

Trace Transform Feature Learning for Offline Jawi Handwritten Recognition

Anton Heryanto Hasan^{a,1}, Khairuddin Omar^{a,2}, Muhammad Faizdul Nasrudin^{a,3}

^aFaculty Technology and Information Science, Universiti Kebangsaan Malaysia, Bandar Baru Bangi, 43650, Malaysia
Corresponding author: ¹anton.heryanto@gmail.com, ²ko@ukm.edu.my, ³mfn@ukm.edu.my

Abstract— Offline Jawi handwritten recognition is very important to allow efficient archiving and retrieving the original documents and increase the availability of the content. It is challenging task and still considered an open problem because the state-of-the-art recognizer performance is considered sub-par. The tradition trace Transform features extractor has potential, however the complexity of parameters tuning in feature engineered approach combine with independent non-learnable sub-word classifier produce sub-par Jawi sub-word recognition accuracy. The proposed trace Transform feature learning address the features extraction complexity by automatically discovers the features according to data. The features extractor and classifier trained end-to-end from raw input data to target class to find the optimum parameters. The trace transform process defined as layer similar with convolution process in Convolution Neural Network. This approach improves data representation and produce better Jawi handwritten recognition performance. trace Transform feature learning are more robust to Affine Transformations compared to the state-of-the-arts Convolution Neural Networks feature learning because its data representation invariant to rotation, slanting and skewing. This proposed feature learning performance evaluated with its performance in sub-word recognition performance using Jawi dataset. In this paper only single layer of trace transform feature learning compare with traditional trace transform feature and Convolution Neural network as the state-of-the-art feature learning. The performances are significantly better compared to traditional trace transform feature and able to compete with convolution neural network in single layer, three layers and comparable with eight layers.

Keywords— Jawi; handwritten recognition; sub-word; end-to-end learning; features learning; trace transform.

Manuscript received 16 Oct. 2019; revised 4 Jul. 2020; accepted 12 Sep. 2020. Date of publication 28 Feb. 2021.
IJASEIT is licensed under a Creative Commons Attribution-Share Alike 4.0 International License.



I. INTRODUCTION

Jawi is a Malay language script adapted from Arabic script. Six additional characters are introduced to support the Malay phonemes. The Jawi characteristic is almost similar to Arabic scripts and other variants such as Urdu and Farsi. However, Jawi had different challenges because of differences in language context and writing styles. Jawi is mostly found in old Malay manuscripts. Information retrieval of Jawi manuscript requires Jawi experts who are very few in numbers. Digitization and pre-processing the script will allow better information retrieval of the manuscripts. The motivation is to create the Jawi information retrieval to analyse the archives of historical manuscripts that will allow rapid perusal by scholars and researchers who wish to consult the original manuscript without physically reading the manuscript as most of them are in poor conditions. Offline Jawi handwritten recognition research is critical. It is a challenging task and still considered open problems because of its cursive nature, various writing styles, dialects, ligatures,

the overlap between characters, sub word part of the Arabic word, and the low quality of older manuscripts [1]. The sub words are forms when the disconnected type characters creating space within the word. It consists of the isolated characters and the sequence of characters.

Robust Feature extraction is essential in Offline handwritten recognition. The Features represent the structure of the object and help the classifier to distinguish the target class. Previous Jawi handwritten [2]–[6] explored various local and global Jawi features based structural and statistical approach, based on a human-oriented perspective. However, the features are limited and unable to represent various shapes of similar characters nor differentiate with different characters. Trace transform feature is the current state of the art feature uses by the best performing Jawi handwritten recognizer [7]. The feature consists of an object signature generated using various functions using the feature engineering approach. The object signature features of identical sub words are invariance regarding size or rotation. Despite its potential, the features engineering approach is

challenging to optimize and sensitive to tuning. It also requires complex feature selection to produce the best features. The feature is hand-crafted based on an engineered model to cover variants of data need to recognize. However, the modeled features parameter tuning is very sensitive and tends only to cover specific data variants. It requires considerable effort to cover all possible variants.

The circular natures of object signature features are not suitable when used by machine learning classifiers. Hence, similarity measurement is used to classify the sub-word [7]. Nevertheless, the recognition performance of the Jawi handwritten sub-word recognizer [7] still considers sub-pars. The use of non-learnable classifiers contributes to low recognition performance because it is profoundly dependable on the robustness of features. As shown by previous research, the potential of Trace Transform could be further optimized using the feature learning approach. Trace transform features adjusted based on data and tasks using a feature learning approach. Thus, the features will be suitable when combining with machine learning classifiers such as Feed Forward Neural Network (FFNN) or Support Vector Machine (SVM). The deep learning approach using feature learning and Feed Forward Neural Network classifier trained end-to-end is the state-of-the arts in images, voice, and natural language processing applications [8].

II. MATERIALS AND METHOD

Feature learning uses machine learning techniques to generate relevant features based on data for targeted tasks instead of modeled the feature based on the assumptions about what is relevant in the dataset. There are two approaches to feature learning methods, which are unsupervised and supervised. The unsupervised feature learning accurately represents the image by mapping the image into a compact feature vector. The supervised feature learning represents the image by classifying the shape from labeled data as a by-product of the discriminative task.

The unsupervised feature learning optimized the model parameters to accurately restores the object after some transformation. There are several unsupervised feature learning methods, which are shallow models such as k-means clustering, local linear embedding (LLE), principal component analysis (PCA), and independent component analysis (ICA). Besides, another method is a deep model, such as Restricted Boltzmann Machines [10]. Self-organizing feature map (SOFM) and Auto-Encoder (AE) are the other methods which have lots of variants [9], including Sparse Auto-Encoders [12], Contractive Auto-Encoders [13], [14], and De-noising Auto-Encoders [15]. The Auto-Encoder (AE) consists of an encoder that converts input into feature vector and a decoder that recover image from feature vector. The encoder and decoder are a feed-forward neural network model with single or multiple hidden layers. The compact feature vector is generated by minimizing the original image and the generated output by minimizing the mean square error. Only the encoder part is used as a feature's extractor [11].

The Restricted Boltzmann Machines or Sparse Coding method represent the image by inferring the latent features variables with observed pixels using a graphical model in a bipartite undirected probabilistic. The compact image representation is builded iteratively using optimum spare

decomposition of its elements to generate a based image dictionary to reconstruct the original image from the close representation [16]. Convolution Neural Networks (CNN) is one of the supervised features of the learning-based architecture of Neural Network. It has a particular convolution layer, which is the convolution process build into the Neural Network layer. Instead of fully connected the node inside the Neural Network Layer, the convolution layer had a local connection between the layer according to the convolution kernel's size. This approach reduces the parameters compare to traditional Fully Connected Neural Networks. The local connection allows local feature extraction, which mimics the mammal visual cortex. The earlier Convolution Neural Network consists of multiple layers of the Convolution layer and the pooling layer, which act as feature learning, and a classifier, a fully connected layer of Feed Forward Neural Network [17]. The strength of the Convolution Neural Network is on the multiple-layer arrangement of its Convolution layers. It represents the object's features in hierarchical form, from the low-level features to medium and high-level features. The latest architecture of Convolution Neural Network consists of multiple layers [18]. The Convolution Neural Network is the state of the arts in features learning. However, to get the best performance from it, a large dataset is required to train many Convolution Neural Network layers. The local connectivity is why it is easy to fool using adversarial surface attack, and sensitive to affine transforms such as rotation and slanting [19]. Trace Transform with suitable function generates robust features which invariant to affine transform [20]. Inspire by the Convolution Neural Network where the convolution process builds into Neural Network as a special layer. This research proposes the Trace Transform feature learning, which uses a special layer where the Trace transform process inside the Neural Network process.

A. Trace Transform

The Trace transform had a similar principle with Radon transform [21]. Image reconstructs by tracing the image using straight line in multiple angles, calculating function over it, and plotting into sinogram. Radon transform calculates integration function and uses significantly in the application of computerized tomography [22]. Thousands of features could be generated using a combination of functions that construct invariance features as a result of image transformation through algebraic invariance [20]. Therefore, a similar group image with a different affine transform state could be classified.

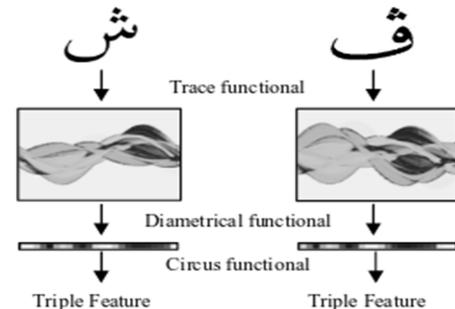


Fig. 1 The trace transform feature extraction of Jawi character. Initially, trace functional processes produce a sinogram, object sign generates a diametrical function and triple features calculated using circus function [7].

The trace transform has three functional types, the trace functional, diametrical functional, and circus functional as shown in figure 1. Function $f(x,y)$ is 2D function calculated over the trace line t with parameter (θ,p) where θ is the angle of the trace line to produce tomography representation known as sinogram [20]. Radon transform is one of the implementations of Trace transform where trace functional calculated using integral function along the trace lines. The diametrical functional calculate function on the trace functional result. This result features known as the object signatures. The circus functional further calculate the function on the result of the diametrical functional. The output is a single coefficient number known as a triple feature. This triple feature uses as identification keys for object identification [23].

$$\Pi(f) = \Phi[P[T[f(r, \theta, t)]]], \quad (1)$$

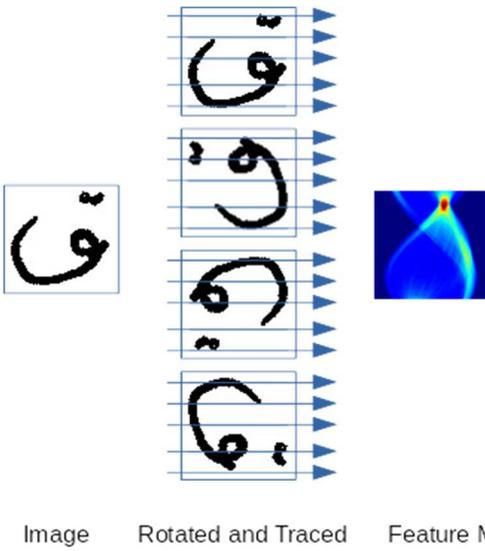


Fig. 2 Feature map of image generated using trace transform.

Trace transform produces affine transform invariance features where groups of images scaled, slanted, and rotated cloud by classified using algebraic invariance by modelled the knowledge image of transformation by tracing the image from various angles position than calculate features could produce invariance representation [24]. In this paper, we applied trace transform [25] and used a set of function [7] to generate a suitable representation for Jawi handwritten. This object signature feature generates from Jawi handwritten image calculated using equation 1. Figure 2 shown the feature map generation and algorithm are as the following:

- Define three parameters of Trace transform:
- Distance among point in trace line as t
- Distance among trace line in image as p
- Degree of trace line in full circle as ϕ in 360 degrees,
- Function that calculates T , P , and Φ ,
- Generate sinogram by calculated the trace function in each ϕ , $T=T[f(r,\theta,t)]$,
- Generate Object signature by calculate diametric function, $P=P[T[f(r,\theta,t)]]$ along trace of sinogram.

Trace transform features has been used for Jawi handwritten recognition [7]. There are 20 object signature

features generated using combinations of the function listed in table 1. First, the trace functional calculates at 60 trace lines and over 60 angles. Finally, object signature features are generated using diametrical function.

TABLE I
THE FUNCTION USES IN TRACE FUNCTIONAL AND DIAMETRICAL FUNCTIONAL FOR JAWI HANDWRITTEN RECOGNITION [7].

| No. | Functions |
|-------|---|
| T_1 | $\int_0^{\infty} f(t) dt$ |
| T_2 | $\int_0^{\infty} f(t) dt$ |
| T_3 | $\int_0^{\infty} f'(t) dt$ |
| T_4 | $\int_0^{\infty} f''(t) dt$ |
| T_5 | L_p quasi - norm ($p = 0.5$) = q^2 and $q = \int_0^{\infty} \sqrt{ f(t) } dt$ |
| T_6 | $f(t - c)$ weight median where c is abscissa median |
| T_7 | $ f(t) (t - c)$ weight median where c is abscissa median |

Figure 2 shows examples of object signatures features generated from identical characters. The object signatures pattern generated is similar. However, these features are not suitable for machine learning classifier. Based on [1] report, it leads to poor classification performances. These conditions happened because the values are different in scale and position.

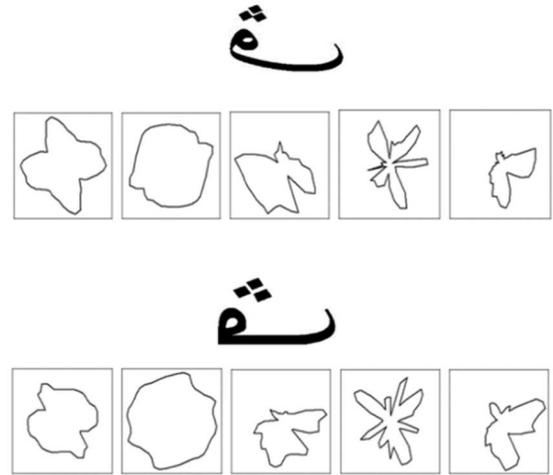


Fig. 3 The object signatures feature of Jawi characters.

To improve the feature generated by the Trace transform. Feature learning is used to generate features based on data and the target task. The feature learning uses weight to adjust the parameter to produce relevant features before feeding them to the classifier.

B. Trace Transform Feature Learning

This paper only presents a single layer implementation of trace transform feature learning. According to the parameter, the multiple angles trace line of image achieves by rotating the image in various angles. The rotation process is costly while training each image. This rotation only can be

conducted in the CPU, further restricting the use of multiple layers of trace transform layer as it causes the training process to be very slow and time-consuming. Therefore, this research scoped the implementation to a single layer trace transform feature learning. The multiple layers of Trace transform are only possible if the rotation is conducted in the GPU.

The trace transform feature learning uses the weight of the node's share by using one weight for each trace line and each theta angle rotation. This share weight approach reduces the parameters of the Neural Networks similar to the convolution layer. Each line's weight decides the importance of the features on that line according to trained data for given tasks.

This Trace transform is considered global features because the transformation is conducted on whole images. However, the uses of shared weight affected the locality of the feature generated. The pixel affected by the shared weight generated the local features, and it localizes the features derived from it.

The implementation of Trace transform feature learning inside Neural Network requires forward and backward trace functional. The forward function infers value in testing and also training. The function used in the forwarding process is similar to the previous function of trace functional. However, the backward process for training in the back-propagation process uses the derivative of the forward function. The function shown in table II is four simple functions based on the original seven functions [1], as shown in table 1.

TABLE II
TRACE TRANSFORM FEATURE LEARNING FORWARD AND BACKWARD FUNCTION.

| No. | Forward Functions | Backward functions |
|-------|-------------------------------|--------------------------------|
| T_1 | $\int_0^{\infty} f(t) dt$ | $\int_0^{\infty} f(x) dt$ |
| T_2 | $\int_0^{\infty} f(t) dt$ | $\int_0^{\infty} f'(t) dt$ |
| T_3 | $\int_0^{\infty} f'(t) dt$ | $\int_0^{\infty} f''(t) dt$ |
| T_4 | $\int_0^{\infty} f''(t) dt$ | $\int_0^{\infty} f'''(t) dt$ |

The trace transform network architecture consists of trace layer, batch normalization, ReLU (rectified linear unit) activation function, max pooling, and dropout regularization. At the trace layer, the input image is pre-resized and rotated for several angle steps. The trace function calculates in each pixel row for each angle step and multiples with shared weight. The classifier inputs are the feature learning layer output. The classifier used is a feed-forward neural network. This structure almost similar to the convolution neural network Layer.

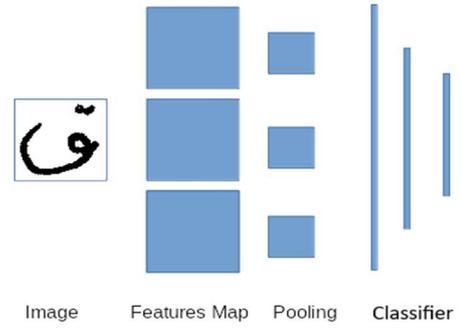


Fig. 4 Trace transform feature learning.

The back-propagation forward pass is straight-forward as it only multiplies the trace features values with share trace line weight. It is shown in figure 5. However, the back-propagation backward process passes the gradient to each feature map, but as the weight is shared, the gradient is summed up for each trace line. Figure 6 shows the process of vectorization.

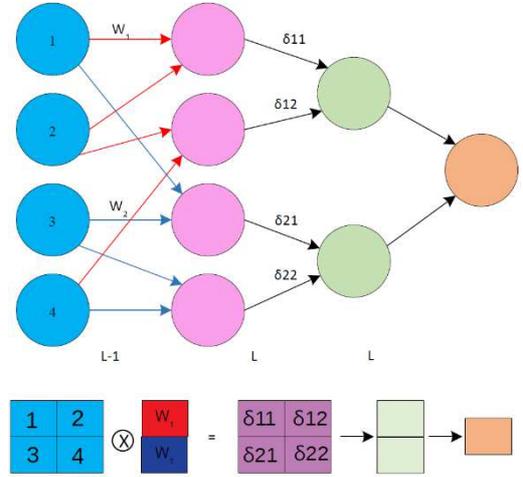


Fig. 5 Trace layer forward pass.

The network forward passes calculate the trace function of a pixel along the trace line and multiply the trace line's weight and produce one neuron feature map. This process is repeated for the number of trace lines and angle steps selected. Figure 5 shows the forward pass of Trace Layer.

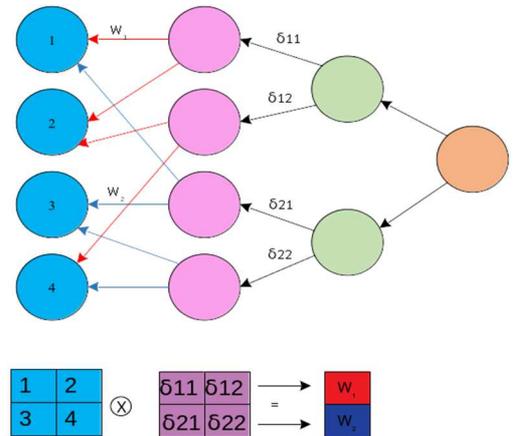


Fig. 6 Trace layer backward pass.

C. Evaluation

To evaluate the model, we used standard Jawi dataset [7] to compare to previous implementations. This dataset is multi-writer Jawi handwritten data based on the new standard of Jawi language. Figure 7 shows a random sample of Jawi handwriting sub-word contains in dataset.

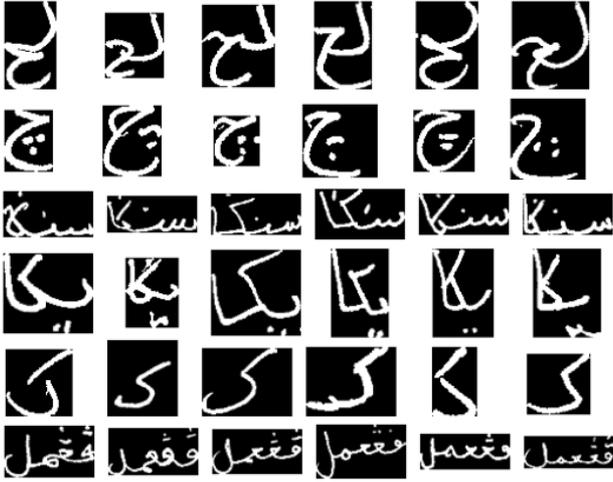


Fig. 7 Jawi Dataset sample [7].

The experiment was conducted on the same dataset used by [7]. Trace transform feature learning is compared with previous research of Trace transform feature engineering with 20 object signatures features and the state of the art of feature learning using Convolution Neural Network. Common Convolution Neural Network architectures including Single Layer Convolution Neural Network, 3 Layer Convolution Neural Network like LeNet [17], and 8 Layer Convolution Neural Network like AlexNet [8]. The methods are listed below:

- Trace transform object signature features with similarity measurement (OS).
- Convolution Neural Network with 1 Layer (CNN1).
- Convolution Neural Network with 3 Layer (CNN3).
- Convolution Neural Network with 8 Layer (CNN8).
- Trace Layer with line weight (TNRW).
- Trace Layer with theta weight (TNTW).
- Trace Layer with line and theta weight (TNRTW).

III RESULT AND DISCUSSION

Table III and figure 8 present the Jawi handwritten sub-word recognition accuracy between evaluated methods. The proposed method of accuracy is better compared to the previous implementation. It is proved that the feature learning approach of trace transform is better than the feature engineering approach of [7]. The performance is comparable with the Convolution Neural Network 3 layer and outperforms the Convolution Neural Network 1 layer even though it is still implemented in single layers.

Integrating the trace functional process inside the Neural Network layer improves the feature generalization as it selects the correct parameter based on data and tasks. Thus, feature correctly representation image. The pooling layer improve the feature representation by reduce more smaller variances and noise and compacted the features.

Trace layer using shared line and theta weight produce the best recognition accuracy of trace layer variants. This result proves feature that invariance to translation and rotation improve image representation

TABLE III
JAWI SUB-WORD RECOGNITION ACCURACY

| Methods | Min | Max | Avg | 1-5 |
|---------|-------|-------|-------|-------|
| TNRWSC | 74.26 | 82.96 | 79.89 | 94.13 |
| TNTWSC | 74.63 | 84.63 | 79.62 | 94.50 |
| TNRTWSC | 75.19 | 84.81 | 80.30 | 94.60 |
| CN1SC | 66.23 | 74.26 | 69.91 | 88.77 |
| CN3SC | 76.95 | 82.22 | 80.19 | 94.15 |
| CN8SC | 88.15 | 93.33 | 91.18 | 98.35 |
| OS | 58.27 | 61.84 | 60.19 | 78.34 |

The advance of Neural network research in architecture and training help to improve the performance of Trace transform features learning as integrated feature extraction combine with Feed Forward Neural Network classifier. The batch normalization significantly improves the result as it normalizes the input value to enable the network to learn more effectively. The rectified linear unit improved the training process as it only passes positive signals and reduced training problems with gradient vanishing and exploding. The final improvement came from drop-out which forced the network to learn more effective.

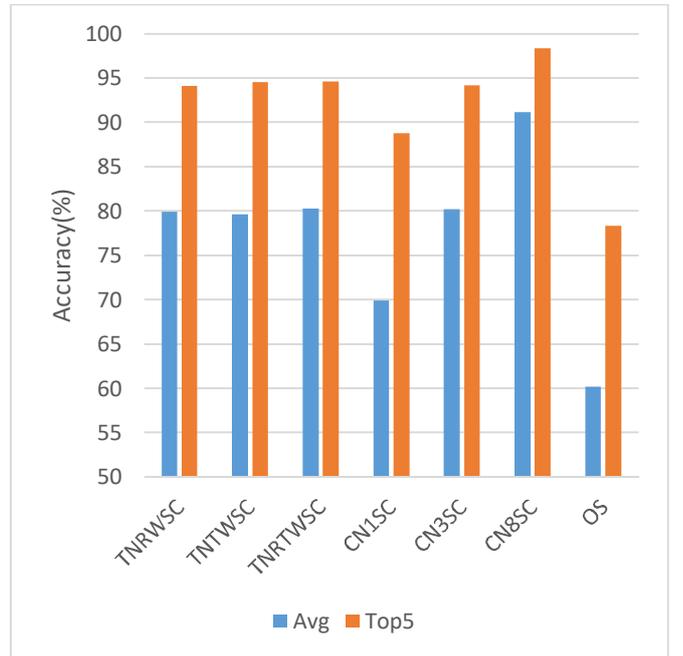


Fig. 8 Recognition performance accuracy.

IV. CONCLUSION

The Trace transform is a useful feature extractor for Jawi handwritten recognizer. This study is subject to feature learning approach improvement on the classification accuracy of in Jawi handwritten sub-word recognition. Its performance shows it could represent different variants of handwriting and its artifact. The Trace transform produces features that

invariant to affine transformation, thus correctly represent the image that re-translated, rotated, scaled, and slanted. Its recognition performance is comparable with the Convolution Neural Network even though the implementation is still in the single-layer architecture. The trace transform feature performance could be improved by implementing the multiple layer trace transform and local feature trace transform. The current implementation with image rotation is time-consuming. Implementation using matrix multiplication for process in Graphics Processing Unit (GPU) will improve its performance and enable multilayer implementation.

ACKNOWLEDGMENT

This research fund by Fundamental Research Grant Scheme (FRGS) with code FRGS/1/2016/ICT02/UKM/01/1. The fund provides the tools to experiment with bigger memory, high-speed solid-state drive and the Graphics Processing Unit (GPU), which significantly reduces experiment running time.

REFERENCES

- [1] Muhammad Faizul Nasrudin, Khairuddin Omar, Muhammad Shanudin Zakaria, and Liong, C.Y., "Handwritten cursive jawi character recognition: A survey," in Proceedings - computer graphics, imaging and visualisation, modern techniques and applications, cgiv, 2008, pp. 247–256.
- [2] Khairuddin Omar, "Jawi Handwritten Text Recognition Using Multi-Level Classifier (in Malay)," PhD thesis, Universiti Putra Malaysia, 2000.
- [3] Mazani Manaf, "Jawi Handwritten Text Recognition Using Recurrent Bama Neural Networks (in Malay)," PhD thesis, Universiti Kebangsaan Malaysia, 2002.
- [4] Anton Heryanto, Mohammad Faizul Nasrudin, and Khairuddin Omar, "Offline jawi handwritten recognizer using hybrid artificial neural networks and dynamic programming," in Proceedings of the international symposium on information technology 2008 2, 2008.
- [5] Remon Redika, Khairuddin Omar, and Mohammad Faizul Nasrudin, "Handwritten jawi words recognition using hidden markov models," in Proceedings of the international symposium on information technology 2008 2, 2008.
- [6] Mohd Sanusi Bin Azmi, "A Novel Feature from Combinations of Triangle Geometry For Digital Jawi Paleography," PhD thesis, Universiti Kebangsaan Malaysia, 2013.
- [7] Mohammad Faizul Nasrudin, "Offline Jawi Handwritten Recognition Using Trace Transform (in Malay)," PhD thesis, Universiti Kebangsaan Malaysia, 2010.
- [8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.
- [9] G. W. Cottrell and P. Munro, "Principal components analysis of images via back propagation," in *Visual communications and image processing '88: Third in a series*, 1988, vol. 1001, pp. 1070–1078.
- [10] N. Jaitly and G. E. Hinton, "Learning a better representation of speech soundwaves using restricted boltzmann machines," 2011.
- [11] N. Y. Hammerla, T. Plötz, S. Vajda, and G. A. Fink, "Towards feature learning for hmm-based offline handwriting recognition," in *International workshop on frontiers in arabic handwriting recognition*, 2010.
- [12] M. Ranzato, C. Poultney, S. Chopra, and Y. LeCun, "Efficient learning of sparse representations with an energy-based model," in *Proceedings of the 19th international conference on neural information processing systems*, 2006, pp. 1137–1144.
- [13] S. Rifai, G. Mesnil, P. Vincent, X. Muller, Y. Bengio, Y. Dauphin, and X. Glorot, "Higher order contractive auto-encoder," in *Joint european conference on machine learning and knowledge discovery in databases*, pp. 645–660.
- [14] S. Rifai, G. Mesnil, X. Muller, X. Glorot, and Y. Bengio, "Contractive auto-encoders: Explicit invariance during feature extraction," in *Proceedings of the 28th international conference on international conference on machine learning*, pp. 833–840.
- [15] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th international conference on machine learning*, 2008, pp. 1096–1103.
- [16] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, p. 607, 1996.
- [17] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.
- [18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.
- [19] S. Luan, B. Zhang, C. Chen, X. Cao, J. Han, and J. Liu, "Gabor convolutional networks," *CoRR*, vol. abs/1705.01450, 2017.
- [20] A. Kadyrov and M. Petrou, "The trace transform and its applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 8, pp. 811–828, Aug. 2001.
- [21] S. R. Deans, *the radon transforms and some of its applications*. Krieger Publishing Company, 1983.
- [22] P. Toft, "The Radon Transform - Theory and Implementation," PhD thesis, Department of Mathematical Modelling, Technical University of Denmark, 1996.
- [23] A. Kadyrov and M. Petrou, "Object descriptors invariant to affine distortions," in *Proceedings of the the british. Machine vision conference bmv2001*, 2001.
- [24] A. Kadyrov and M. Petrou, "Affine parameter estimation from the trace transform," *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, pp. 1631–45, Nov. 2006.
- [25] B. S. Shin, E. Y. Cha, K. W. Cho, R. Klette, and Y. W. Woo, "Effective feature extraction by trace transform for insect footprint recognition," 2008.