Jurnal Matematika MANTIK
Volume 5, Nomor 2, October 2019, pp. 83-89

ISSN: 2527-3159 (*print*) 2527-3167 (*online*)

K-Nearest Neighbor for Recognize Handwritten Arabic Character

Muhammad Athoillah

Universitas PGRI Adi Buana Surabaya, *athoillah.muhammad@gmail.com*

doi: https://doi.org/10.15642/mantik.2019.5.2.83-89

Abstrak: Pengenalan teks tulisan tangan adalah kemampuan sebuah sistem untuk mengenali tulisan tangan manusia dan mengubahnya menjadi teks digital. Pengenalan teks tulisan tangan adalah bagian dari masalah klasifikasi, sehingga algoritma klasifikasi seperti Nearest Neighbor (NN) diperlukan untuk menyelesaikannya. Algoritma NN adalah algoritma yang sederhana namun memberikan hasil yang baik. Berbeda dengan algoritma lain yang biasanya ditentukan oleh beberapa kelas hipotesis, Algoritma NN menemukan label pada titik uji tanpa mencari prediktor dalam beberapa kelas fungsi yang telah ditentukan. Bahasa Arab adalah salah satu bahasa terpenting di dunia. Mengenali karakter Bahasa Arab sangat menarik untuk dijadikan bahan kajian, tidak hanya karena merupakan bahasa utama yang digunakan dalam agama Islam tetapi juga karena jumlah penelitian tulisan tangan Arab yang ada masih jauh jumlahnya bila dibandingkan dengan penelitian pengenalan tulisan Latin atau Cina. Berdasarkan latar belakang tersebut, penelitian ini membangun sebuah sistem untuk mengenali Karakter tulisan Arab dari sebuah citra menggunakan algoritma NN. Hasil penelitian menunjukkan bahwa metode yang diusulkan dapat mengenali karakter dengan sangat baik ditunjukkan melalui rata-rata presisi, recall dan akurasi yang tinggi.

Kata kunci*: Klasifikasi, Karakter Bahasa Arab, Nearest Neighbor, Pengenalan Teks, Tulisan Tangan*

Abstract: Handwritten text recognition is the ability of a system to recognize human handwritten and convert it into digital text. Handwritten text recognition is a form of classification problem, so a classification algorithm such as Nearest Neighbor (NN) is needed to solve it. NN algorithms is a simple algorithm yet provide a good result. In contrast with other algorithms that usually determined by some hypothesis class, NN Algorithm finds out a label on any test point without searching for a predictor within some predefined class of functions. Arabic is one of the most important languages in the world. Recognizing Arabic character is very interesting research, not only it is a primary language that used in Islam but also because the number of this research is still far behind the number of recognizing handwritten Latin or Chinese research. Due to that's the background, this framework built a system to recognize handwritten Arabic Character from an image dataset using the NN algorithm. The result showed that the proposed method could recognize the characters very well confirmed by its average of precision, recall and accuracy.

Keywords*: Arabic Character, Classification, Handwritten, Nearest Neighbor, Text Recognition*

How to cite: *M. Athoillah, "K-Nearest Neighbor for Recognize Handwritten Arabic Character", Mantik, vol. 5, no. 2, pp. 83-89, October 2019.*

1. Introduction

Handwritten text recognition is the ability of a system to recognize human handwritten and convert it into digital text, this study was first conducted by T.L Diamond in 1958 [1]. Recently, the study of handwritten text recognition has been growing rapidly, the huge number of these researches is proved by the number of its publication such as Sadkhan et al. in 2018 who recognized handwritten based on ANN and wavelet Transformation [2], Iqbal and Zafar studied about Offline Handwritten Quranic Text Recognition in 2019 [3], Nguyen et al. who recognized Japanese handwritten using Recurrent Neural Network in 2018 [4] and much more. Handwritten recognition is divided into two categories, namely on-line and off-line recognition. The on-line recognition is a system that has the ability to recognize humanly handwritten directly from smartphones, personal digital assistant (PDA) and others similar device, whilst the off-line recognition is a system that needs to scan any images first to recognize the human handwritten[5].

Arabic is one of the most important languages in the world. Since it is a primary language used in Islam, recognizing Arabic character had become a very interesting task. Study of Arabic text recognition was started by A. Nazif in 1975. In his master's thesis, he built a system to recognize Arabic characters based on extracting strokes called radicals and their positions. He used correlation among the templates of the character image and the radicals, a segmentation phase was included in the cursive text segment [6]. However, the number of recognizing handwritten Arabic research is still far behind the number of recognizing handwritten Latin or Chinese research. This fact is based on the lack of Arabic digital dictionaries and programming tools or the lack of public databases of handwritten Arabic characters that found nowadays[5].

Handwritten text recognition is a form of classification problem, so a classification algorithm is needed to solve it, one of that algorithm is the Nearest Neighbor (NN). Nearest Neighbor algorithms are simple algorithm but give a good result. The idea of this method is to memorize the training data and predict the label of any new instance on the basis of labels from its closest neighbours in the training data. In contrast with other algorithms that usually determined by some hypothesis class, Nearest Neighbor Algorithm finds out a label on any test point without searching for a predictor within some predefined class of functions [7]. By default, the most function that is used to measure the distance between the data (neighbour) is Euclidean. Moreover, the parameter k is added which decided how many neighbours will be chosen for kNN algorithm. The suitable choice of k has a significant impact on the result of the kNN algorithm[8][9].

Due to that's the background, this framework built a system to recognize handwritten Arabic Character scanned from an image dataset using k-Nearest Neighbor (kNN) algorithm. The framework used MADBAse dataset provided by The American University in Cairo. The result of this work was presented through the values of its Precision, Recall and Accuracy while cross-validation method was used to validate the results.

2. Related Work

K-Nearest Neighbor (K-NN) was first introduced in 1951 and 1952 by Fix and Hodges [10][11] and then further study conducted by Cover and Hart in 1967 [12].

The algorithm was inspired by the hypothesis that "things that look alike must be alike" (Cover and Hart). Different with other classification algorithms, KNN is a classification method based on data that were located closest to the object without searching for a predictor within some predefined class of functions to find out the label on test point[7].

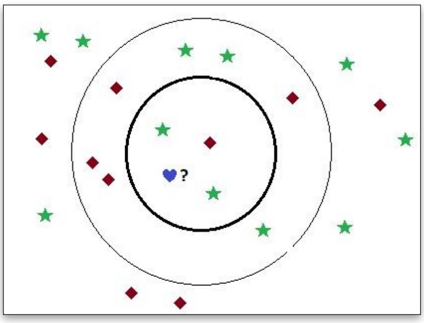

Figure 1. Illustration of K-NN Algorithm

Assume that dataset domain X is given with metric function p, then the function that returns the distance between two elements of X written as $p: X \times X \rightarrow \mathbb{R}$. For instance, if $X = \mathbb{R}^d$ then p can be Euclidean distance

$$p(x, x') = \|x - x'\| = \sum_{i=1}^{d} (x_i - x_i')^2$$

Given a sequence of training data as $S = (x_1, y_1), (x_2, y_2), \dots ,(x_m, y_m)$. For each $x \in X$, let $\pi_1(x), \pi_2(x), \dots , \pi_m(x)$ be a reordering of $\{1,2, \dots , m\}$ according to their distance to $x, p(x, x_i)$. Then, for all $i < m$,

$$p(x, x_{\pi_i(x)}) \leq p(x, x_{\pi_{i+1}(x)})$$

For a number k, the binary classification of k −NN rule is defined as follows:

- **Input**: sample of training $S = (x_1, y_1), (x_2, y_2), \dots ,(x_m, y_m)$
- **Output**: for every point $x \in X$, return the majority label within $\{y_{\pi_i(x)} \,; i \leq k\}$

If $k = 1$, then the 1-NN rule:

$$h_s(x) = y_{\pi_1(x)}$$

Since the output depends on the number of k, therefore the appropriate of k has a significant impact on the result of the kNN algorithm.

3. Implementation

This framework used a dataset from The American University in Cairo called MADBAse which contain images of Arabic handwritten Numeral Character from zero to nine-digit, the images used in this experiment amount to 500 in each object, which means there are 5000 images used in total.

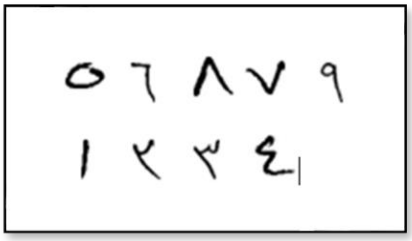

Figure 2. Sample of Dataset

The Image dataset was divided in two-part that was dataset for training process which contains 90 percent of the total image, whilst the rest 10 percent was used for the testing process. This experiment used histogram of the images as input, the histogram was chosen because the colour features which distinguish each image is generally represented by the colour histogram[13]. K-Nearest Neighbor is a simple algorithm, the process of training for this algorithm only contain storing feature vectors and labels of the images, while in classification process, the unlabeled query point is assigned to the label of its k nearest neighbours. When used K-NN, the object was classified based on majority labels among the k nearest neighbours. If k=1, the object was classified as the class from the object nearest to it. If there are only two classes, k must be an odd integer. However, there was still bond when k is an odd integer in case performing multiclass classification. In addition, Euclidean distance function was used to measure the distance between data.

4. Result and Discussion

In this section, the performance of the system to recognize the Arabic handwritten is delivered by computing its precision, recall and accuracy. Since the result of the K-NN algorithm is greatly influenced by the value of k, this framework measured all the performance used 6 sorts of k, namely 1, 3, 5,10 15 and 20. In addition, K-fold cross-validation was applied to validate the result of the system, this process was executed by dividing data into ten batches (k = 10), which are nine batches for training and the rest batch for testing, then the experiment was repeated k times under the condition that training data and testing data were always different during each process. The results are presented in the following table:

Table 1. The Average Result of System with $k = 1$ (%)

Class/Number	Precision	Recall	Accuracy
Zero	93,57	91,00	98,42
One	92,16	94,20	98,58
Two	22,34	20,40	84,68
Three	13,20	15,60	81,52
Four	14,89	15,40	83,02
Five	72,44	68,20	94,06
Six	40,31	25,60	88,72
Seven	17,22	18,60	82,70
Eight	20,79	26,40	82,50
Nine	24,27	21,60	85,20
Average off All	**42,99**	**41,71**	**88,24**

Table 2. The Average Result of System with $k = 3$ (%)

Class/Number	**Precision**	**Recall**	**Accuracy**
Zero	94,81	91,00	98,56
One	92,47	96,20	98,80
Two	23,22	20,80	84,80
Three	14,04	15,60	82,22
Four	17,38	17,80	83,46
Five	72,80	74,40	94,50
Six	44,09	27,60	89,18
Seven	18,31	19,20	83,28
Eight	21,24	27,60	82,42
Nine	26,53	23,40	85,50
Average off All	**44,26**	**43,36**	**88,58**

Table 3. The Average Result of System with $k = 5$ (%)

Class/Number	**Precision**	**Recall**	**Accuracy**
Zero	94,63	91,40	98,58
One	92,41	96,00	98,76
Two	25,82	24,00	85,36
Three	14,29	16,20	82,22
Four	19,07	18,80	83,78
Five	74,24	76,00	94,82
Six	50,80	32,60	89,96
Seven	20,45	19,80	83,94
Eight	20,06	25,80	82,16
Nine	28,45	26,00	85,74
Average off All	**45,75**	**44,51**	**88,84**

Table 4. The Average Result of System with $k = 10$ (%)

Class/Number	**Precision**	**Recall**	**Accuracy**
Zero	95,35	91,20	98,64
One	92,12	95,80	98,70
Two	28,47	26,80	85,50
Three	13,90	14,80	82,58
Four	21,49	21,60	84,20
Five	74,31	79,80	95,00
Six	53,01	35,80	90,08
Seven	23,35	20,80	85,10
Eight	21,66	28,80	82,60
Nine	32,77	29,40	86,56
Average off All	**47,07**	**46,16**	**89,16**

Table 5. The Average Result of System with $k = 15$ (%)

Class/Number	**Precision**	**Recall**	**Accuracy**
Zero	95,30	90,80	98,60
One	92,74	95,20	98,70
Two	31,61	29,00	86,00
Three	16,07	16,40	83,30
Four	22,90	22,80	84,44
Five	72,74	80,40	94,84
Six	51,72	39,80	89,86
Seven	23,58	19,20	85,30

Class/Number	Precision	Recall	Accuracy
Eight	22,22	28,60	82,64
Nine	34,12	30,60	86,88
Average off All	**47,65**	**46,91**	**89,30**

Table 6. The Average Result of System with $k = 20$ (%)

Class/Number	Precision	Recall	Accuracy
Zero	95,79	89,20	98,50
One	93,41	94,60	98,72
Two	34,07	30,60	86,10
Three	17,63	17,40	83,32
Four	22,78	23,20	84,54
Five	72,12	80,60	94,74
Six	49,60	41,60	89,64
Seven	25,15	18,20	86,06
Eight	23,55	28,80	82,92
Nine	34,51	33,00	86,90
Average off All	**48,23**	**47,13**	**89,39**

The result showed from Table 1 until Table 6 that K-NN could recognize the handwritten very well, proved by the lowest average of accuracy is 88,24% while the lowest average of its precision and recall are 42,99% and 41,71%. The highest performance belongs to the recognition of “Number one” and “Number zero”, this is because the shapes of them are the simplest and the most different from the other numbers which are only a straight vertical line for “Number one” and only a dot for “Number zero”. Otherwise, the lowest performance belongs to “Number Three” with the highest point of precision is 17,63%, 17,40% for recall point and 83,32% for its accuracy, that is because “Number Three” has a shape very similar to “Number Two” so those characters are often misclassified. This framework results also showed that the higher value of k provided the higher value of its accuracy, precision and recall. Nevertheless, it cannot be concluded that the higher value of k is always better because there is no clear evidence or definite guidance to determine the best value of k to get the optimal result. In general, a higher value of k makes it less sensitive to noise and cause smoother boundaries. As a result, it is almost impossible to choose the same best value of k for different applications[13].

5. Conclusion

This framework built a system that could recognize handwritten Arabic character using K-Nearest Neighbour (K-NN). Database from The American University in Cairo called MADBase was used in this framework that contains Arabic handwritten Numeral Character Images from zero to nine-digit with a total of 5000 images. Since k values have a lot of effect on the results, then this framework measured all the performance used six types of k values, i.e. 1, 3, 5, 10, 15 and 20. Relating to validation, K-fold cross-validation was applied to validate the result by dividing all data into ten batches then all of these batches were used for train and test process alternately. The result not only showed that the proposed method could recognize the characters very well confirmed by its average of precision, recall and accuracy but also showed that the higher k values produced the higher performance. Even so, it cannot be concluded that the higher k values always present a better result because each data has its own characteristics.

References

[1] T. L. Dimond, “Devices for reading handwritten characters,” in *Papers and discussions presented at the December 9-13, 1957, eastern joint computer conference: Computers with deadlines to meet*, 1957, pp. 232–237.

[2] S. B. Sadkhan and S. F. Jawad-SMIEEE, “Handwritten Recognition based on Hybrid ANN and Wavelet Transformation,” in *2018 Al-Mansour International Conference on New Trends in Computing, Communication, and Information Technology (NTCCIT)*, 2018, pp. 76–80.

[3] A. Iqbal and A. Zafar, “Offline Handwritten Quranic Text Recognition: A Research Perspective,” in *2019 Amity International Conference on Artificial Intelligence (AICAI)*, 2019, pp. 125–128.

[4] H. T. Nguyen, C. T. Nguyen, and M. Nakagawa, “Online Japanese Handwriting Recognizers using Recurrent Neural Networks,” in *2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, 2018, pp. 435–440.

[5] M. Shatnawi, “Off-line handwritten Arabic character recognition: a survey,” in *Proceedings of the international conference on image processing, computer vision, and pattern recognition (IPCV)*, 2015, p. 52.

[6] B. Al-Badr and S. A. Mahmoud, “Survey and bibliography of Arabic optical text recognition,” *Signal Processing*, vol. 41, no. 1, pp. 49–77, 1995.

[7] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[8] Z. Zhang, “Introduction to machine learning: k-nearest neighbors,” *Ann. Transl. Med.*, vol. 4, no. 11, 2016.

[9] F. Amin, “Identifikasi Citra Daging Ayam Berformalin Menggunakan Metode Fitur Tekstur dan K-Nearest Neighbor (K-NN)”, *mantik*, vol. 4, no. 1, pp. 68-74, May 2018.

[10] E. Fix and J. L. Hodges Jr, “Discriminatory analysis-nonparametric discrimination: consistency properties,” California Univ Berkeley, 1951.

[11] E. Fix and J. L. Hodges Jr, “Discriminatory analysis-nonparametric discrimination: Small sample performance,” California Univ Berkeley, 1952.

[12] T. M. Cover and P. Hart, “Nearest neighbor pattern classification,” *IEEE Trans. Inf. theory*, vol. 13, no. 1, pp. 21–27, 1967.

[13] B. Lei, E.-L. Tan, S. Chen, D. Ni, and T. Wang, “Saliency-driven image classification method based on histogram mining and image score,” *Pattern Recognit.*, vol. 48, no. 8, pp. 2567–2580, 2015.

[14] C.-M. Ma, W.-S. Yang, and B.-W. Cheng, “How the parameters of k-nearest neighbor algorithm impact on the best classification accuracy: In case of parkinson dataset,” *J. Appl. Sci.*, vol. 14, no. 2, pp. 171–176, 2014.