ICIC Express Letters
Volume **16**, Number **12**, December **2022**
 ISSN 1881-803X
pp. **1331–1339**

FUSION CONVOLUTIONAL RECURRENT NEURAL NETWORKS FOR THAI AND ENGLISH VIDEO SUBTITLE RECOGNITION

Sarayut Gonwirat[1], Olarik Surinta[2] and Pornntiwa Pawara[3,*]

[1]Department of Computer and Automation Engineering
Faculty of Engineering and Industrial Technology
Kalasin University
Kalasin Sub-District, Muang District, Kalasin 46000, Thailand
sarayut.go@ksu.ac.th

[2]Multi-agent Intelligent Simulation Laboratory (MISL)
Department of Information Technology, Faculty of Informatics
[3]POLAR Lab
Department of Computer Science, Faculty of Informatics
Mahasarakham University
Khamriang Sub-District, Kantarawichai District, Mahasarakham 44150, Thailand
olarik.s@msu.ac.th; *Corresponding author: pornntiwa.p@msu.ac.th

Received April 2022; accepted June 2022

Abstract. *Presently, subtitles are embedded into videos and placed on their bottom line. Locating the subtitle area and recognizing the text in the image is not simple. In this paper, we propose using the fusion convolutional recurrent neural network (CRNN) to recognize multi-language (Thai and English) from the subtitle word images. We fused the state-of-the-art convolutional neural networks (CNNs) with additional fusion operation, followed by the bidirectional long short-term memory (BiLSTM) network. For decoding the output from the text images, we compared two decoding algorithms consisting of connectionist temporal classification (CTC) and word beam search (WBS). We discovered that the WBS outperformed the CTC algorithms in accuracy performance. However, the WBS algorithm computed relatively slowly and is not suggested for application in real-time application. We evaluated our fusion CRNN architecture on the multi-language video subtitle dataset and achieved the CER value of 5.29% and 5.33% when decoding with WBS and CTC algorithms, respectively.*
Keywords: Fusion strategy, Recurrent convolutional neural networks, Word beam search, Connectionist temporal classification, Video subtitle recognition

1. **Introduction.** Text detection and recognition is a sequence to sequence (seq2seq) modeling problem. It has been widely known that recurrent neural networks (RNNs), such as long short-term memory (LSTM), are well suited for processing text recognition task. This is because LSTM has feedback connections that are effectively distinguishing and memorizing required information during training sequential data. Many researchers [1,2] have used LSTM to cope with text recognition in various languages such as English, Chinese, Persian, and Thai.

Yan and Xu [2] combined several techniques to improve Chinese and English subtitle recognition on two benchmark datasets: ICDAR2003 and ICDAR2013. They proposed using connectionist text proposal network for subtitle detection in video, whereas the combination of ResNets, bidirectional gated recurrent unit (BiGRU), and connectionist temporal classification (CTC) loss was applied for feature sequence representative and recognizing the subtitles in the videos. The proposed method also works effectively on

DOI: 10.24507/icicel.16.12.1331

real video transcription. However, this is not the case on the street view text dataset which has heavy text distortion.

Dutta et al. [3] used hybrid convolutional neural network (CNN) and RNN architectures along with residual convolutional blocks, bidirectional LSTM (BiLSTM), and CTC loss and pre-training the network on synthetic data for improving handwritten recognition on off-line images. Kantipudi et al. [4] integrated BiLSTM and CNNs as their feature extractors on the scene image datasets. They also applied contour detection technique to the image to speed up the text detection process and employed CNNs to locate and predict each character. Carbune et al. [5] customized BiLSTM and CTC loss and combined Bézier curves for the input encoder to speed up recognition time.

In [6], the extended version of RNN encoder-decoder network was employed for handling long sequences of text transcription. Instead of detecting the whole words, the model was trained to detect sequences of characters, where the softmax function was more precise for decoding than the sigmoid function. Lei et al. [7] proposed hybrid encoder-decoder networks for solving text recognition in scene images, where various depths of CNN architectures were meant to encode the input images and RNN was applied to decoding the corresponding sequences of characters.

The hybrid RNN+CTC approach also works effectively on Thai character recognition. Chamchong et al. [8,9] focused on recognizing ancient Thai handwritten from Thai archive manuscripts using text block-based technique where convolutional and recurrent layers are used for feature extraction and subsequently combined to CTC loss. Typical data augmentation techniques were also introduced during the training process owing to a limited number of samples.

The main contribution of this research is a fusion strategy of CRNN, as shown in Figure 1, which combines the last convolutional layers from two CNNs using additional fusion operation, followed by the BiLSTM network and decoding algorithm. The fusion CRNN network is proposed to recognize Thai and English languages (totally 157 characters) from the subtitle word images. The results show that the fusion CRNN architecture (VGG-s1+VGG-s2) outperformed the CRNN architectures and achieved the best CER value of 5.29% and 5.33% when using WBS and CTC decoding algorithms, respectively. We suggest that the proposed fusion CRNN architecture recognizes text despite diverse font styles and blurry and noisy images.

Outline of the paper. Section 2 presents the fusion convolutional recurrent neural networks. Section 3 presents the dataset, implementation detail, evaluation metrics, and experimental results. Finally, we conclude and recommend future directions in Section 4.

2. **Fusion Convolutional Recurrent Neural Networks.**

The CNNs have advantages in extracting deep features from the input image and classifying it in one algorithm. First, the CNN considers extracting spatial features from all the input image regions. Second, the fully-connected layers and softmax function are meant to recognize the input image. However, the CNNs were designed to be suitable for recognizing only one class.

For the subtitle word recognition, the output contains many classes. We need to send the deep spatial features directly to the RNN to learn and recognize the sequence of the patterns. Consequently, the outputs of the RNN are sent to the decoding algorithm to create the recognition output.

According to the advantages of CNN and RNN, we then proposed fusion CRNN to address the challenge of video subtitle recognition. The architecture of the proposed fusion CRNN is illustrated in Figure 1 and described in the following section.

2.1. **Convolutional neural networks.**

This section describes two types of CNNs, including sequential CNN and residual CNN.

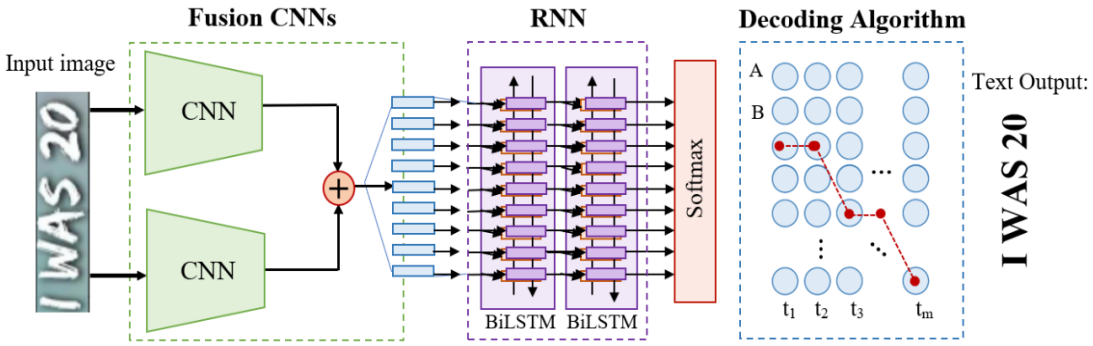

FIGURE 1. Illustration of the proposed fusion CRNN architecture

VGGNets. Simonyan and Zisserman [10] proposed VGGNets for large-scale image recognition. It is a sequential CNN architecture that contains many convolutional layers. The well-known VGGNets are the VGG16 and VGG19. Further, VGG16 contains five blocks of convolutional layers with 16 layers in total. The max-pooling is attached at the end of each block. The feature map size in each block is resized and divided by two. For example, the feature map size of the first block is 224×224 pixels resolution, while the feature map of the second block is 112×112 pixels. In our experiments, we modified VGGNets based on VGG16 by using only four blocks containing only ten convolutional layers. The stride parameter of one is used, which is called VGG-s1. Likewise, in VGG-s2 the stride parameter is two.

ResNets. The ResNets architectures have very deep convolutional layers (i.e., 18, 34, 50, 101, 152 layers) [11]. Due to the deep layers, the skip connection was introduced to connect the current layer with two or three subsequent layers when the size of the current and subsequent layers is equal. This could reduce the computational time when compared with the sequence CNN. The ResNets architectures usually contain five blocks of convolutional layers. In this experiment, however, we modified the network based on the ResNet50 architecture that used only three convolutional blocks with 23 layers in total. Additionally, we used different stride values of 1 and 2, called ResNet-s1 and ResNet-s2, respectively.

2.2. **Recurrent neural networks.** In this section, we briefly present two types of RNNs: BiLSTM and BiGRU.

BiLSTM. The advantage of the LSTM is that it was proposed to face the challenge when training the model with long sequence data. Three gates: input, output, and forget gate, are included in the network to deal with long sequence data that could keep or delete unnecessary data. The BiLSTM was invented using two LSTM architectures to learn the sequence data from forward and backward directions [12].

BiGRU. The GRU network is a light version of the LSTM network that contains only reset and update gates [12]. The reset gate is used to decide how the previous and current information is integrated. The update gate is also used to decide how much the network keeps the previous information in training. Furthermore, BiGRU consists of forward and backward GRUs [13], the same as the BiLSTM.

2.3. **Fusion strategies.** The fusion strategy is an algorithm that fuses deep features of two or more different CNN architectures to exceptional deep features [14,15]. In the first stage, the fully-connected and softmax layers are removed from the CNN architectures. In the second stage, we fused the last layer of two CNN architectures using two operations: addition and concatenation.

Additional Operation. After removing the last layer of each CNN architecture, it should ensure that the feature map sizes of the first CNN and second CNN are identical [15].

The additional operation adds two parameters from two feature maps to create the new feature. Hence, the additional operation produces the exact size of the new feature map.

Concatenation Operation. In the concatenation operation, the sizes of the feature maps (width × height) must be identical, while the number of the feature maps can be different [6]. When performing the concatenation operation, two feature maps are combined. For example, when the first and second feature maps are defined as (width × height × number of feature maps) $4 \times 9 \times 16$ pixels and $4 \times 9 \times 8$ pixels, the new feature maps, after applying concatenation operation, are $4 \times 9 \times 24$ pixels. In our framework, we transform the feature maps to fit the input of the RNN architecture using the convolutional layer of size 1×1.

2.4. **Decoding algorithms.** The decoding algorithm is proposed to transcribe the sequence data to text output. In our study, the sequence of probability distributions of 157 characters, which are the output of the softmax activation function, was used as input of the decoding algorithm.

Connectionist Temporal Classification (CTC). The CTC loss function was proposed to discover the possible alignment path [16], which is the sequence text output. The CTC loss function was designed to compute the conditional probability of a possible alignment path that occurred for each time step along the possible path. Hence, the sequence probabilities given from the softmax function are classified to the correct sequence labels.

Word Beam Search (WBS). The WBS was proposed to solve the decoding of arbitrary character strings that allow arbitrary non-word character strings between words [17]. Multiple text candidates for the final labeling (beams), are calculated and scored. The WBS algorithm starts with an empty beam. The beam can then add some possible characters. Subsequently, the WBS determines to keep the best score. Then, the beam is expanded by all possible characters in each time step and copies the original beam to the following time step. After the final time step, the best beam is returned with the best text output.

3. Experimental Results.

3.1. **Multi-language video subtitle dataset.** The video subtitle images[1] contain 4,224 subtitle images captured from 24 videos [18]. It includes 157 letters in total, including 44 Thai consonants, 19 Thai vowels, 4 Thai tones, 3 Thai punctuation marks, 10 Thai digits, 10 Arabic digits, 26 Roman characters, 26 Roman capital characters, and 15 special characters. The challenge of this subtitle text image dataset is that it has various font styles, brightness, noise, and background. Examples of subtitle text images are shown in Figure 2.

(a)

(b)

(c)

FIGURE 2. Some examples of the Thai-English video subtitle dataset: (a) Thai, (b) English, and (c) mixed languages

[1]The multi-language video subtitle dataset can be downloaded from https://data.mendeley.com/datasets/gj8d88h2g3/2.

3.2. **Implementation detail.** We implemented the fusion CRNNs using the Keras library based on the TensorFlow deep learning framework. All the experiments were trained and evaluated on the Google Colab platform using NVIDIA Tesla P100 GPU with 16GB of RAM. In the training process, first, the pre-trained models of state-of-the-art CNN architectures (VGGNets and ResNets) were examined. We trained and fine-tuned the models with the following parameters: Optimizer = Adam, learning rate = 0.0001, first-moment and second-moment estimate values = 0.9 and 0.999, epsilon = 1e-07, batch size = 32, epoch = 200. Second, we used the fine-tuned models that were trained in the first step. We created the new network by fusing two CNN models and followed by the RNN network. The softmax function was the last layer of our proposed network. In the fused operation, two operations: addition and concatenation, were compared. For RNN architectures, we also experimented with both LSTM and GRU. We trained the fusion CRNNs with 100 epochs.

We divided the dataset into three subsets with a ratio of 70 : 10 : 20. The training, validation, and test sets contained 2,957, 422, and 845 images, respectively. The training and validation sets were combined and used in the 5-fold cross-validation (5-cv) experiments.

3.3. **Evaluation metrics.** We presented three metrics to evaluate the CRNN models: training time, parameter, and character error rate (CER) value [19]. The network with the lowest CER value is the best performance for a word recognition task. We computed the CER value as follows:

$$CER = \frac{I + S + D}{N} \tag{1}$$

where I is the number of character insertions, S is the number of characters substituted, D is the number of characters deleted, and N is the total number of characters in the ground truth text.

3.4. **Experiments on CRNN architectures.** In this experiment, we compared the state-of-the-art VGGNets and ResNets with ChamchongNet [8]. ChamchongNet has a few convolutional layers, including three convolutional with max-pooling layers and two layers of RNN architecture. For the evaluation metrics, the training time (hour:minute), parameters (million: M), and CER value (%) were presented.

Table 1 shows that the best ChamchongNet+BiGRU model spent 40 minutes training because the ChamchongNet model had fewer convolutional layers than the other CRNN models. It has 0.51 million parameters. As a result, it achieved the worst performance on the multi-language video subtitle dataset with the CER value of 10.17%.

On the other hand, the VGG-s2+BiLSTM achieved the best performance with a CER value of only 5.78%. Nevertheless, the VGG-s2+BiLSTM had more parameters and spent

Table 1. Evaluation of the CRNN architectures on the multi-language video subtitle dataset

CNN models	Two RNN layers	Input sizes	Training time		Parameters		CER	
			RNN sizes		RNN sizes		RNN sizes	
			128	256	128	256	128	256
ChamchongNet	BiGRU	32 × 379	00:26	00:30	0.17	0.51	10.40	10.89
ChamchongNet	BiGRU	64 × 755	00:37	00:40	0.17	0.51	10.72	10.17
ChamchongNet	BiLSTM	32 × 379	00:33	00:34	0.21	0.66	12.06	11.79
ChamchongNet	BiLSTM	64 × 755	00:41	00:44	0.21	0.66	11.37	11.95
VGG-s1	BiLSTM	32 × 379	00:57	1:01	9.00	11.14	8.22	6.25
VGG-s2	BiLSTM	64 × 755	1:34	1:41	9.00	11.14	6.90	**5.78**
ResNet-s1	BiLSTM	32 × 379	00:44	00:47	4.60	6.74	12.57	8.71
ResNet-s2	BiLSTM	64 × 755	1:04	1:07	4.60	6.74	6.94	6.44

more time training than the ChamchongNet+BiGRU model. Furthermore, the best performance was performed with input images size of 64×755.

VGGNets have fewer layers but more parameters than ResNets because VGGNets contain many feature maps. ResNets have more layers and propose the skip connection to connect the layer with the following two layers using additional operation when it has the same size. Hence, the parameters are not increased. Indeed, ResNets spent less computation time than VGGNets.

3.5. **Experiments on fusion strategies.** After evaluating various CRNN architectures on the multi-language video subtitle dataset (see Table 1), we fused the CRNN architectures that achieved a low CER value: VGG-s1, VGG-s2, and ResNet-s2 using fusion operation. After applying the fusion operation, the LSTM network was added to the network. We also used an input image with 64×755 pixels resolution.

The results of fusion CRNN models: VGG-s1+VGG-s2, VGG-s1+ResNet-s2, and VGG-s2+ResNet-s2, are presented in Table 2. The fusion of VGG-s1+VGG-s2 using additional operation gave the best fusion CRNN architecture. It achieved a CER value of 5.33%. However, despite the best performance, VGG-s1+VGG-s2 consumed the longest computational training time among these models. For the recognition performance, the fusion CRNN architecture clearly outperformed the single CRNN architecture.

Table 2. The comparison results of fusion CRNN models using additional and concatenation operations

Fusion CRNN models	Additional fusion			Concatenation fusion		
	Training time	Parameters	CER	Training time	Parameters	CER
VGG-s1+VGG-s2	2:24	20.65	**5.33**	2:31	21.85	6.04
VGG-s1+ResNet-s2	1:51	14.56	5.66	1:59	16.32	6.06
VGG-s2+ResNet-s2	1:54	15.69	5.65	1:59	17.65	5.77

Note that, we used the CTC algorithm to decode the output of the subtitle words in experiments because the CTC algorithm became the default algorithm of the CRNN used in every research. We then compare the results of the CTC and WBS algorithms, as shown in the following section.

3.6. **Comparing the decoding algorithms for word recognition.** Another interesting comparison is the performance of the decoding algorithms in terms of computation and CER value. We performed a final experiment in the CTC and WBS decoding algorithms.

The comparative results on the multi-language video subtitle dataset using the CTC and WBS decoding algorithms are presented in Table 3.

When considering the testing time using both decoding algorithms, the single CRNNs consumed less computational time than the fusion CRNN architectures. For CTC, the

Table 3. The comparison results of different CRNN models using CTC and WBS decoding algorithms

CRNN models	CTC			WBS		
	Testing time (∼ms)	CER		Testing time (∼ms)	CER	
		5-cv	Test		5-cv	Test
VGG-s2	1	3.33 ± 0.60	5.78	639	3.29 ± 0.59	5.73
ResNet-s2	1	3.43 ± 0.29	6.44	568	3.40 ± 0.29	6.42
Fusion VGG-s1+VGG-s2	6	3.04 ± 0.37	5.33	1,775	3.01 ± 0.37	**5.29**
Fusion VGG-s2+ResNet-s2	12	2.90 ± 0.30	5.65	2,130	$\mathbf{2.89 \pm 0.29}$	5.63

single CRNNs spent around one millisecond per image, while the fusion CRNNs spent up to 12 milliseconds per image. For WBS, the best testing time was 568 ms and 1,775 ms when tested with the single CRNNs and the fusion CRNNs, respectively.

The WBS slightly outperformed the CTC decoding algorithm regarding recognition performance. The fusion CRNN architecture (VGG-s1+VGG-s2) achieved the best CER value on the test set when fused with two CNNs using additional operation and combined with LSTM. Additionally, VGG-s2+ResNet-s2 outperformed other architectures on the 5-cv.

To recognize the subtitle word in the real-world application, we recommend decoding the output using the CTC algorithm. Examples of the correct recognition using fusion CRNN architecture (VGG-s1+VGG-s2) are shown in Figure 3. Some recognition results and CER values are shown in Table 4.

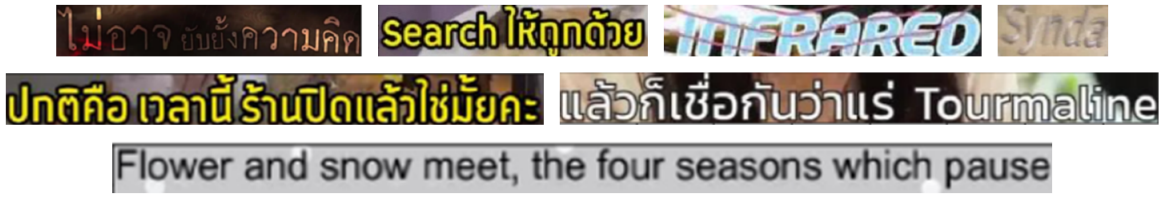

FIGURE 3. Illustration of the correct recognition with a CER value of 0

TABLE 4. The recognition results of the fusion CRNN (VGG-s1+VGG-s2) architecture using the CTC decoding algorithm

#	Input Images and Labels	Recognition Results	CER
1	 Label: Thailand Education Partnership Forum 2018	Thailand Education Partnership Forum 208	2.44
2	 Label: LIFE BECOMES A BETTER ONE	LFE BECOMES A BETTER ONE	4.00
3	 Label: FACEBOOK ลัดดาแลนด์ 2513	FACEBOOK ลัดดาแลนด์ 2513	4.17
4	 Label: 31 ธ.ค. 2019 จีนรายงาน WHO	31ธ.ค 2019 จีนรายงาน WHO	3.85
5	 Label: ฉันเคยมี หัวใจ แต่แล้ว ก็โดนทำร้ายไป	ฉันคยมี หัวใจ แต่แล้ว ก็โดนทำร้ายไป	5.55
6	 Label: ธรรมาชาติบำบัด	ธรรมาชาติบำบัด	50.0

In Table 4, the output of samples #1 and #2 (orange highlight) were incorrect because the input images contain a narrow space between characters "1" and "I". In sample #6 (blue and underline text), a font used in the input image produced almost identical characters between "ธ" and "ร", and also "ค" and "ก".

4. **Conclusions.** In this research, we propose a deep learning method called fusion convolutional recurrent neural network (CRNN) to address the challenge of text image recognition from video subtitle images. We evaluated the proposed CRNN architecture on a multi-language video subtitle dataset, consisting of Thai and English alphabets with 157 different letters (consonants, digits, and symbols).

In the experiment section, we first experimented with three CRNN architectures: VGGNets, ResNets, and ChamchongNet. We discovered that VGGNets combined with BiLSTM (VGG-s2+BiLSTM) achieved the character error rate (CER) value of 5.78%. The ResNets architecture achieved a slightly higher CER value than the VGGNets architecture. However, the VGG+BiLSTM has few convolutional layers and has many hyperparameters compared to the ResNet+BiLSTM, which has more convolutional layers but fewer hyperparameters. Second, we fused two CNN models with fusion operation (addition and concatenation) followed by a BiLSTM network, called fusion CRNN. We found that combining two VGGNets with additional fusion operation achieved a CER value of 5.29%. Fusion CRNN architecture was slightly better than a single CRNN architecture. Finally, we conducted experiments with two decoding algorithms: connectionist temporal classification (CTC) and word beam search (WBS). WBS slightly outperformed CTC. However, WBS computed much slower than CTC. Notably, we prefer the CTC algorithm to decode the output when used in real-world applications.

In future work, we are interested in extracting the deep features using adaptive feature fusion [20] and possibly apply it to verifying handwritten signature [21].

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

REFERENCES

[1] A. U. Rehman, A. K. Malik, B. Raza and W. Ali, A hybrid CNN-LSTM model for improving accuracy of movie reviews sentiment analysis, *Multimed. Tools Appl.*, vol.78, no.18, pp.26597-26613, 2019.

[2] H. Yan and X. Xu, End-to-end video subtitle recognition via a deep residual neural network, *Pattern Recognit. Lett.*, vol.131, pp.368-375, 2020.

[3] K. Dutta, P. Krishnan, M. Mathew and C. V. Jawahar, Improving CNN-RNN hybrid networks for handwriting recognition, *International Conference on Frontiers in Handwriting Recognition (ICFHR)*, Niagara Falls, USA, pp.80-85, 2018.

[4] M. V. V. P. Kantipudi, S. Kumar and A. K. Jha, Scene text recognition based on bidirectional LSTM and deep neural network, *Comput. Intell. Neurosci.*, vol.2021, pp.1-11, 2021.

[5] V. Carbune, P. Gonnet, T. Deselaers, H. A. Rowley, A. Daryin, M. Calvo, L.-L. Wang, D. Keysers, S. Feuz and P. Gervais, Fast multi-language LSTM-based online handwriting recognition, *Int. J. Doc. Anal. Recognit.*, vol.23, no.2, pp.89-102, 2020.

[6] J. Poulos and R. Valle, Character-based handwritten text transcription with attention networks, *Neural Comput. Appl.*, vol.33, no.16, pp.10563-10573, 2021.

[7] Z. Lei, S. Zhao, H. Song and J. Shen, Scene text recognition using convolutional recurrent neural network, *Mach. Vis. Appl.*, vol.29, no.5, pp.861-871, 2018.

[8] R. Chamchong, W. Gao and M. D. McDonnell, Thai handwritten recognition on text block-based from Thai archive manuscripts, *International Conference on Document Analysis and Recognition (ICDAR)*, Sydney, Australia, pp.1346-1351, 2019.

[9] R. Chamchong, U. Saisangchan and P. Pawara, Thai handwritten recognition on BEST2019 datasets using deep learning, *International Conference on Multi-disciplinary Trends in Artificial Intelligence (MIWAI)*, pp.152-163, 2021.

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

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

[12] A. Graves and J. Schmidhuber, Framewise phoneme classification with bidirectional LSTM and other neural network architectures, *Neural Networks*, vol.18, no.5, pp.602-610, 2005.

[13] J. Deng, L. Cheng and Z. Wang, Self-attention-based BiGRU and capsule network for named entity recognition, *arXiv.org*, arXiv: 2002.00735, 2020.

[14] Y.-H. Tu, J. Du, Q. Wang, X. Bao, L.-R. Dai and C.-H. Lee, An information fusion framework with multi-channel feature concatenation and multi-perspective system combination for the deep-learning-based robust recognition of microphone array speech, *Comput. Speech Lang.*, vol.46, pp.517-534, 2017.

[15] X. Liang, P. Hu, L. Zhang, J. Sun and G. Yin, MCFNet: Multi-layer concatenation fusion network for medical images fusion, *IEEE Sens. J.*, vol.19, no.16, pp.7107-7119, 2019.
[16] A. Graves, S. Fernández, F. Gomez and J. Schmidhuber, Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks, *International Conference on Machine Learning (ICML)*, PA, USA, pp.369-376, 2006.
[17] H. Scheidl, S. Fiel and R. Sablatnig, Word beam search: A connectionist temporal classification decoding algorithm, *International Conference on Frontiers in Handwriting Recognition (ICFHR)*, Niagara Falls, USA, pp.253-258, 2018.
[18] 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.
[19] T. Bluche, *Deep Neural Networks for Large Vocabulary Handwritten Text Recognition*, Ph.D. Thesis, Universite' Paris Sud-Paris XI, France, 2015.
[20] S. Zhao, T. Xu, X.-J. Wu and X.-F. Zhu, Adaptive feature fusion for visual object tracking, *Pattern Recognit.*, vol.111, 107679, 2021.
[21] O. AbuAlghanam, L. Albdour and O. Adwan, Multimodal biometric fusion online handwritten signature verification using neural network and support vector machine, *International Journal of Innovative Computing, Information and Control*, vol.17, no.5, pp.1691-1703, 2021.