ICIC Express Letters
Part B: Applications
Volume **16**, Number **1**, January **2025**

ICIC International ©2025 ISSN 2185-2766
pp. **83–91**

MULTIMODAL FUSION OF CONVOLUTIONAL RECURRENT NEURAL NETWORK AND LANGUAGE MODELS FOR TEXT RECOGNITION

Praetawan Jarutan[1], Teerapong Hanyong[2], Zagon Bussabong[1]
Emmanuel Okafor[3] and Olarik Surinta[1,*]

[1]Multi-agent Intelligent Simulation Laboratory (MISL) Research Unit
Department of Information Technology
Faculty of Informatics
Mahasarakham University
Khamriang Sub-District, Kantarawichai District, Mahasarakham 44150, Thailand
{ 65011293002; 64011293003 }@msu.ac.th; *Corresponding author: olarik.s@msu.ac.th

[2]Betimes Solutions Co., Ltd.
84/4 Soi Sukhumvit 62, Sukhumvit Road, Bangchak Sub-District, Phrakanong District
Bangkok 10260, Thailand
teerapong.h@betimes.biz

[3]SDAIA-KFUPM Joint Research Center for Artificial Intelligence
King Fahd University of Petroleum and Minerals
Dhahran 31261, Saudi Arabia
emmanuel.okafor@kfupm.edu.sa

Received April 2024; accepted June 2024

Abstract. *Humans often encounter text in natural scenes daily, such as on traffic signs, billboards, and walls. From a computer vision perspective, two main learning paradigms (text detection and text recognition) are most commonly explored for localizing and predicting text in natural scenes. However, many traditional computer vision algorithms for text recognition in natural scenes struggle with prediction accuracy due to variations in font styles, colors, blurriness, and text distortion. To address these challenges, this paper proposes a text recognition architecture that employs a fusion of multimodal contexts (vision and language models) trained on a multi-language video subtitle dataset, aimed at recognizing text (English letters and Arabic numbers) from video frames (scene images). To achieve this goal, the vision model was developed using a convolutional recurrent neural network (CRNN) integrated with a connectionist temporal classification decoder for feature extraction and text prediction. The language model was created using a sequence-to-sequence model (bidirectional gated recurrent unit: BiGRU) that learns text sequence representations and produces readable text output. The resulting proposed fused modality, known as Fusion-based CRNN with Sequence-to-Sequence (Fusion-CRNN+Seq2Seq), is used for recognizing text from images. The proposed method outperforms all other approaches and achieves the lowest character error rates of 1.36 and 1.22 based on different BiGRU network configurations.*

Keywords: Convolutional recurrent neural networks, Connectionist temporal classification, Vision model, Language model, Sequence-to-sequence, Character error rate, Text recognition

1. **Introduction.** In recent years, text recognition has gained significant interest from academics and industries due to its high performance [1]. Although optical character recognition (OCR) systems have demonstrated high performance, they are primarily effective with printed text. Current OCR systems might cause damage to the text from natural scenes because the existing OCR system is appropriate for text in documents [2].

DOI: 10.24507/icicelb.16.01.83

In computer vision tasks, recognizing the text from natural scenes is challenging due to the diversity of font types, sizes, styles, colors, blurriness, distortion, and text blending with the background. The performance of the text recognition framework could be negatively impacted by these factors [3,4]. A broader aspect of this article will investigate the applicability of deep learning technology for text recognition (reading text from images).

Recurrent convolutional neural networks (RCNNs) and search algorithms, such as connectionist temporal classification (CTC) and beam search, have been proposed to address the problem of long text sequences in modeling of many languages [5]. The CRNN model is not effective for processing text images that are highly diverse in font sizes and styles, irregular, and have low-quality images [6]. Nonetheless, the output of the existing methods is not entirely accurate. Language models are now frequently utilized in natural language processing (NLP) research to analyze language patterns for predicting words, summarizing text, and correcting spelling [7-9]. Hence, applying language models, such as spelling correction, can enhance text recognition performance. Spelling correction can provide automatic detection and correction of text errors [10].

Researchers have proposed a CRNN framework for text recognition. Singkhornart and Surinta [11] combined the VGG19 architecture with a long short-term memory (LSTM) network (CNN-LSTM) to recognize multi-language text from video subtitles. The framework resized input images from 224×224 to 32×379 pixels to fit subtitle images, modifying the input resolution of VGG19 while retaining its feature map layers. It utilized two bidirectional LSTM (BiLSTM) layers before the dense layer. A softmax function and a CTC loss function were applied to generating text output. [11] reported a character error rate (CER) of 9.36 on the video subtitle dataset by integrating VGG19 and BiLSTM in the CNN-LSTM model. Moreover, Gonwirat et al. [5] proposed a Fusion-CRNN model by merging two CNN architectures with recurrent neural networks. Their approach ensures the feature map sizes match both CNNs by using additional operations. Input images were resized to 64×755 pixels to accommodate increased information in the input layer. The softmax activation at the terminal layer computes output probabilities, which are then processed by the word beam search (WBS) algorithm to find the optimal path. The Fusion-CRNN achieved a character error rate (CER) of 5.29 on the video subtitle test set. However, when decoded using the CTC algorithm, the Fusion-CRNN framework yielded a CER of 5.33.

Zhuang et al. [2] combined vision and language models through a text-level strategy paradigm called the TextCLR architecture. This model integrates vision and language feature maps using a gated mechanism to generate text output. The vision model employs residual network (ResNet) and transformer architectures to extract informative visual features. Additionally, position attention modules compute features into character probabilities, while a bidirectional cloze network (BCN) is used for language processing. The effectiveness of the TextCLR model was demonstrated using synthetic datasets. Evaluations on four datasets – ICDAR2013 (IC13), ICDAR2015 (IC15), Street View Text (SVT), and Street View Text Perspective (SVTP) – showed recognition accuracies of 97.7%, 85.0%, 93.8%, and 89.3%, respectively. Also, Lertpiya et al. [10] proposed a two-stage error correction system to detect and correct spelling errors in the Thai language. Their system uses a neural-based method to rectify Thai spelling errors. In the error detection stage, a BiLSTM and binary sequence tagger, comprising embedded word and character layers, identify error segments in the text. The error correction stage employs a sequence-to-sequence (Seq2Seq) neural network to correct the errors. This approach significantly reduced the word error rate (WER) from 2.51 to 2.07, demonstrating improved accuracy in Thai text.

We have discussed the limitations of the related work. Singkhornart and Surinta [11] faced resolution issues due to image resizing, which can lead to information loss and reduced text recognition accuracy. Gonwirat et al. [5] encountered increased computational time and memory usage due to resizing, the complexity of integrating two CNNs with RNNs, and variability in character error rates depending on the decoding methods. Zhuang et al. [2] experienced model complexity from integrating multiple advanced components and potential performance variability with real-world data, as their results are based on synthetic datasets. Lertpiya et al. [10] presented a Thai language-specific error correction system that may not generalize well to other languages and could be limited by the effectiveness of its error detection and correction processes.

Given these limitations, we integrate Seq2Seq models into the multimodal fusion of CRNNs to enhance text recognition performance. This approach improves contextual understanding, allows for dynamic output generation, facilitates effective error correction, and ensures integrated end-to-end learning. Consequently, this integration is expected to yield more accurate and contextually relevant results.

The contribution of this paper is to investigate the framework that enhances the performance of text recognition tasks from text images. The proposed framework includes two main models: the vision and language models. In the vision model, we first propose the Fusion-CRNN model to extract sequential visual features and employ the CTC encoder to find the best text output. Second, we utilize the Seq2Seq language model to spell the text when the text output is inaccurate. The proposed method (Fusion-CRNN+Seq2Seq) outperforms all other approaches and yielded the least character error rate (CER) of 1.36 and 1.22 based on different BiGRU network nodes when evaluated on the subset of the video subtitle dataset.

This paper is organized in the following way. Section 2 presents the proposed Fusion-CRNN+Seq2Seq architecture that factored a bidirectional gated recurrent unit (BiGRU). Section 3 presents the experimental setup and results obtained. The conclusion and areas for future work are described in Section 4.

2. Multimodal Fusion of CRNN and Language Models Framework.

We propose a text recognition framework integrating two modalities (vision and language models) to create a robust system for extracting text from images. The primary goal of this framework is to build a robust text recognition system that addresses the challenge of extracting text from images exhibiting any of the following properties: variations in font styles, colors, blurriness, and distortion of the text embedded in signs. To achieve this goal, given a text image as the input, the vision model is employed to extract deep features and learn the feature representation before computing output probability distributions. Furthermore, the language model relies on the output probabilities from the vision model, which serve as input to the language learning algorithm. The end goal of the language model is to learn rich language feature representations and produce readable text output. The proposed text recognition framework is shown in Figure 1.

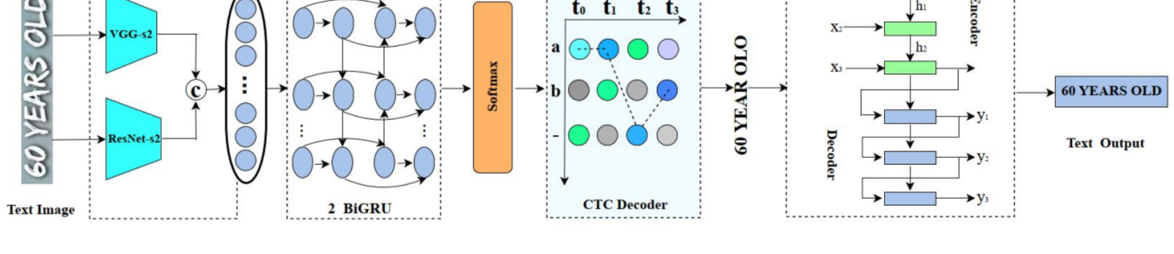


Figure 1. Illustration of the fusion convolutional recurrent neural network and language model framework for text recognition

2.1. **Vision model.** For the vision model, the text images are directly fed into VGG and ResNet models, which combine the robust deep features using a concatenation operation. These combined features are then sent to the BiGRU network to generate the output probabilities before encoding the text output using the CTC decoder. In this context, t_0, t_1, t_2 and t_3 represent the individual time steps in the sequence of output probabilities generated by the vision model. At each time step, the vision model produces a probability distribution over the possible characters in the text sequence.

We briefly explain the vision model in the proposed framework as follows. The vision model is the combination of the Fusion-CRNN architecture and CTC encoder. For this, text images were passed as input to two CNN architectures (VGG-s2 and ResNet-s2) which are the backbone of our method and were used for performing feature extractions. Note that the fully-connected and softmax layers from the original VGG [12] and ResNet [13] were removed from the respective CNN architectures. Hence, the resulting CNN architectures are fused with an additional operation. Second, we added two bidirectional gated recurrent unit (BiGRU) layers and a softmax function to the architecture to generate the output probabilities. Finally, the CTC encoder was connected to the architecture to learn the output probabilities and generate the optimal text output.

VGG-s2 and ResNet-s2. The VGG-s2 and ResNet-s2 earlier reported by Gonwirat et al. [5] are the backbone feature extractions from the text images. In this experiment, the input layer was transformed to a spatial resolution of 64 × 755 pixels to accommodate the text images existing in rectangular shapes. For the VGG-s2, we modified the VGG-s2 based on VGG19 architecture. The VGG-s2 architecture has a consistent number of weight layers in each block and feature maps in each weight layer when compared to the VGG19 architecture. Furthermore, only the stride parameter was set as two (s2). For ResNet-s2, we also created ResNet-s2 based on the ResNet50 architecture. We reduced the convolutional blocks from five to three blocks, resulting in 23 weight layers. Moreover, we adjusted the stride parameter to two strides. We employed additional operations: a type of fusion strategy that merges two or more CNN architectures while removing the last layer of CNN architecture and combining the terminal feature maps of the two CNN architectures to create new feature maps [14], while factoring in the generation of the feature map of the exact dimensions.

Recurrent neural networks (RNN). We provide a brief introduction to two RNN networks (BiLSTM and BiGRU) that were used in the experiments. The BiLSTM and BiGRU are RNN architectures that address the weakness of the original RNN network by learning long-term sequence patterns or long-term dependencies in given sequential data [5,11]. For the BiLSTM, the advantage of using a BiLSTM network is its ability to effectively learn and memorize a growing number of long-term sequence patterns. The BiLSTM consists of a memory cell for retentive capability and three gating mechanisms: an input gate, an output gate, and a forget gate. These gates are responsible for retaining necessary sequence patterns and eliminating unnecessary ones. In contrast, the BiGRU has a simpler structure, comprising only two gating mechanisms: the update gate and the reset gate. This simpler structure allows the BiGRU to process information faster, use fewer resources, and effectively capture past and future information.

Connectionist temporal classification (CTC). The CTC encoder or the CTC loss function is proposed to identify the potential alignment path that represents the optimal output sequence of the text output [15,16]. The likelihood of a specific alignment path occurring at each time step within the given path is computed. Consequently, to calculate the alignment path, the CTC loss function uses two inputs, the matrix of output probabilities from the RCNN network and the corresponding ground truth text. Hence, the potential alignment paths are categorized into accurate sequence text output.

2.2. **Language model.** A language model is a method used to determine the output probabilities of sequence patterns of characters or words occurring in text sentences. Many language models have recently been proposed in natural language processing (NLP) for spelling correction, machine translation, and chatbots. Some well-known techniques include permutation-based learning, discrete learning techniques, and bidirectional encoder representation from transformers (BERT) [17-19].

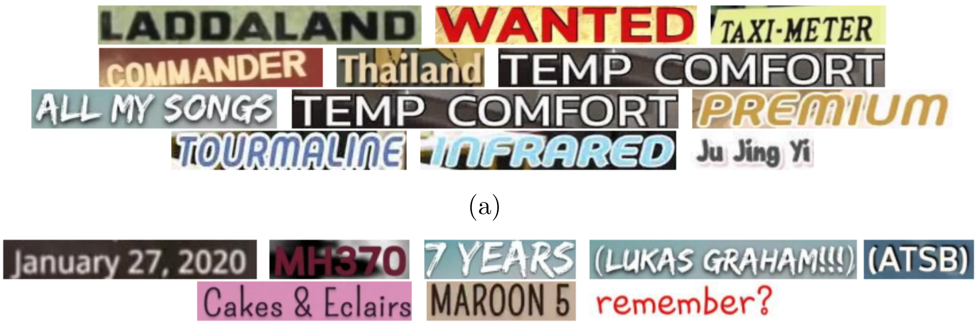
Sequence-to-Sequence. We integrated the sequence-to-sequence (Seq2Seq) language model proposed by Sutskever et al. [20] with the vision model. This research aims to correct spelling errors in the text generated from the vision model. In the Seq2Seq model, the LSTM maps sequence patterns to a fixed-dimension vector. Hence, an extra LSTM block decodes the sequence patterns. To do this, a beam search decoder was proposed to consider every word in the corpus to generate the best text output.

In the encoder, the variables x_1, x_2, and x_3 represent the input sequence to the Seq2Seq model. These variables are the individual tokens (characters) from the text sequence decoded by the CTC decoder from the vision model. For example, if the text sequence is "60 YEAR OLO", x_1, x_2, and x_3 correspond to the tokens "Y", "E", and "A", respectively. Meanwhile, h_1 and h_2 are the hidden states in the Seq2Seq model. The hidden states capture the context and dependencies of the sequence. Furthermore, y_1, y_2, and y_3 represent the output sequence generated by the model. The decoder uses the context vector produced by the encoder to generate these output tokens.

3. Experimental Setup and Results.

3.1. **Multi-language video subtitle dataset.** Multi-language videos contained English, Thai, Arabic numerals, and Thai numerals, collected from the subtitles of the published videos on social media platforms such as Facebook and YouTube [1]. The total numbers of characters were 157 characters, consisting of 44 Thai consonants, 19 Thai vowels, 4 Thai tone marks, 3 Thai punctuation marks, 10 Thai numbers, 10 Arabic numbers, 26 Roman letters, 26 Roman capital letters, and 15 special signs. The challenge of the multi-language video subtitle dataset is that the dataset consists of texts with varying font styles, sizes, and lengths. The subtitle images are shown in different colors, with some having a single color and others having multiple colors. The text in the subtitle image can include contrast text with a complex background, or blurred text. Some texts have edge characteristics, such as narrow and wide borders and surface textures. Furthermore, text images have varying distances between characters, some characters with equal spacing and others with no gaps. In this study, we explored a subset of the described dataset (886 subtitle images with text content consisting of English letters, Arabic numerals, and special symbols). Sample images of the dataset used are shown in Figure 2.

3.2. **Experimental setup.** We implemented the framework of the fusion convolutional recurrent neural network (CRNN) and language model based on the TensorFlow deep learning framework. Consequently, all the experiments were trained and evaluated using the Google Colab platform. Further, we implemented the fusion CRNN architecture based on Gonwirat et al. [5], which used the VGG-s2 and ResNet-s2 architectures. Our experiments involved adjusting the RNN networks, using both long short-term memory (LSTM) and gated recurrent unit (GRU), and adjusting the sizes of the RNN to either 128 or 256 using the equation of 2^n. In addition, the probabilities were decoded into text output using a connectionist temporal classification (CTC) encoder. In this research, the fusion CRNN models were fine-tuned using the following hyperparameters: optimizer = Adam, batch size = 32, epoch = 100. We trained the fusion CRNN with 100 epochs on the multi-language video subtitle dataset comprising 886 subtitle images. Hence, the subset was split into three sets with a ratio of $70 : 10 : 20$.



(a)

(b)

Figure 2. An illustration of the subset of the multi-language video subtitle dataset that contains (a) English letters and (b) Arabic numerals and special symbols

3.3. **Evaluation metrics.** We use the character error rate (CER) value to evaluate the proposed framework. The CER value was calculated by dividing the sum of insertions (I), substitutions (S), and deletions (D) by the total number of characters (N). The equation expressing the CER value is expressed as $CER = (I + S + D)/N$.

3.4. **Experiments on CRNN network and sequence-to-sequence language model.** In this experiment, we compared two text recognition networks: Fusion-CRNN and Fusion-CRNN+Seq2Seq, in which the Seq2Seq is the sequence-to-sequence language model. First, we use a combination of two CNN architectures: VGG-s2 and ResNet-s2, in the Fusion-CRNN model based on the Fusion-CRNN model proposed by Gonwirat et al. [5]. Second, we experimented on the RNN network using two RNN types: LSTM and GRU, with sizes of 128 and 256, respectively. Consequently, we evaluated the effectiveness of the Fusion-CRNN and Fusion-CRNN+Seq2Seq networks by measuring the CER value. The comparative results on the subset of the multi-language video subtitle dataset of the text recognition networks are presented in Table 1.

Table 1. The comparison results of the fusion CRNN network and sequence-to-sequence language model

Fusion CRNN network and language model	**Two RNN layers**	**Training time (Min.)**		**CER value (↓) of RNN sizes**	
		RNN size			
		128	**256**	**128**	**256**
Fusion-CRNN	BiLSTM	08:31	15:32	1.90	1.94
Fusion-CRNN+Seq2Seq	BiLSTM	08:33	15:35	1.44	1.49
Fusion-CRNN	BiGRU	09:03	16:01	1.86	1.66
Fusion-CRNN+Seq2Seq	BiGRU	09:04	16:02	**1.36**	**1.22**

*Note that we applied the CTC algorithm after the Fusion-CRNN network to decoding the text.

As shown in Table 1, we analyzed the CRNN network by examining the fusion of VGG-s2 and ResNet-s2 using the additional operation. The experimental results showed that the Fusion-CRNN+Seq2Seq network performed better than only using the Fusion-CRNN network for text recognition. Furthermore, the GRU network outperformed the LSTM network in both the Fusion-CRNN and Fusion-CRNN+Seq2Seq models. Specifically, the Fusion-CRNN+Seq2Seq model utilizing a GRU layer consisting of 256 nodes gained the lowest error rate and achieved a CER value of 1.22. In comparison, the GRU layer composed of 128 nodes achieved a CER value of 1.36. Furthermore, we compared the training

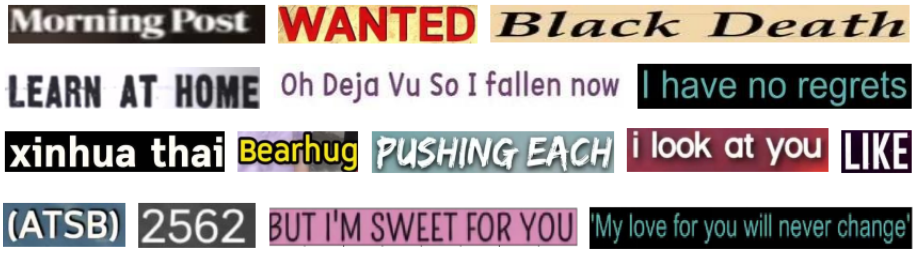


Figure 3. The accurate recognition that achieved a CER value of 0 on the example text images

Table 2. The performance of the Fusion-CRNN and Fusion-CRNN+Seq2Seq models using the CTC decoder in terms of the CER value

#	Input images and labels	Text recognition models and CER values (↓)			
		Fusion-CRNN	**CER**	**Fusion-CRNN +Seq2Seq**	**CER**
1	 *Label:* BUT I'M SWEET FOR YOU	BUT I'M SWEET FOR YOU 16	1.25	But I'M SWEET FOR YOU 6	1.17
2	 *Label:* WHEN LIFE WAS LONELY	WHEN LIFE WAS LONELYIS1	2.06	WHEN LIFE WAS LONELY?	1.12
3	 *Label:* My love for you will never change	'My loe flor you wl never ohange s	4.57	'My love for you will never change'	0
4	 *Label:* yeah when my world is falling apart	yeah when my wod s flalng apats	3.46	Yeah, when my world is floating apart	3.00
5	 *Label:* ONLY TO BURY THEM UNDER THE WEEPING WILLOW OF SEPARATION	WN P PMU THEI MNSE THE WhNG WLGoN PF S MNNiS!NS6	52.0	WN P. PMU THEI MNSE THE WNG WGN PF S	51.8

*Note that we identified any incorrectly recognized text in highlighting light gray and underlining.

time of Fusion-CRNN to Fusion-CRNN+Seq2Seq. We discovered that the addition of the language model did not cause a significant increase in training time. We demonstrated that the proposed Fusion-CRNN+Seq2Seq model successfully achieved a CER value of 0, indicating accurate recognition, as illustrated in Figure 3.

4. **Conclusions.** We propose a text recognition framework that fuses vision and language models. The proposed method, a fusion-based convolutional recurrent neural network (Fusion-CRNN), combines the vision model with a sequence-to-sequence language model, referred to as Fusion-CRNN+Seq2Seq. This framework addresses the challenge of recognizing text in natural scenes characterized by diverse font styles, colors, blurriness, and distortions. The Fusion-CRNN+Seq2Seq framework is divided into two parts. In the first part, the Fusion-CRNN fuses VGG-s2 and ResNet-s2 using an additional operation that enhances feature extraction performance, which is then transferred to the RNN network to generate output probabilities. This study also investigated the impact of a connectionist temporal classification (CTC) decoder for performing the classification to produce the best text output. In the second part, the sequence-to-sequence (Seq2Seq) language model is implemented to transform the sequence of text output generated by

the CTC decoder into readable text. The Seq2Seq model comprises two networks, the encoder and decoder components, which are based on the long short-term memory (LSTM) network. The Fusion-CRNN, which uses the gated recurrent unit (GRU), achieved a character error rate (CER) of 1.66. However, when the proposed Fusion-CRNN+Seq2Seq was trained using the GRU network, the CER value improved to 1.22.

For future work, we plan to implement a 3D-CNN architecture to replace current CNN architectures. The 3D-CNN model will enhance the extraction of sequential visual features and recognize text images more effectively. Our plan also includes incorporating visual attention mechanisms [21] and transformer architectures [7]. We will use the word error rate (WER) for evaluating the text recognition framework. Additionally, we are considering implementing other language models, such as few-shot learning with generative pretrained transformers (GPT-3) [22] and XLNet for language understanding [23].

Acknowledgments. This research project was financially supported by Mahasarakham University.

REFERENCES

[1] B. Shi, M. Yang, X. Wang, P. Lyu, C. Yao and X. Bai, ASTER: An attentional scene text recognizer with flexible rectification, *IEEE Trans. Pattern Anal. Mach. Intelligence*, vol.41, pp.2035-2048, 2019.

[2] J. Zhuang, Y. Ren, X. Li and Z. Liang, Text level contrastive learning for scene text recognition, *International Conference on Asian Language Processing (IALP)*, pp.231-236, 2022.

[3] Z. Wan, F. Xie, Y. Liu, X. Bai and C. Yao, 2D-CTC for scene text recognition, *arXiv Preprint*, arXiv: 1907.09705, 2019.

[4] N. Lu, W. Yu, X. Qi, Y. Chen, P. Gong, R. Xiao and X. Bai, Master: Multi-aspect non-local network for scene text recognition, *Pattern Recognition*, vol.117, 2021.

[5] S. Gonwirat, O. Surinta and P. Pawara, Fusion convolutional recurrent neural networks for Thai and English video subtitle recognition, *ICIC Express Letters*, vol.16, no.12, pp.1331-1339, 2022.

[6] X. Zhang, B. Zhu, X. Yao, Q. Sun, R. Li and B. Yu, Context-based contrastive learning for scene text recognition, *The 36th AAAI Conference on Artificial Intelligence*, pp.3353-3361, 2022.

[7] P. Selvam, J. A. S. Koilraj, C. A. T. Romero, M. Alharbi, A. Mehbodniya, J. L. Webber and S. Sengan, A transformer-based framework for scene text recognition, *IEEE Access*, vol.10, pp.100895-100910, 2022.

[8] S. Fang, H. Xie, Y. Wang, Z. Mao and Y. Zhang, Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition, *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.7098-7107, 2021.

[9] Y. Wang, H. Xie, S. Fang, M. Xing, J. Wang, S. Zhu and Y. Zhang, PETR: Rethinking the capability of transformer-based language model in scene text recognition, *IEEE Transactions on Image Processing*, vol.31, pp.5585-5598, 2022.

[10] A. Lertpiya, T. Chalothorn and E. Chuangsuwanich, Thai spelling correction and word normalization on social text using a two-stage pipeline with neural contextual attention, *IEEE Access*, vol.8, pp.133403-133419, 2020.

[11] T. Singkhornart and O. Surinta, Multi-language video subtitle recognition with convolutional neural network and long short-term memory networks, *ICIC Express Letters*, vol.16, no.6, pp.647-655, 2022.

[12] K. Simonyan and A. Zisserman, Very deep convolutional networks for large-scale image recognition, *The 3rd International Conference on Learning Representations (ICLR)*, pp.1-14, 2015.

[13] K. He, X. Zhang, S. Ren and J. Sun, Deep residual learning for image recognition, *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.770-778, 2016.

[14] L. Alzubaidi et al., Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions, *Journal of Big Data*, vol.8, no.1, 2021.

[15] A. Graves, S. Fernández, F. Gomez and J. Schmidhuber, Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks, *The 23rd International Conference on Machine Learning (ICML)*, pp.369-376, 2006.

[16] W. Hu, X. Cai, J. Hou, S. Yi and Z. Lin, GTC: Guided training of CTC towards efficient and accurate scene text recognition, *AAAI Conference on Artificial Intelligence*, pp.11005-11012, 2020.

[17] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li and P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, *Journal of Machine Learning Research*, 2019.

[18] N. B. Shah, S. Balakrishnan and M. J. Wainwright, A permutation-based model for crowd labeling: Optimal estimation and robustness, *IEEE Transactions on Information Theory*, vol.67, no.6, pp.4162-4184, 2016.

[19] R. Cartuyvels, G. Spinks and M. F. Moens, Discrete and continuous representations and processing in deep learning: Looking forward, *AI Open*, pp.143-159, 2021.

[20] I. Sutskever, O. Vinyals and Q. V. Le, Sequence to sequence learning with neural networks, *The 27th International Conference on Neural Information Processing Systems*, vol.2, 2014.

[21] W. Song and G. Zhang, Risky-driving-image recognition based on visual attention mechanism and deep learning, *Sensors*, vol.22, no.15, 2022.

[22] X. V. Lin et al., Few-shot learning with multilingual generative language models, *Conference on Empirical Methods in Natural Language Processing*, pp.9019-9052, 2022.

[23] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov and Q. V. Le, XLNet: Generalized autoregressive pretraining for language understanding, *The 33rd Conference on Neural Information Processing Systems*, 2019.