Preprocessing of Online Urdu Handwriting for Mobile Devices

Fareeha Anwar[†], Muhammad Adnan Aftab[††],Dr. Syed Afaq Hussain[†††]and Dr.Ayyaz Hussain[†]

[†]Faculty of Basic & Applied Sciences, International Islamic University, Islamabad, Pakistan
[††]Sky United Kingdom,
[†††]College of Computer Science & IT, King Faisal University, Al-Hassa, Saudi Arabia

Summary
Mobile and hand held devices are becoming a necessity these days. New technologies are emerging as more and more mobile devices are used commonly. Touch devices are cheap and affordable these days. Touch screen has increased the mobile devices utilization and more navigation of mobile systems. Keyboard are used since long and an easy way to input for writing text. But it is difficult to input cursive languages. Urdu is one of the example of cursive language. Urdu input is difficult using a traditional keyboard. Therefore, there is need to devise a software, which take online input for Urdu text, and after performing processing task can recognize it more efficiently and give correct results. Preprocessing is one of the most important steps in process of recognition. Good preprocessing will directly results in efficient recognition. In this paper, a novel technique of preprocessing of Urdu text for mobile devices is proposed. Results shows that proposed preprocessing technique results in fast and efficient recognition.

Keywords:
Strokes, Preprocessing, Urdu, Cursive Language, Online Handwriting, Mobile devices

1. Introduction

Taking notes or writing messages using pen or finger will provide ease in every field of life and if mobile devices understand and interpret whatever we write with pen then the life become very easy. Currently different software exists in market that allows users to write in free handwriting and then recognize the words. Most of the work has done in European languages and specially the languages written from left to write.

Online Handwriting recognition is still challenging field for researchers. Cursive languages are hard to recognize. Considerable variations in handwriting style, noise introduced by touch devices, distortion of handwritten characters, different styles of handwritten character etc. make online handwriting difficult to recognize. Recognition process will be fast and accurate if a recognition engine gets clear and noise free input. Therefore preprocessing of input is very important in online handwriting recognition.

Cursive languages and the languages written from right to left are complex and difficult to recognize efficiently. In Pakistan, most of the people are novice to computer or handheld devices. Typing messages or taking notes in Urdu language is not an easy task as Urdu is complex language.

First step in online recognition is data acquisition. Data is acquired in the form of coordinates and then comes preprocessing. The main objective of the preprocessing steps is to normalize words and remove variations, which complicate recognition and reduce the recognition rate. Efficient preprocessing helps in better recognition. In online handwriting recognition, preprocessing plays very important role. Efficient and precise preprocessing of input generates accurate results. Given below are some factors which effects handwriting recognition.

1.1 Factors effecting Online Handwriting recognition.

Number of factors effect efficient handwriting recognition. All factors must handle prudently otherwise; they results in wrong recognition and thus produce low recognition rate. The important factors are

- Different handwriting styles.
- Sensitivity of digitizer
- Different writing speed
- Noise/ irregularity in input

Preprocessing plays vital role in good recognition. Efficient preprocessing overcome all factors stated above and helps in efficient feature extraction and correct recognition. Looking at the consideration and limitations of mobile phones we have devised new preprocessing steps which eliminate noise and helps in fast processing.

2. Related work

Urdu Online handwriting recognition is also very challenging task due to its cursive nature and variety of writing styles and become more difficult in context of mobile devices. Most of the research is done in the field of OCR. Research in the field of intelligent character recognition is less. Even within the context of online

Manuscript received October 5, 2017
Manuscript revised October 20, 2017

handwritten character recognition, studies dealing with Arabic and Urdu characters are scarce [23].

Yanming Zou et al. [1] proposed recognition of Arabic characters and numbers only for mobile devices. They used 3 NN and decision tree to select the NN. Recognition rates were not good but recognition time decrease. They developed algorithm for mobile devices. In the field of online writing recognition for mobile devices, research is scarce.

In 2014, Ibrahim et al [2] devised new algorithm for preprocessing of delayed strokes in Arabic language. The steps performed for preprocessing were de-hooking, removal of duplicate points, smoothing, and interpolation and re sampling. The proposed system recognized very large Arabic vocabulary efficiently. Andreas et al. [3] introduces Entropy based preprocessing for online English handwriting. First sampling and normalization of input was done and then skew and slant was corrected using entropy. This performed well as compared to windows based approach

In 2012, Gupta et al. [4] proposed novel preprocessing algorithm for online Gurmukhi characters. Writing area was divided into 300x300 working area. They followed different sequence. They performed normalization then interpolation and uniforming. Interpolation was done again followed by smoothing, uniforming and sampling. They introduces new algorithm for uniformizing and smoothing. Bezier curves was used for interpolation. This efficient preprocessing steps result decent recognition rates. H. Kaur [5] also worked on online Gurmukhi scripts. He combined offline and online techniques of preprocessing. After normalization and interpolation, slant corrected by detecting baseline. [6]

Mai Al-Ammar et al. [7] recognize online Arabic characters. They normalized stroke to 100x100 size then they computed center of gravity for translation. Douglas and Peucker‘s algorithm used for smoothing and noise removal. Hany Ahmed and Sherif Abdel Azeem [8] achieved 90% recognition rate for online Arabic words. Interpolation performed for each stroke separately and 5-point moving average algorithm used for smoothing. They extracted Baseline using projection profile.

Hosny et al. [9] introduced preprocessing for delayed strokes for recognition of inline Arabic words. Before this they removed duplicate points, performed interpolation, smoothing and sampling of points. Delayed stroke detection and removal helps in efficient recognition. Monji et al. [10] combined offline and online preprocessing steps for Arabic online word recognition.ADAM database used for testing and training. Their system achieved good recognition rates.

Khaled Al-Ghoneim [12] proposed recognition algorithm for isolated Arabic characters for handheld devices. Preprocessing involved translation by finding center of gravity, scaling. Then connected line generated using Bresenham’s Line Generator algorithm. It was only dealing with isolated characters. Khaled et al. [13] applied low pass filter for smoothing. Their focus was on segmentation. Recognition rate was 92% for words.[14]

Khan et al. [15] recognized online isolated Urdu characters. They first resampled the input then applied de-hooking and smoothing. Imran et al. [17] proposed combined technique for preprocessing of online Urdu characters. They combined offline and online preprocessing. They segmented characters based on the threshold value and position with respect to the previous character. They combined offline and online techniques for preprocessing.

S.A. Husain et al. [18] developed a new engine for online handwritten Urdu words. They first performed dehooking and then smooth online handwritten strokes. BPNN was used for recognition of ligature and words. They achieved 93% recognition rates. S. Malik and S.A. Khan [19] also worked on Urdu numerals and two character words. For preprocessing of strokes repetition of strokes was removed and filtering was done to get good results. [20]

Fadiet al.[21] research based on online Arabic handwriting. Noise removal was done using low pass filter then Douglas and Peucker’s method used for smoothing of stoke with tolerance t1. Writing speed normalized using resampling of strokes. Riad and jihad [23] performed geometrical preprocessing for online Arabic handwriting. For Noise removal and smoothing, they used Dynamic Time Warping (DTW).

Based on the literature survey it is clear that most of the steps for preprocessing are almost same in different languages [Gupta]. Basic steps of preprocessing are given below

- Smoothing.
- Removing duplicate points.
- De Hooking.
- Normalization
- Resampling

Most of the work has done for English, Chinese, Japanese, Indian languages but very less research found for Online Urdu handwriting recognition. Our main objective is to device a new Preprocessing algorithm that handle input efficiently and helps to extract good features for recognition.

3. Proposed Preprocessing Algorithm

Preprocessing is perform before feature extraction and recognition. It plays very important role in efficient online handwriting recognition. The objective of preprocessing is to remove noise and make input uniform and smooth. This help in better feature extraction and recognition process boost up. We proposed novel algorithm for preprocessing the major steps in preprocessing are given below in figure 1. After taking input from mobile screen using pen or finger, system perform preprocessing.

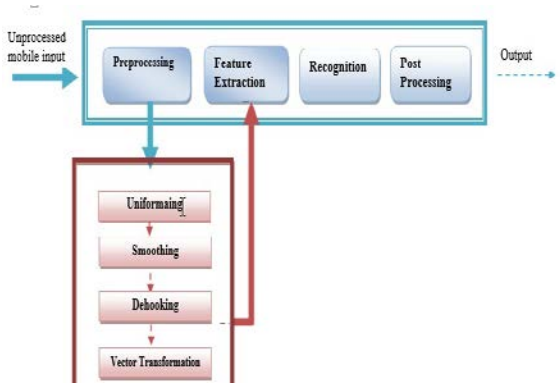

Fig.1. Proposed Preprocessing steps

3.1 Uniforming:

Uniforming is the process to remove the extra points from the user input to get a refined set of equally distant points from each other, for further processing. User input speed can will produce different results. If a user writes slowly, few extra points are produced that will result in slower processing and then recognition. If user gives the input in fast speed then important points are misses that will mislead the process to give wrong recognition. We have developed algorithm for name uniforming. This algorithm helps to have fast processing because there are fewer points to go consider. Algorithm will consider only those points which are at distance greater than threshold distance. Table 1 shows proposed algorithm for uniforming of input.

Table 1: Algorithm for Uniforming

Algorithm: Uniforming

Input : Array p
Output : Array u having uniform points

```
1. set count = number of points
2. set t = 10 (threshold distance)
3. insert first item of p to u (consider
   first point is always uniform)
4. set i =1
5. loop i<count-2
   set p[i] to point
   set p[i+1] to nextPoint
   d= Distance (p[i], p[i+1])
   if d>=t
   insert point to u
6. return u
```

Distance (p1, p2)

```
p1= first point
p2= Second point
X=|p1.x - p2.x|
Y=|p1.y - p2.y|
Distance= √(X² + Y²)
Return Distance
```

3.2 Smoothing:

Smoothing is process to remove the extra noise which comes with the input strokes due to shaking hands or bad writing style. This results in more noise and extra points. Smoothing eliminates extra points to get a smooth set of points. To smooth a point 'p' we have taken 5 points, two its previous points and two next points and then taken mean of all five points to get smooth point. Smoothing is done after uniforming so that most of the input is already smoothed in uniforming step. It will help in fast processing of online Urdu character taken from mobile phones. Details of proposed algorithm is given below in table 2

Table 2: Algorithm for Smoothing

Algorithm: Smoothing

Input : Array P which hold points
Output : Array S having smooth points

1. count is number of items in array
2. if count < 5
return P (less than 5 are already smooth)
3. S an array which will hold smooth points after process
4. insert P[0] to S
5. insert P[1] to S
6. set i = 2
7. loop i < count-2
    1. retrieve P_{i-2}, P_{i-1}, P, P_{i+1}, P_{i+2}
    2. $X_m = \frac{P_{i-2}X + P_{i-1}X + P_iX + P_{i+1}X + P_{i+2}X}{5}$
    3. $Y_m = \frac{P_{i-2}Y + P_{i-1}Y + P_iY + P_{i+1}Y + P_{i+2}Y}{5}$
    4. create point pi by using X_m and Y_m
    5. insert pi to S
8. return **S**

Figure 2 shows result of uniforming and smoothing. One can clearly notice that input word is smooth having no jagging effects and also it eliminates extra points. Therefore to avoid extra work we have performed dehooking after uniforming and smoothing.

Fig. 2 Smoothing of Ligature

3.3 De-hooking and Preservation of shape features

Hooks (ﻉ) re-traces are commonly unwanted parts and of the user input that effect the efficiency of recognition process. Hooks are mostly found at the end of the input but sometimes can be seen at the beginning of the writing as well. Hooks could be due to writing speed, style or inexperience users. If Hooks are not removed, it will be difficult to detect original ligature. Lesser are the unwanted parts, greater is the recognition rate. Urdu language has lots of different variations in writing. It has loops, sharp edges, cusps and curves etc. In order to preserve edges, cusp points, loops present in stroke and to retain the original shape, we have written a new algorithm. Results are achieved by looking at angles. Table 3 shows proposed algorithm for de hooking and preserving of shape features

Table 3: Algorithm for Dehooking and Curve/cusp preservation

Algorithm: Dehooking and Cusp preservation

Input : Array p which hold points
Output : Array d

```
1.  count is number of points in p
2.  if count<=2
        return p
3.  d is an array which will hold points after
    process
4.  Pt1 = p[1]
5.  Ptn =p[count-1]
6.  set t = 25
7.  set dt=12
8.  set i = 1
9.  loop i < count-2
        1.  Last_Θ = Angle(p[i], p[i-1] )
        2.  Θ = Angle(p[i], p[i+1] )
        3.  Δθ = | (Θ - Last_Θ) |
        4.  if Δθ> t
            1.  if d.count> 0
                    d[0]= Pt1
            2.  else
                    Pt1 = p[1]
            3.  Dist_Total= Distance (Ptn,
                Pt1)
            4.  Dist= Distance (p[i], Pt1)

            5.  Δ= Dist / Dist_Total * 100 (0<Δ<100)
            6.  if Δ< 12 and p[i] !=
                circlePoint
                 remove all objects from d
        5.  insert point in d
10. return d
```

Angle(p1, p2)

```
X=|p1.x - p2.x|
Y=|p1.y - p2.y|
```

$\theta = \mathbf{atan}(\mathbf{X}, \mathbf{Y}) * \mathbf{180} / \prod$

```
    Return θ
```

Results of proposed algorithm are given in figure 3. One can clearly observe that in addition with removing extra parts proposed algorithm is also preserving its shape. Cusp points are same. For this we are looking at angles

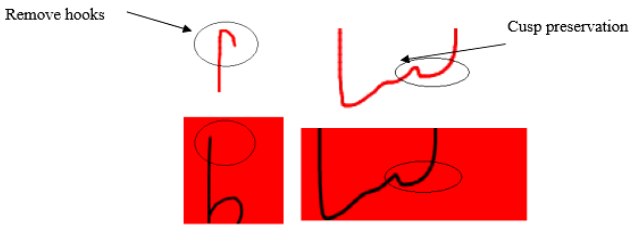

Fig. 3 De hooking and shape preservation

4. Results and Discussions

Proposed Preprocessing algorithm is implemented in online Urdu handwritten character recognition application for mobile phones. We developed an application for I phone users to write Urdu characters without using keyboard. Experiment is done by taking input by 70 users to have a different handwriting styles. Figure 4 shows recognition rates after applying preprocessing on input stroke and without applying preprocessing. If noise and extra points are not removed in the process of preprocessing, recognition rate becomes slower and inefficient especially for cursive language like Urdu. The results clearly show recognition is boosted by applying these algorithms.

For example recognition rate of Urdu Characters having loops and double loops increases approximately 40% when proposed preprocessing algorithm is applied.

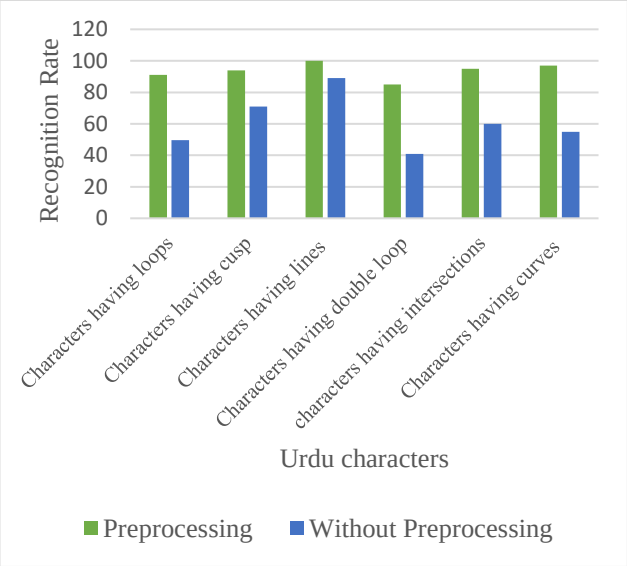

Fig. 4. Recognition rates with and without preprocessing

Proposed preprocessing algorithm refined input stroke and recognition of characters having complex features is very impressive. Table 4 provides recognition rates of characters having complex features. Features are preserved during preprocessing also if feature is drawn incompletely our preprocessing algorithm completes that feature looking at properties of Urdu features. This boosts recognition rate of Urdu characters written on mobile screen.

Table 4: Recognition Rate after Preprocessing

Features	Preprocessing
Loops	91
Cusp	94
Horizontal and Vertical lines	100
Double loop	85
Intersections	95
Curves	97

5. Conclusion

We have proposed new preprocessing algorithm for online Urdu character recognition for mobile devices. Mobile phones have limited memory and processing speed plays very imported role in mobile applications. Results shows that proposed preprocessing algorithm tries to remove all noise and imperfections. Features of Urdu languages e.g. loops, Cusp, curves are also preserved during preprocessing phase. This eliminates feature extraction phase and speeds up recognition. Results also show that recognition rate increases from 50% to 95% using preprocessing technique.

References

[1] YanmingZou; Yingfei Liu; Ying Liu; Wang Kongqiaoanming. Overlapped handwriting input on mobile devices. Proceedings International Conference on Document Analysis and Recognition (ICDAR), Beijing (2011), pp. 369-73

[2] Ibrahim Abdelaziza,,SherifAbdoua, Hassanin Al-Barhamtoshyb. Large Vocabulary Arabic Online Handwriting Recognition System

[3] Andreas Holzinger, Christof Stocker, Bernhard Peisch and Klaus-Martin Simonic. On Using Entropy for Enhancing Handwriting Preprocessing 2012

[4] Gupta, Nainsi; Gupta, Mayank; Agrawal, Rahul. Preprocessing of Gurmukhi Strokes in Online Handwriting Recognition. (December 2012, International Proceedings of Computer Science & Information Tech;2012, Vol. 56, p163

[5] H. Kaur. Interpolation of missing points and Smoothing of character for online handwriting recognition of Gurumukhi Script. Thesis submitted 2011.

[6] Sharma, A., Kumar, R. and Sharma, R. K., 2008. Online Handwritten Gurmukhi Character Recognition Using Elastic Matching, Congress on Signal and Image Processing, vol. 2, pp. 391-396

[7] Mai Al-Ammar, Reham Al-Majed, HatimAboalsamhOnline handwriting recognition for the Arabic letter set. CIT'11 Proceedings of the 5th WSEAS international conference on Communications and information technology USA 2011 Pages 42-49

[8] HanyAhmed, Sherif Abdel Azeem. On-line Arabic Handwriting Recognition System based on HMM. 2011 International Conference on Document Analysis and Recognition

[9] Hosny, I., Abdou, S. and Fahmy, A., 2011. Using advanced hidden markov models for online Arabic handwriting recognition, First Asian Conference on Pattern Recognition, pp.565-569

[10] MonjiKherallah, NajibaTagougui, Adel M. Alimi, Haikal El Abed, Volker Margner 2011 Online Arabic Handwriting Recognition Competition, International Journal on Document Analysis and Recognition (IJDAR)March 2011, Volume 14, Issue 1, pp 15-23

[11] Zhao, H., Linfeng H. and Hao, S., 2011. An online recognition algorithm of handwritten Uyghur characters, Seventh International Conference on Natural Computation, vol. 3, pp. 1616-1619

[12] Khaled Al-Ghoneim. Online Arabic Character Recognition for Handheld Devices. Proceeding IPCV 2008: pp 15-22

[13] Khaled Daifallah, Dr. Nizar Zarka and Hassan Jamous. Recognition-Based Segmentation Algorithm for On-Line Arabic Handwriting. Proceedings ICDAR '09 Proceedings of the 2009 10th International Conference on Document Analysis and Recognition, pp 886-890

[14] Huang, B. Q., Zhang, Y. B. and Kechadi, M. T., 2007. Preprocessing techniques for online handwriting recognition, Seventh International Conference on Intelligent Systems Design and Applications, pp. 793-800.

[15] Khan, K. U. and Haider I., 2010. Online Recognition of Multi-Stroke Handwritten Urdu Characters, IEEE International Conference on Image Analysis and Signal Processing, pp. 284-290.
[16] NabeelShahzad, Brandon Paulson and Tracy Hammond, Urdu Qaeda: Recognition System for Isolated Urdu Characters. Proceedings of Intelligent User Interfaces (IUI 2009) Workshop on Sketch Recognition (Long Talks) , Sanibel Island, Florida, February 8-11, 2009.
[17] Muhammad Imran Razzak, Syed Afaq Hussain, Muhammad Sher, ZeeshanShafiKhan,Combining Offline and Online Preprocessing for Online Urdu Character Recognition. Proceedings of the International MultiConference of Engineers and Computer Scientists Vol I IMECS 2009, March 18 - 20, 2009, Hong Kong, pp 912-915
[18] S.A.Hussain , Anwar F., Asma. "Online Urdu Character Recognition System." MVA2007 IAPR Conference on Machine Vision Applications. 2007. pp. 98-101
[19] Malik, S. Khan, S.A., "Urdu Online Handwriting Recognition", Emerging Technologies, 2005. Proceedings of the IEEE Symposium on Volume, Issue, 17-18 Sept. 2005 pp. 27 – 31,
[20] M. Hussain et. al. "Urdu Character Recognition Using Spatial Temporal Neural Network", Proceedings INMIC 2005.
[21] FadiBiadsy Jihad El-Sana NizarHabash. Online Arabic Handwriting Recognition using Hidden Markov Model…
[22] A. Brakensiek, A. Kosmala, D. Willett, W. Wang and G. Rigoll. Performance Evaluation of a New Hybrid Modeling Technique for Handwriting Recognition Using Identical On-Line and Off-Line Data. Proceedings Document Analysis and Recognition. pp: 446 - 449
[23] Raid Saabni and Jihad El-Sana. Hierarchical On-line Arabic Handwriting Recognition. Proceedings 10th International Conference on Document Analysis and Recognition 2009.Barcelona, Spain.-July 26-July 29, pp 867-871
[24] Aparna, K.H., Subramanian, V., Kasirajan, M., Prakash, G. V., Chakravarthy, V.S., and Madhvanath, S., 2004. Online handwriting recognition for Tamil. Ninth International Workshop on Frontiers in Handwriting Recognition, pp. 438-443.
[25] Wei, W. and Guanglai, G., 2009. Online handwriting Mongolia words recognition with Recurrent Neural Networks, Fourth International Conference on Computer Sciences and Convergence Information Technology, pp. 165-167

Fareeha Anwar is pursuing her PhD in Computer Science. She has received her MS in Computer Science in from International Islamic University Islamabad, Pakistan. Before she did he BS in Computer Science from the same University. She is working as Lecturer in Department of Computer Science & Software Engineering since 2007 at International Islamic University Islamabad. Her research interest includes signal processing, Pattern Recognition, Image Processing and AI.

Muhammad AdnanAftab has graduated from International Islamic University Islamabad, Pakistan. He did his final year project to build a Software Engineering Ontology Analyzer. Currently he is a Software Engine et Sky United Kingdom. He research interest includes AI, Image Processing, Pattern Recognition, Augmented Reality and Open Source Software.

Syed Afaq Hussain is working as Associate Professor at King Faisal University, Al-Hassa, Saudi Arabia. He has received his PhD from Kyoto University Japan. His research areas include Image processing, Computer vision, Pattern recognition, Decision support systems and Digital watermarking. He is member of IEEE, ACM, and PEC.

Ayyaz Hussain has received his PhD from National University of Computer and Emerging Sciences (NU-FAST) Islamabad Pakistan.
He is working as Associate Professor at International Islamic University Islamabad Pakistan.
His research interest include Image Processing, Pattern Recognition, Machine Learning and Data Mining and Computational Intelligence Techniques.