A Unified Architecture for Urdu Printed and Handwritten Text Recognition

Arooba Maqsood[1,2](✉), Nauman Riaz[1](✉), Adnan Ul-Hasan[1], and Faisal Shafait[1,2](✉)

[1] National Center of Artificial Intelligence (NCAI), National University of Sciences and Technology (NUST), Islamabad, Pakistan
{amaqsood.mscs20seecs,nriaz.mscs20seecs,adnan.ulhassan}@seecs.edu.pk

[2] School of Electrical Engineering and Computer Science (SEECS), National University of Sciences and Technology (NUST), Islamabad, Pakistan
faisal.shafait@seecs.edu.pk

Abstract. Urdu text recognition (handwritten or printed) remains a challenging task due to its diverse writing styles and fonts. State-of-the-art Transformer-based OCR systems are computationally expensive because they rely on computationally expensive pretraining over text images. To address this challenge, we propose a robust architecture that utilizes a custom CNN block with a Transformer encoder for image understanding and a pre-trained Transformer decoder on Urdu language modeling. The presented model generalized well even for scarce training data without the need for pre-training on synthetic text images. Experiments show that our proposed architecture outperforms the state-of-the-art methods for Urdu printed and handwritten text recognition on several publicly available datasets including UPTI, NUST-UHWR, and MMU-OCR-21. We also combined printed and handwriting datasets to train our architecture and propose a single unified model; capable of recognizing both printed and handwritten text for maximum variations of fonts and writing styles with state-of-the-art results.

Keywords: OCR · Urdu Text Recognition · Handwriting Recognition

1 Introduction

Optical Character Recognition (OCR) technology has a rich history and is widely used by different industries and organizations to digitize their data in order to perform data analysis, streamline day-to-day operations and automate their processes [1]. Additionally, this technology is being used by the immigration department to recognize passports, the city traffic police to recognize license plates, and banking institutions to process demand drafts and checks automatically and to preserve and digitize historical writing [15]. The OCR systems can also be used to improve assistive technologies for blind and visually impaired people [6].

Urdu OCR has attracted tremendous research interest over the past ten years. According to the Ethnologue[1], Urdu is among the 10 most spoken languages in the world with over 230 million native speakers. Even though there

[1] https://www.ethnologue.com/.

G. A. Fink et al. (Eds.): ICDAR 2023, LNCS 14190, pp. 116–130, 2023.
https://doi.org/10.1007/978-3-031-41685-9_8

is a sizable global audience for the Urdu language and it is written and spoken in many countries, there has been little to no advancement in having its script recognized [10].

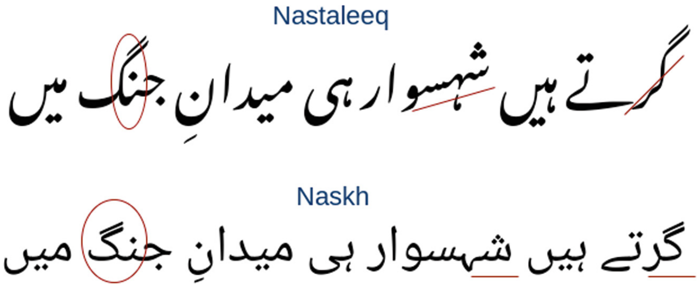

Nastaleeq
گرتے ہیں شہسوار ہی میدانِ جنگ میں
Naskh
گرتے ہیں شہسوار ہی میدانِ جنگ میں

Fig. 1. An example of the Naskh and Nastaleeq scripts depicting the difference in their writing style and alignment. The beginning characters of the Naskh script are aligned along a straight baseline, whereas the characters of the Nastaleeq script have a baseline that is diagonal and shifts from right to left.

The Urdu language has a complex writing script with over 24,000 unique ligatures and different joining rules [13]. One possible reason for the complexity of Urdu script is its writing styles and scripts [15]. Devani, Kofi, Naskh, Nastaleeq, Riqa, and Taluth are only a few of the writing styles used in Urdu. The two most well-known of these are Naskh and Nastaleeq [4]. The majority of the printed material currently in circulation is in the Nastaleeq script, whereas the more prevalent script for digital content is Naskh [6]. Figure 1 illustrates an example sentence to show how these two scripts differ from one another.

It is evident from Fig. 1 that there are notable changes between the two scripts in terms of ligature formation and character shape variation. Due to diverse variations, a system trained on the Nastaleeq script performs poorly in recognizing text in the Naskh [1]. In order to have a larger range of applications, it would be useful to create an Urdu OCR system that can recognize text written in any script. Practically all of the current Urdu OCRs were created for the Nastaleeq scripts and are, therefore, useless when the source images have text in other scripts [6].

Additionally, the handwritten text also poses challenges in its recognition [15]. When it comes to handwriting, humans are extremely inventive, which results in a wide variety of writing styles, character formations, etc. Every person has a unique writing style (refer to Fig. 2), so teaching a model to recognize an unseen handwriting style is a difficult task. Therefore, while addressing text recognition, it is important to take into account several features of handwritten text, including writing styles, the type of paper used, the thickness of strokes, human mistakes, and a number of other issues.

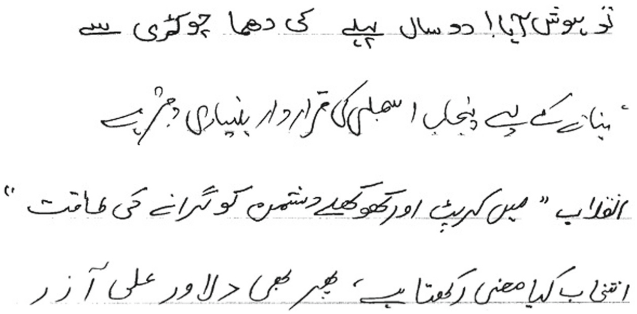


Fig. 2. Sample images of different writing styles for handwritten text.

To the best of our knowledge, there exists no system that can recognize both handwritten and printed real-world Urdu text samples. Many of the current challenges, including speech recognition, machine translation, text summarization, and image captioning, have a single end-to-end solution in deep learning [6]. When developing an end-to-end OCR system, all of the OCR's substitute tasks (i.e. printed text recognition, offline handwriting recognition, etc.) can be contained within a single model.

Printed text is easier to recognize as it is usually well-formed and printed with a consistent font. But on the other hand, handwriting text recognition is a challenging task due to the diverse variability in writing styles. Both forms of text are commonly encountered together in real-world scenarios. This motivates us to develop a unified model that can recognize both forms of text and can improve accuracy and efficiency in text recognition tasks.

Developing a unified model is valuable for digitizing Urdu documents that contain a mix of printed and handwritten text, such as application forms, invoices, receipts, and affidavits. One such system can facilitate a seamless transcription of different Urdu text forms; improving the usability of Urdu language technology for everyday tasks and helping preserve manuscripts and documents, making them easily accessible to a wider audience.

Hence, we aim to propose an end-to-end OCR system optimized as a single, unified entity that will be capable of recognizing both printed and handwritten text for a maximum variation of script and writing styles. The major contributions of our work are:

1. We propose a Transformer based text recognition model that employs a Convolutional Neural Network to extract image embeddings. The whole architecture is trained using CTC loss at the transformer encoder for image understanding and cross-entropy loss at the transformer decoder for language modeling.
2. We propose a unified architecture that gives state-of-the-art results both for printed and handwritten text recognition for the Urdu language.

The paper is mainly divided into the following sections. Section 2 gives a summary of the related work in the field of Urdu text recognition. Section 3 discusses the proposed technique for the task at hand. Section 4 describes the experimental setup including preprocessing steps and implementation details. Section 5 provides the findings and their interpretation. Lastly, Sect. 6 concludes the study and provides future research directions.

2 Related Work

In this section, we present the previous approaches for both Urdu printed and handwritten text recognition.

2.1 Printed Text Recognition

The approaches discussed in [2,3] rely on Recurrent Neural Network (RNN) and Convolutional Neural Network (CNN) blocks for the identification of Urdu ligatures. Naz et al. [2] used the stack of Multi-Dimensional Long Short Term Memory (MDLSTM) layers and feedforward neural networks followed by an output layer for sequence labeling using Connectionist Temporal Classification (CTC) loss for printed Urdu text recognition. Experimentally it was shown that the proposed model achieved an accuracy of 98% on the UPTI dataset. RNNs are widely used in situations when there is a temporal relationship between the inputs, but the nature of this interaction is less clear when it comes to visuals. To address this problem, in another study by Naz et al. [3], the authors proposed a hybrid model that comprises a CNN block followed by the MDLSTM block. The CNN block is added for implicit feature extraction from the images that help learn refined representations from the input image. On the publicly available Urdu Printed Text-line Image (UPTI) dataset, the proposed model had an accuracy of 98.12% for the classification of Urdu ligatures. Despite these changes, the architecture fails to generalize on varying scripts of Urdu. The proposed model lacks an implicit or explicit language model and relies on image signals only.

The authors in [7,9,12,16] developed a combined framework for the detection and recognition of text in video frames and natural scenes. In Mirza et al.'s study [16], text detection in video frames is carried out by fine-tuning Faster R-CNN [5] (model for object detection). Whereas, for text recognition, the authors proposed the UrduNet model (combination of CNN and LSTM blocks). An extensive series of experiments resulted in an 88.3% F1-score for text detection and an 87% recognition rate for text recognition on their custom dataset. In another study for text recognition in video frames, Rehman et al. [9] proposed a simple model that comprises a CNN block that acts as a feature extractor, a bi-directional GRU block for sequence recognition, and finally a classification layer that classifies feature vectors from prior layers into characters. The authors used the AcTiVComp20 and NUST-Urdu Ticker Text (NUST-UTT) [9] datasets for testing the proposed model and achieving encouraging results. Narwani et al. [12]

also focused on text recognition in natural scenes and also proposed an Urdu Scene Text Dataset (USTD) that contains images from real scenes like roads and streets. To prove the validity of their dataset, the authors provided an extensive comparison with baselines for both text detection (including ResNet-50, EAST, and Seglink) and recognition (including variants of CRNN). The end-to-end combination of ResNet-50 with CRNN outperformed with an F1-score of 0.66. Since Urdu Nastaleeq text uses a modified version of Arabic script, there is still a challenge in localizing, detecting, and recognizing it. To further improve the baselines, the authors in [7], enhanced approaches for both text detection and recognition. The Connected Component Analysis (CCA) and Long Short-Term Memory (LSTM) units are used in the initial stage to detect text. The detected text is recognized in the second phase using a hybrid Convolution Neural Network and Recurrent Neural Network (CNN-RNN) architecture. The proposed method performs better than the ones currently in use, with an overall accuracy of 97.47% due to the use of CCA, which has the capability to process higher dimensional data as well.

The different scripts of Urdu have variations in cursive writing styles which pose a challenge in text recognition and most of the systems for Urdu OCR are developed that do not cater to these variations of writing styles. To further aid the research in this direction, the authors of [10] focused on developing a framework that can recognize the text irrespective of its script and writing style. The authors not only created a large-scale multi-font printed Urdu text recognition data set but also presented extensive experimentation using the CNN-based ResNet-18 model with an accuracy of 85 percent. Further addressing the scarcity of multi-font and multi-lines datasets for the Urdu language, the authors of [6] presented a very large 'Multi-level and Multi-script Urdu (MMU-OCR-21)' corpus. The corpus is made up of over 602,472 images in total, including ground truth for text-line and word images in three well-known fonts. Additionally, the authors provided extensive experimentation with text-line and word-level images using a variety of cutting-edge deep learning baselines with encouraging results.

The previous approaches discussed for printed text recognition so far rely only on information from the text image and do not incorporate language modeling. Moreover, to capture the temporal information the authors relied only on networks like RNNs, GRUs, and LSTMs, which suffer from vanishing gradients for very large sequences which is the case in text line recognition. The lack of language understanding and extensive use of RNNs hinders the generalization of these models on varying Urdu text fonts and thus a generalized OCR system cannot be proposed for the Urdu language.

2.2 Handwritten Text Recognition

The authors in [8,11,14] presented analytical approaches based on implicit character segmentation. Similar to the approach in [3], the authors [8] employed a CNN block that works as a feature extractor and an LSTM block for sequence classification of the Urdu characters. The experimentation on custom data of

6,000 unique handwritten text lines gave a character recognition accuracy of 83.69%. The authors proposed to extend their experimentation to other publically available datasets for better generalization of their approach. Similar to the idea of Naz et al. [2], the authors of [14] used a similar approach based on CNN block as a feature extractor and an MDLSTM block as the classifier. The authors provided an extensive comparison of several baseline datasets and achieved satisfying results. Mushtaq et al. [11] also used CNNs as they produce effective results compared to traditional handmade feature extraction methods and do not require explicit feature engineering. The authors achieved a recognition rate of 98.82% on their custom dataset.

Zia et al. [13] demonstrates how convolutional-recursive architecture can be utilized to recognize recursive text effectively. The papers aimed to address the challenges of recognizing the complex ligatures in the Urdu Language by developing a robust architecture based on CNN-RNN blocks aided by an n-gram language model (LM). In this proposed model, the implicit character level segmentation is done using CNN, and RNN acts as the classifier. On top of these blocks, the n-gram language model acts as a spelling corrector. The reported character error rate (CER) for this approach is 5.82%. Additionally, to address the scarcity of Urdu language datasets, the authors also presented a dataset named 'NUST-UHWR'. The authors used a character-level deep learning model on the output of which a word-level n-gram model acts as a spelling corrector. This architecture fails to recognize out-of-vocab words. In order to address these issues, Riaz et al. [15] mapped the problem of handwritten text recognition as a Seq2Seq problem. The authors proposed an encoder-decoder Conv. Transformer model that not only leverages the task at hand by capturing the inter-language dependencies and caters to diverse alignments of characters at the embedding level. Due to the inherent property of how transformers work, the proposed model also learns a language model for language understanding hence eliminating the need for an explicit language model as proposed by Zia et al. [13] and also effectively handles the out-of-vocab words. Riaz et al's [15] approach gave the CER of 5.31% on the publicly available NUST-UHWR [13] dataset. The authors trained the Conv-Transformer architecture from scratch and thus the language model learning for the decoder of the transformer is restricted to text image datasets. The decoder can be pre-trained on an Urdu language modeling task on a large Urdu text dataset like Urdu News Dataset 1M [17].

3 Methodology

Taking our inspiration from [15,20] and addressing the lack of language modeling in current Urdu OCR systems, we propose a composition of CNN along with a transformer architecture as shown in Fig. 3.

In [20], the authors proposed TrOCR, which is a transformer architecture that uses vision transformers as encoders [22,26]. These vision transformers rely on heavy pretraining over synthetic text images before fine-tuning over the respective task of text recognition. The scalability of vision transformers over huge

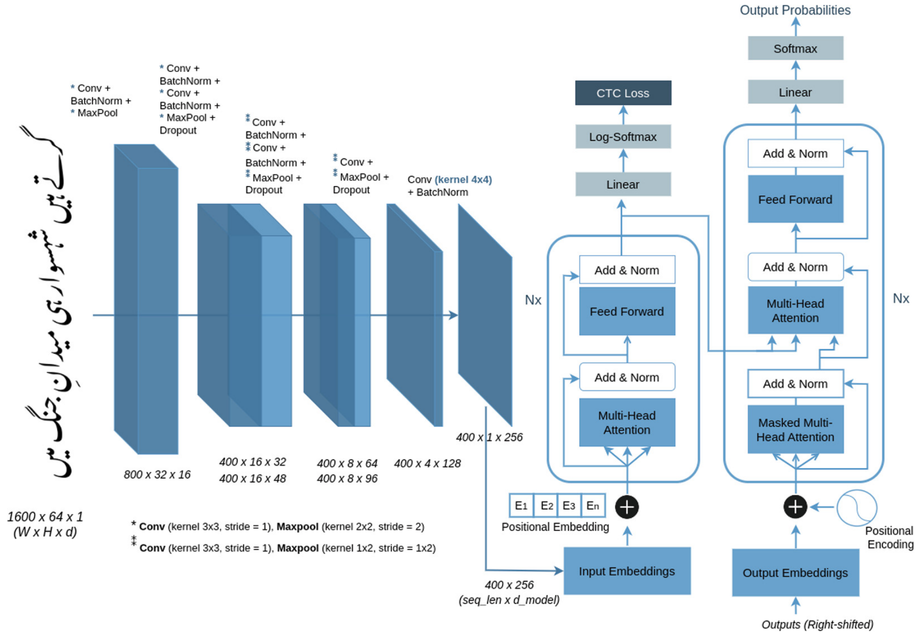


Fig. 3. Overview of the Proposed Architecture. The proposed architecture comprises a custom Convolutinal block coupled with a Transformer encoder for image understanding and a pre-trained Transformer decoder on Urdu language modeling.

datasets gives state-of-the-art results but on the other hand, they struggle in cases where data is not abundant and overfit very quickly lacking generalization. Moreover, due to the n^2 computational complexity of the transformers [19], the image is resized to reduce the spatial resolution leading to reduced training times but on the other hand information loss as well.

To address these issues, we propose a CNN block (as shown in Fig. 3) before the transformer which uses max pooling layers to reduce the spatial resolution of the feature maps. Essentially, the max-pooling layers capture the important features and eradicate the problem of information loss due to resizing of original images. Moreover, the presence of attention layers in the transformer encoder after the convolutional block is used to capture the global context of the text images, and then a transformer decoder is used for language understanding. We use character-level tokenization since the publicly available Urdu text image datasets are smaller in size.

3.1 Convolutional Encoder

We use simple convolutional blocks with max pooling layers to reduce the spatial resolution of feature maps and rely mostly on a transformer encoder for text image understanding. The configuration of CNN is shown in Fig. 3. We follow the standard practices of building convolution blocks using batch normalization and dropout layers. The dimension of the feature map after the convolution block is (W/4, 1, 256) where W is the original image width. This is reshaped to (W/4, 256) and fed into the transformer encoder. The height is reduced to 1 whereas the depth of the feature map is treated as embedding vectors. So we have a sequence length of W/4 and an embedding dimension of 256. These configurations of convolutional blocks give the best results for Urdu text recognition. The sparsity of connections in CNNs leads to better generalization when trained on smaller datasets of text images.

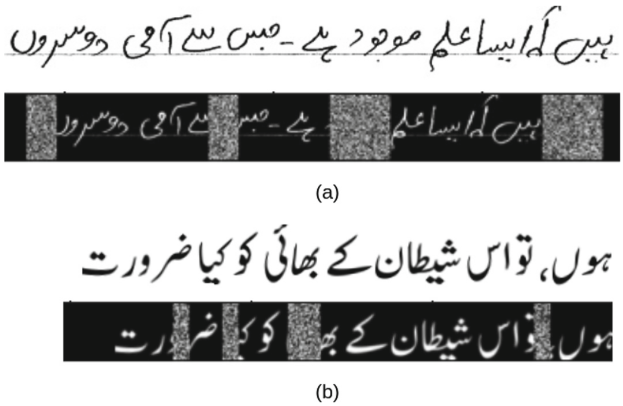
3.2 Transformer Encoder-Decoder

The features extracted from CNN are used as input to the transformer encoder. We use trainable vectors for positional embeddings that are fused with the feature maps at the transformer encoder end. The embeddings generated after the transformer encoder are fed to the decoder as keys and values for cross attention [21] and also through a linear layer that changes the embedding dimension to *vocab* size (refer to Fig. 3). Log Softmax probabilities are generated over the *vocab* dimension and then CTC loss is calculated using the ground truth during training. We use 3 stacks of transformer encoder layers with the embedding dimension of 256 and 8 attention heads for multi-head attention.

We use a pre-trained vanilla transformer decoder on the Urdu language modeling task over the 'Urdu News Corpus 1M Dataset' [17] with sinusoidal position encodings [21]. The configuration of the transformer decoder is the same as the transformer encoder. Softmax probabilities are generated over the vocab dimension and cross-entropy loss is calculated at the decoder end during training. The decoder is trained in a teacher-forcing manner and both the CTC loss and cross-entropy loss are used for backpropagation and training the model. The use of both CTC and cross-entropy Loss leads to effective training of the architecture.

3.3 Data Augmentations

To add diversity and enhance the generalization of the decoder, we use Tiling and Corruption (TACo) technique [24] for data augmentation. As per this technique, tiling is the process of dividing an input image into many, tiny, equal-sized tiles. As part of the corruption stage after tiling, a portion of the tiles is swapped out for corrupted ones. The enhanced image is then created by stitching the tiles back together in the same sequence. A sample instance of this process can be seen in Fig. 4).



(a)

(b)

Fig. 4. The figure shows samples of input images after applying the TACo augmentations. **(a)** shows a respective example from the NUST-UHWR handwriting dataset. **(b)** shows input sample from UPTI2.0 dataset for printed text.

4 Experimental Setup

Different experiments were carried out for Urdu Text Recognition to provide a comparison with the baselines and the current state-of-the-art. The details of the datasets, implementation and hyper-parameters are discussed below.

4.1 Datasets Used

To evaluate the performance of our proposed model, we utilized both printed and handwritten open-source datasets for the Urdu language. The different datasets used for experimentation in this study include:

Urdu News Dataset 1M [17]**:** This dataset provides a text corpus of more than 1 million Urdu news articles for four different subject areas: business and economics, science and technology, entertainment, and sports. The dataset is used for pretraining the decoder for URDU language modeling. We use this pre-trained decoder in our proposed architecture for all experiments.

UPTI2.0 [18]**:** The dataset was formed by collecting sample images from the web, new articles, and books. It contains around 120,000 unique text lines in four different scripts namely Alvi Nastaleeq, Jameel Nori Nastaleeq, Pak Nastaleeq, and Nafees Nastaleeq. This dataset is primarily used for training our architecture and showing generalization capabilities on other datasets like URTI.

URTI [31]**:** The dataset comprises text lines that have been scanned from printed and calligraphic Urdu magazines, newspapers, poems, and novels. There are 694 text lines from novels, 971 text lines from periodicals, 233 text lines from books, and 282 text lines from poetry. Due to the diversity of fonts in this dataset, it is not included in training and is kept only for testing and inference purposes. It serves well to test the generalization capability of our architecture.

NUST-UHWR [13]**:** This dataset is obtained from a variety of websites, including social networking and news websites; and contains a total of 10,606 samples of handwritten Urdu text recognition.

MMU-OCR-21 [6]**:** It is the largest collection of printed Urdu text. The corpus is made up of over 602,472 images in total, including ground truth for text-line and word images in three well-known scripts.

4.2 Implementation Details and Hyperparameters

We evaluate our model separately for printed and handwriting text recognition with state-of-the-art OCR systems at first and then propose a unified training approach to train a single model for inference on both tasks.

For printed text recognition, different configurations of printed text datasets for training and testing are used. At first UPTI 2.0 is utilized for training and URTI for inference. The same dataset splits are used as in [18]. This configuration tests the generalization of different state-of-the-art architectures with ours. Furthermore, our architecture is evaluated on datasets MMU-OCR-2021 with training and testing configurations as in [6]. For unconstrained offline Urdu handwriting recognition, we utilize the NUST-UHWR dataset. The same training and testing splits are employed as presented in [13,15].

Our proposed architecture gives state-of-the-art results for all configurations which inspire us to further test its generalization potential. For this purpose, we combine printed and handwritten text images from UPTI 2.0 and NUST-UHWR respectively for training the architecture. The training is converged against a validation set of equal printed and handwritten text samples from the same datasets. The model is tested against the URTI dataset for printed and NUST-UHWR testing split for handwriting text images. The results are presented and discussed in Sect. 5.

The proposed architecture is trained on RTX 3080 GPU with a batch size of 8. All the text images are resized to (1600 × 64), keeping the aspect ratio. 'GELU' [32] is used as an activation function throughout the architecture with dropout layers having a probability of 0.1 (refer to Fig. 3). We utilize AdamW [23] optimizer with a learning rate of 3×10^3 for updating the weights. During training, we calculate the CTC and cross-entropy loss on the transformer encoder and decoder output respectively. For inference, the decoder part of the transformer gives the best results after utilizing beam search compared to the transformer encoder.

Table 1. Comparison of CERs of various architectures on URTI [31] to show generalization after training on multi-font subset of UPTI2.0 [18].

Subset	Character Error Rate (CER%)				
	BDLSTM [18]	MDLSTM [3]	CLE Nastaliq [28]	TrOCR [20]	**Proposed Printed**
Magazine	50.69	47.44	50.3	18.4	**11.85**
Book	58.10	56.73	50.12	23.76	**19.23**
Poetry	59.40	58.38	64.55	22.47	**17.17**
Novel	57.70	58.99	38.65	28.75	**21.68**

Table 2. Comparison of CERs of CNN+LSTM based architectures with our proposed model on MMU-OCR-21 dataset [6].

Models	Character Error Rate (CER%)		
	train	val	test
CNN+BLSTM+CTC [6]	0.1	7.4	7.2
VGG-16+BLSTM+CTC [6]	35.5	49.0	49.0
Encoder-Decoder [27]	0.1	7.4	7.3
Proposed Printed	2.0	**6.6**	**6.7**

5 Results and Analysis

We perform extensive experimentation and provide results that establish a new state-of-the-art in Urdu printed and handwritten text recognition. To first test the generalization of our model in comparison with other baselines for printed text (referred as proposed printed), we carry out training on 80,000 images containing each script in equal proportion from UPTI 2.0 that cover 18,000 ligatures [18]. The training converged against a validation set of 10,000 images. Then we benchmarked our model against various other architectures on the URTI dataset.

The results are shown in Table 1. BDLSTM, MDLSTM, and CLE Nastaliq did not generalize well when tested on variations of scripts. The absence of CNN and language modeling in all these models leads to reducing the generalizability and yielding unsatisfactory results. TrOCR is trained from scratch and it quickly overfits during training. Transformers in general require heavy pretraining before fine-tuning over a specific task. This is quite evident from our results.

Our proposed printed model gives the best CER on different scripts compared to other architectures with a significant margin (as given in Table 1). The use of Convolution before the Transformer proves to work best for small datasets, improving generalization.

Next, we benchmark our printed model against various CNN-based architectures (refer to Table 2) on the MMU-OCR-21 dataset [6]. The CNN+BLSTM [6] and VGG16+BLSTM [6] use CTC loss for transcription without any language modeling. Encoder-Decoder [27] architecture comprises a CNN + LSTM encoder

Table 3. Comparison of CER between baselines and our proposed HWR model for handwritten text recognition on NUST-UHWR test split [13].

Models	(CER%)	
	Val	Test
BLSTM [25]	27.39	27.05
Modified CRNN [30]	18.57	19.34
MDLSTM [33]	14.11	19.15
CNN-RNN [29]	13.25	14.12
BiGRU [34]	13.50	13.28
TrOCR [20]	20.12	21.34
Conv. Recursive [13]	7.25	7.35
Conv. Transformer [15]	6.0	6.4
Proposed HWR	**5.9**	**6.2**

Table 4. Performance of our proposed model on URTI dataset [31] for printed text recognition and NUST-UHWR test split [13] for handwriting text recognition after unified training on single dataset.

Dataset		CER(%)
URTI dataset [31]	Magazine	8.11
	Book	13.32
	Poetry	12.39
	Novel	22.34
NUST-UHWR test split [13]		6.6

and an LSTM language modeling decoder. Our proposed printed architecture gives superior results on validation and testing sets in comparison. Higher CER on the training set is due to the TACo augmentations we use during training. The printed model achieves state of the art for Urdu printed text recognition.

For offline handwriting text recognition, we propose to use the same proposed architecture (referred as proposed HWR) using the NUST-UHWR [13] dataset. The same splits were used as in [13, 15]. The results are shown in Table 3. Our proposed HWR model performs superior in terms of CER compared to other architectures.

The proposed printed and HWR model gives better generalization when trained on small datasets for both Urdu printed and handwriting recognition. This encourages us to train this model on a unified dataset of printed and handwritten text to define a single model (namely proposed unified) for both tasks. We take the same 80,000 multi-font printed text images from UPTI 2.0 as described in previous experiments and append the UHWR train split of 8483 handwriting text images to create the unified dataset. We train our architecture

Table 5. CER of our proposed unified model on URTI keeping fixed samples from UPTI 2.0 and gradually increasing NUST UHWR samples on each stage for training.

URTI dataset [31]	Character Error Rate (CER%)		
	25% handwriting sample	50% handwriting sample	75% handwriting sample
Magazine	10.12	10.43	9.73
Book	16.73	14.24	13.87
Poetry	16.74	16.32	14.34
Novel	21.69	23.54	23.43

on this dataset and test it against the URTI and UHWR test split. The results are shown in Table 4. The unified training on a single architecture gives state-of-the-art results for both printed and handwriting recognition. A marginal decrease in CER over the URTI dataset can be seen. Additionally, the URTI dataset comprises text lines that have been scanned from machine-printed magazines, newspapers and novels, and calligraphic handwritten Urdu poems, this indicates that the training and testing data distributions are not entirely the same. This leads to significant variation in CERs of different splits of the URTI dataset.

We perform analysis on the unified architecture by keeping the printed text samples fixed and training it on fewer handwriting samples. The handwriting samples are gradually increased for each stage. Each stage is then tested over the URTI dataset as shown in Table 5. URTI consists of a diverse range of fonts including calligraphy. The results in the table show that with an increase in the handwriting dataset for training the model consistently performs better. The diversity of different handwriting styles aids the model in generalizing better to diverse sets of fonts in URTI.

We demonstrate that utilizing both handwriting and printed text for unified training leads to better results for both tasks despite the different distributions in URTI and diverse writing styles in UHWR, our architecture was able to generalize substantially better than current state-of-the-art Urdu OCR models. These findings suggest that a real-world application using such an architecture would produce correct results due to its superior generalization.

6 Conclusion

In this paper, we present an end-to-end Transformer based OCR model for text recognition that utilizes a CNN architecture to extract image embeddings and a pre-trained transformer decoder for language modeling. To the best of our knowledge, we are the first ones to propose a single end-to-end framework that recognizes both printed and handwritten Urdu text images. Vision transformers despite being the new advancement in computer vision suffer from overfitting when it comes to small datasets. They rely on computationally expensive pre-training over synthetic text images and fail if trained from scratch on datasets with moderate sizes. This effect is greatly reduced when we utilize the generalization capabilities of a CNN to extract image embeddings and then pass them through the transformer network to benefit from the attention mechanism. The hyperparameters of our model are tuned for efficient results on smaller datasets. The scalability of our model on larger datasets is yet to be tested which is proposed as a future direction.

References

1. Khan, N.H., Adnan, A.: Urdu optical character recognition systems: present contributions and future directions. IEEE Access **6**, 46019–46046 (2018)
2. Naz, S., Umar, A.I., Ahmed, R., Razzak, M.I., Rashid, S.F., Shafait, F.: Urdu Nasta'liq text recognition using implicit segmentation based on multi-dimensional long short term memory neural networks. In: SpringerPlus, 5(1), pp. 1–16 (2016)
3. Naz, S., et al.: Urdu nastaliq recognition using convolutional-recursive deep learning. Neurocomputing **243**, 80–87 (2017)
4. Naz, S., Hayat, K., Razzak, M.I., Anwar, M.W., Madani, S.A., Khan, S.U.: The optical character recognition of urdu-like cursive scripts. Pattern Recogn. **47**(3), 1229–1248 (2014)
5. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, 28 (2016)
6. Nasir, T., Malik, M.K., Shahzad, K.: MMU-OCR-21: towards end-to-end urdu text recognition using deep learning. IEEE Access **9**, 124945–124962 (2021)
7. Umair, M., et al.: A multi-layer holistic approach for cursive text recognition. Appl. Sci. **12**(24), 12652 (2022)
8. Hassan, S., Irfan, A., Mirza, A., Siddiqi, I.: Cursive handwritten text recognition using Bi-directional LSTMs: a case study on urdu handwriting. In: 2019 International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML), pp. 67–72. IEEE (2019)
9. Rehman, A., Ul-Hasan, A., Shafait, F.: High performance Urdu and Arabic video text recognition using convolutional recurrent neural networks. In: Barney Smith, E.H., Pal, U. (eds.) ICDAR 2021. LNCS, vol. 12916, pp. 336–352. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-86198-8_24
10. Rehman, A.U., Hussain, S.U.: Large scale font independent Urdu text recognition system. In: arXiv, preprint: arXiv:2005.06752 (2020)
11. Mushtaq, F., Misgar, M.M., Kumar, M., Khurana, S.S.: UrduDeepNet: offline handwritten Urdu character recognition using deep neural network. Neural Comput. Appl. **33**(22), 15229–15252 (2021)
12. Narwani, K., Lin, H., Pirbhulal, S., Hassan, M.: Towards AI-enabled approach for Urdu text recognition: a legacy for Urdu image apprehension. In: IEEE Access (2022)
13. Zia, N., Naeem, M.F., Raza, S.M.K., Khan, M.M., Ul-Hasan, A., Shafait, F.: A convolutional recursive deep architecture for unconstrained Urdu handwriting recognition. Neural Comput. Appl. **34**(2), 1635–1648 (2022)
14. Husnain, M., et al.: Recognition of Urdu handwritten characters using convolutional neural network. Appl. Sci. **9**(13), 2758 (2019)
15. Riaz, N., Arbab, H., Maqsood, A., Nasir, K.B., Ul-Hasan, A., Shafait, F.: Conv-transformer architecture for unconstrained Off-Line Urdu handwriting recognition. Int. J. Document Anal. Recogn. (IJDAR) **25**, 373–384 (2022)

16. Mirza, A., Zeshan, O., Atif, M., Siddiqi, I.: Detection and Recognition of Cursive Text from Video Frames. In: EURASIP J. Image Video Process. **2020**(1), 1–19 (2020)
17. Hussain, K., Mughal, N., Ali, I., Hassan, S., Daudpota, S.M.: Urdu News Dataset 1M. In: Mendeley Data, 3 (2021)
18. Naeem, M.F., Awan, A.A., Shafait, F., ul-Hasan, A.: Impact of ligature coverage on training practical Urdu OCR systems. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), vol. 1, pp. 131–136. IEEE (2017)
19. Riaz, N., Latif, S., Latif, R.: From transformers to reformers. In: 2021 International Conference on Digital Futures and Transformative Technologies (ICoDT2), pp. 1–6. IEEE (2021)
20. Li, M., et al.: TrOCR: transformer-based optical character recognition with pre-trained models. In: arXiv, preprint arXiv:2109.10282 (2021)
21. Vaswani, A., et al.: Attention is All You Need. In: Advances in Neural Information Processing Systems, 30 (2017)
22. Dosovitskiy, A., et al.: An Image is Worth 16x16 words: Transformers for Image Recognition at Scale. In: arXiv preprint arXiv:2010.11929 (2020)
23. Loshchilov, I., & Hutter, F.: Decoupled Weight Decay Regularization. In: arXiv preprint arXiv:1711.05101 (2017)
24. Chaudhary, K., Bali, R.: Easter2. 0: Improving convolutional models for handwritten text recognition. In: arXiv preprint arXiv:2205.14879 (2022)
25. Ul-Hasan, A., Ahmed, S.B., Rashid, F., Shafait, F., Breuel, T.M.: Offline printed Urdu nastaleeq script recognition with bidirectional LSTM networks. In: 2013 12th International Conference on Document Analysis and Recognition, pp. 1061–1065. IEEE (2013)
26. Bao, H., Dong, L., Piao, S., Wei, F.: Beit: BERT pre-training of Image Transformers. In: arXiv preprint arXiv:2106.08254 (2021)
27. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence-to-sequence learning with neural networks. In: Advances in Neural Information Processing Systems, vol. 27 (2014)
28. Hussain, S., Niazi, A., Anjum, U., Irfan, F.: Adapting tesseract for complex scripts: an example for Urdu nastalique. In: 2014 11th IAPR International Workshop on Document Analysis Systems, pp. 191–195. IEEE (2014)
29. Safarzadeh, V.M., Jafarzadeh, P.: Offline Persian handwriting recognition with CNN and RNN-CTC. In: 2020 25th International Computer Conference, Computer Society of Iran (CSICC), pp. 1–10. IEEE (2020)
30. Shi, B., Bai, X., Yao, C.: An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE Trans. Pattern Anal. Mach. Intell. **39**(11), 2298–2304 (2016)
31. Shafait, F., Keysers, D., Breuel, T.M.: Layout analysis of Urdu document images. In: 006 IEEE International Multitopic Conference, pp. 293–298. IEEE (2006)
32. Hendrycks, D., Gimpel, K.: Gaussian Error Linear Units (GELUs). In: arXiv preprint arXiv:1606.08415 (2016)
33. Graves, A., Schmidhuber, J.: Offline handwriting recognition with multidimensional recurrent neural networks. In: Advances in Neural Information Processing Systems, vol. 21 (2008)
34. Chen, L., Yan, R., Peng, L., Furuhata, A., Ding, X.: Multi-layer recurrent neural network based offline Arabic handwriting recognition. In: 2017 1st International Workshop on Arabic Script Analysis and Recognition (ASAR), pp. 6–10. IEEE (2017)