De Sousa Neto, AF.; Bezerra, BLD.; Toselli, AH.; Lima, EB. (2022). A robust handwritten recognition system for learning on different data restriction scenarios. Pattern Recognition Letters. 159:232-238. https://doi.org/10.1016/j.patrec.2022.04.009

Pattern Recognition Letters
journal homepage: www.elsevier.com

A Robust Handwritten Recognition System for Learning on Different Data Restriction Scenarios

Arthur Flor de Sousa Neto[a], Byron Leite Dantas Bezerra[a,**], Alejandro Héctor Toselli[b], Estanislau Baptista Lima[a]

[a] *Universidade de Pernambuco, Recife, Pernambuco, Brazil*
[b] *Universitat Politècnica de València, València, Spain*

ABSTRACT

Handwritten Text Recognition (HTR) systems have gained interest in fields of academic research and commercial applications. Deep learning techniques, and more precisely Convolutional Neural Networks (CNNs), have enabled many recent successes in the computer vision community. However, due to high computational costs, applying CNNs to many real applications is challenging since the specific training data is restricted in many cases. Therefore, in this paper, we present a Gated-CNN-BGRU optical model capable of dealing with this complex challenge. The proposed model was evaluated on five well-known datasets in HTR (Bentham, IAM, RIMES, Saint Gall, and Washington). Additionally, we redefine the training and validation partitions for each dataset, progressively varying the percentage of data between both partitions to create a total of 50 scenarios with different data volumes. The experiment validates and shows that the proposed model presents statistically significant results, surpassing the current models by an average of 2.96 and 8.91 percentage points in character and word recognition accuracy. In the most complex scenario of using 49 images for training, we achieved character and word precision of 87.25% and 71.54% respectively. That means an improvement of 78.32 and 53.54 percentage points, respectively, of the state-of-the-art optical models.

1. Introduction

Offline *Handwritten Text Recognition* (HTR) intends to transcribe cursive texts from images to the digital medium (ASCII, Unicode). Thanks to the expansion of deep learning methods, the HTR research area has evolved in recent years, bringing more robust and accurate models for next generation of text recognition systems (Bezerra et al., 2017). Thus, images are the source of information for HTR systems, which can be applied to transcribe images of historical manuscripts (Sánchez et al., 2016), document forms (Palehai and Fanany, 2017), medical prescriptions (Kamalanaban et al., 2018), and so on. Recent successes in this research field can be attributed to developments in *Convolutional Recurrent Neural Networks* (CRNN), in which the convolutional block is responsible for extracting the features from text images. Next, the recurrent block propagates and decodes the extracted features through a *Connectionist Temporal Classification* (CTC), resulting in the final recognized text hypothesis (Puigcerver, 2017; Bluche and Messina, 2017; Neto et al., 2020).

However, despite the improvements achieved in this field, applying deep learning methods to various tasks is challenging due to the high computational costs involved and the restriction of training data. In addition to the scarcity of data for training deep models, another issue is the vanishing gradient problem of deep learning models (according to the complexity of network topology) (Glorot and Bengio, 2010).

With all these concerns in mind, we propose in this paper an extension of the optical model for HTR systems based on the *Gated Convolutional Recurrent Neural Network* approach with *Connectionist Temporal Classification* (Gated-CRNN-CTC) (Bluche and Messina, 2017). In order to deal with different complex scenarios with availability of data for training, we extend the work of Neto et al. (2020) through an optical model for HTR systems based on this Gated-CRNN-CTC. In this way we simplified some layers and parameters of the proposed model, that give it a greater robustness and stabil-

**Corresponding author: Tel.: +55-081-3184-7548;
e-mail: byron.leite@upe.br (Byron Leite Dantas Bezerra)

ity throughout the training. The main objectives are:

- Achieve high stability and precision in scenarios of extremely low data volumes on handwritten lines available;
- Keep a low number of trainable parameters (thousands) through models of low complexity.

In order to evaluate the effectiveness of the proposed optical model, we used the state-of-the-art models of Puigcerver (2017), and Bluche and Messina (2017) to compare with our proposed model. Several experiments were carried out on 5 well-known datasets: Bentham (Gatos et al., 2014), IAM (Marti and Bunke, 2002), RIMES (Grosicki et al., 2008), Saint Gall (Fischer et al., 2010) and Washington (Fischer et al., 2011). In addition to the official partitions of each dataset, we defined arrangements on their corresponding sample sets, with different proportions of training and validation data, varying from 90%/10% to 10%/90%, to analyze the performance of our model for different data volume scenarios. The results demonstrated that our model outperforms state-of-the-art methods on all the datasets mentioned through extensive experimental evaluations. The scripts of the proposed model are available at: `https://github.com/arthurflor23/handwritten-text-recognition`.

The remaining of this paper is structured as follows. In section 2, state-of-the-art models are described. Then, in section 3, the proposed optical model is presented. In section 4, the experimental methodology is detailed. In section 5, the experimental results obtained in each dataset are reported. In section 6, the results and insights are discussed. Finally, section 7 presents the conclusions.

2. Related Works

The basic pipeline of the optical models for text recognition explored in this paper involves three steps: (i) line images are fed into the convolutional block (CNN) to extract characteristics; (ii) by considering the CNN's output information as a sequence in the writing direction, a recurrent neural network (RNN) block processes it in both directions (bidirectional mapping); and finally (iii) the Connectionist Temporal Classification (CTC) calculates the loss value (training mode), or decodes the output in the final text (inference mode). The state-of-the-art for optical models is presented in the following subsections.

2.1. Convolutional Recurrent Neural Networks

Puigcerver (2017) presented a traditional CRNN architecture approach, which has a high number of trainable parameters (around 9.6 million), but also a high performance in text recognition. The architecture is composed of 5 convolutional and 5 recurrent layers.

The convolutional block is composed of layers with kernel 3x3 and number of filters with the following scheme: 16n (16, 32, 48, 64, 80). Max-pooling with kernel 2x2 is applied in the first three layers, and *Dropout* (with rate 0.2) is applied in the last three layers to avoid overfitting. Besides, the *Glorot Uniform* initialization and the *Leaky Rectifier Linear Unit* (LeakyReLU) activation are used throughout the layers. In addition, *Batch Normalization* is also used in all convolutional layers to normalize the non-linear activation functions.

Finally, the recurrent block contains the *Bidirectional Long Short-Term Memory* (BLSTM) implementation combined with *Dropout* (rate 0.5) (Pham et al., 2014). The number of hidden units in the LSTM cells is set to 256. A fully connected layer with *Dropout* (rate 0.5) is the last layer with output size equal to the character set size plus one (it includes the CTC blank symbol).

2.2. Gated Convolutional Recurrent Neural Networks

The work Bluche and Messina (2017) presented the Gated-CRNN, a variation of CRNN optical model for HTR systems. The Gated-CRNN aims to extract the most relevant features in the convolutional block, maintaining a low number of trainable parameters while achieving good performance in the training process. The Gated-Convolutional layer uses all input features (x) to perform a sigmoid activation (s) and the result is a pointwise multiplication between the input and the output features according to:

$$y = s(x) \odot x \tag{1}$$

The Gated-CRNN architecture in Bluche and Messina (2017) has few parameters (around 730 thousand), and the optical model is composed of 8 convolutional (3 gated included) and 2 recurrent layers.

Specifically, the convolutional block is composed of mini-blocks with convolutions and gated mechanisms, except for the first and last layers where only a regular convolution is used. Thus, the pipeline of convolutions is as follows: (i) 3x3 kernel (8 features); (ii) 2x4 kernel and 3x3 gated mechanism (16 features); (iii) 3x3 kernel and 3x3 gated mechanism (32 features); (iv) 2x4 kernel and 3x3 gated mechanism (64 features); and (v) 3x3 kernel (128 features). Besides, the *Glorot Uniform* initialization and the *Hyperbolic Tangent* (tanh) activator are applied throughout the layers.

Finally, the recurrent block contains a BLSTM implementation alternated with fully connected layers with tanh activation. The number of hidden units in the LSTM cells is set to 128. A final fully connected layer is set with an output size equal to the character set size plus one.

3. Proposed Model

The proposed model is inspired by Puigcerver (2017) and Bluche and Messina (2017) architectures, which aim to: (i) to achieve better results than the Puigcerver (2017) model while keeping a low number of trainable parameters as the model in Bluche and Messina (2017); and (ii) to achieve high stability and performance with few training data.

The *gated mechanism* used to compose our Gated-CRNN architecture was presented in Dauphin et al. (2017), which also aims to extract the most relevant features in the convolutional block, although it is applied differently to Bluche and Messina (2017). In this case. unlike Bluche and Messina (2017), half of the input features (h_1) goes to the sigmoid activation (s), while

the other half does not (h_2). The result is a pointwise multiplication between the two halves:

$$y = s(h_1) \odot h_2 \tag{2}$$

Therefore, the proposed architecture has few parameters (around 920 thousand), which brings a compact model and fast computation. The optical model contains 11 convolutional (5 gated included) and 2 recurrent layers as shown in Figure 1.

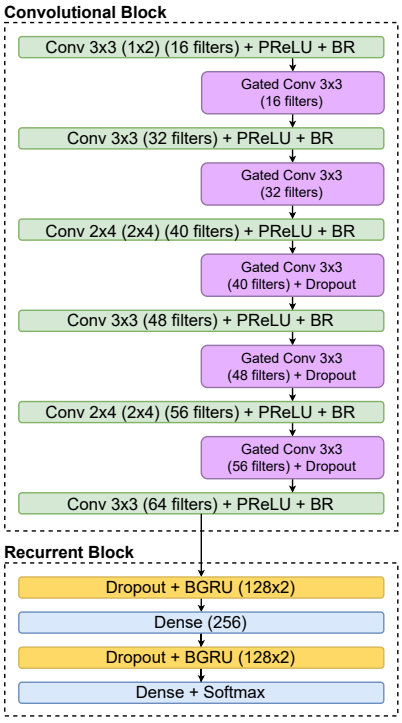

Fig. 1. The proposed optical model architecture.

The convolutional block is in turn composed of mini-blocks with convolutions and gated mechanisms. Thus, the pipeline of one of these mini-blocks is as follows: (i) 3x3 kernel and 3x3 gated mechanism (16 features); (ii) 3x3 kernel and 3x3 gated mechanism (32 features); (iii) 2x4 kernel and 3x3 gated mechanism (40 features); (iv) 3x3 kernel and 3x3 gated mechanism (48 features); (v) 2x4 kernel and 3x3 gated mechanism (56 features); and (vi) 3x3 kernel (64 features). Besides, *He Uniform* initialization with *Parametric Rectified Linear Unit* (PReLU) activation is applied throughout the layers (He et al., 2015), and *Batch Renormalization* (Ioffe, 2017) is applied in all traditional convolutional layers followed by *Dropout* (rate 0.2) in the last three gated mechanisms. Those last components prevent overfitting the model. In addition, we do not use *Max-Pooling* between the blocks, and we decrease the value of the kernel strides of the first convolutional layer unlike the work of Neto et al. (2020).

Finally, the recurrent block contains two *Bidirectional Gated Recurrent Unit* (BGRU) layers with *Dropout* (rate 0.5) (Pham et al., 2014) combined with a Fully Connected layer. The number of hidden units in the GRU cells is set to 128. A final Fully Connected layer has an output size equal to the character set size plus one. This last block emits the most likely characters that make up the output text hypothesis.

4. Material and Methods

In order to compare the proposed model with the state-of-the-art in different scenarios, we conducted segmentation-free HTR experiments on the following standard datasets: Bentham (Gatos et al., 2014), IAM (Marti and Bunke, 2002), RIMES (Grosicki et al., 2008) Saint Gall (Fischer et al., 2010) and Washington (Fischer et al., 2011).

4.1. Datasets

The Bentham dataset (Gatos et al., 2014) is a historical collection of manuscripts written by the English philosopher Jeremy Bentham (1748-1832). The official experimental partition of the 11,473 text lines available in this dataset divides them into 9,198 for training, 1,415 for validation, and 860 for testing.

The IAM handwriting dataset (Marti and Bunke, 2002) issued by the Institut für Informatik und Angewandte Mathematik (Switzerland) contains forms with English manuscripts. This dataset has 9,862 text lines in total, whose official partitioning assigns 6,161 for training, 1,840 for validation (sets 1 and 2), and 1,861 for testing.

The *Reconnaissance et Indexation de données Manuscrits et de fac similÉS* (Rimes) dataset (Grosicki et al., 2008) is a collection of letters written in French by several writers. The official partition, with 12,111 text lines, was set to 11,333 for training and 778 for testing. There is no validation partition defined in this case.

The Saint Gall dataset (Fischer et al., 2010) brings manuscripts written in Latin from the 9th century. This dataset has 1410 text lines officially partitioned into 468 for training, 235 for validation, and 707 for testing.

Finally, the Washington dataset (Fischer et al., 2011) was built from scanned letters handwritten in English by George Washington (and his associates) in the 18th century. This dataset has 656 text lines officially partitioned into 325 for training, 168 for validation, and 163 for testing. Table 1 summarizes the default data distribution for each dataset.

Table 1. Default partitions of each dataset

Dataset	Training	Validation	Test	Total
Bentham	9198	1415	860	11,473
IAM	6161	1840	1861	9862
RIMES	11,333	-	778	12,111
Saint Gall	468	235	707	1410
Washington	325	168	163	656

In order to simulate and investigate the impact of restricted data to HTR systems, we performed an experimental methodology with diverse data scenarios, generated from different amounts of data between training and validation partitions.

Thus, we progressively and randomly vary the percentage of data between both partition sets in steps of 10 percentage points from 90%/10% to 10%/90%. It is worth mentioning that the test partition was kept as the original in each set. Table 2 details the data volume scenarios with the arrangements in the training and validation partitions.

Table 2. Arrangements of data in training and validation partitions of each dataset.

Arrangement	Bentham		IAM		RIMES		Saint Gall		Washington	
Train/Valid. (%)	Train	Valid.	Train	Valid.	Train	Valid.	Train	Valid.	Train	Valid.
90/10	9552	1061	7201	800	10,200	1133	633	70	444	49
80/20	8490	2123	6401	1600	9066	2267	562	141	394	99
70/30	7429	3184	5601	2400	7933	3400	492	211	345	148
60/40	6368	4245	4801	3200	6800	4533	422	281	296	197
50/50	5307	5307	4001	4001	5667	5667	352	352	247	247
40/60	4245	6368	3200	4801	4533	6800	281	422	197	296
30/70	3184	7429	2400	5601	3400	7933	211	492	148	345
20/80	2123	8490	1600	6401	2267	9066	141	562	99	394
10/90	1061	9552	800	7201	1133	10,200	70	633	49	444

4.2. Experimental Evaluation

The most common metrics for HTR systems are *Character Error Rate* (CER) and *Word Error Rate* (WER). These metrics are calculated using the Levenshtein Distance (LD) (Levenshtein, 1966), also known as edit distance (insertion, deletion, and substitution), between ground truth and predicted strings. The Normalized Levenshtein Distance (NLD), which is more immune to any bias resulting from the sentence length, can be computed by:

$$NLD(a_T, a_R) = \frac{LD(a_T, a_R)}{\mid a_T \mid}, \tag{3}$$

where a_T is the target string (ground truth), a_R is the recognized string, $\mid a_T \mid$ is the length of the string a_T, and LD is the Levenshtein Distance. Furthermore, through the Average Normalized Levenshtein Distance (ANLD) defined as:

$$ANLD = \frac{\sum_{i=1}^{T} NLD(a_T^i, a_R^i)}{T}, \tag{4}$$

where T is the number of test samples. Experimental results (see Sec.5) are reported in terms of *Character/Word Precision Rates* CPR/WPR (which are the corresponding complements of CER and WER), where a high value indicates high recognition rate.

Finally, for statistical testing, we performed five training runs of the optical models in each different arrangement of the datasets (default partitioning included), and used Wilcoxon signed-rank test (Wilcoxon, 1992) with 5% significance. Thus, we considered the null hypothesis $H_0 : \mu_1 \leq \mu_2$, and alternative hypothesis $H_1 : \mu_1 > \mu_2$, where μ_1 is the precision rate of the proposed model and μ_2 is the precision rate of the other model in comparison. That means that the p-value must be lower than $\alpha = 0.05$ to assume that the proposed model offers a significantly higher precision rate. In addition, statistical testing is applied to both character and word levels.

4.3. Experimental Setup

The optical models proposed by (Puigcerver, 2017) and (Bluche and Messina, 2017) were evaluated following a different experimental setup from that of their original works. Therefore, to make a fair statistical comparison, we have adopted the same workflow and hyper-parameters for all models in each dataset.

In this context, we do not apply any image preprocessing to the input data other than the standardization (Z-score normalization) and resizing to 1024x128x1 (Height x Width x Channel) with padding. In addition, data augmentation was applied to the training partition (during the training step) to increase the variety of images. To this end, morphological and displacement functions with random parameters were used, such as erosion (up to 5x5 kernel), dilation (up to 3x3 kernel), rotation (up to 3 degrees), resizing (up to 5%), and displacement of height and width (up to 5%).

The optical models were trained in order to minimize the validation CTC loss function. To this end we used RMSprop optimizer with a learning rate of 0.001 and mini-batches of 8 images per step. Reduce Learning Rate on Plateau (factor 0.2), and Early Stopping mechanisms were also applied after 10 and 100 epochs, respectively, without improving the validation loss.

The Kaldi ASR (Povey et al., 2011) along with SRI Language Modeling (SRILM) (Stolcke, 2002) toolkits were applied to weight the CRNN model outputs with a language model trained on external data. The language model, based on statistical characters N-grams, can be efficiently trained using only plain text from the ground-truth transcripts of the training partition of each dataset (Sánchez et al., 2019).

Finally, the optical and character language models were subsequently used by the Kaldi decoder, with a search beam of 30, to produce the best hypotheses for the test-set lines images.

The whole training process was conducted on the Google Colab platform, which offers Linux operating system with 12GB memory and GPU NVIDIA Tesla T4 16GB.

5. Results

In this section, we present our results based on our data arrangements in training and validation partitions of each dataset. In the Bentham dataset, the best results were obtained using a 9-gram language model, considering the full text of the test set. The arrangement that provided the best precision result was 90%/10%, where the proposed model achieved a CPR/WPR of 96.73%/90.67%, while Puigcerver's model got 95.05%/87.51% and Bluche's model 93.46%/78.96%, correspondingly. Even with close results in the first arrangements (default partition included), a decrease in the volume of training data leads to an increase in the difference in performance between the optical

models. Figure 2 shows the results obtained throughout the different arrangements for each optical model.

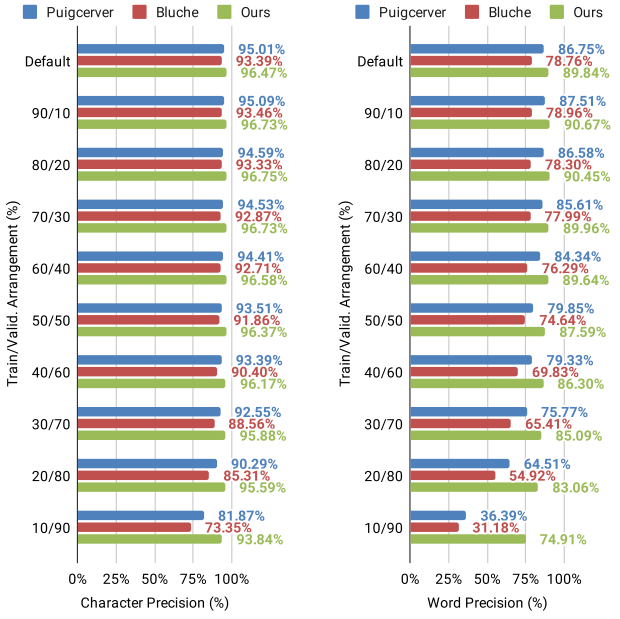

Fig. 2. Results for different arrangements of training and validation partitions of the Bentham dataset (higher values are better).

In the IAM dataset, the best results were obtained using an 8-gram language model. The proposed model achieved a CPR/WPR of 96.41%/87.07%, while Puigcerver's model got 95.57%/85.38% and Bluche's model 93.43%/76.54%, respectively. Figure 3 shows the results obtained throughout the different arrangements defined for the IAM dataset.

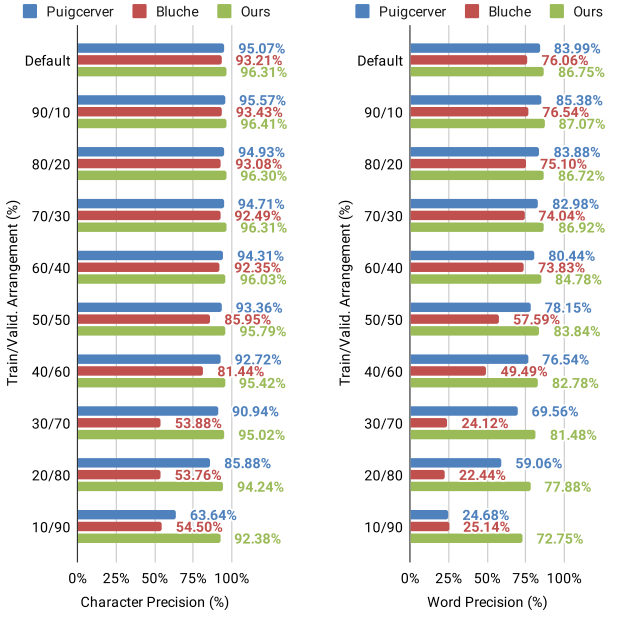

Fig. 3. Results for different arrangements of training and validation partitions of the IAM dataset (higher values are better).

The RIMES dataset does not have an official default validation partition defined, so we use only the arrangements for analysis. In this way, the best results were obtained using a 12-gram language model. Thus, the proposed model achieved a CPR of 97.77% with a WPR of 90.47%, while Puigcerver's model got 96.90% with 89.67% and Bluche's model 95.60% with 82.87%, respectively. Figure 4 shows the corresponding results obtained for each of optical model and arrangement.

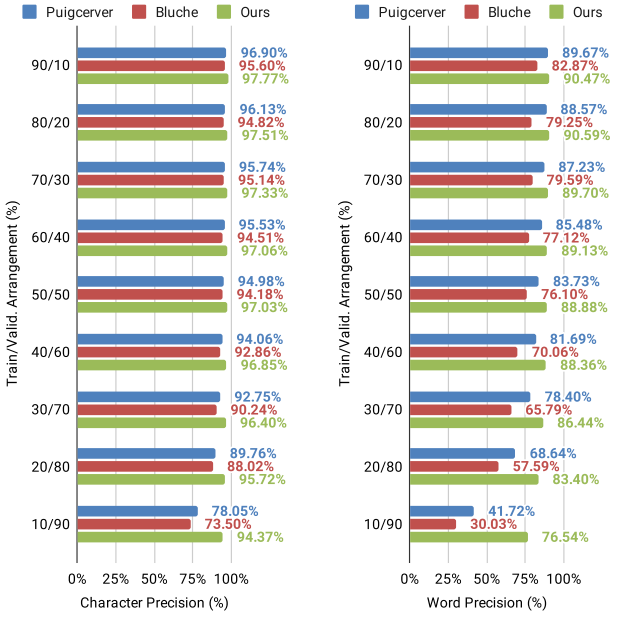

Fig. 4. Results for different arrangements of training and validation partitions of the RIMES dataset (higher values are better).

The Saint Gall dataset is the only one that does not have punctuation marks in the text. Then, we found that 11-gram language model produces the best results. The arrangement that provided the best precision results were 70%/30%, for the proposed model and Bluche's model, while 80%/20% for Puigcerver's. Thus, the proposed model achieved a CPR of 96.20% with a WPR of 83.02%, while Puigcerver's model was 96.31% with 78.28% and Bluche's model 95.03% with 77.86%, respectively.

Furthermore, Puigcerver's and Bluche's models achieved low results in the last arrangement 10%/90%, due to the low volume of data. In addition, we must also mention here about the premature overfitting of Puigcerver's model in all executions in this specific scenario. Figure 5 shows the results obtained throughout the different arrangements for each optical model.

The Washington dataset has the least amount of data compared to the others, and as expected, this scenario highlights the challenge of dealing with overfitting, for which early stopping is quickly activated. Also employing an 11-gram language model, the proposed model reached a CPR of 96.82% with a WPR of 92.72%, while Puigcerver's got model 89.20% with 75.43% and Bluche's model 92.11% with 77.90%, respectively.

In particular, given that the Washington dataset has the most restrictive data volume, overfitting cases emerged for several of the data arrangement scenarios. Fortunately, the proposed model proved to be stable in these different restrictive scenarios. The 10%/90% arrangement, for example, has only 49 sam-

Fig. 5. Results for different arrangements of training and validation partitions of the Saint Gall dataset (higher values are better).

ples for training, and the proposed model still achieved a CPR of 87.25% with a WPR of 71.54%, while the Puigcerver's and Bluche's models did not exceed 20% in both metrics. Figure 6 shows the results obtained throughout the arrangements.

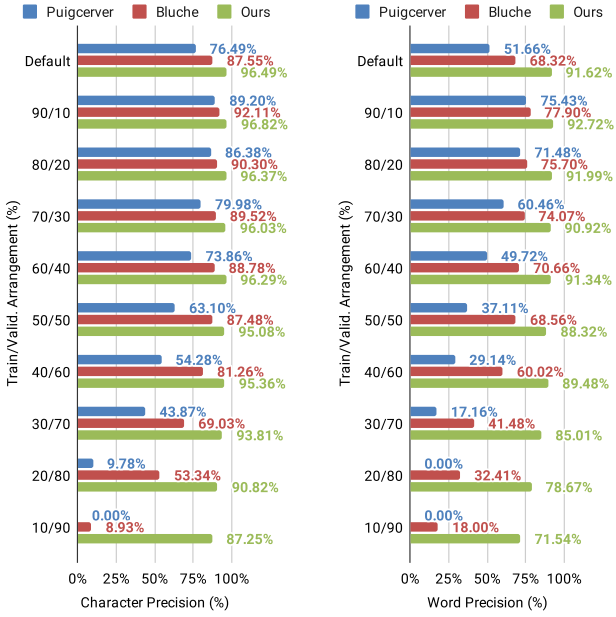

Fig. 6. Results for different arrangements of training and validation partitions of the Washington dataset (higher values are better).

To summarize results of the performed entire experiment, we compute the weighted average of all results over the datasets. In general, the proposed optical model achieved a CPR around 96.81% with a WPR of 89.03%, while Puigcerver's got 93.84% with 81.31% and Bluche's 93.85% with 78.93%, respectively. That means an improvement of about 2.96 and 8.91 percentage points in CPR and WPR, respectively. Figure 7 shows the averaged results.

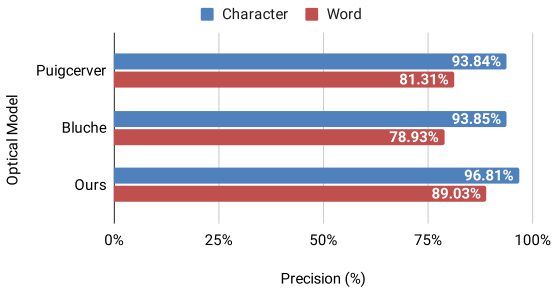

Fig. 7. The weighted average of precision over all datasets (higher values are better).

Finally, the statistical analysis was applied to all results obtained from each dataset. Thus, we computed the character precision p-value and word precision p-value lower than 0.04 in all datasets. That is below the standard $\alpha = 0.05$ (p-value $<$ 5e-2), meaning that we can assume that the proposed optical model, based on Gated-CRNN, has a higher precision rate, consequently, a lower CER and WER.

6. Discussion

The improvements of the proposed optical model can be explained mainly by the combination of gated mechanism, BGRU in the recurrent block, and recent deep learning techniques. That allows not only to achieve high performance of the optical model in scenarios with a considerable amount of data, but also in more complex scenarios with a reduced amount of data.

In scenarios with reduced data volume, it was possible to compare the stable performance of the proposed model with the current ones in the literature. Thus, the low achieved results are caused by ill-learnt models with poor generalization and, in some cases, by premature overfitting. Another important fact to mention is that the obtained results are better compared to the default partitions of datasets. In general, default partitioning is defined by the study purpose of each dataset, so the specific selection of images may be more complex in the training or validation partition. Finally, the arrangements in this work were applied through the pseudo-randomness of the data.

Another important aspect to evaluate in deep neural networks is their complexity, which directly impacts its application performance in terms of, for example, the computational memory usage of the model and its decoding time. The number of trainable parameters of the proposed optical model is close to Bluche's model (thousands), and it is equivalent to about 10% of the one of the Puigcerver's model (millions). In decoding time (decoding phase), we calculated the average decoding time of each optical model across all runs of the experiment. Thus, Bluche's model was the fastest one, while the proposed model was the slowest one. Table 3 shows the number of parameters and the average decoding time using a standard notebook with a dual-core CPU (Intel i7-7500U).

Therefore, the achieved results show a good performance over several data scenarios, mainly with small training data.

Table 3. Number of parameters and average decoding time.

Optical Model	# of params	Decoding Time
Puigcerver	9.4 M	157 ms/line
Bluche	0.7 M	111 ms/line
Ours	0.9 M	189 ms/line

However, it is important to mention that in the decoding phase, the proposed model is a bit slower than the other state-of-the-art models because its network pipeline has more internal layers processing an image.

7. Conclusions

This paper presents a Gated-CNN-BGRU architecture, based on Gated-CRNN for offline Handwritten Text Recognition (HTR) systems, which focuses on improving the state-of-the-art recognition rate, especially in low data volume scenarios. We evaluated the proposed optical model and the current state-of-the-art models on five known public benchmark datasets in the HTR field (Bentham, IAM, RIMES, Saint Gall, and Washington), which allowed an analysis from several perspectives. In addition to facilitating better analysis, mainly in low data volume scenarios, we evaluated performance models for different arrangements of data partition between training and validation, varying from 90%/10% to 10%/90%.

The proposed optical model demonstrated robustness in recognizing text lines, and we highlight its good performance even with small training data compared to the other models. In all the executions and results of the experiment, we surpass the state-of-the-art optical models by 2.96 and 8.91 percentage points on average in characters and words precision (recognition rate), respectively. It is also worth mentioning that training only on 49 samples, the performance achieved was around 87.25% and 71.54% in character and word precision. If these are compared with the Puigcerver's and Bluche's optical models, we outperform them by around 78.32 and 53.54 percentage points in this specific scenario.

In the future, we want to explore alternatives of convolutional layers for the optical model to make it even more compact and improve performance in the decoding time. Moreover, we will evaluate other study scenarios, such as offline handwriting recognition at the paragraph and page levels.

Acknowledgments

This study was financed in part by the founding public agencies: Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001, and CNPq.

References

Bezerra, B.L.D., Zanchettin, C., Toselli, A.H., Pirlo, G., 2017. Handwriting: Recognition, Development and Analysis. Nova Science Pub Inc.

Bluche, T., Messina, R., 2017. Gated convolutional recurrent neural networks for multilingual handwriting recognition. 14th IAPR International Conference on Document Analysis and Recognition (ICDAR) , 646–651doi:10.1109/ICDAR.2017.111.

Dauphin, Y.N., Fan, A., Auli, M., Grangier, D., 2017. Language modeling with gated convolutional networks, in: Proceedings of the 34th International Conference on Machine Learning - Volume 70, JMLR.org. p. 933–941.

Fischer, A., Frinken, V., Fornés, A., Bunke, H., 2011. Transcription alignment of latin manuscripts using hidden markov models, in: Proceedings of the 2011 Workshop on Historical Document Imaging and Processing, Association for Computing Machinery, New York, NY, USA. p. 29–36.

Fischer, A., Indermühle, E., Bunke, H., Viehhauser, G., Stolz, M., 2010. Ground truth creation for handwriting recognition in historical documents. ACM International Conference Proceeding Series , 3–10.

Gatos, B., Louloudis, G., Causer, T., Grint, K., Romero, V., Sánchez, J.A., Toselli, A.H., Vidal, E., 2014. Ground-truth production in the transcriptorium project, in: 11th IAPR International workshop on document analysis systems (DAS), pp. 237–241.

Glorot, X., Bengio, Y., 2010. Understanding the difficulty of training deep feedforward neural networks, in: In Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS'10), Society for Artificial Intelligence and Statistics, New Jersey, USA. pp. 249–256.

Grosicki, E., Carre, M., Brodin, J.M., Geoffrois, E., 2008. Rimes evaluation campaign for handwritten mail processing, in: ICFHR 2008 : 11th International Conference on Frontiers in Handwriting Recognition, Concordia University, Montreal, Canada. pp. 1–6.

He, K., Zhang, X., Ren, S., Sun, J., 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 1026–1034.

Ioffe, S., 2017. Batch renormalization: Towards reducing minibatch dependence in batch-normalized models, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA. pp. 1942–1950.

Kamalanaban, E., Gopinath, M., Premkumar, S., 2018. Medicine box: Doctor's prescription recognition using deep machine learning. International Journal of Engineering and Technology(UAE) 7, 114–117.

Levenshtein, V., 1966. Binary codes capable of correcting deletions, insertions and reversals. Soviet Physics Doklady 10, 707.

Marti, U.V., Bunke, H., 2002. The iam-database: An english sentence database for offline handwriting recognition. International Journal on Document Analysis and Recognition 5, 39–46.

Neto, A.F.S., Bezerra, B.L.D., Toselli, A.H., Lima, E.B., 2020. HTR-Flor: a deep learning system for offline handwritten text recognition, in: 2020 33rd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), IEEE Computer Society, Los Alamitos, CA, USA. pp. 54–61. doi:10.1109/SIBGRAPI51738.2020.00016.

Palehai, D., Fanany, M.I., 2017. Handwriting recognition on form document using convolutional neural network and support vector machines (cnn-svm). 5th International Conference on Information and Communication Technology (ICoIC7) doi:10.1109/ICoICT.2017.8074699.

Pham, V., Bluche, T., Kermorvant, C., Louradour, J., 2014. Dropout improves recurrent neural networks for handwriting recognition, in: 2014 14th International Conference on Frontiers in Handwriting Recognition, pp. 285–290. doi:10.1109/ICFHR.2014.55.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlíček, P., Qian, Y., Schwarz, P., Silovský, J., Stemmer, G., Vesel, K., 2011. The kaldi speech recognition toolkit. IEEE 2011 Workshop on Automatic Speech Recognition and Understanding .

Puigcerver, J., 2017. Are multidimensional recurrent layers really necessary for handwritten text recognition? 14th IAPR International Conference on Document Analysis and Recognition (ICDAR) , 67–72doi:10.1109/ICDAR.2017.20.

Stolcke, A., 2002. Srilm – an extensible language modeling toolkit. Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP 2002) , 901–904.

Sánchez, J.A., Romero, V., Toselli, A.H., Villegas, M., Vidal, E., 2019. A set of benchmarks for handwritten text recognition on historical documents. Pattern Recognition 94, 122–134. doi:https://doi.org/10.1016/j.patcog.2019.05.025.

Sánchez, J.A., Romero, V., Toselli, H.A., Vidal, E., 2016. Icfhr2016 competition on handwritten text recognition on the read dataset. 15th International Conference on Frontiers in Handwriting Recognition (ICFHR) , 630–635doi:10.1109/ICFHR.2016.0120.

Wilcoxon, F., 1992. Individual Comparisons by Ranking Methods. Springer New York, New York. chapter 1. pp. 196–202. doi:10.1007/978-1-4612-4380-9_16.