

A compact deep learning model for Khmer handwritten text recognition

Bayram Annanurov¹, Norliza Mohd Noor²

¹Department of Computer Science, Paragon International University, Cambodia

²Department of Engineering, Razak Faculty of Technology and Informatics, Universiti Teknologi Malaysia, Malaysia

Article Info

Article history:

Received Sep 7, 2020

Revised May 19, 2021

Accepted May 25, 2021

Keywords:

Character recognition

Convolutional neural networks

Deep learning

Handwriting recognition

Multilayer neural networks

ABSTRACT

The motivation of this study is to develop a compact offline recognition model for Khmer handwritten text that would be successfully applied under limited access to high-performance computational hardware. Such a task aims to ease the ad-hoc digitization of vast handwritten archives in many spheres. Data collected for previous experiments were used in this work. The one-against-all classification was completed with state-of-the-art techniques. A compact deep learning model (2+1CNN), with two convolutional layers and one fully connected layer, was proposed. The recognition rate came out to be within 93-98%. The compact model is performed on par with the state-of-the-art models. It was discovered that computational capacity requirements usually associated with deep learning can be alleviated, therefore allowing applications under limited computational power.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Norliza Mohd Noor

Department of Engineering, Razak Faculty of Technology and Informatics

Universiti Teknologi Malaysia Kuala Lumpur Campus

Jalan Sultan Yahya Petra, 54100 Kuala Lumpur, Malaysia

Email: norliza@utm.my

1. INTRODUCTION

Khmer is an official language of Cambodia, spoken by about 16 million people. It has an Alpha syllabary (Abugida) writing structure: words are comprised of syllables, most of which consist of a radical for a consonant and additional score for vowels. The modern Khmer alphabet consists of 33 consonants.

There is a great demand for a recognition system reflecting Khmer writing specifics due to the constant accumulation of documents in such spheres as government, healthcare, finance, education. Until the early 2000 s, most records in the government and private sectors in Cambodia have been held on handwritten documents and hand-filled forms. One has to manually browse through the entire mass of paper to reach any of these records. The bulk of such tasks is extremely complex to carry out on daily basis, even with help of a systematic archiving system. Having an effective deep learning [1] application for digitizing handwritten text is particularly important for promoting the development of public and private services. Such an application also needs to be inexpensive and applicable in developing economies.

As opposed to other common alphabetical systems, there is a very small amount of research on Khmer text recognition. Most of the efforts have been done only within the past decade [2]-[8]. Sok and Taing [3] and Srun and Vyshyakov [4], [5], [7] studied recognition of the Khmer printed text. Ye *et al.* [8] developed an online recognition method for printed text in the Khmer, Bangla, and Myanmar alphabets. The amount of work in the field, as well as the nature of the collected data for relevant experiments, describes the current state of the art for Khmer handwritten text recognition (HTR). Most of the data used in the past

experiments were printed (Machine-derived) text, which greatly impedes the development of an accurate application.

As an extension of previous experiments [9], [10], current work implemented CNNs for Khmer HTR. A novel, compact model 2+1CNN was proposed to be used alongside the models used in literature (LeNet-5 [1], AlexNet [11], visual geometry group 16 (VGG16), VGG19 [12], ResNet [13]). 2+1CNN is designed for binary classification while existing models were optimized accordingly due to the adapted one-against-all tactic used throughout the work.

To increase overall performance, an independent network was trained and evaluated for each class. One particular class was taken as "positive" and all others – as "negative" while training each network. That is, given a set of classes $C = \{c_1, c_2, \dots, c_k\}$, the samples of class c_j were isolated and all samples of other classes $c_1, c_2, \dots, c_{j-1}, c_{j+1}, \dots, c_k$ were considered as "not c_j " (or c_j'). Training cable news network (CNN) with this setting yielded a classifier model $F_j(\cdot)$. The output of the training process was the combination of all trained classifiers:

$$F(\cdot) = \{F_j(\cdot) | j = 1..k\} \quad (1)$$

Intuitively, the final model was designed to iterate the question "Are you of class c_j ?" instead of asking directly "What class are you?" This work aims to design a compact model for the Khmer HTR system. Lack of appropriate datasets contributes to its difficulty. Only datasets collected in preliminary experiments [9], [10] were used.

2. RELATED WORK

2.1. Recognition of Khmer handwriting

Meng and Morariu [2] described how to combine feedforward artificial neural network (ANN) with a self-organizing map (SOM) to design a recognition system for printed Khmer characters. Sok and Taing [3] described their experiment with SVM on printed Khmer characters. Font size-based accuracy and CPU load were presented as efficiency assessment. Authors also listed some scarce work done towards Khmer optical character recognition (OCR) to emphasize on lack of research for the Khmer language. Backpropagation was used by Srun [4] to train a classifier to recognized Khmer characters. For the experiments, Srun sampled printed text. Preprocessing consisted of resizing images to standard dimensions. Thumwarin *et al.* [6] in their studies implemented finite impulse response (FIR) to extract features from handwritten Khmer characters and sent their results to a Euclidean-based classifier. The work relies on temporal information, which is impossible to collect from a scanned image of a manuscript. Another problem that the method requires extra hardware for collecting temporal information. Another work by Srun and Vishnyakov [7] included the implementation of classifiers in TESSERACT and further improvement of recognition quality of scanned characters. The earliest mention of Khmer HTR in a computerized setting dates as early as 2008 in work by Ye *et al.* [8], which proposed a recognition system of scripts like Myanmar, Khmer, and Bangla [8]. Research data was collected by the means of drawing characters with a mouse, which is also a drawback of the work. Unlike many previous attempts, data used in the current work reflects the nature of common handwriting which makes resultant models more realistic. Khmer datasets acquired in previous attempts are compared in Table 1.

Table 1. Data sets were acquired for Khmer HTR

Literature	Dataset	Data and size
Sok and Taing [3]	Printed and scanned text	Khmer Characters, 3000
Ye <i>et al.</i> [8]	Collected by mouse, stylus pen	Khmer, 135, Myanmar Characters, 107
Thumwain <i>et al.</i> [6]	Scanned text	Khmer letters and digits, 6750
Kruy and Kameyama [14]	Printed and scanned text	Khmer words, 1104
Meng and Morariu [2]	Printed and scanned text	Khmer Characters, 215
Kheang <i>et al.</i> [15]	Printed and scanned text	Khmer words, 110713
Srun [4]	Printed and scanned text	Khmer Characters, 33

2.2. Convolutional neural networks

A convolutional neural network (CNN, or ConvNet) is a special kind of deep, feed-forward artificial neural network. In an image processing application, CNNs learns directly from images. Key concepts, important in the description of CNN, are local receptive fields (LRF), shared weights and biases, activation,

and pooling. CNNs also differ from each other in the method and objective of training, e.g., prediction, object discovery, segmentation.

According to Cun [1], [16], CNN is a variation of multilayer perceptron which require minimal preprocessing requirements. The connectivity pattern between neurons in a CNN is inspired by the biological processes of the animal visual cortex, where each cortical neuron responds to signal from only a restricted area of the visual field (*receptive field*). Matsugu *et al.* [17] described that receptive fields that connect to different neurons, partially overlap. This leads to having the entire visual field covered and, therefore, to smooth vision.

Figure 1 shows an example of a three-dimensional neuron arrangement in a convolutional neural network. Every layer takes a three-channel image, where each pixel has a separate value for red, green, and blue components. The image is split to form output in form of a 3D matrix of neurons. Data used in this study was preprocessed into grayscale images.

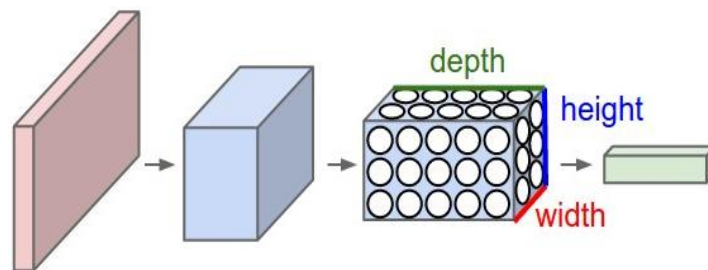


Figure 1. 3-D neuron arrangements in a CNN [18]

The convolution operation is performed on the input data. This step models the response of an individual biological neuron to visual input. The activation step applies a transformation to the output of each neuron by using activation functions. Rectified linear unit (ReLU), is an example of a commonly used activation function. It takes the output of a neuron and maps it to the highest positive value. If the output is negative, the function maps it to zero.

The output of the activation step can be further transformed by applying a pooling step. Pooling reduces the dimensionality of the feature map by condensing the output of small regions of neurons into a single output. This helps to simplify the consequent layers and reduces the number of parameters that the model needs to learn. CNN layers are configured by these three concepts. A CNN can have tens or hundreds of hidden layers that each learns to detect different features of an image. In such feature maps, every hidden layer increases the complexity of the learned image features. For example, the first hidden layer learns how to detect edges, and the last layer learns how to detect more complex shapes.

In CNN inputs from a small local receptive field (LRF) are connected to one neuron hidden layer. LRF is translated across an image to create a feature map from the input layer for being used in the hidden layers. Convolutions are used to implement this process efficiently [19]. A convolution operation is applied to the input of each layer. The convolution mimics the reaction of neurons to visual input. CNN architecture also includes *pooling layers*, that are used to group the outputs of one layer into a single neuron in the next layer [11], [20]. The cluster of neurons is designed in form of square batches of any size $n \times n$, where $n = 2, 3, 4, \dots$.

In some cases, pooling batches need to be moved beyond the boundaries of a sample image, which may cause ambiguity in the training process as well as computational and programmatic complexity. Extending the image by several rows and columns of pixels to match the size of pooling batches (*padding*) helps to overcome such a problem. The values used for the extra pixels may be taken differently: average overall spectrum of pixel values (*average padding*), zeros (*zero paddings*). Denoting filter size as F , input size as W , resulting in image size as R , padding size as P , and stride size as S , it is obvious that the size of the sample after each pooling layer will become is being as forms, which can also be deducted for two dimensions:

$$R = \begin{cases} \frac{W+2P-F}{S}, & \text{if } S|(W+2P-F) \\ \frac{W+2P-F}{S} + 1, & \text{otherwise} \end{cases} \quad (2)$$

3. RESEARCH METHOD

Current experiments were based on the same data set and most preprocessing steps [9], [10]. Later, the potential to highly increase the recognition rate of neural networks was explored [21]. Figure 2 shows the development of the Khmer HTR framework. Data collection and preliminary experiments were completed in our previous work [9], [10]. In preliminary experiments, the number of features was reduced by 90% using three independent methods: correlation-based feature selection (CORR), two-dimensional Fourier transform (FT2D), and Gabor filters (GF). The result of each method was classified with an artificial neural network (ANN). The original data, without feature space transformation, was classified for comparison of performance. Gabor Filters yielded the highest improvement in recognition. Such a fact suggested that filters may play an important role in feature extraction. The current study is based on convolutional models, which rely on a wider variety of filters. In the course of current work, Models LeNet-5, AlexNet, VGG16, VGG19, ResNet50 have been modified for binary classification.

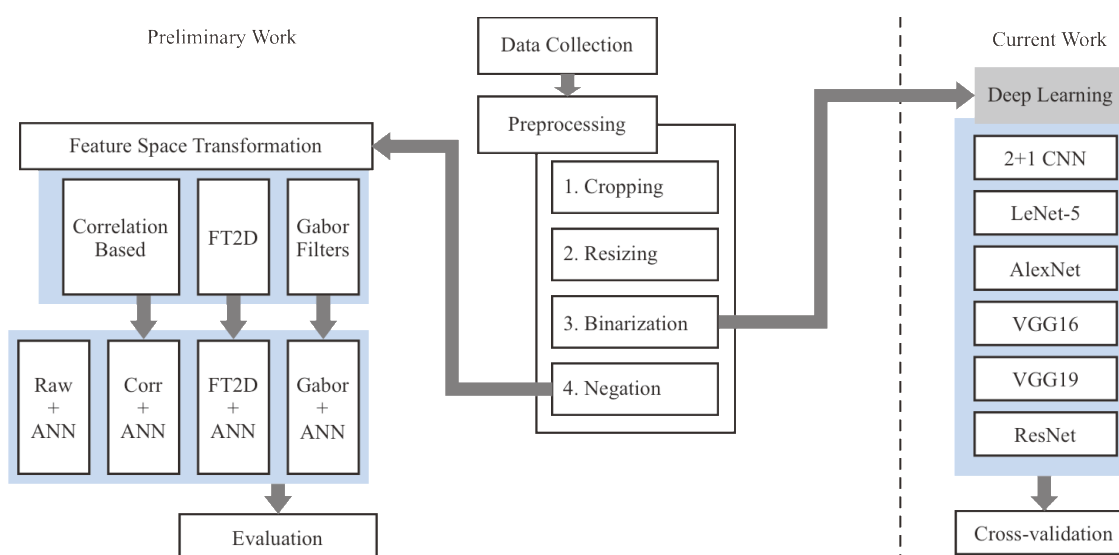


Figure 2. Research framework

3.1. One-against-all tactic

Khmer samples of one consonant were taken as positive class and the ones of remaining consonants – as negative class. Such practice is called a *two-way classification* for having only two classes to recognize from: “*positive*” and “*negative*”. It has been adopted at all stages of the work. The performances of all 33 classifiers (one per each consonant) have been averaged to obtain the performance of each method. The final classification model for each method is the assembly of the classifiers as described in (1). Such a tactic was adopted since it has proven to be highly effective in comparison to direct multi-class classification in many other [22] applications. Each Khmer character has been treated based on the corresponding root radical (consonant) as a sample of that consonant. Since 17 vowels were combined with each consonant, there were 17 samples in each class.

3.2. The proposed model

This study introduces a convolutional neural net with a compact architecture: two convolutional layers and one fully-connected layer. The model is referred to as “2+1CNN”, for brevity. The model is built ground-up and is initialized with random weights. 2+1CNN is based on a one-against-all tactic and designed for binary classification.

3.3. Proposed model architecture

2+1CNN was proposed as a compact model for Khmer HTR and is expected to ease the burden of computational requirements while staying close to the guidelines of previously designed successful architectures [1], [11]-[13], [23]. Local receptive fields of size 5×5 have been used in convolutional layers. Maximal pooling out of 2×2 patches has been used after each convolutional layer. Convolutional and pooling layers were kept as simple as possible to reduce the number of computations per filter. The input size was kept the same as in the previous research [11].

To prevent overfitting, 50% of the nodes in the fully connected layer are dropped out in random order. Rectified linear unit (ReLU) is used as an activation function, due to the simplicity of differentiation and its behavior close to other activation functions. Hyper-parameters used in 2+1CNN are being as:

- Input images are pre-processed, resized to 224×224.
- First convolutional layer with ReLU as activation function, 5×5 filters with stride size 1.
- First pooling layer with 2×2 filters, stride size 1.
- Second convolutional layer with ReLU activation, 5×5 filters, stride size 2.
- Second pooling layer with 2×2 filters, stride size 1.
- The dropout stage randomly erases 50% of the perceptron, to reduce overfitting.
- The fully connected layer is made of 463 perceptron's with the ReLU activation function. The choice for the number is based on average (number of features + number of samples) / 2.

Table 2 illustrates the structure of 2+1CNN. The values R, W, P, and F were obtained per (2). Filter sizes are chosen to minimize the number of computations required during model training. Figure 3 gives the visualization of a sample as it is traversed through each layer in 2+1CNN. Represented layers are input, convolution, pooling, convolution, pooling. All other models used in this work (LeNet, AlexNet, VGG16, VGG19, RESNET) were also modified so that the number of output classes was reduced to two. This modification was done to implement binary classification due to the adopted one-against-all tactic. While 2+1CNN is built ground-up, transfer learning was used to retrain the State-of-the-Art methods on Khmer samples. Due to limitations of available processing power and a high amount of data, training of all classifiers has been limited to 500 iterations.

Table 2. Structure of 2+1CNN

	Input (W×W)	Padding (P×P)	Filter size (F×F)	Stride size (S×S)	Output (R×R)
Data	224×224				
Conv-1, ReLU	224×224	2×2	5×5	1×1	223×223
Pooling-1	223×223	0×0	1×1	2×2	111×111
Conv-2, ReLU	111×111	2×2	5×5	2×2	55×55
Pooling-2	55×55	0×0	1×1	2×2	27×27
Dropout	27×27	-	-	-	27 ² /2
FC Layer	365	-	-	-	1

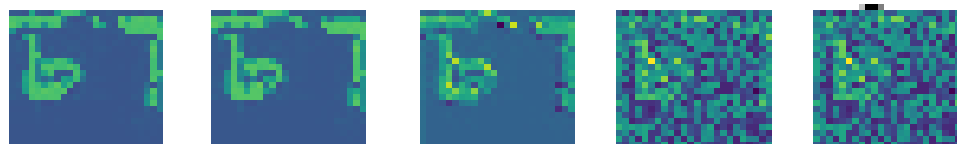


Figure 3. Visualization of a sample within 2+1CNN

3.4. Performance evaluation

The performance of each classifier was quantified by the recognition rate on the testing data set: the ratio of the number of samples recognized correctly to the total number of samples. To ensure the robustness of each model, cross-validation was applied in four-folds. To measure the performance of an assembly of classifiers, the average of their recognition rates was taken, as per (1).

3.5. System specifications

System hardware used in experimentation: Windows 7 64 bit, 4GB RAM, Intel Core i3-220 2.20 GHz CPU. CNN architecture was implemented in Keras, with TensorFlow backend.

4. RESULTS AND DISCUSSION

DL models were applied to each of the existing data sets, individually and are compared by recognition rate at each data set. The main finding of the study is that the compact model 2+1CNN is highly effective. The recognition rate came out to be more than 94% on average, on par with the other models. This proves the concept of an ad-hoc CNN-based recognition system, that can be designed in a setting with low computational and capabilities. The implications are important for the applications in growing economies, like Cambodian, where developers and data engineers have limited access to high-performance technology.

Table 3 compares the hardware used in previous experiments to that of the current work. The overall comparison of models is given in Table 4. Table 5 shows the comparison of current work against previous attempts. It highlights the theoretical progress in the field of handwritten text recognition for Abugida writing systems, including Khmer. In previous attempts, data was collected either by scanning printed text or drawing with a computer mouse, which poses difficulty representing common handwriting. The results of the current HTR task were achieved on a hardware system of lesser specifications.

Table 3. System requirements for deep learning applications

Literature	System setting	Model	Data Set, size	Result
Current work	Intel core i3, 2.20GHz, RAM:4GB	2+1CNN, LeNet, AlexNet VGG16, VGG19, Resnet	Khmer Chars, 3366	Accuracy: 94.9%, 97.1%, 97.6%, 96.4%, 95.8%, 100%
[11]	GTX 580, GPU 3GB	AlexNet	ImageNet, 1.4M	Error: 15.3%
[12]	4×NVIDIA Titan Black GPU	VGG16, VGG19	ImageNet, 1.4M	Error: 12%
[24]	8×GPU	ResNet	ImageNet, 1.4M, CIFAR-10, 50k	Error: 3.57%, 6.97%
[25]	GeForce Titan X Pascal GPU	LeNet	IAM, 115k, RIMES, 12k	Error: 12.7%, 6.6%
[26]	Intel Core i3 3.30 GHz, 12 GB RAM, GPU: Nvidia 1050Ti 4GB	ResNet	Bangla, 200k	Error: 5.5%
[27]	GTX TITAN X GPU	ResNet	ICDAR- 2013, 462	Accuracy: 97.03%
[28]	Intel i7-4600U, 16 GB RAM	ResNet	ICDAR- 2011, 7166	F-score: 90.18-96.88
[29]	GTX Titan X	ResNet, VGG16	Georgian HWT, 200k	Accuracy: 95%, 89%
[30]	Intel Core i5-6500 4GHz, RAM: 8GB, Nvidia GTX 1070 8GB	AlexNet	Iranshahr, 15k	Accuracy: 99.13%

Table 4. Overall summary on CNN-based model

Model	Average recognition rate (%)	Convolutional layers	Fully-Connected layers
2+1 CNN	94.9	2	1
LeNet-5	97.2	2	3
AlexNet	97.6	5	3
VGG16	96.4	13	3
VGG19	95.8	16	3
ResNet50	100	49	1

Table 5. Previous attempts to develop a classifier for abugida-type texts

Literature	Dataset	Data and size	Classifier	Accuracy
Sok and Taing [3]	Printed and scanned text	Khmer Characters, 3000	SVM	98%
Ye <i>et al.</i> [8]	Mouse drawn, stylus	Khmer, 135, Myanmar, 107	Stock methods	Writing speed
Thumwain <i>et al.</i> [6]	Scanned text	Khmer symbols, 6750	Distance-Based	98%
Kruy and Kameyama [14]	Printed and scanned text	Khmer words, 1104	SIFT, distance-based	98%
Meng and Morariu [2]	Printed and scanned text	Khmer Characters, 215	ANN	65%
Kheang <i>et al.</i> [15]	Printed and scanned text	Khmer words, 110713	WFST	~73%
Srun [4]	Printed and scanned text	Khmer Characters, 33	ANN	97%
Annanurov and Noor [9], [10]	Handwritten Characters	Khmer Syllables, 3366	ANN + feature extraction	Higher performance
2+1 CNN	Handwritten Characters	Khmer Syllables, 3366	2+1 CNN	94.9%

5. CONCLUSION

This work aimed to develop a compact and effective model for offline recognition of Khmer handwritten characters. In general, recognition rates came out to be 93-98%. The 2+1CNN model was built ground-up and had performance over 94%, which is at the same level as other, more sophisticated models. The results also helped towards closing the research gap in the field since, at the time of experiments, Khmer HTR has not yet been approached with deep learning. The main contribution is the compact Khmer HTR model (2+1CNN) with low computational requirements, which is based on open-source software and does not require any proprietary packages. These aspects ease its implementation, therefore, allowing swift digitization of document corpora in rural and developing areas. The developed models may be applied in a high-end OCR application targeted to the general public, as well used in more sophisticated applications with only the back-end part, aiming to digitize documents. Further works may include recognition based on the information about the layout of documents, forms, tables.

ACKNOWLEDGEMENTS

This work was partially funded by Universiti Teknologi Malaysia and the Ministry of Higher Education Malaysia.

REFERENCES

- [1] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278-2324, Nov. 1998, doi: 10.1109/5.726791.
- [2] H. Meng and D. Morariu, "Khmer character recognition using artificial neural network," in *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific*, 2014, pp. 1-8, doi: 10.1109/APSIPA.2014.7041824.
- [3] P. Sok and N. Taing, "Support Vector Machine (SVM) based classifier for Khmer Printed Character-set Recognition," in *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific*, 2014, pp. 1-9, doi: 10.1109/APSIPA.2014.7041823.
- [4] S. Srun, "Applying Backpropagation for Khmer Printing Character Recognition," *Proceedings of Japan-Cambodia Joint Symposium on Information Systems and Communication Technology 2011*, Phnom Penh, 2011, pp. 135-136.
- [5] S. Srun and U. Vishnyakov, "An Approach for Quality Enhancement of the Text Recognition," *Intellectual CAD*, vol. 4, 2009.
- [6] P. Thumwarin, S. Khem, K. Janchitraponvej, and T. Matsuura, "On-line writer dependent character recognition for Khmer based on FIR system characterizing handwriting motion," *2008 SICE Annual Conference*, 2008, pp. 73-78, doi: 10.1109/SICE.2008.4654625.
- [7] S. Srun, "Applying Tesseract for Khmer Optical Character Recognition," in *ASEAN-UEC Symposium*, 2015.
- [8] Y. K. Thu, O. Phavy, and Y. Urano, "Positional gesture for advanced smart terminals: Simple gesture text input for syllabic scripts like Myanmar, Khmer and Bangla," in *2008 First ITU-T Kaleidoscope Academic Conference - Innovations in NGN: Future Network and Services*, 2008, pp. 77-84, doi: 10.1109/KINGN.2008.4542252.
- [9] B. Annanurov and N. M. Noor, "Handwritten Khmer text recognition," in *2016 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)*, 2016, pp. 176-179, doi: 10.1109/WIECON-ECE.2016.8009112.
- [10] B. Annanurov and N. M. Noor, "Feature selection for Khmer handwritten text recognition," in *2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, 2017, pp. 626-630, doi: 10.1109/EIConRus.2017.7910634.
- [11] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012.
- [12] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *CoRR arXiv: 1409.1556*, 2014.
- [13] Z. Y. He, "A New Feature Fusion Method for Handwritten Character Recognition Based on 3D Accelerometer," *Applied Mechanics and Materials*, vol. 44-47, pp. 1583-1587, 2010, doi: 10.4028/www.scientific.net/AMM.44-47.1583.
- [14] V. Kruiy and W. Kameyama, "Preliminary Experiment on Khmer OCR," *8th International Conference of Frontiers of Information Technology*, 2010.
- [15] S. Kheang, K. Katsurada, Y. Iribe, and T. Nitta, "Solving the Phoneme Conflict in Grapheme-to-Phoneme Conversion Using a Two-Stage Neural Network-Based Approach," *IEICE Transactions on Information and Systems*, vol. E97.D, no. 4, pp. 901-910, 2014, doi: 10.1587/transinf.E97.D.901.
- [16] Y. LeCun, "Deep learning & convolutional networks," in *2015 IEEE Hot Chips 27 Symposium (HCS)*, 2015, pp. 1-95, doi: 10.1109/HOTCHIPS.2015.7477328.
- [17] M. Matsugu, K. Mori, Y. Mitari, and Y. Kaneda, "Subject independent facial expression recognition with robust face detection using a convolutional neural network," *Neural Networks*, vol. 16, no. 5-6, pp. 555-559, 2003, doi: 10.1016/S0893-6080(03)00115-1.
- [18] A. Karpathy, A., "Connecting images and natural language," Thesis, Dept. of Computer Science, Stanford University, 2016.
- [19] K. Gregor and Y. LeCun, "Emergence of Complex-Like Cells in a Temporal Product Network with Local Receptive Fields," *CoRR arXiv:1006.0448*, 2010.
- [20] D.C. Cireşan, U. Meier, J. Masci, L.M. Gambardella, and J. Schmidhuber. "Flexible, high performance convolutional neural networks for image classification," in *IJCAI International Joint Conference on Artificial Intelligence*, 2011, doi: 10.5591/978-1-57735-516-8/IJCAI11-210.
- [21] B. Annanurov and N.M. Noor, "Khmer handwritten text recognition with convolution neural networks," *ARPN Journal of Engineering and Applied Sciences*, vol. 13, no. 22, pp. 8828-8833, 2018.
- [22] R. Venkatesan and M. J. Er, "A novel progressive learning technique for multi-class classification," *Neurocomputing*, vol. 207, pp. 310-321, 2016, doi: 10.1016/j.neucom.2016.05.006.
- [23] A. Krizhevsky, "Convolutional deep belief networks on cifar-10," in *Unpublished manuscript*, U.o. Toronto, Editor. 2010, Available: <https://www.cs.toronto.edu/~kriz/conv-cifar10-aug2010.pdf>.
- [24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.
- [25] J. Sueiras, V. Ruiz, A. Sanchez, and J. F. Velez, "Offline continuous handwriting recognition using sequence to sequence neural networks," *Neurocomputing*, vol. 289, pp. 119-128, 2018, doi: 10.1016/j.neucom.2018.02.008.

- [26] M. Al Rabbani Alif, S. Ahmed, and M. A. Hasan, "Isolated Bangla handwritten character recognition with convolutional neural network," in *2017 20th International Conference of Computer and Information Technology (ICCIT)*, 2017, pp. 1-6, doi: 10.1109/ICCITECHN.2017.8281823.
- [27] R. Zhang, Q. Wang, and Y. Lu, "Combination of ResNet and Center Loss Based Metric Learning for Handwritten Chinese Character Recognition," in *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, 2017, pp. 25-29, doi: 10.1109/ICDAR.2017.324.
- [28] K. R. Ayyalasomayajula, F. Malmberg, and A. Brun, "PDNet: Semantic segmentation integrated with a primal-dual network for document binarization," *Pattern Recognition Letters*, vol. 121, pp. 52–60, 2019, doi: 10.1016/j.patrec.2018.05.011.
- [29] D. Soselia, M. Tsintsadze, L. Shugliashvili, I. Koberidze, S. Amashukeli, and S. Jijavadze, "On Georgian Handwritten Character Recognition," *IFAC-PapersOnLine*, vol. 51, no. 30, pp. 161–165, 2018, doi: 10.1016/j.ifacol.2018.11.279.
- [30] R. Sabzi *et al.*, "Recognizing Persian handwritten words using deep convolutional networks," in *2017 Artificial Intelligence and Signal Processing Conference (AISP)*, 2017, pp. 85-90, doi: 10.1109/AISP.2017.8324114.

BIOGRAPHIES OF AUTHORS



Dr. Bayram Annanurov completed his Ph.D. at the Universiti Teknologi Malaysia in 2016. His main research area is Deep Learning. He is currently teaching programming and optimization at Paragon International University in Phnom Penh, Cambodia.



Dr. Norliza Mohd Noor is a professor at Razak Faculty of Technology and Informatics, Universiti Teknologi Malaysia, Kuala Lumpur Campus. Her research areas are image analysis and machine learning.