2020 11th International Conference on Electrical and Computer Engineering (ICECE) | 978-1-6654-2254-3/20/$31.00 ©2020 IEEE | DOI: 10.1109/ICECE51571.2020.9393079

Handwritten Numerals Recognition by Employing a Transfer Learned Deep Convolution Neural Network for Diverse Literature

Md. Mehedi Hasan*, Azmain Yakin Srizon†, Abu Sayeed‡ and Md. Al Mehedi Hasan§
Department of Computer Science & Engineering
Rajshahi University of Engineering & Technology, Rajshahi, Bangladesh
Email: *mmehedihasann@gmail.com, †azmainsrizon@gmail.com, ‡abusayeed.cse@gmail.com, §mehedi_ru@yahoo.com

Abstract—**Having around 6,500 languages worldwide, handwritten numerals recognition has been a domain of research for decades now as numerals are a common phenomenon among all these diverse languages. Previously, researchers have contributed significantly to recognize the handwritten numerals of diverse literature. Different approaches have been discovered to be feasible for language-specific numerals recognition. However, finding a common architecture to recognize numerals have been a goal from the very beginning. But despite having many efforts, discovering a common architecture for high recognition of numerals of diverse literature has always been a challenging task to solve and not many contributions have been made in this regard. Therefore, in this research, we focused on seven benchmark datasets of six languages and proposed a modified DenseNet-201 architecture. Our proposed architecture achieved an overall accuracy of 99.04%, 99.33%, 98.83%, 99.50%, 99.83%, 99.54%, and 99.74% for Bengali (CMATERdb 3.1.1), Devanagari (CMATERdb 3.2.1), Arabic (CMATERdb 3.3.1), Telugu (CMATERdb 3.4.1), Nepali, ARDIS II, and ARDIS III datasets respectively which outperformed all notable previous works by a noteworthy margin.**

Index Terms—**Numerals Recognition, Pattern Recognition, Transfer Learning, Deep Convolutional Neural Network, Modified DenseNet-201 Architecture, CMATERdb, Nepali Numerals, ARDIS**

I. Introduction

Numerals recognition has grown as a benchmark dilemma through the last decade as numerals are the only common entities among different kinds of literature or languages worldwide. Numerals recognition is basically a part of the pattern recognition problem but the main concept of numeral recognition lies in designing an OCR-based system to convert a numeric image into a machine editable character [1]. Moreover, numerals recognition is also utilized for recognizing the numbers on the license plate, postal codes, numbers on the bank checks, and other form processing [2], [3]. Although the primary goal is to recognize numerals from numeric images, the work can be expanded to handwritten numerals recognition as various writing fashions of numerous individuals can assist in creating a model that can eventually both recognize handwritten and numeric digits in general. Designing architecture to discover a solution to this dilemma has been an area of interest for decades now. Previously, many studies have been conducted on different numerals dataset and noteworthy works among them have been described in brief in the literature review section.

In this research, we worked with seven popular and publicly available numeral datasets of different languages. Considered language in this research is Bengali, Devanagari, Arabic, Telugu, Nepali, and two forms of Latin. More about the datasets can be discovered in the data description section. We started with some preprocessing steps and applied a modified DenseNet-201 architecture on each of these seven datasets. The experimental analysis showed that our approach gained better performance for all the datasets under consideration. Lastly, we compared our results with the previous works and concluded that for each of the datasets our approach achieved better overall accuracy which outperformed all previous researches.

II. Literature Review

Previously, various researchers have contributed significantly to the recognition of numerals. In this section, we have mentioned some recent and noteworthy works. Regional weighted run-length features were utilized for offline recognition of numerals in one research work in 2016 [4]. Another research at the same time proposed a study of moment-based features on handwritten digit recognition [5]. In 2017, Mojette transform was applied for recognition of Indic script numerals [6]. Another research during a similar timeframe suggested a grid-based Hausdroff distance approach for Devanagari numerals recognition [7]. In 2019, a holistic approach was suggested for numerals recognition as well [8]. However, in 2020, the most recent work on numerals recognition have been published [9] where the authors have not only utilized all the approaches described above but also considered some previously discovered efficient methods i.e., Gabor Filter-based approach [10], HOG [11], GLCM [12], GLRLM [13] and RuLBP [14]. They also proposed PLSS, HOPP, and HOPP+PLSS approach to achieve better performance [9].

III. Materials and Methods

A. Dataset Description

In this research, we considered seven benchmark datasets. Handwritten digits or numerals dataset for Bangla (CMATERdb 3.1.1), Devanagari (CMATERdb 3.2.1), Arabic (CMA-

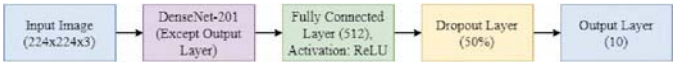

Fig. 1: Proposed modified DenseNet-201 architecture

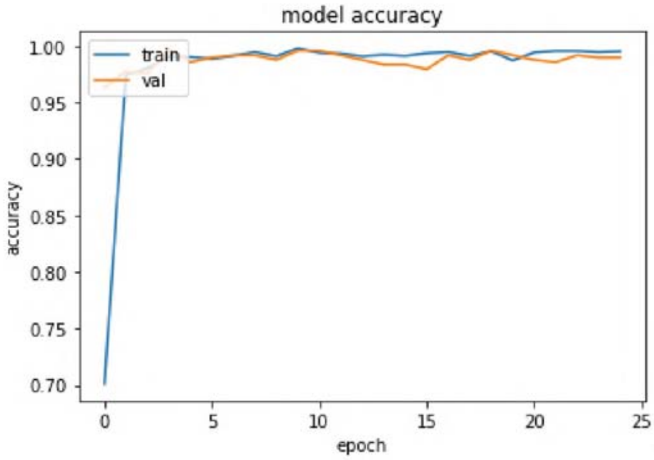

Fig. 2: Training accuracy of our proposed model for Bengali Numerals (CMATERdb 3.1.1) dataset

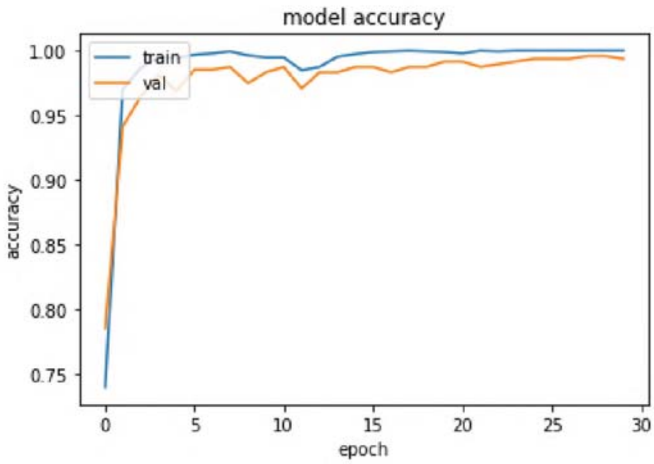

Fig. 3: Training loss of our proposed model for Devanagari Numerals (CMATERdb 3.2.1) dataset

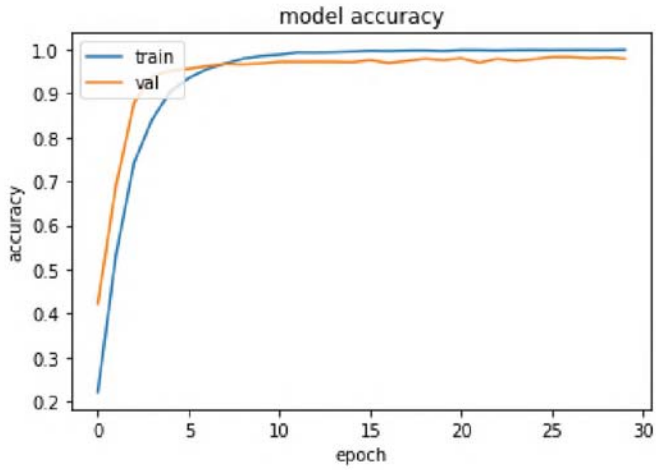

Fig. 4: Training accuracy of our proposed model for Arabic Numerals (CMATERdb 3.3.1) dataset

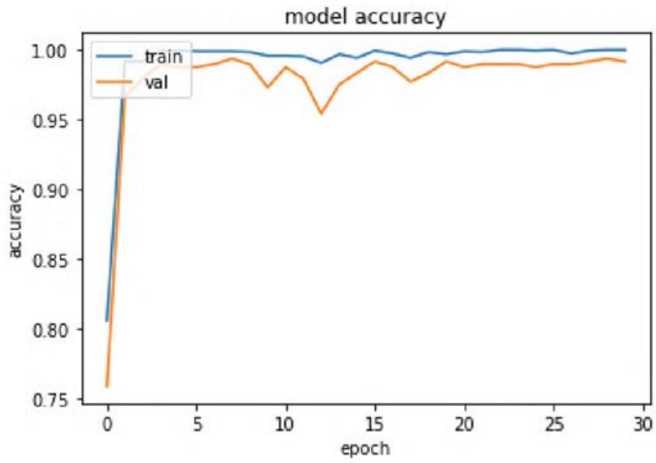

Fig. 5: Training accuracy of our proposed model for Telugu Numerals (CMATERdb 3.4.1) dataset

TERdb 3.3.1), and Telugu (CMATERdb 3.4.1) were collected from the publicly available and popular CMATERdb benchmark dataset [15]. Nepali handwritten numerals dataset was collected from a Kaggle repository which was introduced in a research work [16]. Finally, Latin numerals were collected from the ARDIS dataset [17] which has two types of datasets. ARDIS II dataset has a degraded and noisy background whereas the ARDIS III dataset has a clean background. The number of total, training, validation, and testing samples for all seven datasets have been shown in TABLE-I.

B. Transfer Learning

Transfer learning centers on collecting data obtained while resolving one barrier and performing it to different but comparable difficulty [18]. For occurrence, the information collected while learning to recognize cars could employ while the perception of trucks. This department of study bestows a striking similarity to the enduring history of cerebral investigation on the transfer of knowledge, though established relationships among the two spaces are incompetent. From a pragmatic perspective, transferring or communicating information from earlier performed tasks for the training of current tasks has the capacity to dramatically improve the individual achievement of an agent.

C. Proposed Modified DenseNet-201 Architecture

The general infrastructure of the Convolutional Neural Network (CNN) can be found in [19]. Nonetheless, in a conventional Convolutional Neural Network, all layers are constantly associated which makes the network challenging to spread broader and deeper, as it may grow beyond challenges of either failing or gradient missing requirements.

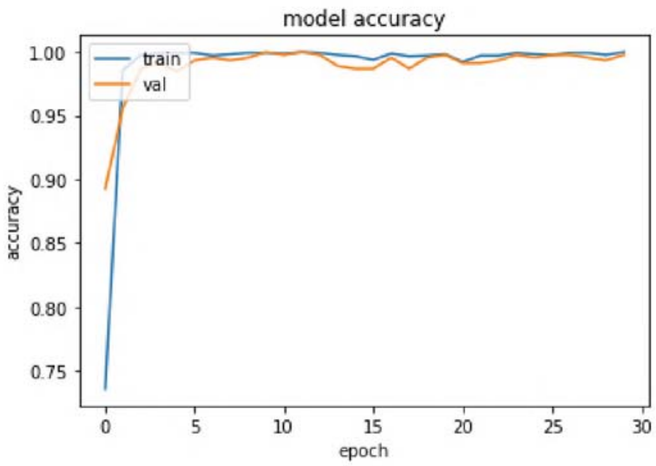

Fig. 6: Training accuracy of our proposed model for Nepali Numerals dataset

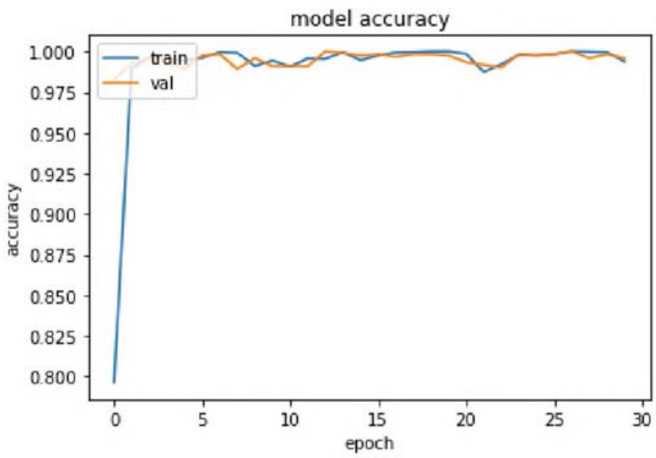

Fig. 7: Training accuracy of our proposed model for ARDIS II Numerals dataset

After that, ResNet introduced an approach to engaging the shortcut attachment by skipping at least two layers. Then, DenseNet additionally extended the model by concatenating all the characteristic designs sequentially instead of summation of the output characteristic graphs from all previous layers. In this research, we've organized a modified DenseNet-201 architecture which is illustrated in Figure-1. After the implementation of DenseNet-201 basic layers, we performed a fully connected layer of size 512 followed by a dropout layer of 50%. Finally, an output layer is associated with having a size of 10. While training by this architecture, no layer was kept frozen. This architecture was performed on all the seven datasets.

IV. Experimental Analysis

A. Preprocessing

Because of providing pictures to a convolutional neural network, a heavy preprocessing of the images was skipped as Convolutional Neural Network is a sophisticated network that can obtain relevant features from raw photographs. However, some preprocessing measures were demanded. The input photographs were in various sizes, hence, they were resized to 224x224x3.

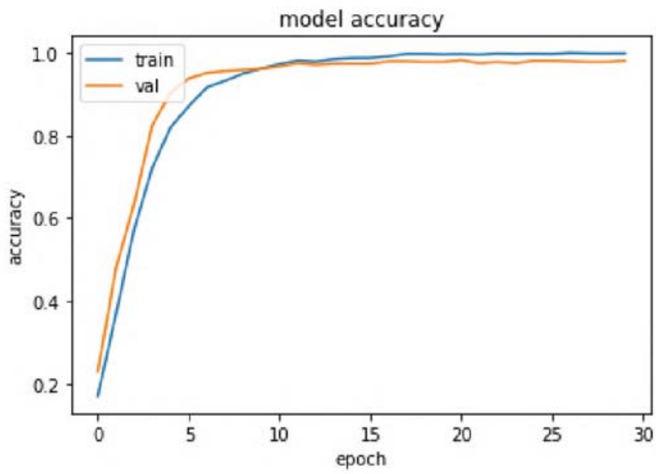

Fig. 8: Training accuracy of our proposed model for ARDIS III Numerals dataset

TABLE I: Number of total, training, validation and test samples for all seven datasets

Dataset	Total	Train	Validation	Test
Bengali [15]	6000	2990	490	2520
Devnagari [15]	3000	1920	480	600
Arabic [15]	3000	1920	480	600
Telugu [15]	3000	1920	480	600
Nepali [16]	2880	1840	460	580
ARDIS II [17]	7600	4860	1220	1520
ARDIS III [17]	7600	4860	1220	1520

B. Experimental Settings

The model was trained for 25 epochs for the Bengali (CMATERdb 3.1.1) dataset and 30 epochs for all other six datasets with a batch size of 24 as after that the validation loss grew approximately constant for the rest of the epochs. Adam optimizer [20] with the learning rate of 0.0001 was exercised to maximize the error function. A categorical cross-entropy function was applied for the loss or error function. For bypassing overfitting, the dropout method was employed.

C. Result Analysis

After resizing the images, the datasets were split into train, validation, and test sets. The train and test set for CMATERdb 3.1.1 or Bengali numerals dataset were provided separately. However, other datasets were split into 80% and 20% in terms of training and testing set. The number of samples in training, validation, and testing sets can be discovered in TABLE-I. After that, we applied the proposed modified DenseNet-201 architecture on the datasets and measured the performance. Figure-2, 3, 4, 5, 6, 7, and 8 illustrate the training

TABLE II: Comparison of performances in terms of overall accurcay among our proposed approach and previous approaches

Method	Dimension	Bengali	Devanagari	Arabic	Telugu	Nepali	ARDIS II	ARDIS III
Singh et al. [4]	140	94.81%	95.83%	96.23%	99.06%	95.25%	90.56%	96.20%
Singh et al. [5]	130	92.83%	92.53%	91.67%	96.67%	95.45%	91.25%	95.94%
Singh et al. [6]	190	93.00%	93.80%	92.25%	94.93%	91.18%	88.48%	93.75%
Bhowmik et al. [7]	300	84.40%	86.50%	90.36%	93.83%	89.00%	86.50%	94.32%
Bhowmik et al. [8]	250	92.45%	90.53%	93.40%	95.56%	93.09%	89.94%	95.86%
Bhowmik et al. [10]	180	94.13%	95.36%	96.30%	98.66%	94.79%	91.64%	93.80%
HOG [11]	324	96.70%	95.60%	94.20%	97.60%	97.50%	92.81%	95.43%
GLCM [12]	76	60.58%	62.10%	73.50%	63.20%	59.60%	49.97%	55.63%
GLRLM [13]	45	76.58%	73.14%	78.23%	78.83%	69.23%	51.09%	57.37%
RuLBP [14]	256	65.36%	68.20%	76.80%	75.90%	19.50%	44.62%	48.25%
PLSS [9]	20	92.56%	90.62%	91.60%	95.75%	92.22%	86.62%	90.34%
HOPP [9]	150	97.86%	96.62%	97.63%	98.53%	97.78%	93.86%	97.16%
PLSS+HOPP [9]	170	98.50%	98.01%	98.40%	99.03%	98.40%	94.27%	97.51%
Proposed	224	**99.04%**	**99.33%**	**98.83%**	**99.50%**	**99.83%**	**99.54%**	**99.74%**

and validation accuracy of the seven datasets while training. Finally, we compared the results with the previous works in TABLE-II and concluded that our approach achieved better performance for all the seven datasets and outperformed all previous approaches in terms of overall accuracy.

V. Conclusion

In this study, we considered seven benchmark datasets of six different languages. After some preprocessing steps, we applied our proposed modified DenseNet-201 architecture on all seven datasets. Experimental analysis and comparison with previous results revealed that our approach achieved better overall accuracy for all seven datasets. In the future, we have plans of working on more numerals datasets of other languages. Moreover, it would be interesting to discover how the model works if separate datasets were utilized for training and testing for a specific language. Furthermore, there are also scopes of research in terms of feature extractions and finding out the causes of misclassification among the datasets.

References

[1] G. Vamvakas, B. Gatos, and S. J. Perantonis, "Handwritten character recognition through two-stage foreground sub-sampling," *Pattern Recognition*, vol. 43, no. 8, pp. 2807–2816, 2010.

[2] S. Basu, N. Das, R. Sarkar, M. Kundu, M. Nasipuri, and D. K. Basu, "Recognition of numeric postal codes from multi-script postal address blocks," in *International Conference on Pattern Recognition and Machine Intelligence*. Springer, 2009, pp. 381–386.

[3] R. Plamondon and S. N. Srihari, "Online and off-line handwriting recognition: a comprehensive survey," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 1, pp. 63–84, 2000.

[4] P. K. Singh, S. Das, R. Sarkar, and M. Nasipuri, "Recognition of offline handwriten devanagari numerals using regional weighted run length features," in *2016 International Conference on Computer, Electrical & Communication Engineering (ICCECE)*. IEEE, 2016, pp. 1–6.

[5] P. K. Singh, R. Sarkar, and M. Nasipuri, "A study of moment based features on handwritten digit recognition," *Applied Computational Intelligence and Soft Computing*, vol. 2016, 2016.

[6] P. K. Singh, S. Das, R. Sarkar, and M. Nasipuri, "Recognition of handwritten indic script numerals using mojette transform," in *Proceedings of the First International Conference on Intelligent Computing and Communication*. Springer, 2017, pp. 459–466.

[7] S. Bhowmik, S. Sen, N. Hori, R. Sarkar, and M. Nasipuri, "Handwritten devanagari numerals recognition using grid based hausdroff distance," in *Computer, Communication and Electrical Technology: Proceedings of the International Conference on Advancement of Computer Communication and Electrical Technology (ACCET 2016), West Bengal, India, 21-22 October 2016*. CRC Press, 2017, p. 15.

[8] S. Bhowmik, S. Malakar, R. Sarkar, S. Basu, M. Kundu, and M. Nasipuri, "Off-line bangla handwritten word recognition: a holistic approach," *Neural Computing and Applications*, vol. 31, no. 10, pp. 5783–5798, 2019.

[9] S. Ghosh, A. Chatterjee, P. K. Singh, S. Bhowmik, and R. Sarkar, "Language-invariant novel feature descriptors for handwritten numeral recognition," *The Visual Computer*, pp. 1–23, 2020.

[10] Y. Hamamoto, S. Uchimura, M. Watanabe, T. Yasuda, Y. Mitani, and S. Tomita, "A gabor filter-based method for recognizing handwritten numerals," *Pattern recognition*, vol. 31, no. 4, pp. 395–400, 1998.

[11] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1. IEEE, 2005, pp. 886–893.

[12] F. Albregtsen *et al.*, "Statistical texture measures computed from gray level coocurrence matrices," *Image processing laboratory, department of informatics, university of oslo*, vol. 5, no. 5, 2008.

[13] F. Albregtsen, "Statistical texture measures computed from gray level run-length matrices," *University of Oslo*, 1995.

[14] S. Ghosh, D. Lahiri, S. Bhowmik, E. Kavallieratou, and R. Sarkar, "Text/non-text separation from handwritten document images using lbp based features: An empirical study," *Journal of Imaging*, vol. 4, no. 4, p. 57, 2018.

[15] N. Das, J. M. Reddy, R. Sarkar, S. Basu, M. Kundu, M. Nasipuri, and D. K. Basu, "A statistical–topological feature combination for recognition of handwritten numerals," *Applied Soft Computing*, vol. 12, no. 8, pp. 2486–2495, 2012.

[16] A. K. Pant, S. P. Panday, and S. R. Joshi, "Off-line nepali handwritten character recognition using multilayer perceptron and radial basis function neural networks," in *2012 Third Asian Himalayas International Conference on Internet*. IEEE, 2012, pp. 1–5.

[17] H. Kusetogullari, A. Yavariabdi, A. Cheddad, H. Grahn, and J. Hall, "Ardis: a swedish historical handwritten digit dataset," *Neural Computing and Applications*, pp. 1–14, 2019.

[18] J. West, D. Ventura, and S. Warnick, "Spring research presentation: A theoretical foundation for inductive transfer," *Brigham Young University, College of Physical and Mathematical Sciences*, vol. 1, no. 08, 2007.

[19] Y.-D. Zhang, C. Pan, X. Chen, and F. Wang, "Abnormal breast identification by nine-layer convolutional neural network with parametric rectified linear unit and rank-based stochastic pooling," *Journal of computational science*, vol. 27, pp. 57–68, 2018.

[20] Z. Zhang, "Improved adam optimizer for deep neural networks," in *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*. IEEE, 2018, pp. 1–2.