

## Improvement in Efficiency of The State-Of-The-Art Handwritten Text Recognition Models

Sai Suryateja S<sup>1</sup>, Veerajay P<sup>1</sup>, Vijay Kumar Naidu P<sup>1</sup>, Ravi Kumar C V<sup>2\*</sup>

<sup>1</sup>4<sup>th</sup> year B Tech, School of Electronics Engineering, Vellore Institute of Technology, Vellore, India

<sup>2</sup>Assistant Professor, School of Electronics Engineering, Vellore Institute of Technology, Vellore, India

**Article History:** Received: 10 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 28 April 2021

### Abstract

In the past few years, the research in the discipline of Handwritten Text Recognition (HTR) has been fast-tracked as many researchers in computer vision are pursuing this discipline. Most of the deep learning models are likely to have vanishing gradient errors when processing paragraph images like scanned images. The most crucial problem with these models is that they have many parameters, which require a large amount of data and resources. So, the most recent offline HTR follows the Convolutional Recurrent Neural Network (CRNNs). Recently developed neural network architecture was used to get better results, namely Gated Convolutional Neural Network (Gated-CNN), which has fewer layers and parameters. The HTR based on Gated-CNN can outperform the CRNN based HTR. This research surpasses the state-of-the-art HTR system on five different handwritten datasets: Bentham, IAM, RIMES, Saint Gall, and Washington. This research needs low computational resources to use in real life, such as smartphones and robots.

**Keywords:** Deep learning; Handwritten Text Recognition; Natural Language Processing; Convolutional Recurrent Neural Networks; Gated Convolutional Neural Network

### 1. Introduction

The discipline of handwritten text recognition (HTR) has a broad range of applications in both the academic and industrial sectors. The HTR converts handwritten text to numeric codes (ASCII or Unicode) either by static or dynamic mode of information [1]. So, images can be considered the information for offline text recognition, which in turn can help to digitalize the manuscripts [2], medical records [3], applications [4], and many more. These applications accentuate the development of HTR for different languages and scripts.

The offline HTR was initially designed as sequence matching, i.e., features extracted from the input images, arranged as a sequence, are emulated with an output sequence, which directs it to a combination of characters. Initially, the most successful approach to solving the issue of HTR was Hidden Markov Model (HMM) [5]. But, the model failed because it could not use context information because Markov assumed that each observation only depends on its current state.

In the past few years, the research in HTR has illustrated rigorous advancements over HMM. Deep learning methods such as Convolutional Recurrent Neural Networks (CRNN) have repeatedly improved and gave practical results in the industrial application [6]. Long Short-Term Memory plays the role of sequence decoder in the CRNN model [7]. To increase the accuracy of HTR, the Multidimensional LSTM (MDLSTM) [8] is used to improve the RNN architecture efficiency by implementing multidimensional data. Bidirectional LSTM (BLSTM) [9] results from the latest studies for the HTR problem because MDLSTM has high complexity and computational cost. The BLSTM offers comparative results with MDLSTM with less computational cost and complexity.

The models using BLSTM, such as CNN-BLSTM, provide excellent results but have obscurities in recollecting extended contexts because of the vanishing gradient problem. Moreover, the current optical models have very high parameters which require a lot of trainable data. It is a considerable problem for real-world applications [10]. To overcome the issue of a large set of parameters, the Gated-CNN-BLSTM method is used to reduce the parameters but would affect the model's performance [11].

To increase the accuracy of the offline HTR systems, we use Gated Convolutional Recurrent Neural Network (Gate-CRNN) architecture which uses a gated mechanism introduced by Dauphin [12]. The

model also has a bidirectional gated recurrent unit (BGRU). The proposed optical model, Gated-CNN-BGRU would require fewer parameters (thousands) to achieve a high accuracy rate.

The proposed model is trained and tested on five well-known datasets, namely, Bentham, IAM, RIMES, Saint Gall, and Washington [15]. The results of the proposed models are then compared with the work of Puigcerver [13], Bluche [14], and Flor [15].

## 2. Materials and Methods

The well-known datasets viz. Bentham, IAM, RIMES, Saint Gall and Washington [15] were used to compare the results of the proposed model with the well-known models viz. Puigcerver [13], Bluche [14] and Flor [15].

### 2.1. Proposed Model

The model proposed in this research paper is derived from the well-known models viz. Puigcerver, Bluche and Flor. The model has used the gated mechanism to reduce the future context.

The proposed model compared to **Puigcerver model**[13] has reduced the trainable parameters and increased the efficiency of the model. The trainable parameters are reduced because gated mechanism introduced by Dauphin is used in the proposed model. Maxpooling is used to overcome the problem of overfitting while in the proposed model we used it at the end of the convolutional block rather than using it after each convolutional layer as in Puigcerver model, because it would decrease the parameters while giving the equivalent results. The He uniform is used as an initializer in the proposed model rather than glorot uniform, which improves the distribution. The parametric rectifier linear unit is used as an activator rather than leaky rectifier linear unit to increase the accuracy according to the trainable parameters rather than the fixed function. Batch renormalization was used rather than batch normalization to ensure that all the layers are trained on internal representations that are used during inference. Bidirectional gated recurrent unit is used instead of bidirectional long short term memory unit because BGRU doesn't use memory unit (exposes full hidden content without control).

The proposed model compared to **Bluche model** [14] has increased the efficiency of the model. The gated mechanism is changed from Bluche gated mechanism to dauphin gated mechanism which decreased the trainable parameters. The Bluche gated mechanism works as a point wise product of the original feature (X) and the sigmoid activation (S) of the original feature which is given below:

$$Y = S(X) \odot X$$

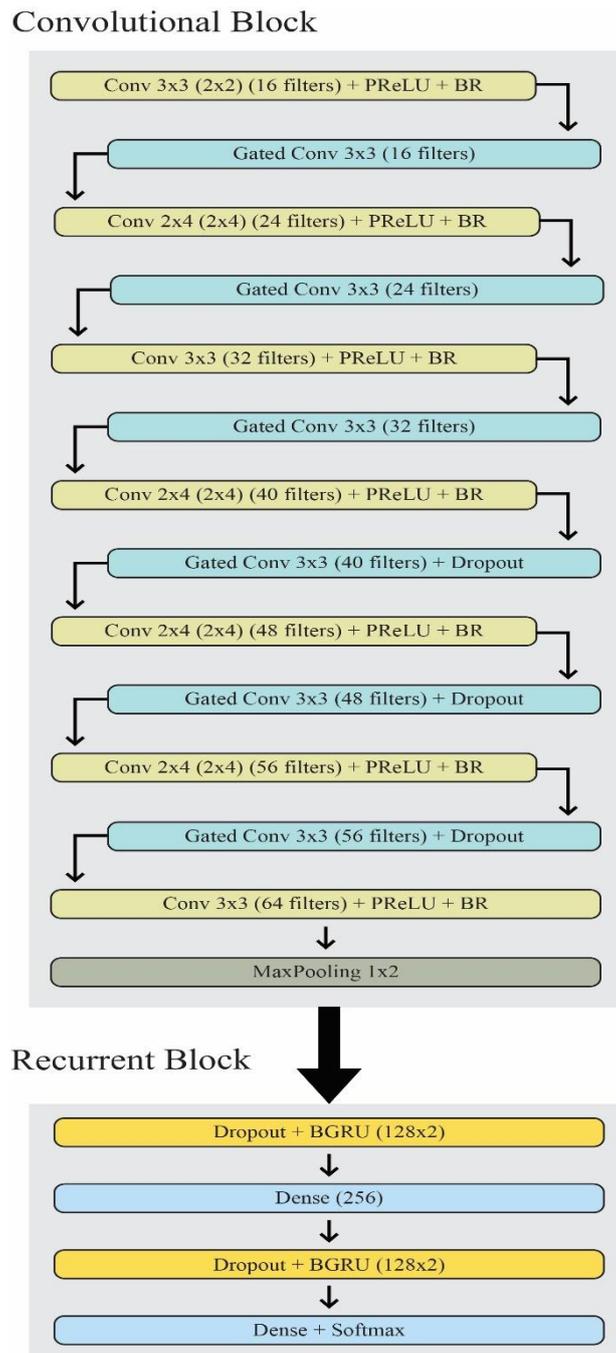
The gated mechanism is similar to Bluche model with a minor difference of the formula. The original features are divided into half and the sigmoid function is applied to the first half (H1) and then pointwise product is carried out between the sigmoid function (S) and the second half (H2) of the original features.

$$Y = S(H1) \odot H2$$

Maxpooling is used to overcome the problem of overfitting while in the proposed model we used it at the end of the convolutional block. The He uniform is used as an initializer in the proposed model rather than Glorot uniform which improves the distribution. The parametric rectifier linear unit is used as an activator rather than hyperbolic tangent to increase the accuracy according to the trainable parameters rather than the fixed function. Batch renormalization is used to ensure that all the layers are trained on internal representations that are used during inference. Bidirectional gated recurrent unit is used instead of bidirectional long short term memory unit because BGRU doesn't use memory unit (exposes full hidden content without control).

The proposed model compared to **Flor model**[15]has increased the efficiency of the model. The proposed model uses the same gated mechanism introduced by Dauphin. Maxpooling is used to overcome the problem of overfitting while in the proposed model we used it at the end of the convolutional block. The He uniform is used as an initializer in the proposed model rather than Glorot uniform which improves the distribution. The convolutional layers are increased with different number of filters to increase the efficiency by detecting important features.

The proposed model is inspired by all three viz. Puigcerver; Bluche and Flor models. The model is using Gated mechanism and architecture similar to Flor with minor changes to increase the accuracy of the model with fewer parameters (approx. 830,000) [15]. Fig.4 depicts the architecture proposed includes 7 convolutional layers, 6 gated convolutional layers, and 2 BGRU.



**Fig.1: Proposed Architecture**

## 2.2. Datasets

All the datasets are partitioned in three subsets i.e. training, validation and testing. The datasets viz. Bentham, IAM, RIMES, Saint Gall and Washington are having their own partitioning methodology. Table I shows the text line image partitioning for all the dataset used.

**Table 1: Description of Datasets**

Dataset	Sentence Length		Average		Partitioning			Total
	Min.	Max.	Characters	Words	Training	Validation	Testing	
Bentham	2	105	48	8	9,195	1,415	860	11,470
IAM	20	81	44	7	6,161	900	1,861	8.922

RIMES	2	110	47	7	10,193	1,133	778	12,104
Saint Gall	8	74	56	8	468	235	707	1,410
Washington	4	62	42	7	325	168	163	656

**2.2.1. Bentham**

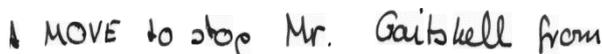
The dataset is written by an English philosopher Jeremy Bentham. The Bentham dataset is a collection of historical manuscripts which are in the form of gray scale images with dark backgrounds and noise in texts. This dataset has about 11,500 lines of text. The partitioning subsets comprises of 9195 images for training, 1415 images for validation and 860 images for testing. The main challenge with this dataset is that it has large amount of punctuation marks in the text lines.



**Fig.2: Sample of Bentham Database**

**2.2.2. IAM**

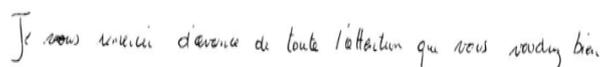
The dataset was prepared by the Institut für Informatik und Angewandte Mathematik (IAM, Department of Computer Science and Applied Mathematics). The dataset comprises of 1539 gray scale scanned text pages of handwritten English. The IAM dataset is a collection of 9000 lines of text written by 657 writers. The dataset was prepared for HTR systems to be independent of the handwriting of the writer, so, the lines written by a single writer belongs to a single subset. The partitioning subsets comprises of 6161 images for training, 900 images for validation and 1861 images for testing. The main challenge with this dataset is that it has many writers and some of the images have cursive handwriting which is very hard for recognition.



**Fig.3: Sample of IAM Database**

**2.2.3. RIMES**

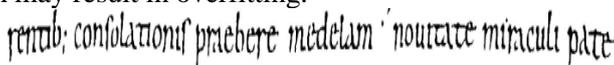
The dataset was compiled by the Reconnaissance et Indexation de données Manuscrites et de facsimilés (RIMES, Recognition and Indexing of Handwritten Documents and Faxes). The RIMES dataset comprises of 12,000 handwritten lines taken from 5600 mails written in French language. The images of the text lines are having more readable writing and have clear background. The dataset was prepared for HTR systems to be independent of the handwriting of the writer, so the text lines written by a single writer belongs to a single subset. The partitioning subsets comprises of 6161 images for training, 900 images for validation and 1861 images for testing. The main challenge in this dataset is that there are many local dialect based words.



**Fig.4: Sample of RIMES Database**

**2.2.4. Saint Gall**

The dataset is written by a Latin person in 9<sup>th</sup> century. The Saint Gall dataset is a collection of historical manuscripts written in Latin. This dataset has about 6,000 unique words and 48 unique characters. This dataset has about 1,410 lines of text. The partitioning subsets comprises of 468 images for training, 235 images for validation and 707 images for testing. The advantage of this dataset is that the text line images are already normalized and binarized. The main challenge with this dataset is that it has very less data which may result in overfitting.

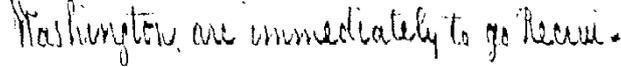


**Fig.5: Sample of Saint Gall Database**

**2.2.5. Washington**

The dataset was built from English papers written by George Washington in 18<sup>th</sup> century. The Washington dataset comprises of historical manuscripts written by two writers and have much lesser

data than Saint Gall. This dataset has about 1,189 unique words and 68 unique characters. This dataset has about 656 lines of text. The partitioning subsets comprises of 325 images for training, 168 images for validation and 163 images for testing. The advantage of this dataset is that the text line images are already normalized and binarized. The main challenge with this dataset is that it has very less data which may result in overfitting.



**Fig.6: Sample of Washington Database**

### 2.3. Experimental Setup

The Puigerver's model used images of whole paragraphs with each case having its own hyperparameters. The Bluche's model used a large private training set of 132,000 images. The Flor's model used the text line images. So, to have a fair correlation between the models and compare the statistical results, we will use the same workflow and hyperparameters for all datasets and models. This idea is inspired by the work of [10].

The experimental setup starts with training the optical models and CTC functions to improve the loss value. RMSprop optimizer [17] is used at each step with 16 images as mini-batch and 0.001 as learning rate. To improve the loss value, after 15 epochs with no improvements, Reduce learning rate on plateau is applied with a factor of 0.2 and after 20 epochs with no improvements, Early Stopping is applied. Word Beam Search [18] is used in this paper as CTC function. For encoding and decoding, a charset of 150 is taken which consists of all the useful characters from ASCII table.

The images for this project must be normalized to have better understanding of the model. So, to normalize all the images takes place in four parts i.e. (i) Illumination Compensation [19] for balancing brightness and contrast. (ii) Deslating [20] the cursive writing images. (iii) Resizing and padding all images to 1024x128x1. (iv) Data Augmentation such as displacement transformation and morphological scaling for all input images is done in 3 parts i.e. (a) Rotating and scaling of the image by  $3^0$  and 5% respectively. (b) Height and width shifting by 5% each. (c) Erosion and Dilation upto 5x5 kernels and 3x3 kernels respectively. To enhance the results, N-gram statistical characters Language model is used with a free to use software application viz. SRILM Toolkit [21]. The language model uses text rather than images, so it is easily trainable. The project uses another free to use online simulator viz. Google Colaboratory for running all the project files using GPU for stronger computational power.

### 2.4. Exploratory evaluation

The experimental evaluation of the models is done using 2 metrics viz., Character Error rate and Word Error rate. They are calculated using Levenshtein distance [22] between predictions and ground truth. To declare the proposed model to have lower error rate, we must have p-value less than alpha i.e. 0.05.

## 3. Results

The models applied on the well-known datasets viz. Bentham, IAM, RIMES, Saint Gall and Washington to obtain better results than the previous models viz. Puigerver, Bluche and Flor. To get lower CER and WER compared to the previously declared models, our proposed model has p-value lower than 0.01. The p-values of all the models are significantly reduced and are given in Table 2.

**Table 2: CER and WER of Datasets**

Optical Model +	Full Text		Only Words	
	CER	WER	CER	WER
<b>Bentham Test Partition (char 9-gram)</b>				
<b>Puigerver</b>	4.65% ( $\pm 0.07$ )	12.07% ( $\pm 0.17$ )	3.95% ( $\pm 0.06$ )	9.07% ( $\pm 0.17$ )
<b>Bluche</b>	6.71% ( $\pm 0.09$ )	16.82% ( $\pm 0.20$ )	5.77% ( $\pm 0.08$ )	13.76% ( $\pm 0.21$ )
<b>Flor</b>	3.98% ( $\pm 0.06$ )	9.76% ( $\pm 0.14$ )	3.33% ( $\pm 0.06$ )	6.65% ( $\pm 0.13$ )
<b>Proposed Model</b>	<b>2.69% (<math>\pm 0.05</math>)</b>	<b>8.47% (<math>\pm 0.15</math>)</b>	<b>2.97% (<math>\pm 0.05</math>)</b>	<b>3.83% (<math>\pm 0.16</math>)</b>

IAM Test Partition (char 8-gram)				
Puigcerver	4.94% ( $\pm 0.05$ )	13.73% ( $\pm 0.12$ )	4.31% ( $\pm 0.04$ )	12.10% ( $\pm 0.13$ )
Bluche	6.60% ( $\pm 0.06$ )	17.89% ( $\pm 0.15$ )	6.13% ( $\pm 0.06$ )	17.64% ( $\pm 0.16$ )
Flor	3.72% ( $\pm 0.04$ )	11.19% ( $\pm 0.11$ )	3.37% ( $\pm 0.04$ )	10.92% ( $\pm 0.12$ )
Proposed Model	<b>2.37% (<math>\pm 0.06</math>)</b>	<b>9.85% (<math>\pm 0.13</math>)</b>	<b>2.70% (<math>\pm 0.06</math>)</b>	<b>6.33% (<math>\pm 0.15</math>)</b>
RIMES Test Partition(char 12-gram)				
Puigcerver	3.75% ( $\pm 0.06$ )	11.67% ( $\pm 0.18$ )	3.23% ( $\pm 0.05$ )	9.89% ( $\pm 0.18$ )
Bluche	5.22% ( $\pm 0.07$ )	14.79% ( $\pm 0.18$ )	4.78% ( $\pm 0.07$ )	14.63% ( $\pm 0.21$ )
Flor	3.27% ( $\pm 0.05$ )	11.14% ( $\pm 0.19$ )	2.63% ( $\pm 0.04$ )	8.71% ( $\pm 0.18$ )
Proposed Model	<b>2.67% (<math>\pm 0.06</math>)</b>	<b>10.18% (<math>\pm 0.19</math>)</b>	<b>1.83% (<math>\pm 0.05</math>)</b>	<b>8.68% (<math>\pm 0.21</math>)</b>
Saint Gall Test Partition (char 11-gram)				
Puigcerver	5.95% ( $\pm 0.03$ )	23.70% ( $\pm 0.03$ )	5.95% ( $\pm 0.03$ )	23.37% ( $\pm 0.03$ )
Bluche	6.01% ( $\pm 0.04$ )	23.73% ( $\pm 0.15$ )	6.01% ( $\pm 0.04$ )	23.73% ( $\pm 0.15$ )
Flor	5.26% ( $\pm 0.03$ )	21.14% ( $\pm 0.13$ )	5.26% ( $\pm 0.03$ )	21.14% ( $\pm 0.13$ )
Proposed Model	<b>3.87% (<math>\pm 0.04</math>)</b>	<b>18.63% (<math>\pm 0.15</math>)</b>	<b>3.82% (<math>\pm 0.04</math>)</b>	<b>18.63% (<math>\pm 0.15</math>)</b>
Washington Test Partition(char 10-gram)				
Puigcerver	19.29% ( $\pm 0.13$ )	32.92% ( $\pm 0.20$ )	18.70% ( $\pm 0.13$ )	34.26% ( $\pm 0.22$ )
Bluche	10.92% ( $\pm 0.11$ )	21.98% ( $\pm 0.18$ )	10.38% ( $\pm 0.11$ )	21.27% ( $\pm 0.19$ )
Flor	3.00% ( $\pm 0.04$ )	7.87% ( $\pm 0.16$ )	2,58% ( $\pm 0.04$ )	7.59% ( $\pm 0.11$ )
Proposed Model	<b>2.99% (<math>\pm 0.04</math>)</b>	<b>7.56% (<math>\pm 0.16</math>)</b>	<b>2.62% (<math>\pm 0.04</math>)</b>	<b>6.58% (<math>\pm 0.11</math>)</b>
ALL Test Partition				
Puigcerver	7.72%	18.82%	7.23%	17.74%
Bluche	7.09%	19.04%	6.61%	18.21%
Flor	3.85%	12.22%	3.43%	11.00%
Proposed Model	<b>2.92%</b>	<b>10.94%</b>	<b>2.79%</b>	<b>8.81%</b>

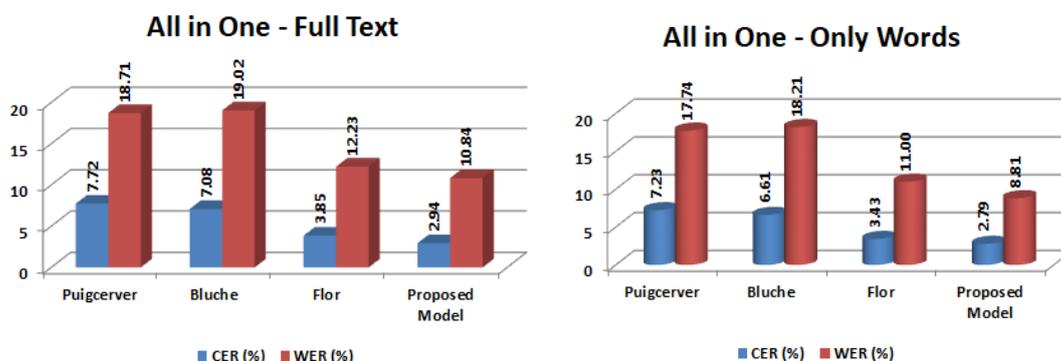


Fig. 7: CER and WER comparison for ALL Test Partition

The improvements of the proposed model compared to previous models are because of three reasons viz. (i) latest deep learning techniques and toolkits, (ii) gated mechanism for convolutional block and (iii) bidirectional gated recurrent units in the recurrent block. The results are clear that the performance

was improved compared to all the previously introduced models, but the trainable parameters are only lower than Puigcerver model while greater than both Bluche and Flor models.

#### 4. Conclusion

In the present work, we have improved the Gated-CNN-BGRU introduced by Flor, after which 2 steps of language processing to get results similar to the handwritten images. All the optical models used the same parameters for a fair comparison. Well-known datasets viz. Bentham, IAM, RIMES, Saint Gall, and Washington were used to analyze all the possible perspective such as small datasets such as the Washington dataset, with no punctuation marks like in Saint Gall dataset, accented language like in the RIMES dataset, cursive writing like in the IAM dataset and full of punctuation marks like in Bentham dataset.

We can say that we have taken the low trainable parameters concept from the Bluche model, the typical loss value concept from the Puigcerver model, and the Flor model's Gated mechanism. The hyperparameters are cost-friendly and result-oriented.

The main contributions to this research are as follows:

- A new architecture, Gated-CNN-BGRU, improves the results based on CNN-BLSTM.
- With less training data, to be able to handle different noises, styles, and variations.
- The numbers of parameters are reduced from the traditional model (CNN-BLSTM) when using the Gated-CNN-BGRU model to reduce the computational cost and smaller model.

We are focused on improving the model for lines and paragraphs and even pages for future work. To improve the proposed model, a new convolutional layer must be introduced.

#### 5. Conflicts of Interest:

The authors declare no conflict of interest.

#### 6. Author Contributions:

Sai Suryateja S: Conceptualization, methodology, software, validation, original draft preparation, Veeraj P: Conceptualization, data curation, writing –review and editing, Vijay Kumar Naidu P: Conceptualization, formal analysis, methodology, Ravi Kumar C V: Supervision, project administration and funding acquisition.

#### References

1. B. L. D. Bezerra, C. Zanchettin, A. H. Toselli, and G. Pirlo, *Handwriting: Recognition, Development, and Analysis*. Nova Science Pub Inc, July 2017.
2. J. A. Sánchez, V. Romero, A. H. Toselli and E. Vidal, "ICFHR2016 Competition on Handwritten Text Recognition on the READ Dataset," *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, Shenzhen, China, 2016, pp. 630-635, DOI: 10.1109/ICFHR.2016.0120.
3. E.Kamalanaban, D., Gopinath, M., &Premkumar, S. (2018). Medicine Box: Doctor's Prescription Recognition Using Deep Machine Learning. *International Journal of Engineering & Technology*, 7(3.34), 114-117. doi:http://dx.doi.org/ 10.14419/ijet.v7i3.34.18785
4. Darmatasia and M. I. Fanany, "Handwriting recognition on form document using convolutional neural network and support vector machines (CNN-SVM)," *2017 5th International Conference on Information and Communication Technology (ICoICT)*, Melaka, Malaysia, 2017, pp. 1-6, DOI: 10.1109/ICoICT.2017.8074699.
5. Alejandro H. Toselli and Enrique Vidal. 2015. Handwritten Text Recognition Results on the Bentham Collection with Improved Classical N-Gram-HMM methods. In *Proceedings of the 3rd International Workshop on Historical Document Imaging and Processing (HIP '15)*. Association for Computing Machinery, New York, NY, USA, 15–22. DOI: <https://doi.org/10.1145/2809544.2809551>
6. FedorBorisyyuk, Albert Gordo, and ViswanathSivakumar. 2018. Rosetta: Large Scale System for Text Detection and Recognition in Images. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery &Data Mining (KDD '18)*. Association for

- Computing Machinery, New York, NY, USA, 71–79. DOI:<https://doi.org/10.1145/3219819.3219861>
7. SeppHochreiter, Jürgen Schmidhuber; Long Short-Term Memory. *Neural Comput* 1997; 9 (8): 1735–1780. doi: <https://doi.org/10.1162/neco.1997.9.8.1735>
  8. Graves, Alex & Fernandez, Santiago & Schmidhuber, Juergen. (2007). Multidimensional Recurrent Neural Networks. *Computing Research Repository - CORR*. 549-558. [10.1007/978-3-540-74690-4\\_56](https://doi.org/10.1007/978-3-540-74690-4_56).
  9. M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," in *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673-2681, Nov. 1997, doi: [10.1109/78.650093](https://doi.org/10.1109/78.650093).
  10. Conover, W.J., 1971. Chapter 6: Statistics of the Kolmogorov-Smirnov type. *Practical nonparametric statistics*.
  11. Moysset, B., Messina, R. Are 2D-LSTM really dead for offline text recognition?. *IJDAR* 22, 193–208 (2019). <https://doi.org/10.1007/s10032-019-0325-0>
  12. Dauphin, Y.N., Fan, A., Auli, M. & Grangier, D.. (2017). Language Modeling with Gated Convolutional Networks. *Proceedings of the 34th International Conference on Machine Learning*, in *Proceedings of Machine Learning Research* 70:933-941 Available from <http://proceedings.mlr.press/v70/dauphin17a.html> .
  13. J. Puigcerver, "Are Multi- dimensional Recurrent Layers Really Necessary for Handwritten Text Recognition?," in 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 2017 pp. 67-72. doi: [10.1109/ICDAR.2017.20](https://doi.org/10.1109/ICDAR.2017.20)
  14. T. Bluche and R. Messina, "Gated Convolutional Recurrent Neural Networks for Multilingual Handwriting Recognition," 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 2017, pp. 646-651, DOI: [10.1109/ICDAR.2017.111](https://doi.org/10.1109/ICDAR.2017.111).
  - A. F. de Sousa Neto, B. L. D. Bezerra, A. H. Toselli and E. B. Lima, "HTR-Flor: A Deep Learning System for Offline Handwritten Text Recognition," 2020 33rd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), Porto de Galinhas, Brazil, 2020, pp. 54-61, DOI: [10.1109/SIBGRAPI51738.2020.00016](https://doi.org/10.1109/SIBGRAPI51738.2020.00016).
  15. Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labeling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning (ICML '06)*. Association for Computing Machinery, New York, NY, USA, 369–376. DOI:<https://doi.org/10.1145/1143844.1143891>
  16. Tieleman T, Hinton G. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*. 2012 Oct;4(2):26-31.
  17. H. Scheidl, S. Fiel and R. Sablatnig, "Word Beam Search: A Connectionist Temporal Classification Decoding Algorithm," 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR), Niagara Falls, NY, USA, 2018, pp. 253-258, DOI: [10.1109/ICFHR-2018.2018.00052](https://doi.org/10.1109/ICFHR-2018.2018.00052).
  18. Chen, K.N., Chen, C.H. and Chang, C.C., 2012. Efficient illumination compensation techniques for text images. *Digital Signal Processing*, 22(5), pp.726-733.
  19. Vinciarelli, A. and Luettin, J., 2001. A new normalization technique for cursive handwritten words. *Pattern recognition letters*, 22(9), pp.1043-1050. <http://www.speech.sri.com/projects/srilm/>
  20. Levenshtein, V.I., 1966, February. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady* (Vol. 10, No. 8, pp. 707-710).