International Journal on Advanced Science Engineering Information Technology

Vol.11 (2021) No. 5
ISSN: 2088-5334

Classification of Acute Myeloid Leukemia Subtypes M1, M2 and M3 Using K-Nearest Neighbor

Nurcahya Pradana Taufik Prakisya [a,*], Febri Liantoni [a], Yusfia Hafid Aristyagama [a], Puspanda Hatta [a]

[a] *Department of Computer and Informatics Education, Universitas Sebelas Maret, Surakarta, Indonesia*
Corresponding author: [*]nurcahya.ptp@staff.uns.ac.id

***Abstract*— Leukemia is a malignant disease caused by the massive and rapid development of white blood cells in the bone marrow. These excessive white blood cells begin to interfere with the body's mechanism rather than fighting infection. Acute Myeloid Leukemia (AML) is one of the four main types of leukemia with eight subtypes, M0 to M7. AML M1, M2, and M3 have similarities, making them more difficult to distinguish from the other types. Furthermore, they are usually identified by calculating the ratio of myeloblast, promyelocyte, and monoblastic. This research aims to apply the k-Nearest Neighbor (k-NN) in classifying these cell types. k-NN is an algorithm used for classification based on a similarity measure. In cases of finding the best number of neighborhoods, trial and error were conducted. The features needed for classification are cell area, perimeter, roundness, nucleus ratio, mean and standard deviation. Four distance metrics such as Euclidean, Manhattan, Minkowski, and Chebyshev were used in this research. The results show that the Euclidean, Manhattan, Chebyshev, and Minkowski distance successfully identified 207 out of 300 objects at K=18, 197 out of 300 objects at K=13, 209 out of 300 correct objects at K=9, and 210 out of 300 objects at K=7. In conclusion, Minkowski was chosen as the best distance metric for KNN in classifying leukemia-forming blood cells. Furthermore, the accuracy, recall, and precision values of KNN with Minkowski distance obtained from 5-fold cross-validation were 80.552%, 44.145%, and 42.592%, respectively.**

***Keywords*— Acute myeloid leukemia; classification; K-Nearest Neighbor; white blood cell.**

Manuscript received 19 Aug. 2019; revised 6 Jun. 2020; accepted 13 Jan. 2021. Date of publication 31 Oct. 2021.

I. INTRODUCTION

Leukemia is a progressive malignant disease that occurs due to an excessive number of immature or abnormal white blood cells or leukocytes in the human body. They are created in the bone marrow and other blood-forming organs. Furthermore, these increased numbers can suppress normal blood cells' production, which leads to other blood-related diseases. In general, this disease consists of two types, acute and chronic leukemia. Both of which depends on how fast blast cells in the blood can multiply. Acute leukemia is characterized by the swift development of blast cells in the blood. It may lead to death in a matter of weeks or even days unless treated immediately [1]. However, the case is different in chronic leukemia as the blast cells multiply much slower in order to have a longer life expectancy [2].

This disease can also be further classified based on blast cell descendants, namely myeloid and lymphoid leukemia. Myeloid leukemia consists of white blood cells from the myeloid stem cell descendant [2]. In comparison, lymphoid leukemia is defined by the number of white blood cells from the lymphoid descendant. Based on the speed of immature cell development and cell descendant, leukemia is grouped into four main types: chronic myeloid, chronic lymphocytic, acute myeloid, and acute lymphocytic [3].

Acute Myeloid Leukemia (AML) is a blood cancer originating from the myeloid descendant. Patients would need to get the right treatment immediately because immature myeloid cells have fast growth. Furthermore, it is divided into eight groups of diseases based on the number of components of the white blood type that make up AML, namely M0, M1, M2, M3, M4, M5, M6, M7 [3]. These subtypes are shown in Table 1 below.

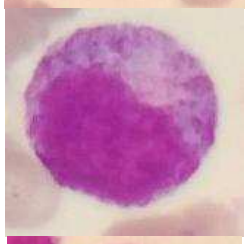
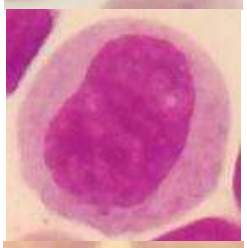
TABLE I
AML SUBTYPES

AML Subtype	Name
M0	Undifferentiated acute myeloblastic leukemia
M1	Acute myeloblastic leukemia with minimal maturation
M2	Acute myeloblastic leukemia with maturation
M3	Acute promyelocytic leukemia
M4	Acute myelomonocytic leukemia
M5	Acute myelomonocytic leukemia with eosinophilia

AML Subtype	Name
M6	Acute erythroid leukemia
M7	Acute megakaryoblastic leukemia

In AML M1, M2, and M3, three white blood cell types are used to determine what illness the patient suffered. They include *myeloblast, promyelocyte, and monoblast* [4]. The list of white blood cells sample images are shown in Table 2.

TABLE II
AML CONSTITUENT WHITE BLOOD CELLS

Type	Images
Myeloblast	
Promyelocyte	
Monoblast	
Others	

The total ratio of these three cells mentioned above becomes the criteria for determining AML M1, M2, or M3 classification. Table 3 shows the AML subtypes based on the present percentage of cell types. However, the values with additional asterisks are the most critical factors for determining the AML subtypes. While the remaining values represent the percent of the number of cell types that act as supports in the calculation [4].

TABLE III
AML CELLS FACTOR

Type	AML M1	AML M2	AML M3
Myeloblast	>89%*	30-89%*	<30%*
Promyelocyte	<10%	>10%*	>20%*
Monoblast	<10%	<20%*	<10%
Others	<10%	<10%	<10%

In AML M1, the presence of myeloblast in the blood must be more than 90%. In contrast, others should be less than 10%. Furthermore, the percentage of myeloblasts in the blood should be at 30% - 89% of all non-erythroid cells, and promyelocyte has to be more than 10% and monoblast less than 20%. The percentage of myeloblasts in AML M3 must be less than 30%. In the contrary, the appearance of white blood cells needs to be dominated by promyelocyte immature cells. Furthermore, monoblast and any other types of white blood cells are less than 10%.

One way to identify AML is to make observations manually, which is quite a time consuming [5]. However, this method is also vulnerable to misidentification. By using advanced technology, this obstacle can be removed with ease [6]. The purpose of this research is to speed up the classification process for the types of white blood cells and reduce the level of errors that might occur when identifying them. Therefore, the proposed method to achieve these goals is to use the k-Nearest Neighbor algorithm on normalized white blood cell object to image the feature data set. At present, k-NN is still very reliable in terms of classification. Besides being fast, the effort needed is also relatively low [7].

II. MATERIAL AND METHOD

The research started with data acquisition and ended with result analysis. Furthermore, each set of extracted features was normalized as input data in classification. Figure 1 shows the whole step of research.

A. Data Acquisition

1) Data Preparation and labeling: The AML M1, M2 and M3 data were acquired from dr. Sardjto Hospital, Yogyakarta. They were basically 33 images taken from three blood preparations. Each image was segmented to get the white blood cell objects. Subsequently, they were later extracted to obtain the features carried out by Harjoko et al., in previous research. In total 734 numerical data previously used in Harjoko et al., analysis was extracted.

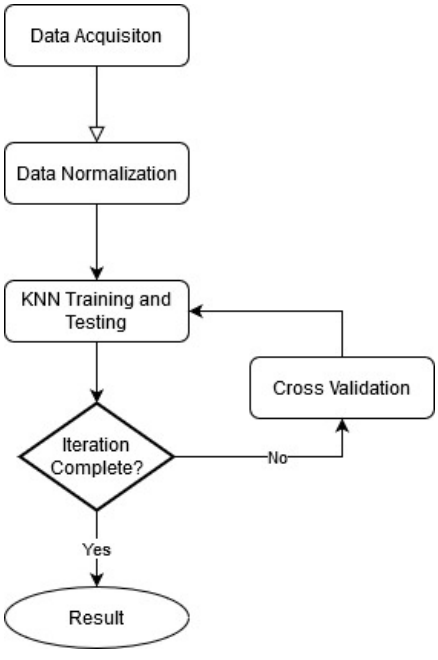


Fig. 1 Research steps

Every single object from AML preparations was labeled with three white blood cell types. There are three labels whose names match the original cell name: *myeloblast, promyelocyte, monoblast.* However, the other cell types which are not included in these labels are grouped with the name *support cell*. Table 4 shows the breakdown of the number of objects in each AML preparation.

TABLE IV
THE NUMBER OF DATA USED FROM EACH CELL TYPES

Type	AML M1	AML M2	AML M3	Subtotal
Myeloblast	201	159	17	377
Promyelocyte	6	22	101	129
Monoblast	0	19	10	29
Others	17	25	157	199
TOTAL	**224**	**225**	**285**	**734**

AML M1 has the highest number of myeloblast cells compared to other cell types. The total number of cells, which are the properly extracted objects on AML M1 was 224. Furthermore, the number of myeloblasts, promyelocyte, monoblast, and other support cell types in the AML M1 are 201, 6, 0, and 17 objects.

The number of myeloblast cells appears to decrease in the AML M2 preparation. On the contrary, monoblast cells begin to appear. Furthermore, the number of myeloblasts, promyelocyte, monoblast, and other support cell types in the AML M2 are 159, 22, 19, and 25 objects. Thus, the total data taken was 225.

AML M3 preparations contain 285 white blood cells where promyelocyte and support cells are more dominant among all existing cell types. The number of myeloblasts, promyelocyte, monoblast, and other support cell types in the AML M3 is 17, 101, 10, and 157 objects, respectively.

2) Input determination: Six features can be used as inputs for K-NN training [4]:

- *Cell area*: the number of pixels that form an area of white blood cell, including nucleus and cytoplasm.
- *Perimeter*: the outermost part of the cell object that is located right next to the background image.
- *Roundness*: a degree of curvature measurement of an object that forms a circle.
- *Nucleus Ratio:* the value obtained from the ratio of the nucleus area is divided by the cell body area.
- *Mean*: In this case, this is the average distribution of grey color intensity values of each pixel in a grayscale image.
- *Standard deviation:* measurement of the variation or dispersion of a set of value relative to its mean. It is also known as the square root of the variance.

B. Data Normalization

Data has a diverse variation and range; therefore, it needs to be normalized before entering the training stage. Data normalization aims to change the various number formats in the dataset to a common scale without distorting differences in the range of values. The formula for data normalization can be shown as follows:

$$normalization = \frac{data[x] - \min(data)}{\max(data) - \min(data)} \quad (1)$$

Where *data[x]* is the value of the x-index data, *min(data)* is the smallest value on each feature, and *max(data)* is the largest value of the data set from each feature [2][4].

C. K-Nearest Neighbor

The k-Nearest Neighbor is a supervised classification algorithm based on training data that has the closest distance to the object. The purpose of K-NN is to classify the new object based on attributes and K samples of training data with some nearest neighbors included in the contribution of the voting process [6]. Furthermore, the number of k depends on the case where k-NN is applied. If the number is large, the time and memory cost will also be huge, but if it is little, the nearby gauge will, in general, be extremely poor attributable to the meager information condition [8]. Therefore, it is essential to find the best value of K. For that to be done, trial and error needs to be conducted [9].

Training data is projected into multi-dimensional space, where each dimension has a feature data. This space is divided into several sections consisting of collections of learning data. Furthermore, a point in this space is marked as class c if its the most common classification to the closest K of that point.

In the training phase, K-NN stores features and class vectors of training data. While in the testing phase, these same features are calculated for the testing data. When new data is entered, its class is still unclear. However, the distance of this new one to all the learning data vectors must be calculated, of which the closest number of K is taken. The new point is classified to be included in most classifications of these points [10]. K-NN can be modeled in Figure 2 below.

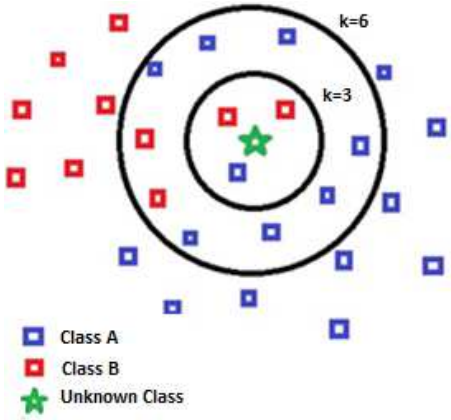


Fig. 2 K-nearest neighbor sample model

Figure 2 displays two classes, A and B. There is also a test data located right in the center of the circle. If k = 3 is used, then it can easily be seen that the data's closeness is more inclined to class A. However, if k = 6 is used, then the test data will be recognized as class B because it has a greater closeness to class B

The K-NN algorithm accuracy is greatly influenced by the absence or presence of irrelevant features. It is also influenced by the weight of a feature that is not equivalent to its relevance towards classification [11]. Furthermore, this algorithm is based on selecting and giving weight on features to find the

best classification performance. Until today, k-NN can still be used to classify diseases by pattern mainly related to texture[12].

D. Euclidean Distance

Euclidean distance is the most common distance metric used for KNN. It is a straight line distance between two data points (x_i, y_i) where $x_i, y_i \in R$ appears in the N-dimensional vector plane [13]. The equation for Euclidean distance is written below

$$d_{euclidean}(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \quad (2)$$

The distance between two points is simply calculated by finding the root squared difference of *x* and *y*. This formula is similar to the Pythagorean theorem formula.

E. Manhattan Distance

Manhattan distance is also known as the City Block distance, and it measures two points along the x and y axes at right angles. It is the absolute sum of the lengths of the line segment between the points in a plane with points at (x_i, y_i) where $x_i, y_i \in R$ appears in the N-dimensional vector plane [14]. The equation for Manhattan distance can be represented in Equation 3.

$$d_{manhattan}(x, y) = \sum_{i=1}^{n} |(x_i - y_i)| \quad (3)$$

F. Minkowski Distance

Minkowski distance is a metric in a normed vector space that can be considered a generalization of both the Euclidean and Manhattan distances. It is used as the dissimilarity measurement between two vectors at $x = (x_1, x_2, \ldots, x_n)$ and $y = (y_1, y_2, \ldots, y_n)$ where $x_i, y_i \in R$ appears in the N-dimensional vector space [15]. The equation for Minkowski distance can be represented in Equation 4.

$$d_{minkowski}(x, y) = \left(\sum_{i=1}^{n} |x_i - y_i|^l\right)^{1/l} \quad (4)$$

Furthermore, since it is a generalized distance metric, we can manipulate the above formula by substituting '*l*' to calculate the distance between two data points in different ways. For $l = 2$, the Minkowski distance gives the Euclidean distance. For $l = \infty$, the Minkowski distance gives the Chebychev distance [16].

G. Chebychev Distance

Chebychev distance is also known as chessboard distance or $l\infty$ metric, that is defined on a vector space where the distance between points (x, y) and $x_i, y_i \in R$ is the maximum absolute distance in one dimension of two N dimensional points[17]. The equation for Chebychev distance can be shown in Equation 5.

$$d_{chebychev}(x, y) := \max_i(|x_i - y_i|) \quad (5)$$

H. Validation

This method is vital in ensuring that the classification model is clean, correct, and reliable. K-fold cross-validation was used as a method for this study. This method is one of the most common cross-validation methods because it folds data into numbers of k by repeating (iterating) the training and testing process [18]. For every single iteration, one-fold is used as a test set, and the rest is used as a train set. However, test data takes turns by order of the k index. Figure 3 is an example of a 5-fold cross-validation.

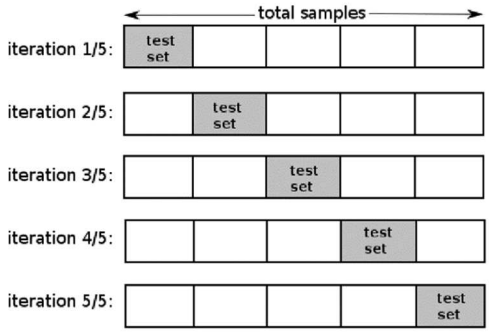


Fig. 3 K-fold cross-validation

Figure 3 shows a set of data that is divided into five segments or folds. In the first iteration, the first segment is used as the test data. Furthermore, the number of test data set is *1/5*n,* where *n* is the data set total number while the other four segments are used as a train set.

In the next iteration, the second fold of the data set is used as a test set, while the rest is used as a train set, including the very first fold. This iteration is done five times as *k*=5.

III. Results and Discussion

Data was acquired by featured extraction from 734 data conducted by Harjoko et al. Table 5 shows the detailed data recap based on the highest, the lowest and average value of each parameter and cell types.

TABLE V
DETAILED DATA RECAP

Cell type		*Myeloblast*	*Promyelocyte*	*Monoblast*	*Support*
Cell area (pixels)	Avg.	7377,05	13581,201	14923,72	9714,42
	Highest	14675	24099	23756	17010
	Lowest	2193	5876	8640	2216
Perimeter (pixels)	Avg.	322,4827	454,659	470,482	378,412
	Highest	493	703	678	537
	Lowest	180	277	337	176
Roundness (scale 0 – 1)	Avg.	0,8629	0,8153	0,8359	0,8245
	Highest	0,9719	0,9811	0,97184	0,9611
	Lowest	0,5939	0,4547	0,56533	0,5157
Nucleus Ratio (scale 0 – 1)	Avg.	0,8314	0,6016	0,6093	0,5238
	Highest	1	1	0,85866	1
	Lowest	0,5	0,3166	0,41671	0,0482
Mean (greyscale)	Avg.	135,3601	144,8222	149,6375	147,8820
	Highest	163,0481	167,7399	168,0211	175,6501
	Lowest	97,1572	111,9015	120,6002	114,3517
Standard Deviation (greyscale)	Avg.	20,0017	24,9116	24,2194	31,0159
	Highest	36,1478	33,8457	30,9531	46,0831
	Lowest	7,0259	14,1436	17,4565	13,1931

From Table 5, myeloblast has unique criteria. Its size of the area is relatively the smallest compared to other cell types. Its

has an average of 7377.0530 pixels. Furthermore, myeloblast's maximum area is only 14675 pixels, which makes it the cell with the highest roundness and nucleus ratio compared to the other three. Table 5 shows that its roundness average is of 0.8629, and its nucleus ratio average is 0.8314.

Promyelocyte and monoblast cell types are closely related. They both have an average value of four geometrical features that are not much different. However, both have differences in the value of color features, such as mean and standard deviation. The cytoplasmic color of monoblasts is purer blue while in the cytoplasm the promyelocyte cells appear pinker because there are visible granules.

Support cells have a relatively diverse range of feature values, and that is because there are more than only just one form of cells in the support cells type. Examples include lymphocytes, myelocytes, plasma and segment cells. They were deliberately grouped into a new type for easy classification.

Before entering the training phase, the raw data had to be normalized first. This method needed to be done because the extracted feature data still had a wide range of values. The range and data type of each feature can be described as follows:

- The cell area and perimeter have a range of original integer values.
- The roundness and nucleus ratio has a range of real number values between 0 and 1.
- Mean and standard deviation in the form of real numbers with range limitations between 0 and 255.

After the normalization, all features range were changed from varying scale from between 0 to 1. This scale would simplify the calculation process in classification.

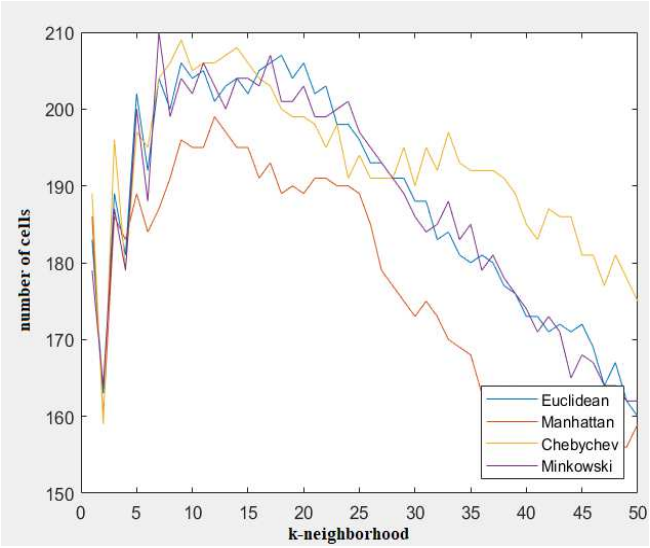


Fig. 4 Line graph of 50 K-nearest neighbor.

In the first stage, selecting the best distance metrics was carried out by dividing training and testing data into 434 and 300 through random data sharing. Four distance metrics were tested to find out the best one based on the maximum number of correctly predicted objects and the minimum K number. Furthermore, each metrics were tested in increasing value of K-neighborhood. It increased gradually starting from 0 and ending at 50. The result of 50 times k-NN iteration is shown in Figure 4.

X-axes is the number of k-neighborhood, and y-axes are the number of correctly predicted objects. The number of correct objects is perpendicular regardless of the size of k-neighborhood. Furthermore, the blue, red, yellow, and purple lines represent K-NN testing with Euclidean, Manhattan, Chebyshev and Minkowski distances, respectively.

Figure 4 shows that the Euclidean distance successfully identified 207 out of 300 objects at K=18. The second metric, Manhattan distance, only correctly identified 197 out of 300 objects at K=13. Chebyshev distance got 209 out of 300 correct objects at K=9. Meanwhile, Minkowski distance could acquire 210 out of 300 objects at K=7. Thus, Minkowski was chosen for the next steps as the best distance metric for classifying AML cells.

It was then later validated by using k-fold cross-validation in stage two. Also, the number of k-folds used in this study was 5. Therefore, each fold was 1/5 of the total data. Thus, it can be said that every fold could hold 147 data.

In the first iteration of 5-fold cross-validation, Fold number 1, which had 147 data, was used as the test data set. The rest four folds which contained 587 total data, were used as the train data set. This experiment was repeated five times, according to the proposed architecture. The test data partition position shifted in each iteration, in such a way that in the second iteration, the position of the test data set would be in the second fold and so on.

The experimental results show that some data can be appropriately identified. Every data that has been tested, whether correctly or incorrectly predicted, were counted. Table 6 shows the prediction results from 5-fold cross-validation.

TABLE VI
5-FOLD CROSS-VALIDATION

Fold	Correctly predicted	Incorrectly predicted	Subtotal
1	119	28	147
2	124	23	147
3	127	20	147
4	127	20	147
5	119	27	146
Total			**734**

There are some mispredicted data seen in Table 6. Misclassifications occurred because the features possessed by some cells were very similar such that they had very close degrees of neighborliness. Furthermore, these data are aggregated by category, i.e. true positive and negative, false positive and negative.

A true positive is an outcome where the objects correctly predict the positive class. Similarly, a true negative is an outcome where the model correctly predicts the negative class. Furthermore, a false positive is an outcome where the model incorrectly predicts the positive class, and a false negative is an outcome where the model incorrectly predicts the negative class [5]. Table 7 shows the confusion matrix from k-NN.

TABLE VII
CONFUSION MATRIX

		Actual values *Myeloblast*	*Promyelocyte*	*Monoblast*	*Others*
Predicted values	***Myeloblast***	347	19	5	20
Predicted values	***Promyelocyte***	13	96	15	8
Predicted values	***Monoblast***	0	3	2	0
Predicted values	***Others***	17	11	7	171

Confusion matrix was subsequently used as a basis in calculating the value of accuracy, recall and precision. Each class has the same accuracy value that totaled 83.9237% from this experiment. Table 8 shows detailed recall values for each class. The average recall value obtained from the table is 64.822%.

TABLE VIII
DETAILED RECALL OF EACH CLASS

Type	Recall
Myeloblast	92.0424403 %
Promyelocyte	74.4186047 %
Monoblast	6.8965517 %
Others	85.9296482 %
Average	**64.822 %**

Table 9 shows detailed precision values for each class. The average precision value obtained from the table was 77.788%.

TABLE IX
DETAILED PRECISION OF EACH CLASS

Type	Precision
Myeloblast	88.7468031 %
Promyelocyte	72.7272727 %
Monoblast	66.6666667 %
Others	83.0097087 %
Average	**77.788 %**

IV. CONCLUSION

Out of the four metrics offered in this study, Minkowski distance was chosen as the best metric capable of identifying white blood cell types forming leukemia M1, M2 and M3. This result is proved by the acquisition of the highest number of correctly predicted objects and the lowest k-neighborhood value obtained compared to the other three metrics in the test step with 210 out of 300 objects at k-neighborhood = 7. KNN with Minkowski distance was further analyzed with cross-validation to obtain accuracy, recall and precision. Although it can predict all object classes well, proved by 83.923% accuracy, it is less able to determine all the relevant class in the data set. Furthermore, the recall and precision obtained was 64.882% and 77.788%. The error occurred due to the variations in white blood cell that were too diverse. Some of which even had an only small portion of true positive results. They had a similar characteristic which makes the classification process more difficult. The given suggestions for the next research are the use of deep learning or genetic algorithm to classify blood cell types. Also, the amount of data needs to be increased so that the research validity can be better.

ACKNOWLEDGMENT

We would like to thank dr. Sardjito Hospital Yogyakarta and Harjoko for the permission to use the Acute Myeloid Leukemia: M1 M2 and M3 extracted features data.

REFERENCES

[1] A. Setiawan, A. Harjoko, T. Ratnaningsih, E. Suryani, Wiharto, and S. Palgunadi, "Classification of cell types in Acute Myeloid Leukemia (AML) of M4, M5 and M7 subtypes with support vector machine classifier," *2018 Int. Conf. Inf. Commun. Technol. ICOIACT 2018*, vol. 2018-Janua, no. Cml, pp. 45–49, 2018.

[2] P. Sachin and R. Y. Kumar, "Detection and Classification of Blood Cancer from Microscopic Cell Images Using SVM KNN and NN Classifier," *Int. J. Adv. Res.*, vol. 3, no. 6, pp. 315–324, 2017.

[3] E. Suryani, Wiharto, S. Palgunadi, and N. P. T. Prakisya, "Classification of Acute Myelogenous Leukemia (AML M2 and AML M3) using Momentum Back Propagation from Watershed Distance Transform Segmented Images," in *Journal of Physics: Conference Series*, 2017, vol. 801, no. 1.

[4] A. Harjoko, T. Ratnaningsih, E. Suryani, Wiharto, S. Palgunadi, and N. P. T. Prakisya, "Classification of acute myeloid leukemia subtypes M1, M2 and M3 using active contour without edge segmentation and momentum backpropagation artificial neural network," in *MATEC Web of Conferences*, 2018, vol. 154.

[5] S. Rajpurohit, S. Patil, N. Choudhary, S. Gavasane, and P. Kosamkar, "Identification of Acute Lymphoblastic Leukemia in Microscopic Blood Image Using Image Processing and Machine Learning Algorithms," *2018 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2018*, no. Cll, pp. 2359–2363, 2018.

[6] M. H. Waseem *et al.*, "On the Feature Selection Methods and Reject Option Classifiers for Robust Cancer Prediction," *IEEE Access*, vol. 7, pp. 141072–141082, 2019.

[7] S. S. Devi, A. Roy, M. Sharma, and R. H. Laskar, "kNN Classification Based Erythrocyte Separation in Microscopic Images of Thin Blood Smear," *Proc. - Int. Conf. Comput. Intell. Networks*, vol. 2016-Janua, pp. 69–72, 2016.

[8] M. P. Vaishnnave, K. Suganya Devi, P. Srinivasan, and G. Arutperumjothi, "Detection and classification of groundnut leaf diseases using KNN classifier," *2019 IEEE Int. Conf. Syst. Comput. Autom. Networking, ICSCAN 2019*, pp. 1–5, 2019.

[9] N. Zhang, W. Karimoune, L. Thompson, and H. Dang, "A between-class overlapping coherence-based algorithm in KNN classification," *2017 IEEE Int. Conf. Syst. Man, Cybern. SMC 2017*, vol. 2017-Janua, pp. 572–577, 2017.

[10] B. Harijanto, E. L. Amalia, and M. Mentari, "Recognition of the character on the map captured by the camera using k-nearest neighbor," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 732, p. 012043, 2020.

[11] H. Wisnu, M. Afif, and Y. Ruldevyani, "Sentiment analysis on customer satisfaction of digital payment in Indonesia: A comparative study using KNN and Naïve Bayes Sentiment analysis on customer satisfaction of digital payment in Indonesia: A comparative study using KNN and Naïve Bayes," 2020.

[12] N. Krithika and A. Grace Selvarani, "An individual grape leaf disease identification using leaf skeletons and KNN classification," *Proc. 2017 Int. Conf. Innov. Information, Embed. Commun. Syst. ICIIECS 2017*, vol. 2018-Janua, pp. 1–5, 2018.

[13] A. Singh and B. Pandey, "An Euclidean Distance based KNN Computational Method for Assessing Degree of Liver Damage," *Int. Conf. Inven. Comput. Technol.*, 2016.

[14] J. Williams and Y. Li, "Comparative Study of Distance Functions for Nearest Neighbors," *Adv. Tech. Comput. Sci. Softw. Eng.*, no. January, 2010.

[15] M. Klimo, O. Škvarek, P. Tarabek, O. Šuch, and J. Hrabovsky, "Nearest neighbor classification in minkowski quasi-metric space," *DISA 2018 - IEEE World Symp. Digit. Intell. Syst. Mach. Proc.*, pp. 227–232, 2018.

[16] B. Khaldi, F. Harrou, F. Cherif, and Y. Sun, "Improving robots swarm aggregation performance through the Minkowski distance function," *2020 6th Int. Conf. Mechatronics Robot. Eng. ICMRE 2020*, pp. 87–91, 2020.

[17] F. H. K. Zaman, I. M. Yassin, and A. A. Shafie, "Ensembles of large margin nearest neighbour with grouped lateral patch arrangement for face classification," *IRIS 2016 - 2016 IEEE 4th Int. Symp. Robot. Intell. Sensors Empower. Robot. with Smart Sensors*, no. December, pp. 6–12, 2017.

[18] S. Yadav and S. Shukla, "Analysis of k-Fold Cross-Validation over Hold-Out Validation on Colossal Datasets for Quality Classification," *Proc. - 6th Int. Adv. Comput. Conf. IACC 2016*, no. Cv, pp. 78–83, 2016.