Pixel Value Graphical Password Scheme: K-Means as Graphical Password Fault Tolerance

Original Scientific Papers

Mohd Afizi Mohd Shukran
Professor, Department of Computer Science,
Faculty of Defence Science and Technology, National Defence University of Malaysia
Kem Sungai Besi, Kuala Lumpur, Malaysia
afizi@upnm.edu.my

Mohd Sidek Fadhil Mohd Yunus
Senior Lecturer, Department of Computing,
Faculty of Arts, Computing and Industry Creative,
Sultan Idris Education University
Tanjung Malim, Malaysia
msidek@fskik.upsi.edu.my

Mohd Rizal Mohd Isa
Senior Lecturer, Department of Computer Science,
Faculty of Defence Science and Technology, National Defence University of Malaysia
Kem Sungai Besi, Kuala Lumpur, Malaysia

Fatimah Ahmad
Professor, Department of Computer Science,
Faculty of Defence Science and Technology, National Defence University of Malaysia
Kem Sungai Besi, Kuala Lumpur, Malaysia

Muhammad Naim Abdullah
Lecturer, Department of Computing, Academic Affairs,
University Malaysia of Computer Science and Engineering (UNIMY),
Selangor, Malaysia

Syed Muzzameer Syed Zulkiplee
Department of Computer Science,
Faculty of Defence Science and Technology, National Defence University of Malaysia
Kem Sungai Besi, Kuala Lumpur, Malaysia

Mohammad Adib Khairuddin
Senior Lecturer, Department of Computer Science,
Faculty of Defence Science and Technology, National Defence University of Malaysia
Kem Sungai Besi, Kuala Lumpur, Malaysia

Mohd Nazri Ismail
Professor, Department of Computer Science,
Faculty of Defence Science and Technology, National Defence University of Malaysia
Kem Sungai Besi, Kuala Lumpur, Malaysia

Mohd Fahmi Mohamad Amran
Senior Lecturer, Department of Computer Science,
Faculty of Defence Science and Technology, National Defence University of Malaysia
Kem Sungai Besi, Kuala Lumpur, Malaysia

Norshahriah Wahab
Senior Lecturer, Department of Computer Science,
Faculty of Defence Science and Technology, National Defence University of Malaysia
Kem Sungai Besi, Kuala Lumpur, Malaysia

Nur Adnin Ahmad Zaidi
Department of Computer Science,
Faculty of Medicine and Defence Health, National Defence University of Malaysia
Kem Sungai Besi, Kuala Lumpur, Malaysia

Abstract *– Pixel value access control (PVAC) was introduced to deliver a secure and simple graphical password method where it requires users to load their image as their password. PVAC extracts the image to obtain a three-octet 8-bits Red-Green-Blue (RGB) value as its password to authenticate a user. The pixel value must be matched with the record stored in the database or otherwise, the user is failed to authenticate. However, users which prefer to store images on cloud storage would unintentionally alter and as well as the pixel value due to media compression and caused faulty pixels. Thus, the K-Means clustering algorithm is adapted to fix the issue where the faulty pixel value would be recognized as having the same pixel value cluster as the original. However, most of K-Means algorithm works were mainly developed for content-based image retrieval (CBIR) which having opposite characteristics from PVAC. Thus, this study was aimed to investigate the crucial criteria of PVAC and its compatibility with the K-Means algorithm for the problem. The theoretical analysis is used for this study where the suitable characteristics of K-Means are analyze based on PVAC requirements. The compliance analysis might become a referencing work for digital image clustering techniques adaptation on security system such as image filtering, image recognition, and object detection since most of image clustering works was focused on less sensitive image retrieval.*

Keywords*: Cybersecurity, PVAC, Pixel Value, Graphical Password, Clustering, K-Means*

1. INTRODUCTION

The pixel value access control (PVAC) is a graphical password method that utilizing pixel value extracted from a digital image, referred to as Password Pixel or *PassPix*, to authenticate a username. PVAC was produced through the design and development of Pixel Value Graphical Password [1] idea in 2014 was motivated to solve the human memory burden on memorizing a strong and challenging password. PVAC is suited to be implemented as the guardian mechanism for server access which would operate in a cloud-based environment which is demanding nowadays especially during the current pandemic outbreak. This trend would bring the same way as the way PVAC users storing their *PassPix* where unfortunately would altering the pixel value unintentionally and caused users failed to authenticate and depict the concept of cloud facility.

As a study conducted in 2019 [2], an image that is transferred and accessible through WhatsApp [3] application would change an image attribute where the dimension is reduced and caused pixel value is transformed. Therefore, theoretically by adapting the K-Means [4] clustering algorithm would solve this issue theoretically by allowing PVAC to identify the faulty pixel value was residing in the same cluster as the pixel value it's supposed to be [5]. The idea is, when PVAC encounters a fail pixel value query during the authentication process, the Query By Example [6-7] or QBE method will be activated to query the specific similarity index (SI) range using the K-Means clustering algorithm.

However, most of K-Means-related works were based on how efficient the algorithm performed on content-based image retrieval system (CBIRS) [8] rather than access control system such as PVAC where both kinds of systems were having contrast criteria. Thus the adaptation of the K-Means clustering algorithm for PVAC must be involving a comparison analysis and algorithm fitting works.

2. THE PVAC

The current PVAC model [9] described that a user requires enrolling to the server first to create an access record with a unique username and upload an image as *PassPix*. As the provided username does not exist in the database, the PVAC will extract the pixel value from the fed *PassPix* as the password for the username and store them in the database. As for the authentication (Log-in) process, a registered user must repeat the same procedure as the enrolment (sign-up) process where the username and the *PassPix* are being fed again to the PVAC. PVAC will extract the newly fed *PassPix* for its pixel value again and perform the record query from the database. As the fed data and the database record are a match, PVAC will grant access to the user to the server. The process shows likelihood as the common log-in process as well as the user interface for PVAC by replacing the alphanumeric password with the *PassPix* as shown in (Fig. 1).

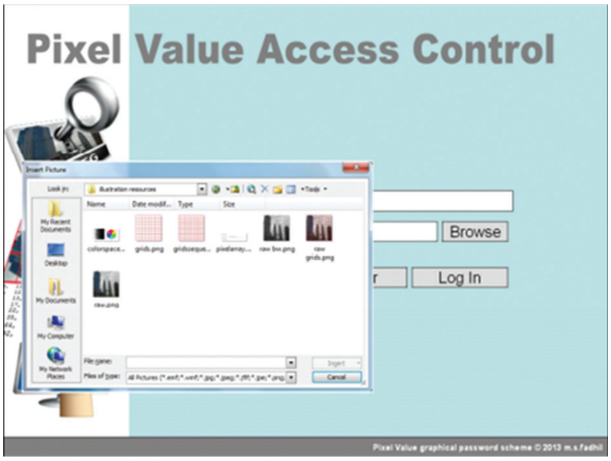

Fig. 1. The user interface for PVAC log-in

The pixel value extraction module plays a vital role in PVAC where it computes the RGB value that is presented as three octet 8-bits numbers (0,0,0 to 255,255,255). In general, a digital image is constructed by number of pixels that arranged in two-dimension resolution with X-axis and Y-axis which presented as an image attribute as Sum of X-axis pixel by Sum of Y-axis pixel (example: 200px * 200px, 1080px * 750px, or 2400px * 1030px). For example, the attribute resolution is 850px * 480px means the image is constructed 850 pixels on X-axis and 480 pixels on Y-axis that make 408,000 pixels in total (Σ_{px}). Every single pixel is holding the RGB color combination that made up the color of the pixel. The RGB color presentation is the standard color presentation as it widely applies for digital images, web colors, OS settings, and others where it is commonly referred to as RGB color-wheel [10]. The color of a pixel is built up by combining the hue and saturation (also referred to as strength for each color) of red, green, and blue that also create other colors as well such as yellow, violet, cyan, and others.

2.1. THE PIXEL VALUE

In a digital image, the colored pixel is arranged in two-dimension based where the combination of all colored pixels creates a region area and as well as to object visible on screen. The pixel value of a digital image is calculated by total up the strength value of red, green, and blue from each value and divide it with total pixels to get the average strength of red, green, and blue color (referred to as a color histogram) which can be calculated as:

$$\frac{\sum(R,G,B)}{\Sigma px} \tag{1}$$

In PVAC, before the extraction module performing the pixel value extraction process, the *PassPix* fed by users is divided into the logical grid where the dimension of the logical grids is a variable that the value could be modified by the system provider. As in a study on fake *PassPix* attempts [11], the logical grid is a feature for PVAC to extending the password space where it enables a *PassPix* could produce a single pixel value to an extremely long pixel value. The study also shows that it requires tremendous works to reconstructing the *PassPix* as the fake *PassPix* (as in case the pixel value is sniffed or leak from the server), compared to utilizing a single pixel value as the password. Currently, the fake *PassPix* reconstruction is a failed attempt that proved that the logical grid is a proven method. For example, if a developer decided to apply for 8 grids on the X-axis and 3 grids on the Y-axis, the total dimension (d) is 24. The pixel value on every grid is extracted which makes the total pixel value utilize on the PVAC is 24-pixel values. By using the same equation (1), the pixel value is calculated within the grid size which the Σ_{px} is the sum of pixels in the respective grid as well as the ∑(R, G, B) value. The pixel value is arranged into a single string object through the array process as the password is kept in the database. In other words, the extraction module on PVAC is performing three processes:

i. Logical grid process,

ii. Grid pixel value extraction process and

iii. Password array process.

The size of the logical grid (Gpx) is computed as:

$$G_{px} = \frac{x_{px}}{x_n} * \frac{y_{Px}}{y_n} \tag{2}$$

Where,
Xp_x is image dimension on x axis
Yp_x is image dimension on y axis
x_n is number of grids determined on x axis
y_n is number of grids determined on y axis

Then, PVAC is computing the pixel value as:

$$\frac{\Sigma_g(R,G,B)}{G_{px}} \begin{bmatrix} x_1y_1 & x_ny_1 \\ x_1y_n & x_ny_n \end{bmatrix} \tag{3}$$

Where,
$\sum_g$ is the sum of RGB value in a grid
x_n is the last grid column on x axis
y_n is the last grid row on y axis

2.2. FAULTY *PASSPIX* PROBLEM

As reported by Widjaja et al. [12] in 2018, almost 2 billion cloud storage active users recorded globally with a countless amount of file size stored, and this trend is forecast to increase over the next few years. Since the world is having a global COVID19 pandemic outbreak, the trend is not just extremely increasing, but also demanding. In addition, Wu et al. [13] mentioned, with cloud storage practices, users do not have to worry about data losses due to the failure of physical storage, data breach, or even Ransomware where all of the risks were mitigated to the service provider. This trend makes no exception for the PVAC users where the *PassPix* is stored cloudily for the above-mentioned reasons.

However, the cloud storage conditions and settings are cloudy where the effect on files is uncertain as found by Li et al. [14] where service providers might apply some compression to prevent certain deficiencies such as data processing consumption, storage insufficiency, and service delays. Such conditions would alter the file 8-bit attribution, especially multimedia files (digital images, digital sound, and digital video) as well as *PassPix* files. That will cause the PVAC to simply discarded the damaged *PassPix* and the affected users unable to authenticate.

Conceptually, a digital image clustering algorithm is an admissible method to integrate with PVAC since both methods are processing the digital image computation that involving the process of pixel value extraction. Despite those promising conceptual ideas, as digital image clustering was commonly designed for CBIR, the adaptation of digital image clustering is necessarily compliant with PVAC rules and regulations.

2.3. CRITERIA FOR PVAC

As the PVAC is meant to apply for client-server access control (user authentication security), there are a few requirements and rules that the digital image clustering method is strictly needed to comply with. Based on that, PVAC security-sensitive requirements are the major factor that regulated the digital image clustering algorithm selection. PVAC requires a digital image clustering algorithm that can compute the pixel value differently as specific as possible to prevent the PVAC could be tolerable to the fake *PassPix* authentication. In other words, the recognition ranges between the faulty *PassPix* and the original *PassPix* are subject to limit from the fake *PassPix*. By concept, during the query process, only *PassPix* that resides within the tolerable range is recognized as authentic *PassPix*. All of the PVAC criteria needed as fault tolerance mechanism are concluded in Table 1.

Table 1. The required PVAC rules and regulation

Requirements		Descriptions
Security	Tolerable Range	To reduce the vulnerability risk coming from fake PassPix
Features	Feature Extraction	PVAC is employing the pixel-based extraction method.
	Color-Space	PVAC is working on RGB colour-space.
	Logical-Grids Extractions	To preserve the password space strength, the clustering data is analyzed in two-dimension data.

3. KMEANS CLUSTERING ALGORITHM

In Partition-Based DIC strategy, there a variety of algorithms that performed the clustering process in many ways. Among them, the K-Means algorithm gains the most attention among researchers [15-17]. Nayini et al. [18] stated that K-Means is the most interesting DIC algorithm because of its capability to clustering large datasets with low computational complexity. Slamet et al. [15] suggested that the K-Means algorithm is a great help for efficiently understanding a complex dataset. In other words, most researchers are interested in K-Means due to their computational simplicity.

Basic K-Means at first was introduced by Macqueen [4] to classify a set of objects into predetermined groups. When the dataset receives a new object, he suggests the iteration process predetermine the group properties such as partition and centroid position. K-Means was equipped with the pixel value extraction function that translates a digital image into a string of pixel values which is similar to the PVAC pixel value extraction function as presented in equation (1). Then, the extracted object in the form of pixel value data is assigned to a cluster through the original K-Means algorithm.

Number (n) of the object (P) is the number of data (x) extracted using equation (1) and the Centroid seed (S) is the initial user to determine the centroid (μ) point for placing the centroids. The S value is randomly picked by a user could be any number and normally not greater than the n value.

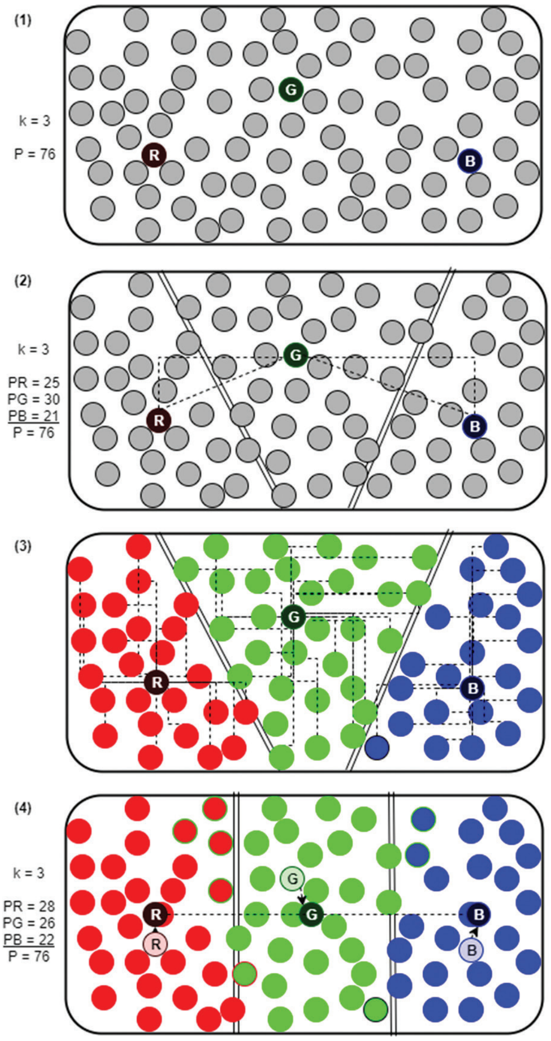

Fig. 2. 3 steps K-Means as in object matrix

For example, if a dataset containing 2,000 objects (n) to group into 20 clusters (k), the 20 S value is picked from 1 to 2,000. The number of seed must be equal to number or cluster (k), also a value that determined by users. However, as emphasized by Jin & Han [17] there is no absolute fail-proof framework to handpick the seed value.

The second step is assigning membership where each object (x) is assigned to the nearest centroid (μ) using the Euclidean distance (∈) that formulate as:

$$\epsilon = \sqrt{\sum_{k=1}^{n}(x-\mu)^2} \tag{4}$$

A partition or cluster border is built perpendicular to the mean point of Euclidean distance between centroids and objects that reside between centroid and cluster border is converged into a cluster. The next step is the iteration process where the centroid in every cluster is repositioned to the mean point of every object. When the centroid moved, the convergence process as in step 2 is performed again until no object is transferring to a different cluster. The iteration is a repetition process that happens every object transition in the object matrix. This 3-steps K-means object matrix is exemplified in (Fig. 2) for 76 objects that need to divide into 3 groups.

4. COMPLIANCE ANALYSIS OF K-MEANS ON PVAC

Referring to equation (3), PVAC is extracting the pixel value in the two-dimension logical grid to extend the password space and password strength as well. The multigrid pixel value is arranged into a string object through an array function. The efficiency of clustering for such data has been performed by Kumari et al. [19] on network packets using the K-Means algorithm where each packet is encapsulating the 8-bit data and multiple packets are forming informative data. The authors referring the technique as multiset clustering which every object is referred to as a subset in which every subset is holding series of packets. The term multiset clustering is applied as two-dimensional data as extracted from default PVAC pixel value extraction. The password derived from the array process on PVAC pixel value password creation is fit as the multiset structure object is an advantage. PVAC is avoiding the risk of adopting a harsh and complex multi-dimension algorithm that excessively consume computational resource.

In this study, the analysis is based on the result produced by the prototype of K-Means patch PVAC which was set up as an exhibit in (Fig.3).

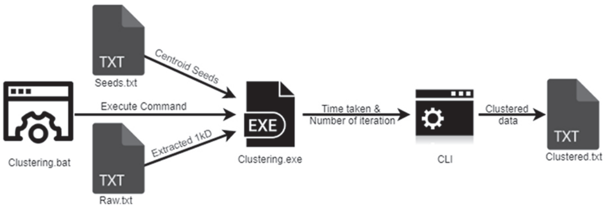

Fig. 3. Clustering experiment setup

There are 1,000 images use in this study which is randomly picked from various research dataset and gathered as one uncategorized dataset. To investigate the K-Means algorithm compliance to the PVAC feature requirement, the modified PVAC prototype is a challenge to perform clustering on the 1,000 images. The Clustering.exe is developed with the multiset clustering function which is required as the simulation of PVAC style data as discussed theoretically. Technically, the multiset clustering that is set on Clustering.exe is a set of Euclidean distance for every pixel value in each grid where every object is returning the set of Euclidean distance of every pixel value. The pixel value from every grid is locked to the object centroid or mean point for all grids to avoid the grid is scattered over the dataset. Then the RGB value in every grid is calculated to get the RGB distance different from the cluster centroid (*μc*) and the Euclidean distance of every grid pixel value becomes the attribute for the object.

4.1. FEATURE REQUIREMENT FOR PVAC

The desired observation for this study is the multiset clustering functionality that determined the feature requirement compliance and the accuracy of clustering result which determined the security requirement for PVAC. The data used for clustering is produced by the original or default PVAC *PassPix* extraction module without any additional coding. The *PassPix* extracted by PVAC in multiset clustering work as a subset and containing series of Euclidean distance of every grids pixel value that refers as Euclidean of a subset (∈⊂) where the whole object becomes one cluster. The ∈⊂ calculation for every object is derived from equation (4) that the *k* value is removed from the equation since the object's grids are not segmented and the object is calculated as in a matrix form. The ∈⊂ is calculated as:

$$\in\subset \;=\; \sqrt{\sum_{\subset}^{n}([G_1\; G_n] - \mu_c)^2}$$

Where,
n is the sum of grids extracted by PVAC
G_1 is the first grid
G_n is the last grid
⊂ is the subset or object in 1kD
μ_c is the centroid of the object cluster (5)

The multiset clustering is working as every grid is calculated as an object. Then, all of the grid is arranged in a dimensional sequence to form a set of grids to become an object before being calculated with default Euclidean distance (∈). By using equation (5), the K-Means algorithm can perform the clustering for 2 dimension data as extracted by default PVAC PassPix extraction module.

As a result, by functionality, the modeled K-Means clustering algorithm is accepted to perform the clustering process as required by PVAC as listed in table 1 previously. The ability of the K-Means clustering algorithm to calculate multiset object and cluster assignment task resulting K-Means clustering algorithm is fulfilling the all feature requirement of PVAC as concluded in table 2.

Table 2. The PVAC features requirement complied with K-Means multiset clustering

Feature Requirements	K-Means multiset clustering
Feature Extraction	K-Means process the data extracted by the default PVAC PassPix extraction mechanism using pixel-based extraction
Color-Space	Same as feature extraction, K-Means process the data extracted by the default PVAC PassPix extraction mechanism that produced the RGB color-space
Logical-Grids Extractions	Clustering.exe is patched with multiset clustering that worked on 2 Dimension data extracted by PVAC

4.2. SECURITY REQUIREMENT FOR PVAC

The second element is the query accuracy experiment; the result from the experiment is the key factor to determine the security requirements for the PVAC pixel fault tolerance mechanism. The query-based experiment is a way to investigate whether the K-Means algorithm would be able to query for similarity of faulty pixel *PassPix* with the clustered original *PassPix* data.

By default, when the query is performed, it will group all objects that are similar to the queried image as in CBIR. However, for this study, the desired query output was the closest distance between the queried image and the clustered database. The accuracy would prevent the fake *PassPix* is unable to bypass the fault tolerance mechanism and create the PVAC vulnerability.

As a simulation for the unintentionally images pixel value altering, all original *PassPix* dataset is transferred to WhatsApp repository and download it to local storage media as a compressed dataset. K-Means were challenged to query all of the compressed images with the clustered database produced from the previous experiment. This experiment is aimed to obtain the accuracies rate of all tested DIC algorithms where the accuracy is determined by the ability to return the queried compressed object with the rightful clustered object.

Observation from the result shows that K-Means with a higher number of clusters produce a higher accuracy rate as concluded in table 3.

Table 3. Query accuracy result

Number of *k*	Accuracy rate
k = 10	78.2 %
k = 20	78.3 %

The K-Means with the *k* parameter set to 20 is producing a 0.1% higher accuracy rate than 10 clusters. This can be concluded that the number of *k*'s is affected the accuracy rates where more *ks* produce more accurate query result. To estimate the effect of more k settings, a line graph is plotted as shown in (Fig. 4).

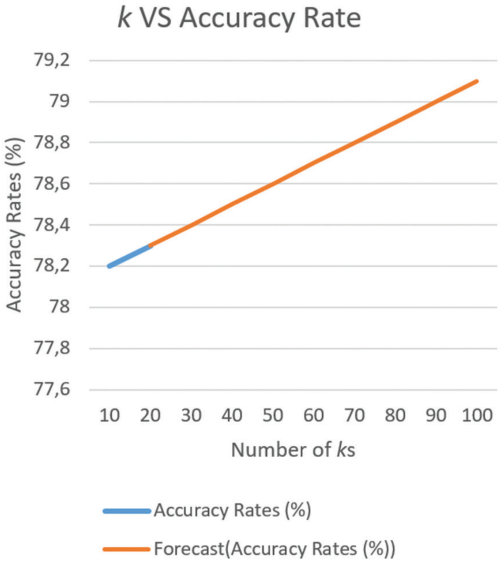

Fig. 4. *k* VS Accuracy rates

The k vs Accuracy rates graph shows that, estimated about 79.1% accuracy rate scored if the K-Means was set with 100 ks based on 10 *ks* and 20 *ks* accuracy rates. That proves that, for the 1,000 objects dataset, more ks are required to be set to increase the accuracy rate.

However, the number of *k* also affected the time taken by K-Means to perform the clustering process for such a dataset. K-Means with 10 *ks* setting taking 10 seconds to complete while K-Means with 20 ks taking 22 seconds. To gain more accurate results, more ks are required which would cause more time consumption as projected in (Fig. 5).

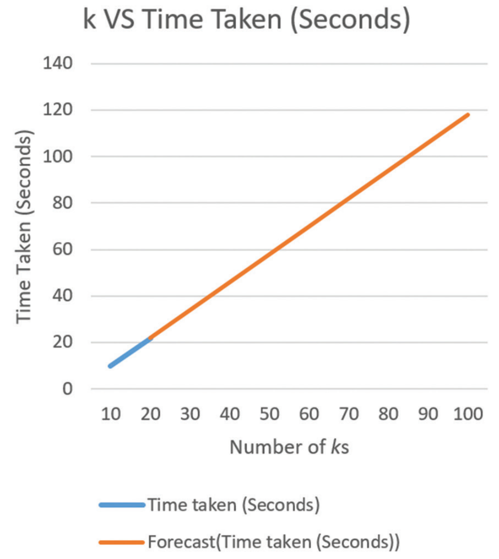

Fig. 4. *k* VS Time Taken

From the graph that forecasts based on *k* = 10 and *k* = 20 data, it can be concluded that the accuracy rate and the time consumption are both have to trade-off where to more time is needed to get a higher accuracy rate. It required about 2 minutes to complete the 100 ks clustering. Further works and effort on this issue might solve the issue as there is still more room for improvement.

5. CONCLUSION

Through this study, we can conclude that the K-Means clustering algorithm is a suitable algorithm to be adapted as a fault tolerance mechanism for PVAC where the multiset clustering enhancement and PVAC feature extraction can accomplish the clustering process.

This study also shows that, by applying more k, the accuracy of querying clustered databases is increased. This finding will reduce the possibility of depending on the similarity range module that would cause the range to exploit and be vulnerable. Besides, since the K-Means clustering algorithm is the most interesting algorithm among researchers, there are several variants derived from the K-Means algorithm which would enhance the performance and accuracy of the PVAC fault tolerance mechanism as well.

6. REFERENCES:

[1] M. A. M. Shukran, M. S. F. M. Yunus, "Method and System For Authenticating User Using Graphical Password For Access Control", Malaysia Patent MY-167835-A, 2018.

[2] M. A. M. Shukran, M. S. F. M. Yunus, M. N. Abdullah, M. N. Ismail, M. R. M. Isa, "Pixel Value Graphical Password: A PassPix Clustering Technique For Password Fault Tolerance", International Journal of Recent Technology and Engineering, Vol. 8, No. 3, 2019, pp. 2973-2975.

[3] WhatsApp Inc. WhatsApp Features, https://www.whatsapp.com/features/, (Accessed: 2019).

[4] J. MacQueen, "Some methods for classification and analysis of multivariate observations", Proceedings of the 5th Berkeley symposium on mathematical statistics and probability,1967.

[5] M. S. F. M. Yunus, "A Novel Graphical Password Clustering Method for Fault Tolerance Mechanism", National Defence University of Malaysia, Faculty of Defense Science and Technology, Kuala Lumpur, Malaysia, PhD Thesis, 2020.

[6] M. M. Zloof, "Query by example", Proceedings of the national computer conference and exposition, 19-22 May 1975, pp. 431-438.

[7] H. Kamper, A. Anastassiou, K. Livescu, “Semantic query-by-example speech search using visual grounding”, Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, Brighton, UK, 12-17 May 2019, pp. 7120-7124.

[8] M. Azam, N. Bouguila, “Bounded Laplace Mixture Model with Applications to Image Clustering and Content Based Image Retrieval”, Proceedings of 17th IEEE International Conference on Machine Learning and Applications, Orlando, FL, USA, 17-20 December 2018.

[9] M. S. F. M. Yunus, M. A. M. Shukran, M. N. Abdullah, “Pixel-based Graphical Password Scheme: Password from Digital Image File”, UPNM Press, Kuala Lumpur, 2019.

[10] M. A. M. Shukran, N. M. S. Ahmad, S. Ramli, F. Rahmat, “Melanoma Cancer Diagnosis Device Using Image Processing Techniques”, International Journal of Recent Technology and Engineering, Vol. 7, No. 5S7, 2019, pp.490-494.

[11] M. A. M. Shukran, M. S. F. M. Yunus, “Pixel Value Graphical Password Scheme: Fake Passpix Attempt on Hexadecimal Password Style”, International Journal of Information and Communication Sciences,Vol. 3, No. 3, 2018, p.104.

[12] A. E. Widjaja, J. V. Chen, B. M. Sukoco, Q. A. Ha, “Understanding users' willingness to put their personal information on the personal cloud-based storage applications: An empirical study”, Computers in Human Behavior, Vol. 91, 2019, pp. 167-185.

[13] K. Wu, J. Vassileva, Y. Zhao, “Understanding users' intention to switch personal cloud storage services: Evidence from the Chinese market”, Computers in Human Behavior, Vol. 68, 2017, pp. 300-314.

[14] C. Li, J. Bai, C. Yi Y. Luo, “Resource and Replica Management Strategy for Optimizing Financial Cost and User Experience in Edge Cloud Computing System”, in Information Sciences, 2019.

[15] C. Slamet, A. Rahman, M. A. Ramdhani, W. Darmalaksana, ”Clustering the Verses of the Holy Qur'an using K-Means Algorithm”, Asian Journal of Information Technology, Vol. 15, No. 24, 2016, pp. 5159-5162.

[16] S. Irfan, G. Dwivedi, S. Ghosh, “Optimization of K-means clustering using genetic algorithm”, Proceedings of the International Conference on Computing and Communication Technologies for Smart Nation, Gurgaon, India, 12-14 October 2017.

[17] X. Jin, J. Han, “K-means clustering”, Encyclopedia of Machine Learning and Data Mining, 2017, pp. 695-697.

[18] S. E. Y. Nayini, S. Geravand, A. Maroosi, “A novel threshold-based clustering method to solve K-means weaknesses”, Proceedings of the International Conference on Energy, Communication, Data Analytics and Soft Computing, Chennai, India, 1-2 August 2017.

[19] R. Kumari, M. K. Singh, R. Jha, N. K. Singh, “Anomaly detection in network traffic using K-mean clustering”, Proceedings of the 3rd International Conference on Recent Advances in Information Technology, Dhanbad, India, 3-5 March 2016.