A Study on Idiosyncratic Handwriting with Impact on Writer Identification

Chandranath Adak*†, Bidyut B. Chaudhuri‡, Michael Blumenstein*†
*Centre for AI, School of Software, University of Technology Sydney, Australia-2007
†IIIS, School of ICT, Griffith University, Gold Coast, Australia-4222
‡CVPR Unit, Indian Statistical Institute, Kolkata, India-700108
Chandranath.Adak@student.uts.edu.au

Abstract—In this paper, we study handwriting idiosyncrasy in terms of its structural eccentricity. In this study, our approach is to find idiosyncratic handwritten text components and model the idiosyncrasy analysis task as a machine learning problem supervised by human cognition. We employ the Inception network for this purpose. The experiments are performed on two publicly available databases and an in-house database of Bengali offline handwritten samples. On these samples, subjective opinion scores of handwriting idiosyncrasy are collected from handwriting experts. We have analyzed the handwriting idiosyncrasy on this corpus which comprises the perceptive ground-truth opinion. We also investigate the effect of idiosyncratic text on writer identification by using the SqueezeNet. The performance of our system is promising.

Index Terms—Handwriting idiosyncrasy, Inception network, SqueezeNet, Writer identification.

I. INTRODUCTION

Handwriting is a nexus of graphical marks/symbols where a human writer can be perceived as a "stochastic pattern generator" of such symbols [1]. The structural pattern of such mark/symbol varies with individual due to personal writing style. On an individual handwritten sample, it can be noted that some marks/symbols are written in an unconventional way and referred to as "handwriting idiosyncrasy"[1]. According to the Oxford dictionary[2], the word "idiosyncrasy" has been originated from the Greek "*idiosunkrasia*", i.e. *idios* (own, private) + *sun* (with) + *krasis* (mixture), which suggests "a distinctive or peculiar feature or characteristic" of an individual.

In this paper, the *idiosyncratic handwriting* interprets individual eccentricities in penning a character. In Fig. 1.(a), a typical idiosyncratic style of writing English character 't' is shown within a red box. Here, the horizontal stroke of 't' is scribbled with a continuity of the vertical stroke. Hence, this 't' apparently looks like cursive lowercase 'f' ('*f*') and depicts individual idiosyncrasy. Also, Fig. 1.(b) presents an example of Bengali idiosyncratic handwriting, where the red-boxed character 'প' is drawn by noodle-like continuous stroke and creating a queer pattern. Although in this paper, we mainly consider a complex Indic script Bengali (endonym, *Bangla*) [2], [3], an example in English is presented in Fig. 1.(a) for mass readership.
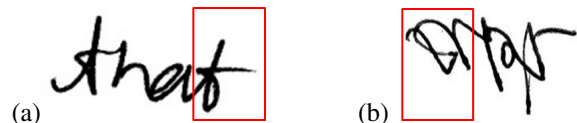

Fig. 1. Examples of highly idiosyncratic handwritten character in (a) English and (b) Bengali script, enclosed in red colored boxes.

In Fig. 2, we present an example of a Bengali text line portion where two idiosyncratic characters ('র' and 'ই') occur multiple times. Here, a bigger unusual *circular loop* (O) instead of the lower dot component of character 'র' and a black ribbon ('ᚷ') like shape in the upper zone of character 'ই' make these highly idiosyncratic.
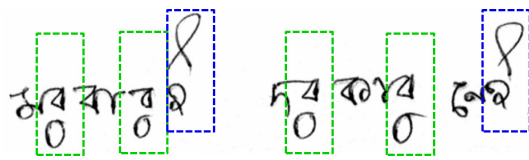

Fig. 2. Two idiosyncratic characters marked by green and blue dotted boxes.

In Fig. 3, we show some examples of idiosyncratic text-patches from the Bengali handwritten old manuscript of Nobel Laureate *R.N. Tagore*. Fig. 3.(a) depicts the character 'প' written in a single stroke without lifting the pen, Fig. 3.(b) presents the character 'চ' looks more likely Bengali numeral '৩' and Fig. 3.(c) shows character 'ন' appears similar to the character 'ম'.
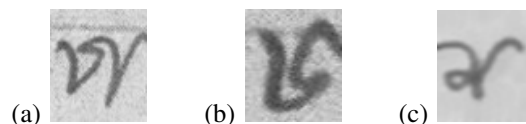

Fig. 3. Highly idiosyncratic text-patches from *R.N. Tagore*'s manuscript.

In the *Educational research* domain, very few studies have been performed on idiosyncratic writing [4]. On the other hand, in the research field of *Document Image Analysis*, some works have mentioned handwriting idiosyncrasy with a motivation of data synthesis [5], [6]. We have not found any other references relevant to our work.

With the assistance of some handwriting experts, we have noted some possible influences of idiosyncratic handwriting generation as follows:

i) *Stroke end/start position:* The current character shape may depend on the stroke ending position of the previous character and stroke starting positing of the next character.

ii) *Positional occurrence:* A character shape sometimes becomes eccentric due to its start/middle/end position in a word.

iii) *Character stroke component:* A single stroke is considered as the scribbling between a pen-down to an immediately next pen-up. Multiple pen-down/ups introduce several character stroke components. Sometimes, to maintain a continuous writing stroke, writer reduces the number of pen-down/ups and may invite idiosyncrasy.

iv) *Cursiveness:* The cursive Bengali writing influences mostly to the individual writing styles and eccentricities due to lack of proper handwriting teaching/learning curriculum in school [7]. The students learn cursive writing by their own adaptive skills and sometimes by mimicking the handwritings of some family members, teachers, friends, and scholars. Therefore, sometimes family/cultural background becomes influential.

v) *Illness:* Some illness such as Parkinson's disease, Alzheimer, Dyslexia, Tourette syndrome etc. may be the cause of idiosyncratic handwriting.

This paper is an early attempt of computer-based analysis of idiosyncratic handwriting. Here, we formulate our idiosyncrasy analysis task as a feature-based classification problem and use an auto-derived feature-based *Inception network* [8]. We also take assistance from several human handwriting experts to provide some cognitive opinion score on handwritten text-patches as ground-truth.

The study of idiosyncratic handwriting can be used for:

i) invoking the artificial intelligence into the machine to understand the idiosyncrasy in handwriting,

ii) sorting the handwritten samples on the basis of idiosyncrasy, which can be utilized in the OCR (*Optical Character Recognition*) preprocessing stage,

iii) scrutinizing the forensic and biometric aspects of handwriting,

iv) investigating distinct penning styles of some famous writers and scholars for the cultural heritage value,

v) examining early/advanced stage of any physical or neurological disorder which may affect writing styles.

Besides handwriting idiosyncrasy analysis, we further conduct an experiment to see the impact of idiosyncratic handwritten text on writer identification. We note that human handwriting experts emphasize some particular characters/text-patches for unknown writer inspection [9]. Here, we try to adopt this technique to investigate on highly idiosyncratic text-patches. The writer identification task is perceived as a multiclass classification problem [10] and the auto-derived feature-based SqueezeNet [11] is used here to tackle this problem.

The rest of the paper is structured as follows. The proposed method is described in *Section* II. Then, *Section* III provides the experimental results followed by discussions. Finally, the conclusion is drawn in *Section* IV.

II. PROPOSED METHOD

A handwritten document is scanned to obtain a digital image and then it undergoes several preprocessing stages. Very small noisy components are removed by a single pass connected component labeling algorithm [12]. Non-textual components such as drawings/doodles are removed using the technique of [13]. Crossed-out/struck-out text, if present in the document image, is also removed by the method of [14]. The characters are segmented using the water reservoir principle-based scheme of [15]. For our task, instead of normalizing the segmented character, we use a fixed size text-patch. The text-patch is selected empirically as a $116{\times}116$ neighbor window around the center of gravity of a segmented character. For our task, it is not mandatory that a text-patch should always contain a single properly segmented character. We feed these text-patches for idiosyncrasy analysis followed by the study on writer investigation.

A. Idiosyncrasy Analysis

In this subsection, at first we discuss our perception of idiosyncrasy analysis task as a machine learning problem, then we describe our approach to tackle this problem.

1) Problem formulation: The idiosyncratic variation of handwriting is huge in quantity. The sense of idiosyncrasy in writing patterns can be well-examined by human handwriting experts. Here, we attempt to impart the machine with the experts' cognitive experiences instead of some statistical hand-crafted features/characteristics.

The individual text-patches are shown to the human experts to study the idiosyncrasies. The experts are then requested to provide a subjective idiosyncratic score within a continuous range of $[I_1, I_2]; I_1, I_2 \in \mathbb{R}$. The subjective scores are marked in a continuous range to avoid the restriction of scoring on a fixed discrete opinion scale, e.g. *Likert scale* [16]. For each
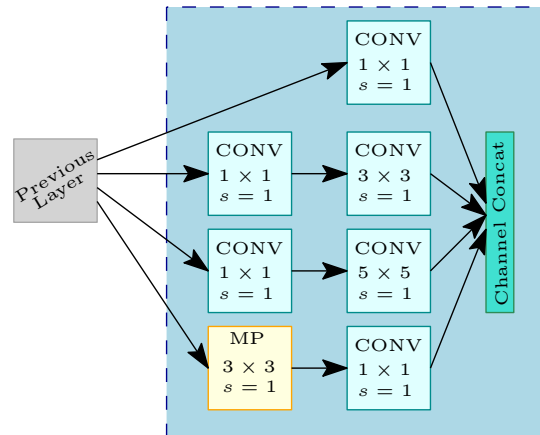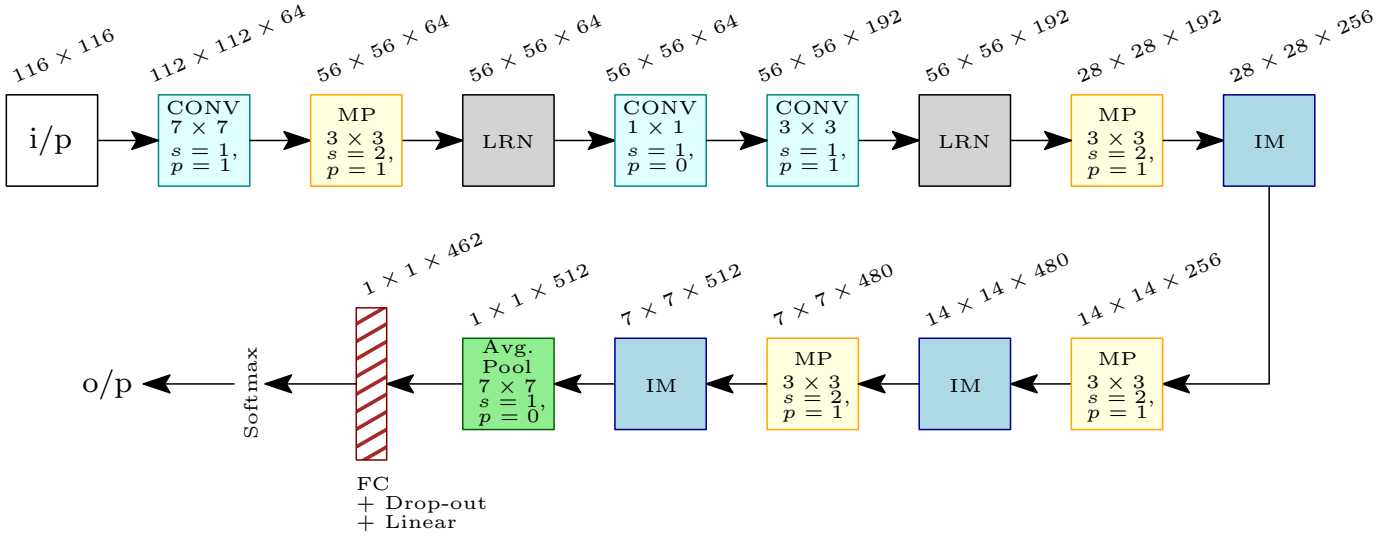

Fig. 4. Inception Module (IM).

Fig. 5. Inception Network.

text-patch, multiple individuals provide their opinion scores. The arithmetic mean of all these scores is calculated and labeled with each text-patch as its *mean opinion score*.

After the data collection, the idiosyncratic score range $[I_1, I_2]$ is partitioned into n_I number of bins of equal span. Therefore, all the *idiosyncratic mean opinion score*s of entire text-patch set fall into these n_I bins. In other words, all the text-patches are categorized into n_I classes with respect to the idiosyncratic score.

Thus, we map the idiosyncrasy analysis job into a classification problem. Now, the task is to label a given text-patch into n_I classes of idiosyncratic opinion score. Such categorization of opinion scores is a well-known topic in the field of video quality assessment [17].

The idiosyncrasy analysis task is performed by an auto-derived feature-based Inception network, discussed in the following subsection.

2) Idiosyncratic text identification: The idiosyncratic text-patches are marked or classified with respect to the idiosyncratic opinion scores. Some auto-derived features are employed for this purpose and discussed in this subsection.

The standard CNN architecture [18] can be deemed as a knot of two parts: *front* and *rear*. The *front* part is the convolutional part regarded as an automated feature extractor, which takes an image input and produces the feature maps. The *rear* part is typically an MLP (*Multi-Layer Perceptron*) used as a classifier. The auto-extracted features are fed to the MLP for obtaining the classified output through the stacked fully connected layers.

For our idiosyncratic text analysis, we adopt the deep neural architecture of [8] which has reported the concept of *inception module*. In Fig. 4, we present the inception module (IM) used in this paper. Here, the "CONV" represents convolutional layer, "Max Pool" denotes max-pooling layer, s symbolizes the stride value. The employed filter size and stride value are provided in Fig. 4.

We adopt the strategy of inception module and work with a comparatively smaller version of [8] due to lesser class classification and reduced memory complexity. Our adopted version of the *Inception Network* is shown in Fig. 5. This architecture takes a fixed-sized image input (i/p). We feed the text-patch image of size 116×116. In the front part of the Inception network, the initial convolutional layer (CONV) contains 64 feature maps of size 112×112. Each of these feature maps is connected with a 7×7 neighborhood of the i/p image. Here, stride (s) = 1 and padding (p) = 1. The following max-pooling (MP) layer, LRN (*Local Response Normalization*), CONV layers are displayed in Fig. 5 with the corresponding employed filter size, the number of feature maps, map size, stride, padding etc. The actual inner-view of inception module (IM) of Fig. 5 is presented in Fig. 4. In total, here we employ three inception modules. We apply ReLU (*Rectified Linear Unit*) [19] as an activation function for all the convolutions including those inside the inception module.

After the last inception module, the rear part of the inception network contains an average pooling (Avg. Pool) layer which obeys the technique of [20]. The following layer is fully connected (FC) with a dropout of 40% [21]. The linear layer with softmax activation is used for predicting the class output (o/p). The output size of each layer is shown in Fig. 5.

B. Writer Identification

Writer identification is a task of allocating a writer-id on a questioned handwritten sample. In other words, writer identification is basically a multiclass classification problem to detect a writer-class among multiple classes [10]. Here also, we use an auto-derived feature-based classification strategy. For writer identification, we employ the SqueezeNet [11] architecture, since this architecture provides good accuracy for general image processing tasks with lesser parameter compared to the AlexNet [22].
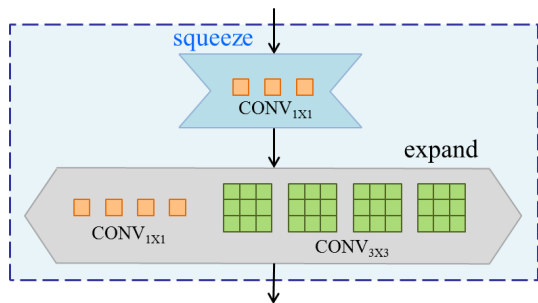
Fig. 6. Fire module.

In SqueezeNet, the key innovation is the *fire module* ($Fire$) [11]. A fire module comprises two layers: *squeeze* and *expand*, as shown in Fig. 6. The squeeze layer contains $Ns_{1\times1}$ number of 1×1 convolutional filters ($CONV_{1\times1}$), while the expand layer contains $Ne_{1\times1}$ number of 1×1 convolutional filters and $Ne_{3\times3}$ number of 3×3 convolutional filters ($CONV_{3\times3}$). Here, Fig. 6 presents $Ns_{1\times1} = 3$, $Ne_{1\times1} = 4$, and $Ne_{3\times3} = 4$.

For our task, the SqueezeNet takes a text-patch of size 116×116. The first convolutional layer (CONV) comprises of 96 feature maps of size 112×112. Every feature map is connected with a 7×7 neighborhood of the input (i/p) image patch. Here, stride (s) = 1 and padding (p) = 1. The following max-pooling (MP), CONV layers are shown in Fig. 7 with the corresponding employed filter size, the number of feature maps, map size, stride and padding measure. Here, we use eight fire modules ($Fire_{i|i=1,2,\ldots,8}$) in total. The counts of ($Ns_{1\times1}$, $Ne_{1\times1}$, $Ne_{3\times3}$) in $Fire_1$, $Fire_2$, ..., $Fire_8$ are (16, 64, 64), (16, 64, 64), (32, 128, 128), (32, 128, 128), (48, 192, 192), (48, 192, 192), (64, 256, 256), (64, 256, 256), respectively. The average pooling (Avg. Pool) layer, after the last convolutional layer, follows the technique of [20]. At last, softmax is used to predict the output class (o/p). We use ReLU (*Rectified Linear Unit*) [19] as activation function at the end of both squeeze and expand layers of all fire modules. After the eighth fire module ($Fire_8$), a dropout [21] of 50% ratio is applied.

III. EXPERIMENTS AND DISCUSSION

In this section, we discuss our employed database and analyze the efficacy of our system through experiments.

A. Database Employed

For the experimentation, we required a database having a subjective opinion score of handwriting idiosyncrasy. We did not find any database with proper ground-truth. Therefore, we generated our own corpus containing subjective opinion ground-truth score of idiosyncrasy. For this, we gathered the Bengali writings, on which we requested handwriting experts to put some subjective score while examining the idiosyncrasy, as discussed in *Section* II-A1.

For the idiosyncratic score, we selected the range [0, 10], i.e. $I_1=0$, $I_2=10$ (refer to *Section* II-A1). This range was

partitioned into 10 bins, i.e. n_I =10. Therefore, idiosyncrasy analysis tasks can be perceived as 10-class classification problems. The idea behind such fixing the class number was adopted from [17]. The class-1 (ID$_1$) was represented by the highest score, i.e. most likely to be an idiosyncratic writing, and class-10 (ID$_{10}$) by the lowest score, i.e. least idiosyncratic handwriting.

To obtain the idiosyncratic opinion score, an expert was given cropped Bengali handwritten text-patches irrespective of character class. The expert put the cognitive score by his/her own expertise. For a text-patch, at least 30 human handwriting experts provided the subjective idiosyncratic opinion scores, and the arithmetic mean opinion score was selected as the golden standard for each text-patch.

For subjective opinion score collection, we employed two publicly available databases and one in-house database. The database details are as follows.

i) *ICDAR-2013 segmentation* [23] (say, D$_{DAR}$): This database comprises 50 Bengali handwritten pages, made available during ICDAR-2013 handwriting segmentation contest [23].

ii) *CMATERdb1.1.1* [24] (say, D$_{CM}$): This is a subpart of the public database CMATERdb1 [24] and contains 100 Bengali handwritten pages.

iii) *In-house database* (say, D$_{new}$): This is an offline database of Bengali handwriting written by 100 writers. Each writer wrote 3 pages to contribute a total of 300 pages.

Here, for D$_{new}$ generation, a typical 0.5 mm ball-point smooth black inked pen of a particular brand/model was used to write on a 75 GSM (g/m^2) blank white page placed on a common good writing surface. The writers were of both genders in the age group of 9 to 67 years, having various academic backgrounds from primary school to university level. The writers were general healthy human-being without having any known serious illness (e.g., Dyslexia, Parkinson's disease etc.) of obstructing general writing. All the handwritings were performed in the healthy awaken state of the writers.

B. Results and Evaluation

In this subsection, we discuss the experimental results and our system performances.

1) Idiosyncrasy analysis performance: Here the idiosyncratic analysis task was formulated as a 10-class classification problem where the job was to classify the handwritten patches into idiosyncratic classes (ID$_1$, ID$_2$, ..., ID$_{10}$).

We conducted our experiment of idiosyncratic handwriting analysis separately on the previously mentioned three databases D$_{DAR}$, D$_{CM}$, and D$_{new}$. Each of these databases with subjective idiosyncratic opinion score was divided into training, validation, and test set in 2:1:1 ratio.

The performance of our system for idiosyncratic handwritten text-patch identification is presented in TABLE I. Here we show the overall average *F-Measure* (%) of all the classes. Since the idiosyncratic text-patch ground-truths are dependent on experts' cognitive analysis, all classes may not be well balanced. Therefore, our database is quite imbalanced.
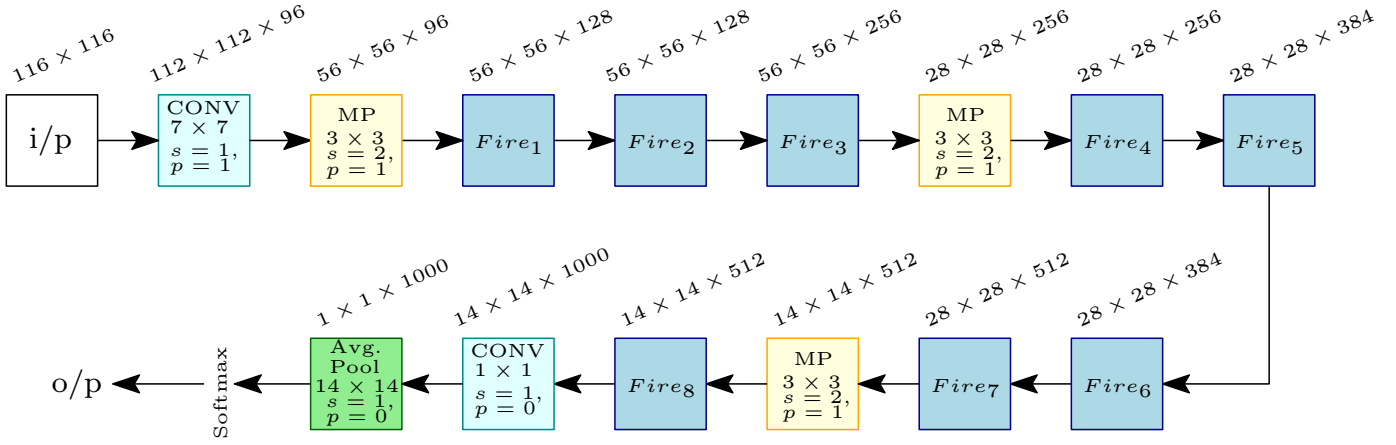
Fig. 7. SqueezeNet architecture.

For classification into such classes, the *balanced accuracy* provides better performance measure [25]. In TABLE I, we also show the performance of idiosyncrasy analysis in terms of balanced accuracy (%).

TABLE I
IDIOSYNCRATIC HANDWRITING ANALYSIS

Database	F-Measure (%)	Balanced Accuracy (%)
D_{DAR}	**91.37**	**93.54**
D_{CM}	89.48	91.33
D_{new}	85.96	88.16
$D_{DAR} + D_{CM} + D_{new}$	89.03	90.89

TABLE II
WRITER IDENTIFICATION PERFORMANCE

Idiosyncratic class (ID)	Accuracy (%)		
	Top-1	*Top-2*	*Top-5*
ID_1	93.23	93.72	96.57
ID_2	92.85	93.46	95.81
ID_3	91.66	92.18	95.10
ID_4	89.94	90.43	93.76
ID_5	87.38	87.89	91.27
$ID_{1,2}$	94.03	94.76	97.54
$ID_{1,3}$	93.62	93.90	96.79
$ID_{2,3}$	92.98	93.54	95.42
ID_{1-3}	**94.46**	**94.92**	**98.27**
ID_{1-4}	93.71	94.48	97.68
ID_{1-5}	93.28	94.15	96.92
ID_{1-10}	90.43	91.59	94.26

On idiosyncrasy analysis, our system obtained the highest performance on database D_{DAR} and lowest performance on D_{new}. For idiosyncratic handwritten text-patch identification, we obtained 91.37%, 89.48%, 85.96% of F-measure and 93.54%, 91.33%, 88.16% of balanced accuracy on D_{DAR}, D_{CM}, D_{new} databases, respectively. Overall, union of all our employed datasets attained system performance of 89.03% F-measure and 90.89% balanced accuracy.

2) Writer identification performance: In this subsection, we present the experimentations performed for writer identification task to investigate whether there exists any influence of idiosyncratic handwriting on this task.

The databases D_{DAR} and D_{CM} were not suitable for writer identification task, since writer information was missing. The D_{new} contained the writer information and was suitable for our study. To inspect the effect of idiosyncratic handwriting on writer identification, all the obtained text-patches (refer to *Section* II) of D_{new} were subdivided into training, validation, and test set in 2:1:1 ratio.

As previously mentioned in *Section* III-A and *Section* III-B1, the handwritten text-patches were classified into 10 idiosyncratic classes (ID_1, ID_2, ..., ID_{10}). The handwritten samples of class ID_1 were highly idiosyncratic whereas ID_{10} contained least idiosyncratic handwritings. We intended to analyze the writer identification performance using only the highest idiosyncratic handwriting (class ID_1) of our database D_{new}. Besides, we performed the writer identification experiments on different idiosyncratic classes. We also experimented with all possible unary, binary, ternary, quaternary, ..., n-ary combinations of idiosyncratic classes ($ID_{n|n = 1, 2, ..., 10}$). Here, in TABLE II, we present the major results of our writer identification method on some important idiosyncratic classes.

For writer identification performance analysis, we employed the *Top-N* criterion where a questioned document's writer belonged to a reduced set of 'N' ($\ll$ total number of writers). Here, we selected *Top-1*, *Top-2* and *Top-5* criteria for providing the writer identification performance in terms of *accuracy* (%). In D_{new} database, the samples of multiple writer classes were quite balanced.

In TABLE II, $ID_{1,2}$ denotes union of text-patches of the idiosyncratic class-1 (ID_1) and class-2 (ID_2), i.e., $ID_{1,2} \equiv ID_1 \cup ID_2$. Likewise, $ID_{1,3} \equiv ID_1 \cup ID_3$, $ID_{1-3} \equiv ID_1 \cup ID_2 \cup ID_3$, $ID_{1,2,4} \equiv ID_1 \cup ID_2 \cup ID_4$ etc., and ID_{1-10} denotes the union of text-patches of all idiosyncratic classes. As a matter of fact, experimenting on ID_{1-10} is synonymous to the traditional writer identification experiment on the full page of writing.

With reference to TABLE II, we obtained 93.23% *Top-1* writer identification accuracy while employing only the

highest idiosyncratic handwriting (class ID_1). On this ID_1, the attained *Top-2* and *Top-5* accuracies were 93.72% and 96.57%, respectively.

Among binary combinations (e.g. $ID_{1,2}$, $ID_{1,3}$, $ID_{2,3}$ etc.) of the idiosyncratic classes, highest *Top-1* accuracy (94.03%) was obtained on $ID_{1,2}$. Among ternary combinations (e.g. ID_{1-3}, ID_{2-4}, $ID_{1,2,4}$ etc.), highest 94.46% of *Top-1* accuracy was obtained on ID_{1-3}. It was the best *Top-1* accuracy among all possible n-ary combinations of idiosyncratic classes.

Union of all the idiosyncratic classes (ID_{1-10}) produced 90.43% of *Top-1* accuracy. In other words, classical experimentation on all the text-patches attained 90.43% *Top-1* writer identification accuracy. However, each of the upper three idiosyncratic classes (ID_1, ID_2, and ID_3) performed better than the traditional approach of considering all text components. Thus, from our experimental results, it can be noted that highly idiosyncratic handwriting can improve the writer identification performance. Here, experimenting on ID_1 and ID_{1-3}, we obtained respective 2.8% and 4.03% additional *Top-1* writer identification accuracy when compared to ID_{1-10}. This baseline performance added value to our idiosyncratic analysis.

C. Comparison

To the best of our knowledge, this work is the earliest attempt of its kind on idiosyncratic handwriting analysis. We did not find any related work for making a comparison.

However, it was interesting to see that highly idiosyncratic handwritings produced up to 4.03% additional *Top-1* writer identification accuracy on our experimentations when compared to the same writer identification method without using idiosyncratic analysis (refer to *Section* III-B2).

IV. CONCLUSION

In this paper, we scrutinize the idiosyncrasy of individual handwriting and show its facilitative nature for the writer identification task. The idiosyncratic handwriting analysis task is modeled as a machine learning problem with the aid of cognitive score of handwriting experts. Here, an Inception network is employed for idiosyncratic text inspection and the SqueezeNet is used for writer identification. For both of these tasks, auto-derived deep features are used. Experimentation of idiosyncratic handwriting analysis is performed on two publicly available and one in-house offline Bengali handwritten databases with collected subjective opinion ground-truth. Overall 90.89% balanced accuracy has been obtained for the idiosyncrasy analysis task on these three datasets. The writer identification job has experimented on the in-house database. For unary idiosyncratic classes, we have obtained highest 93.23% *Top-1* writer identification accuracy (on ID_1). The best *Top-1* accuracy of 94.46% has been obtained while combining three highly unary idiosyncratic classes (ID_1, ID_2, and ID_3). Our experiments suggest that prior analyzing of idiosyncratic handwriting can improve the writer identification performance up to 4.03% (*Top-1* accuracy).

Although in this paper, we have worked on Bengali handwriting, our method can be extended to some other scripts. In future, we will try to exploit the idiosyncrasy analysis for multilingual handwriting. We also plan to hybridize different neural networks for a comparative study.

REFERENCES

[1] M. Bulacu, L. Schomaker, "Text-Independent Writer Identification and Verification Using Textural and Allographic Features", IEEE Trans. on PAMI, vol. 29, no. 4, pp. 701-717, 2007.
[2] R.D. Banerji, "The Origin of the Bengali Script", University of Calcutta, 1919.
[3] B.B. Chaudhuri, U. Pal, "A Complete Printed Bangla OCR System", Pattern Recognition, vol. 31, no. 5, pp. 531-549, 1998.
[4] J.J. Schneider, "Contexts, Genres, and Imagination: An Examination of the Idiosyncratic Writing Performances of Three Elementary Children within Multiple Contexts of Writing Instruction", Research in the Teaching of English, vol. 37, no. 3, pp. 329-379, 2003.
[5] T.S.F. Haines, O.M. Aodha, G.J. Brostow, "My Text in Your Handwriting", ACM Trans. on Graphics, vol. 35, no. 3, artl. 26, 18 pages, 2016.
[6] J. Chen, W. Cheng, D.P. Lopresti, "Using Perturbed Handwriting to Support Writer Identification in the Presence of Severe Data Constraints", Proc. DRR XVIII, vol. 7874, 78740G, 2011.
[7] C. Adak, B.B. Chaudhuri, M. Blumenstein, "Offline Cursive Bengali Word Recognition using CNNs with a Recurrent Model", Proc. ICFHR, pp. 429-434, 2016.
[8] C. Szegedy et al., "Going Deeper with Convolutions", Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 1-9, 2015.
[9] R.A. Huber, A.M. Headrick, "Handwriting Identification: Facts and Fundamentals", CRC Press, 1999.
[10] U. Porwal et al., "Multiclass Learning for Writer Identification Using Error-Correcting Codes", Proc. DAS, pp. 16-20, 2014.
[11] F.N. Iandola et al., "SqueezeNet: AlexNet-level Accuracy with 50x Fewer Parameters and <0.5MB Model Size", CoRR abs/1602.07360.
[12] F. Zhao, H.Z. Lu, Z.Y. Zhang, "Real-time Single-pass Connected Components Analysis Algorithm", EURASIP Journal on Image and Video Processing, vol. 21, pp. 1-10, 2013.
[13] C. Adak, B.B. Chaudhuri, "Extraction of Doodles and Drawings from Manuscripts", Proc. Int. Conf. on Pattern Recognition and Machine Intelligence (PReMI), LNCS #8251, pp. 515-520, 2013.
[14] B.B. Chaudhuri, C. Adak, "An Approach for Detecting and Cleaning of Struck-out Handwritten Text", Pattern Recognition, vol. 61, pp. 282-294, 2017.
[15] U. Pal, S. Datta, "Segmentation of Bangla Unconstrained Handwritten Text", Proc. ICDAR, pp. 1128-1132, 2003.
[16] R. Likert, "A Technique for the Measurement of Attitudes", Archives of Psychology, New York, no. 140, pp. 1-55, 1932.
[17] Q.H.-Thu, M.N. Garcia, F. Speranza, P. Corriveau, A. Raake, "Study of Rating Scales for Subjective Quality Assessment of High-Definition Video", IEEE Trans. on Broadcasting, vol. 57, no. 1, pp. 1-14, 2011.
[18] Y. LeCun et al., "Gradient-based Learning Applied to Document Recognition", Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, 1998.
[19] V. Nair, G.E. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines", Proc. ICML, pp. 807-814, 2010.
[20] M. Lin et al., "Network in Network", CoRR, abs/1312.4400, 2013.
[21] N. Srivastava, G.E. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting", JMLR, vol. 15, no. 1, pp. 1929-1958, 2014.
[22] A. Krizhevsky et al., "ImageNet Classification with Deep Convolutional Neural Networks", Proc. NIPS, vol. 1, pp. 1097-1105, 2012.
[23] N. Stamatopoulos et al., "ICDAR 2013 Handwriting Segmentation Contest", Proc. ICDAR, pp. 1402-1406, 2013.
[24] R. Sarkar et al., "CMATERdb1: A Database of Unconstrained Handwritten Bangla and Bangla-English Mixed Script Document Image", IJDAR, vol. 15, no. 1, pp.71-83, 2012.
[25] K.H. Brodersen et al., "The Balanced Accuracy and Its Posterior Distribution", Proc. ICPR, pp. 3121-3124, 2010.