., Robinson, R., Stahovich, T., and Alvarado, C. The effect of task on classification accuracy: Using gesture recognition techniques in free-sketch recognition. *Computers & Graphics 34*, 5 (2010), 499--512.

[36] Findlater, L., Froehlich, J., Fattal, K., Wobbrock, J. O., and Dastyar, T. Age - Related Differences in Performance with Touchscreens Compared to Traditional Mouse Input. In *Chi'13* (2013), 1--4.

[37] Gader, P. D., Mohamed, M. A., and Keller, J. M. Fusion of handwritten word classifiers. *Pattern Recognition Letters 17*, 6 (1996), 577--584.

[38] Gathercole, S. E. Cognitive approaches to the development of short-term memory. *Trends in Cognitive Sciences 3* (1999), 410--419.

[39] Gesell, A., and Ilg, F. L. *The Child from Five to Ten*. Harper & Brothers, 1946.

[40] Hammond, T., Logsdon, D., Paulson, B., Johnston, J., Peschel, J. M., Wolin, A., and Taele, P. A Sketch Recognition System for Recognizing Free-Hand Course of Action

Diagrams. In *Proceedings of the International Conference on Innovative Applications of Artificial Intelligence*, AAAI Press (2010), 1781--1786.

[41] Herold, J., and Stahovich, T. F. The 1 Recognizer: a fast, accurate, and easy-to-implement handwritten gesture recognition technique. In *Proceedings of the International Symposium on Sketch-Based Interfaces and Modeling*, Eurographics Association (June 2012), 39--46.

[42] Hiniker, A., Sobel, K., Hong, S. R., Suh, H., Irish, I., Kim, D., and Kientz, J. A. Touchscreen Prompts for Preschoolers: Designing Developmentally Appropriate Techniques for Teaching Young Children to Perform Gestures. *Proceedings of the 14th International Conference on Interaction Design and Children* (2015), 109--118.

[43] Hong, J. I., and Landay, J. A. SATIN: a toolkit for informal ink-based applications. In *Proceedings of the International Symposium on User Interface Software and Technology*, ACM Press (New York, Nov. 2000), 63--72.

[44] Hourcade, J. P., Perry, K. B., and Sharma, A. PointAssist: Helping Four Year Olds Point with Ease. In *Proceedings of the International Conference on Interaction Design and Children (IDC '08)*, ACM Press (New York, New York, USA, June 2008), 202--209.

[45] Hse, H. H., and Richard Newton, A. Recognition and beautification of multi-stroke symbols in digital ink. *Computers and Graphics 29*, 4 (2005), 533--546.

[46] Hu, J., Brown, M. K., and Turin, W. Hmm based on-line handwriting recognition. *IEEE Trans. Pattern Anal. Mach. Intell. 18*, 10 (Oct. 1996), 10391045.

[47] Hu, J., Lim, S. G., and Brown, M. K. Writer independent on-line handwriting recognition using an hmm approach. *Pattern Recognition 33* (2000), 133--147.

[48] Jego, J. F., Paljic, A., and Fuchs, P. User-defined gestural interaction: A study on gesture memorization. In *Proceedings of the IEEE Symposium on 3D User Interfaces* (2013), 7--10.

[49] Jiang, W., and Sun, Z.-x. HMM-Based Online Multi-Stroke Sketch Recognition. In *Proceedings of the Fourth International Conference on Machine Learning and Cybernetics (ICMLC '05)*, no. August (2005), 18--21.

[50] Johnson, R. B., and Turner, L. A. *Data Collection Strategies in Mixed Methods Research*. 2002, 297319.

[51] Julia, L., and Faure, C. Pattern recognition and beautification for a pen based interface. In *Proceedings of the International Conference on Document Analysis and Recognition* (1995), 58--63.

[52] Kakebeeke, T. H., Caflisch, J., Chaouch, A., Rousson, V., Largo, R. H., and Jenni, O. G. Neuromotor development in children. Part 3: motor performance in 3- to 5-year-olds. *Developmental medicine and child neurology 55*, 3 (2013), 248--256.

[53] Kamar, E., Hacker, S., and Horvitz, E. Combining human and machine intelligence in large-scale crowdsourcing. *AAMAS '12 Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS '12)* (2012), 467--474.

[54] Kato, Y., S.-H., and Ejima, T. In *International 1989 Joint Conference on Neural Networks*, 576.

[55] Kim, H.-h., Taele, P., Seo, J., Liew, J., and Hammond, T. EasySketch2 : A novel sketch-based interface for improving children's fine motor skills and school readiness. In *Proceedings of the International Symposium on Sketch-Based Interfaces and Modeling*, E. Association, Ed. (2016), 69--78.

[56] Kim, H.-h., Taele, P., Valentine, S., McTigue, E., and Hammond, T. KimCHI: a sketch-based developmental skill classifier to enhance pen-driven educational interfaces for children. In *Proceedings of the International Symposium on Sketch-Based Interfaces and Modeling (SBIM '13)*, ACM Press (New York, New York, USA, July 2013), 33--42.

[57] Kratz, S., and Rohs, M. The $3 Recognizer: Simple 3D Gesture Recognition on Mobile Devices. In *Proceedings of the International Conference on Intelligent User Interfaces* (2010), 419--420.

[58] Kristensson, P.-O., and Zhai, S. SHARK: A Large Vocabulary Shorthand Writing System for Pen-Based Computers. *Proceedings of the 17th annual ACM symposium on User interface software and technology - UIST '04 6*, 2 (2004), 43--52.

[59] LaLomia, M. User acceptance of handwritten recognition accuracy. In *Extended Abstracts of the ACM SIGCHI Conference on Human Factors in Computing Systems* (1994), 107--108.

[60] LeCun, Y., Boser, B. E., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W. E., and Jackel, L. D. Handwritten Digit Recognition with a Back-Propagation Network. *Advances in Neural Information Processing Systems 2* (1990), 396--404.

[61] Lee, S. W. Off-line recognition of totally unconstrained handwritten numerals using multilayer cluster neural network. *IEEE Transactions on Pattern Analysis and Machine Intelligence 18*, 6 (1996), 648--652.

[62] Lee, W., Burak Kara, L., and Stahovich, T. F. An efficient graph-based recognizer for hand-drawn symbols. *Computers and Graphics (Pergamon) 31*, 4 (2007), 554--567.

[63] Li, Y. Protractor: a fast and accurate gesture recognizer. *Proceedings of the 28th international conference on Human factors in computing systems* (2010), 2169--2172.

[64] Long, A. C., Landay, J. a., and Rowe, L. a. Implications For a Gesture Design Tool. *Proceedings of the International Conference on Human Factors in Computing Systems* (1999), 40--47.

[65] McKnight, L., and Fitton, D. Touch-screen technology for children. In *Proceedings of the International Conference on Interaction Design and Children*, ACM Press (New York, New York, USA, June 2010), 238--241.

[66] Mitchell, T. M. *Machine Learning*. 1997.

[67] Mohamad, R. A.-H., Likforman-Sulem, L., and Mokbel, C. Combining slanted-frame classifiers for improved hmm-based arabic handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence 31* (2009), 1165--1177.

[68] Myers, B. A., Giuse, D. A., Dannenberg, R. B., Vander Zanden, B., Kosbie, D. S., Pervin, E., Mickish, A., and Marchal, P. Garnet: Comprehensive Support for Graphical, Highly Interactive User Interfaces. *Computer 23*, 11 (1990), 71--85.

[69] Myers, B. A., McDaniel, R., Michish, A., and Klimovitski, A. The Design for the Amulet User Interface Toolkit. *Human Computing Journal*, January (1995), 9.

[70] Nacenta, M. A., Kamber, Y., Qiang, Y., and Kristensson, P. O. Memorability of pre-designed and user-defined gesture sets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*, ACM Press (New York, New York, USA, Apr. 2013), 1099.

[71] Nacher, V., Jaen, J., Navarro, E., Catala, A., and Gonzalez, P. Multi-touch gestures for pre-kindergarten children. *International Journal of Human Computer Studies 73* (2015), 37--51.

[72] Nakai, M., Akira, N., Shimodaira, H., and Sagayama, S. Substroke approach to hmm-based on-line kanji handwriting recognition. In *Proceedings of Sixth International Conference on Document Analysis and Recognition* (Sep. 2001), 491--495.

[73] Newell, K. M. Motor skill acquisition. *Annual Review of Psychology 42* (1991), 213--237.

[74] Nielsen, M. A. Neural networks and deep learning.

[75] Olsen, L., Samavati, F. F., and Sousa, M. C. Fast Stroke Matching by Angle Quantization. *International Conference on Immersive Telecommunications* (2007).

[76] Paulson, B., and Hammond, T. PaleoSketch : Accurate Primitive Sketch Recognition and Beautification. *Architecture*, September (2008), 1--10.

[77] Piaget, J. Piagets Theory. In *Handbook of Child Psychology*, P. Mussen, Ed. Wiley & Sons, New York, NY, USA, 1983.

[78] Ploetz, T., and Fink, G. Markov models for offline handwriting recognition: A survey. *IJDAR 12* (12 2009), 269--298.

[79] Prat, F., Marzal, A., Martn, S., Ramos-Garijo, R., and Castro-Bleda, M. J. A template-based recognition system for on-line handwritten characters. *J. Inf. Sci. Eng. 25* (05 2009), 779--791.

[80] Punch, S. Research with Children: The same or different from research with adults? *Childhood 9*, 3 (2002), 321--341.

[81] Read, J. C., MacFarlane, S., and Casey, C. 'Good enough for what?': acceptance of handwriting recognition errors by child users. In *Proceedings of the International Conference on Interaction Design and Children*, ACM Press (New York, New York, USA, July 2003), 155.

[82] Rosenbloom, L., and Horton, M. E. The Maturation of Fine Prehension in Young Children. *Developmental Medicine & Child Neurology 13*, 1 (1971), 3--8.

[83] Roy, P. P., Bhunia, A. K., Das, A., Dey, P., and Pal, U. HMM-based Indic handwritten word recognition using zone segmentation, 2015.

[84] Rubine, D. Specifying gestures by example. *ACM SIGGRAPH Computer Graphics 25*, 4 (1991), 329--337.

[85] Rust, K., Malu, M., Anthony, L., and Findlater, L. Understanding childdefined gestures and children's mental models for touchscreen tabletop interaction. *Proceedings of the Internatioanl Conference on Interaction Design and Children (IDC '14)* (2014), 201--204.

[86] Santrock, J. W. *Life-span development*. McGraw-Hill, 2006.

[87] Schneck, C. M., and Henderson, A. Descriptive analysis of the developmental progression of grip position for pencil and crayon control in nondysfunctional children. *The American journal of occupational therapy. : official publication of the American Occupational Therapy Association 44*, 10 (1990), 893--900.

[88] Seifert, K., and Hoffnung, R. J. *Child and Adolescent Development*. Houghton Mifflin, Boston, 1987.

[89] Sezgin, T. M., and Davis, R. HMM-based efficient sketch recognition. In *Proceedings of the International Conference on Intelligent User Interfaces*, ACM Press (New York, New York, USA, Jan. 2005), 281--283.

[90] Shaw, A., and Anthony, L. Analyzing the Articulation Features of Children's Touchscreen Gestures. In *Proceedings of the International Conference on Multimodal Interaction (ICMI '16)*, ACM Press (2016), 333--340.

[91] Shaw, A., and Anthony, L. Toward a Systematic Understanding of Childrens Touchscreen Gestures. In *Extended Abstracts of the ACM SIGCHI Conference on Human Factors in Computing Systems* (2016), to appear.

[92] Shrivastava, V., and Sharma, N. Artificial Neural Network Based Optical Character Recognition. *Signal and Image Processing: An International Journal 3*, 5 (2012), 73--80.

[93] Singh, S., and Amin, A. Neural Network Recognition of Hand-printed Characters. *Neural Computing & Applications 8*, 1 (Mar. 2014), 67--76.

[94] Smithies, S., Novins, K., and Arvo, J. A handwriting-based equation editor. In *Proceedings of Graphics Interface* (1999), 84--91.

[95] Soni, N., Gleaves, S., Neff, H., Morrison-Smith, S., Esmaeili, S., Mayne, I., Bapat, S., Schuman, C., Stofer, K. A., and Anthony, L. Do user-defined gestures for flatscreens generalize to interactive spherical displays for adults and children? In *Proceedings of the 8th ACM International Symposium on Pervasive Displays*, PerDis 19, Association for Computing Machinery (New York, NY, USA, 2019).

[96] Taranta II, E. M., and LaViola Jr., J. J. Penny pincher: a blazing fast, highly accurate $-family recognizer. 195--202.

[97] Ting, K. M. Confusion Matrix. In *Encyclopedia of Machine Learning*, C. Sammut and G. I. Webb, Eds. Springer US, Boston, MA, 2010, 209.

[98] Tu, H., Ren, X., and Zhai, S. Differences and Similarities between Finger and Pen Stroke Gestures on Stationary and Mobile devices. *ACM Transactions on Computer-Human Interaction 22*, 5 (2015), 1--39.

[99] Valentine, S., Vides, F., Lucchese, G., and Turner, D. Mechanix: A Sketch-Based Tutoring System for Statics Courses. *24th Annual Conference on Innovative Applications of Artificial Intelligence* (2012), 2253--2260.

[100] Vatavu, R.-D. The effect of sampling rate on the performance of template-based gesture recognizers. In *Proceedings of the 13th international conference on multimodal interfaces - ICMI '11*, ACM Press (New York, New York, USA, Nov. 2011), 271.

[101] Vatavu, R.-d. Improving Gesture Recognition Accuracy on Touch Screens for Users with Low Vision. In *Proceedings of the International Conference on Human Factors in Computing Systems*, ACM Press (2017), 4667--4679.

[102] Vatavu, R. D., Anthony, L., and Brown, Q. Child or adult? Inferring Smartphone users' age group from touch measurements alone. In *INTERACT*, vol. 9299 (2015), 1--9.

[103] Vatavu, R.-D., Anthony, L., and Wobbrock, J. O. Gestures as point clouds: a $P recognizer for user interface prototypes. In *Proceedings of the ACM International Conference on Multimodal Interaction (ICMI '12)*, ACM Press (New York, New York, USA, Oct. 2012), 273--280.

[104] Vatavu, R.-D., Anthony, L., and Wobbrock, J. O. Relative accuracy measures for stroke gestures. In *Proceedings of the ACM International Conference on Multimodal Interaction*, ACM Press (New York, New York, USA, Dec. 2013), 279--286.

[105] Vatavu, R.-D., Anthony, L., and Wobbrock, J. O. Gesture Heatmaps: Understanding Gesture Performance with Colorful Visualizations. In *Proceedings of the ACM International Conference on Multimodal Interaction*, ACM Press (New York, New York, USA, Nov. 2014), 172--179.

[106] Vatavu, R. D., Cramariuc, G., and Schipor, D. M. Touch interaction for children aged 3 to 6 years: Experimental findings and relationship to motor skills. *International Journal of Human Computer Studies 74* (2015), 54--76.

[107] Verma, B., Gader, P., and Chen, W.-T. Fusion of multiple handwritten word recognition techniques. *Pattern Recognition Letters 22*, 9 (2001), 991--998.

[108] Wacom. Professional Hybrid Creative Tablet Users Manual About the Cintiq Companion Hybrid.

[109] Willems, D., Niels, R., van Gerven, M., and Vuurpijl, L. Iconic and multi-stroke gesture recognition. *Pattern Recognition 42*, 12 (2009), 3303--3312.

[110] Wobbrock, J. O., Morris, M. R., and Wilson, A. D. User-defined gestures for surface computing. *Proceedings of the International Conference on Human Factors in Computing Systems (CHI 09)* (2009), 1083.

[111] Wobbrock, J. O., Wilson, A. D., and Li, Y. Gestures without libraries, toolkits or training: a $1 recognizer for user interface prototypes. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST '07)*, ACM Press (New York, New York, USA, Oct. 2007), 159--168.

[112] Woodward, J., Shaw, A., Aloba, A., Jain, A., Ruiz, J., and Anthony, L. Tablets, tabletops, and smartphones: cross-platform comparisons of childrens touchscreen interactions. In *Proceedings of the International Conference on Multimodal Interaction (ICMI '17)*, ACM Press (2017), 5--14.

[113] Woodward, J., Shaw, A., Luc, A., Craig, B., Das, J., Hall, P., Hollay, A., Irwin, G., Sikich, D., Brown, Q., and Anthony, L. Characterizing How Interface Complexity Affects Children's Touchscreen Interactions. In *Proceedings of the ACM International Conference on Human Factors in Computing Systems (CHI '16)*, ACM Press (San Jose, CA, USA, 2016), 1921--1933.

[114] Xiaolin Li, and Dit-Yan Yeung. On-line handwritten alphanumeric character recognition using dominant points in strokes. *Pattern Recognition 30*, 1 (1997), 31--44.

[115] Ye, Y., and Nurmi, P. Gestimator: Shape and Stroke Similarity Based Gesture Recognition. 219--226.

[116] Yin, J., and Sun, Z. An Online Multi-stroke Sketch Recognition Method. In *Affective Computing and Intelligent Interaction*, Springer (2005), 803--810.

[117] Zhai, S., Kristensson, P. O., Appert, C., Andersen, T. H., and Cao, X. Foundational Issues in Touch-Surface Stroke Gesture Design An Integrative Review. *Foundations and Trends in HumanComputer Interaction 5*, 2 (2012), 97--205.

[118] Zhang, W., Chen, J., Yang, Y., Tang, Y., Shang, J., and Shen, B. A practical comparison of de novo genome assembly software tools for next-generation sequencing technologies. *PloS ONE 3*, 6 (2011), e17915.

BIOGRAPHICAL SKETCH

Alex Shaw received his B.S. in Computer Science from Auburn University in 2013. He received his PhD in Computer Science from the University of Florida in 2020. His research interests include human-computer interaction, machine learning, and gesture recognition.