

A Toolkit for Teaching Arabic Handwriting

Hassanin Al-Barhamtoshy
Faculty of Computing &
Information Technology
King Abdulaziz University

Sherif Abdou
Faculty of Computers and
Information
Cairo University

Fakhraddin A. Al-Wajih
Faculty of Computers and
Information
Cairo University

ABSTRACT

The percentage of people who produce a neat and clear handwriting is declining sharply. The traditional approach for handwriting teaching is to have a dedicated teacher for long hours of handwriting practice. Unfortunately, this is not feasible in many cases. In this paper we introduce an automated tool for teaching Arabic handwriting using tablet PCs and on-line handwriting recognition techniques. This tool can simulate the tasks performed by a human handwriting teacher of detecting the segments of hypothesized writing errors and producing instructive real time feedback to help the student to improve his handwriting quality. The tool consists of two main components, the guided writing component and the free writing component. In the guided writing mode the student is required to write over transparent images for the training examples to limit his hand movements. After the student acquires the basic skills of handwriting he can practice the free writing mode where he writes with his own style, as he usually does in his daily handwritings. The first version of the tool was tested in several schools for children with edge ranging 4-11. The results are promising and show that this tool can help students to analyze their own writing and understand how they can improve it.

General Terms

Pattern recognition, handwriting recognition, handwriting verification.

Keywords

Handwriting teaching tool, SVMs for handwriting recognition, handwriting segmentation, guided writing, free writing.

1. INTRODUCTION

The ratio of persons who produce a neat and clear handwriting is declining sharply. The problem can be traced to the early stages of handwriting learning. Many students struggle to produce neat, expressive written work which may be due to physical or cognitive difficulties. Young children want to write well, but are often frustrated by their own lack of coordination and discouraged because it requires so much more effort to please either the teacher or themselves than they thought it would [9].

It is generally recognized that correct stroke making techniques are essential to good hand writing skills. These techniques can be successfully acquired only by practicing regularly and for long time periods. To date, methods of training handwriting in school mainly utilize the "blackboard and paper" approach. This consists of blackboard based demonstrations by teacher followed up by paper based examples and exercises for students. The 'blackboard and paper' approach has a number of serious drawbacks. The forming of each character is a dynamic process and students

need to become proficient in both the process and the final result. The process is only visible each time the teacher is actually writing on the blackboard. This process is not present whilst students are concentrating on their own work. Therefore the students are practicing the sequence, order and direction of the strokes of the character (s) based on what they can remember from watching the teacher's action. This is not reliable as the students may not remember all the process steps for the strokes. In a typical class room when writing or drawing the teacher's body will be between the students and the blackboard, and therefore vision of some students will be obscured at certain stages. Printed sheets with letters for tracing or copy are currently popular amongst certain education authorities. These suffer from the deficiency of only addressing the final look of the characters and not the process of their formation. When teachers come to assess handwritten work, they only see the final result, not the process that was used. It is impossible to tell whether or not the correct stroke making techniques have been applied. For example, the teacher will not be able to tell whether a pencil was lifted too many times when forming the character, or whether the character was stroked in an incorrect direction, or whether the writing was fluent or jerky.

Having a dedicated teacher for long hours of practice is not usually available. For example in Egypt the number of the admitted students in the elementary schools stage can reach one million students per year. Schools simply do not have sufficient resources to teach all children the handwriting skills with required interaction and attention. In recent years some educational software for teaching handwriting to children has been developed to provide students with extra self-practice besides the class teachers [7]. The handwriting lessons in most of this software mainly deal with showing writing models drawn on the computer screen associated with instructions to help the student to imitate the displayed model. Most of the educational load is on the student to learn his own errors and try to improve them using the try and error approach. A useful handwriting teaching tool should help the students learn to analyze their own writings and help them understand what writing errors they have done and how they can improve the quality of their writings.

Conventional mouse based personal computers have been used in some instances for teaching handwriting. These systems do not solve all of the problems outlined above. The mouse driven PC programs that teach writing do not reflect the way most students will write such characters in real life. Writing strokes with a mouse on a horizontal surface whilst watching them appearing on a vertical screen does not accurately mirror the process of writing with a pen, pencil or brush onto paper. Further these programs do not indicate the sequence and directions of students' strokes.

Recently, systems with combined LCD display and digitizers have been available. With these systems children can write with a pen directly on-screen without having to lift up their heads to look at what has been written. With these new hardware tools, we have reached the technological capability needed to build interactive systems to assist in teaching handwriting to children. Although these new systems provide a learning environment very close to the real one for handwriting teaching they still have limited capabilities when compared with human teachers [4]. Most of the currently available tools for handwriting training only give a very rough estimate of the overall quality of the student writing [6]. They measure how close the student writing to some ideal handwriting samples. Though this approach can judge the student handwriting quality and can evaluate his progress after some amount of practice, it can not provide any feedback about the regions of handwriting errors in the student writing. Also it does not provide any information on the types of errors the student have done and how he can avoid them in his next trails. This type of information is very crucial for any useful handwriting training tool.

In this paper we introduce an automated tool for teaching Arabic handwriting for children using tablet PCs and on-line handwriting recognition technology. The aim of this tool is to help young children to become good writers with fluent movements and a good quality of writing in shorter time frame. This handwriting teaching tool is able to recognize the student handwriting, detect the segments of hypothesized writing problems and produce instructive feedback to help the students to improve their handwriting. Also they can practice their handwriting whenever and wherever they want.

In the following sections, section 2 includes the description of our “Handwriting Teaching Tool” and its overall architecture. Section 3 describes the used preprocessing operations on the input handwriting signal. The tool is composed of two main modules, guided writing module which is described in section 4 and the free writing module that is described in section 5. Section 6 describes the handwriting data corpus that we used for training the tool models. Section 7 includes the results of several evaluations that we made for the first prototype version of our tool. Section 8 includes the final conclusions and our prospected future work and enhancements for our handwriting training tool.

2. TOOL DESCRIPTION

Following the methods used in schools for teaching handwriting, we designed our tool to consist of two main components, the *guided writing* component and the *free writing* component. . Figure (1) includes a flow chart of the main modules of the handwriting tool. The *guided writing* component is a preliminary level of education where students write characters or words on a transparent image for the training examples. This approach is equivalent to the method of writing over dotted images, which is frequently used in the initial lessons of handwriting teaching. After the student acquires the basic skills of handwriting he can move on to the second level of practice using the *free writing* mode. In this mode students are shown an image or a video animation of a handwritten example, then they are asked to write that example on an empty panel that contains a single line. That panel is similar to writing books used at schools. In the free writing mode the student has more freedom to write with his own style, as he usually does in his daily handwritings, then the tool evaluate his handwriting and give him feedback messages about his errors. To boost the performance of the handwriting recognition system several preprocessing operations are required on the raw input from the tablet

device. The following sections describe the detailed implementation of each one of these modules.

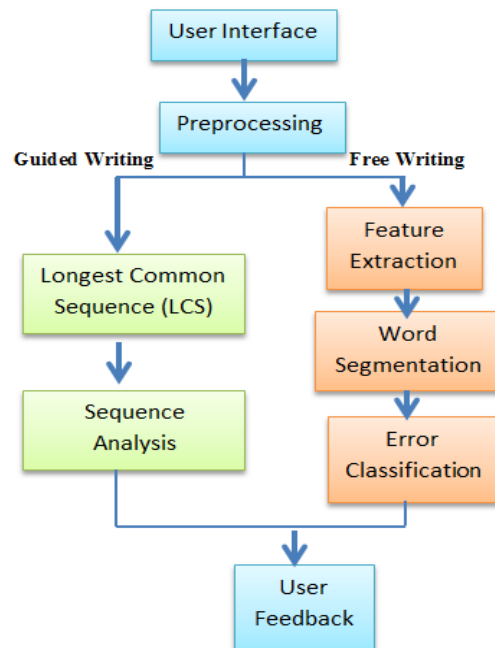


Fig 1 Handwriting Training Tool structure

3. PREPROCESSING

The first phase in the system modeling is preprocessing the input data. The basic problems involved in handwriting are: the noise during digitization, irregularity generated by inexperienced writers having an erratic handwriting and variations in handwriting styles. In our system, the following operations are applied:

- Duplicate Points Removal: By checking whether the coordinates of any two points in a stroke are the same. If so, one of them is removed.
- Interpolating Points: To add any missing points by linear Interpolation [3].
- Smoothing: To eliminate hardware imperfections and trembles in writing each point is substituted with the weighted average of its neighboring points [1].
- Re-sampling: Due to the variation in writing speed, the acquired points are not distributed evenly along the stroke trajectory. This operation is used to get a sequence of points which is equidistant [2].
- Strokes Reordering : Multiple-stroke character, such as “Ta ط” “Kaf ك” “Alif ا” may be written with several orders and directions. Moreover children tend to use much more delayed strokes than the standard ones. In many cases they return to complete parts of the characters that they previously wrote or even rewrite several copies over the previously wrote characters. This would affect on-line handwriting recognition. Building multiple models for the same character is inefficient approach. Normalizing the order of the strokes is an option to reduce the stroke order variance. We developed a new algorithm for handling the delayed stokes, more details can be found at [5].

4. GUIDED WRITING

This module is developed for kids who are in the earliest stages of motor-skill learning. The guided writing training includes three levels of lessons. The first level is the training

on writing a single letter with its different shapes. The second level is the training on writing complete words in progressive approach, i.e stroke by stroke, where the user is forced to follow the order and direction of writing the strokes of the reference template. In the third level the user practice writing complete words with no restriction for the strokes orders and direction.

In this mode the tool displays a transparent image for an ideally handwritten training example. The user is required to write over this transparent image. On the transparent image the tool sets specific control points. These points aren't visible to the user but they are used for tracking the user handwriting. The tool evaluates the user performance by measuring the difference between his handwriting and the reference writing. This process is done by initially aligning the user handwriting points with the reference template points to determine the parts that the user managed to correctly follow the reference writing and the parts that he deviated from it. To do this alignment we used the Longest Common Subsequence Similarity (LCSS) measure [8]. The LCSS, is a variation of edit distance measure used in speech recognition and text pattern matching. The basic idea is to match two sequences by allowing some elements to be unmatched. The advantage of the LCSS method is that some elements may be unmatched or left out (e.g. outliers), whereas in Euclidean distance and Dynamic Time Warping (DTW), all elements from both sequences must be used, even the outliers.

Figure 2 shows an example for two aligned patterns using the LCSS method. If X and Y are two sequences of length m and n , respectively and $c[i,j]$ denotes the longest common subsequences $\{x_1, \dots, x_i\}$ and $\{y_1, \dots, y_j\}$, the length of the longest common subsequence of X and Y can be recursively defined as follows:

$$c[i,j] = \begin{cases} 0 & \text{if } i = 0 \text{ or } j = 0 \\ c[i-1, j-1] + 1 & \text{if } i, j > 0 \text{ and } x_i = y_i \\ \max(c[i, j-1], c[i-1, j]) & \text{if } i, j > 0 \text{ and } x_i \neq y_i \end{cases} \quad (1)$$

Then the similarity between X and Y is estimated by:

$$LCSS(x,y) = 1 - \frac{m+n-2l}{m+n} \quad (2)$$

Where l is the length of the longest common subsequence.

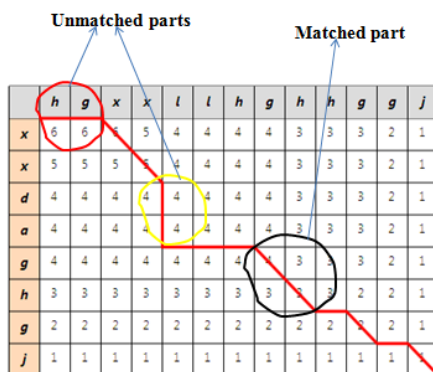


Fig 2: Longest Common Subsequence Matrix

When applying LCSS to the case of handwriting patterns we set a threshold that considers two points are equal if their distance are less than this threshold. This threshold can be tuned to control the degree of difficulty of the tool whether requiring the student to follow restrictively the reference

pattern or can allow minor deviations from it. Figure 3 show a guided writing example of the tool where the unmatched parts of the user handwriting and the reference pattern are marked with the red and yellow colors respectively.



Fig 3: Screen shot for Guided Writing

5. FREE WRITING

After the student acquires the basic skills of handwriting using the guided writing mode he should move on to the second level of practice using the free writing mode. In this mode students can practice handwriting on an empty panel that contains a single line similar to the writing handbooks used at school. The tool simulates the teacher functions and get to the student instantly feedback about his errors and how he can correct these errors. Figure 4 shows a sample of the free writing training examples.

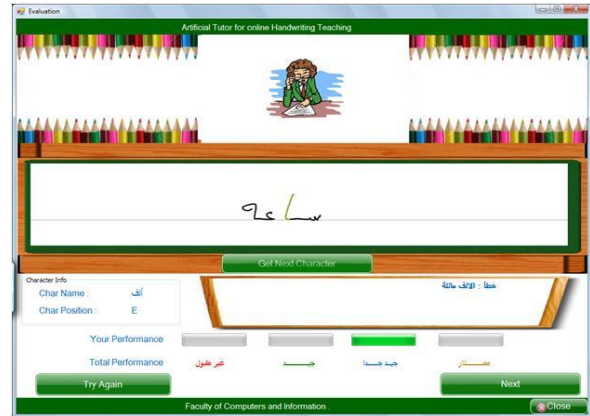


Fig 4 :The free writing tool

Before analyzing the user input for checking handwriting errors the required features are extracted and the sample is segmented to the characters level, if it was a word example, then these segmented characters passes through group of classifiers. Each one of these classifiers checks for the existence of a specific type of handwriting errors in the user handwriting. In the following sections we describe these steps in details.

5.1 Feature Extraction

We investigated several features and found this set of six local spatial features, which are commonly used in signature verification applications [5], to achieve the best performance in our tool. These features are the x and y coordinates, difference between two consecutive points, $(\delta x, \delta y)$, the sine and cosine of the angle with the x -axis ($\sin \alpha$ and $\cos \alpha$), and

the sine and cosine of the curvature ($\sin\beta$ and $\cos\beta$) as illustrated in Figure 5.

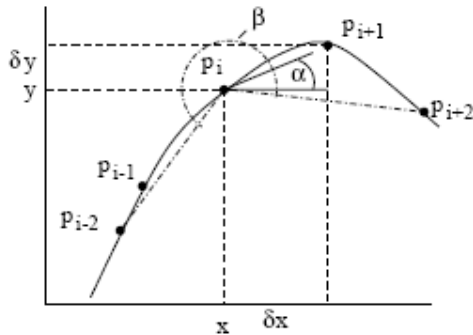


Fig 5 :Features Extraction

These features have different offsets in Y-axis and in order to obtain normalized features a z-score transformation is applied [11].

5.2 Word Segmentation

We use the Hidden Markov Models (HMM) in our tool in the alignment mode to find the optimum segmentation points of a word to its composing characters. The HMM technique provide solutions for most of the difficulties inherent in recognizing Arabic script including letter connectivity, position-dependent letter shaping, and delayed strokes. Figure 6 show an example for the segmentation of the word “ولد” “boy” to its composing three characters “د ل و”. In our tool we report the handwriting errors for each one of these characters separately. The HMM is a flexible tool that can search all the possible segmentation hypotheses for a word to find the optimum one, with highest match with the training data that the model has seen before.

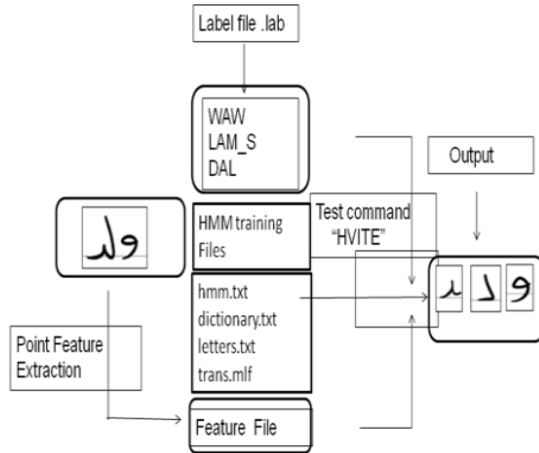


Fig 6 :Segmentation process of the word “ولد” (Boy)

A left to right HMM is used, with different number of states per model according to how complex the model shape is. For example, the “أ ALF” model has only 3 states, “د DAL” has 5 states, “ج GEM” has 7 states...etc. The segmentation HMM model that we use in our tool includes 115 different models that cover all the letters of the Arabic language, with their position dependent shapes, and the digits 0-9. Initially we built a mono-grapheme HMM model using the Maximum Likelihood (ML) training to maximize the probability of the training samples generated by the model. Then we expanded this initial model to a more sophisticated HMM model that is the tri-graphemes context-dependent model. For more details on this HMM model training the reader can refer to [18].

5.3 Errors Classification

This is the main component in the handwriting teaching tool. It is responsible for analyzing the segmented characters of the student handwriting and give him feedback on his performance. We collected a large data set of children handwriting samples from all the target grades. With help of some handwriting experts we made an analysis for this dataset to get knowledge about the type and rate of handwriting errors in the children handwriting. We found that some errors can be detected using simple geometric rules [5]. Some other errors required the design of more intelligent classifiers. We experimented with several types of classifiers and got the best performance with Support Vector Machines (SVMs) [10][11]. This is due to the generalization capabilities of SVMs even with limited training data. Our handwriting training tool provides feedback for more than two hundred types of errors and many classes of these errors are still represented with few samples in our training database.

We use binary SVMs to classify each handwriting error type against the ideal writing shape. Each segmented character of the user handwriting input passes through a group of classifiers each of them determine whether the inspected character contains a specific error or not.

The decision boundary of the SVM is calculated by:

$$f(x) = \sum_{i=1}^n (\alpha_i y_i K(x, x_i)) + b \quad (3)$$

Where y_i is the label of training sample, x_i is support vector, x is the sample to be classified and $K(x, x_i)$ is the kernel function. In our tool we use SVMs to classify the whole sequence of the user handwriting. To enable SVMs to perform classifications of unequal length sequences directly at the sequence level, a sequential kernel function need to be used [12][13]. In our tool we use Gaussian kernel with Time Warping (DTW). The Euclidean distance in the usual Gaussian kernel is replaced with dynamic time warping distance [14][15]. The DTW approach provides a way of computing a ‘distance’ between two sequences with different lengths. The DTW kernel function takes the form of:

$$K(a, b) = \exp(-\gamma DTW(a, b)), \gamma > 0 \quad (4)$$

To obtain a good performance, some parameters in SVMs have to be chosen carefully. These parameters include the regularization parameter C, which determines the trade-off between minimizing the training error and minimizing model complexity; and parameter γ of the kernel function that implicitly defines the nonlinear mapping from input space to some high-dimensional feature space. (In this work we entirely focus on the Gaussian kernel)[16]. Tuning these parameters is done by minimizing the estimated generalization error using the k-fold cross-validation error.

6. TRAINING AND EVALUATION DATA CORPUS

This data corpus included three types of collected samples for Separate letters, Single words and Sentences. The list of words and sentences were selected to be simple enough for children. We wanted to make the child concentrate on the handwriting practice and not spend much effort in understanding the meaning of the training examples. The data included samples that represent the left and right handed subjects. Also the data included balanced numbers of male and female samples. The data was collected for the two styles of the tool exercises, the guided and free writing. The data was collected from 9 schools and from 340 student. It is

known from classical studies of human behavior that the process of learning handwriting skills begins around age five and finishes approximately at age fifteen. In this project we collected data from students in the age range 7-10 as we expect this would be the optimum range for improving handwriting skills. The collected data size is around 20,000 samples that included 100,000 characters. Table (1) include the details of the collected data corpus.

Table 1 :Details of the collected handwriting corpus

	No. of children	Samples	Right Handed students	Left Handed students
Male	197	70235	170	27
Female	144	32517	110	34
Total	341	102752	280	61

A small portion of the corpus, around 10% , were selected to be manually segmented and annotated for the purpose of HMM models initialization. In this process each word is divided to its composing characters. If the character was written using multiple strokes they are grouped together and attached with the character label. We developed a special tool for data segmentation and annotation. This tool allows the user to do the segmentation by hand using the touch screen pen which accelerated the segmentation process. Figure.7 shows a screen shot for the data annotation tool.

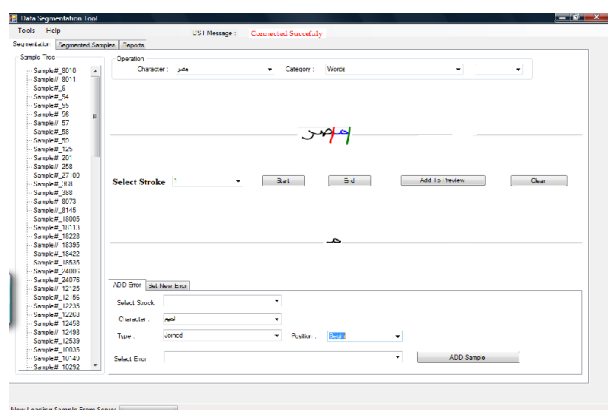


Fig 7 :The data annotation tool

7. SYSTEM EVALUATIONS

For evaluating the handwriting training tool kit test we selected fifty children from an elementary school, their ages were in the range 4-11 years old. We evaluated both of the two components of the tool which are the guided writing and the free writing modules.

7.1 Guided Writing module evaluations

For testing this module we selected the younger students since they are better representatives for the target users if that module. We run a subjective test that covered all the included training examples in the tool. An expert teacher judged the tool feedback whether it is helpful for the student or misleading, i.e the number of correctly marked error segments after excluding the false alarms. For that evaluation we collected handwritings for 350 words that the tool marked

in them 132 segments as errors, Table 2 includes the expert judgment for them.

Table 2 :Guided writing evaluations

Correctly Marked Segments	Incorrectly Marked Segments	Missed segments
105	27	12

After analyzing the incorrectly judged segments we found that in many of them the student didn't follow the displayed directions for writing. The preprocessing operation of *stroke reordering* was integrated in the tool to handle this type of unordered writings. To evaluate the performance of the *stroke reordering* operation we evaluated separately the examples in which the students followed the displayed writing order and the ones in which the students wrote with different orders. Table 3 displays this result.

Table 3 :Guided writing evaluations with

Writing Order	Correctly Marked Segments	Incorrectly Marked Segments
Followed reference order	85	7
Different from reference order	20	20

From the results in table 3 we can see that in 75% of the incorrectly marked segment were in cases that the student did not follow the reference writing order. This mean the used stroke reordering method still need more improvements as it worked correctly only in 50% of the cases when the student didn't follow the reference ordering. For adults we can enforce some sort of handwriting restrictions, such as writing word parts in single strokes and forbidding back movements, to reduce the cases of delayed strokes. For children such kind of restrictions would be very hard and for sure they will not be able to follow them. Which emphasize the importance of resolving the problem of the delayed strokes.

7.2 Free Writing Evaluations

For this module we evaluated the two main components the HMM segmentation component and the SVMs errors classification using a testing data set of 490 words. Table 4 summarize the evaluation for the HMM segmentation.

Table 4 :Evaluation results for the HMM segmentation

The Training Data	Accuracy
Children only	66%
ALTEC database	85%
ALTEC + Children adaptation	94%

The initial HMM model was trained using the children training data set, the data set described in section 6. This model, achieved a segmentation accuracy of 66% [5]. When we trained the HMM model using a large scale handwriting

database, the ALTEC handwriting database [17], which included samples from 1000 writers with 150,000 words we achieved a segmentation accuracy of 85%. This boosted performance is due to using this large scale database with wide varieties of writing qualities and styles. Finally we adapted this model using the children handwriting training dataset. This adapted models provided the segmentation accuracy of 94%.

For the correctly segmented words we measured the accuracy of the SMVs handwriting errors classifiers. Table 5 displays samples of the reported handwriting errors by the tool and the achieved accuracy for recognizing each one of them.

Table 5 :Evaluation results for handwriting errors recognition

Correct sample	Error Description	Error sample	SVM	
			Acc	F-score
و	تدويرة حرف الواو صغيرة	و	0.98	0.98
س	اربع سنن في حرف السين	سس	0.86	0.84
س	تدويرة حرف السين غير مكتملة	س	0.95	0.94
ر	حرف الراء شبه الدال	ر	0.90	0.88
ر	لا يوجد انحاء في جسم الراء	ر	0.98	0.97
ح	لا يوجد انحاء في حرف الحاء	ح	0.96	0.95
ص	حرف الصاد بدون سنة	ص	0.96	0.96
ع	حرف العين شبه رقم اربعة	ع	0.88	0.71
ف	لا يوجد عنق لحرف الفاء	ف	0.91	0.91
ه	تدويرة حرف الهاء مفتوحة	ه	0.98	0.975

From the results displayed in table 5 we can see that the tool managed to detect some handwriting errors with accuracy 98%. The overall accuracy of the tool for detecting the different types of handwriting errors, which are currently implemented in the tool, is in the range 80%-98% with average accuracy 91%. For some types of handwriting errors the tool managed to detect them with very high accuracy but for other errors types the accuracy still need more

enhancements to be practical for enhancing the students' handwriting skills.

In the last evaluation we measured the impact of using the tool for improving the handwriting skills of the tool users. We selected a group of 50 students from different ages 6-10 years old. We asked a handwriting expert to evaluate samples of their handwritings before and after using the toolkit for several training sessions. Figure 8 shows the improvements rates for this group of students. We noticed that the most significant improvement was for the 6-7 years old children. They managed to copy the ideal writings with perfect performance. They required around three hours of practice to reach that level with no teacher guidance. The older children had harder time trying to change their writing style and didn't show much improvement compared to the younger ones.

8. CONCLUSION AND FUTURE WORK

Handwriting does not have to be a battleground. By targeting specific and narrow objectives, praising efforts that are well-done as well as pointing out errors to be corrected, and scheduling regular, supervised practice, progress can be made much more rapidly than if students are left on their own to complete handwriting workbooks. Through this project we were able to explore, and also enjoy, an important problem which is teaching handwriting for kids. The Arabic language had its own challenges of cursive writing, the many dots and delayed strokes. Also the recognition of the handwriting of children is much more challenging than adult handwriting due to the increased irregularities, the lossy control of the pen movements and the fragmented writings. We implemented some standard techniques for Arabic handwriting recognition and also developed new techniques that can handle the challenging handwriting of children. We tested the components of the application in real usage scenarios. The tool works with reasonable accuracy considering it is the first version. Also considering this application is a new one and there are no similar products in the market that we can compare with.

This handwriting teaching tool can increase the effectiveness of classroom teachers in several ways. It can provide positive, independent, individualized, and effective practice for students, and it can give the teacher detailed feedback on each student's progress. It can help free teachers' time by enabling students who need more individualized instruction to work independently with effective learning tools on a computer, while other students in the classroom receive more interaction and attention from the teacher. The tool was developed for the Arabic language but can easily be ported to other languages since all the language related information are stored in external databases.

In our future work we plan to extend the application to include non-native Arabic students and increase the types of handled handwriting errors. Also will develop a version on mobile devices to approach the illiterate communities.

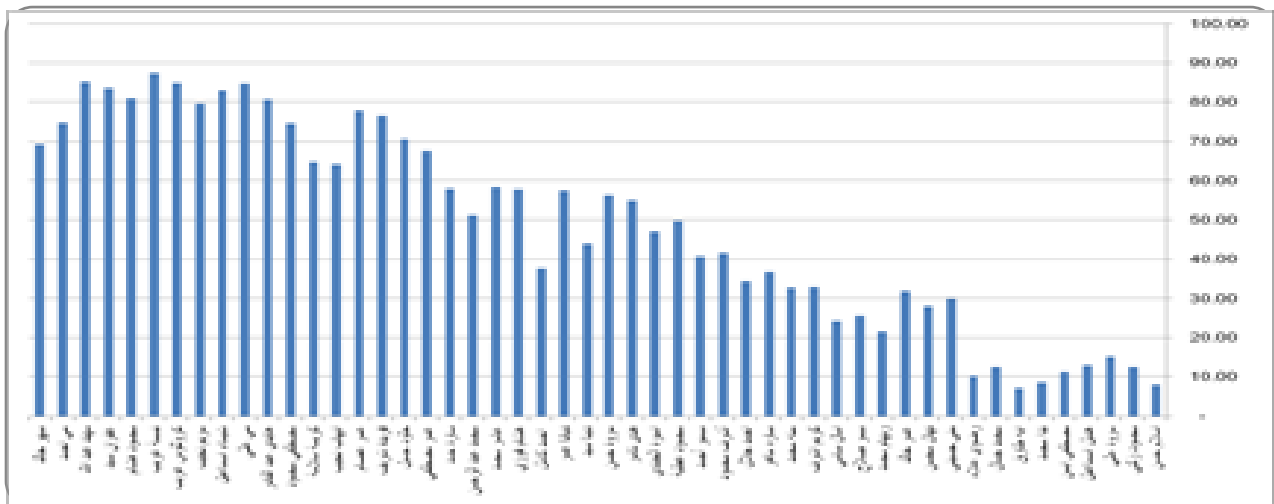


Fig 8 :Students improvement rates

9. ACKNOWLEDGMENT

Authors of this paper thankful to King AbdElaziz University (KAU) through their grant's number D1432-18-611. Their financially and support during the period of this research took place is greatly acknowledged.

10. REFERENCES

- [1] E. Tapia, "Understanding mathematics: A system for the recognition of on-line handwritten mathematical expressions," Ph.D. dissertation, the University of Fu Berlin, 2004.
- [2] Hu J., Lim S.G., Brown M.K. Writer independent on-line handwriting recognition using an HMM approach (2000) *Pattern Recognition*, 33 (1), pp. 133-147.
- [3] Bing Quan Huang, Y. B. Zhang, M. Tahar Kechadi. "Preprocessing Techniques for Online Handwriting Recognition". In *Collection of Intelligent Text Categorization and Clustering*, 2009: 25~45.
- [4] Greer, T. , Lockman, J. J. (1998). *Using Writing Instruments: in variances in Young Children and Adults*, *Child Development*, vol. 69, no. 4, pp. 88-902.
- [5] S. Abdou, A. Elgammal, and A. Fahmy, "A Tool for Arabic Handwriting Training". *The Fifth Conference of Learning International Networks Consortium (LINC)*, 23-26 May 2010, MIT, Cambridge, Massachusetts, USA.
- [6] Marse, J. F. et al, (1991). *Handwriting Training: Computer-Aided Tools for Premedial Teaching, Development of Graphic Skills*. J. Wann, A.M. Wing, N. Suvik, eds., pp. 249-258, Academic Press.
- [7] Muroya et al, (2001) U. S. Patent No. 5,040,222.
- [8] Narao Nakatsu, Yahiko Kambayashi and Shuzo Yajima, "A longest common subsequence algorithm suitable for similar text strings" *Acta Informatica*, Volume 18, Number 2 (1982), 171-179.
- [9] Walter B. Barbe, Virginia H. Lucas and Thomas M. Waslylyk , "Basic Skills for Effective Communication", copyright 1994.
- [10] C. Cortes, V. Vapnik, in *Machine Learning*, pp. 273{297 (1995).
- [11] N. Cristianini, J. Shawe-Taylor, *An introduction to support Vector Machines: and other kernel-based learning methods*. Cambridge University Press, New York, NY, USA (2000).
- [12] C. Watkins, "Dynamic alignment kernels," *Tech. Rep., Royal Holloway College, University of London*, 1999.
- [13] W. M. Campbell, "Generalised linear discriminant sequence kernels for speaker recognition," in *Proc. ICASSP*, 2002.
- [14] C. Bahlmann, B. Haasdonk, and H. Burkhardt, "On-line handwriting recognition with support vector machines – a kernel approach," in *Proc. 8th International Workshop on Frontiers in Handwriting Recognition*, 2002, pp. 49–54.
- [15] V. Wan, J. Carmichael, "Polynomial Dynamic Time Warping Kernel Support Vector Machines for Dysarthric Speech Recognition with Sparse Training Data", *Proc. Interspeech'05*, pp 3321-3324, 2005.
- [16] Duan K., Keerthi S.S., Poo A.N. Evaluation of simple performance measures for tuning SVM parameters (2003) *Neurocomputing*, 51, pp. 41-59.
- [17] S. Abdou, M. W. Fakher, I. Hosney, F. Alwajeesh "A general purpose large scale Arabic Online Handwriting Corpus" *The Egyptian Society of Language Engineering Conference*, December 2010.
- [18] I. Hosseney, S. Abdou, A. Fahmy, "Using Advanced Hidden Markov Models for Online Arabic handwriting recognition", *ACPR2011*, Bekeen, China, 29-30 Nov 2011.