International Journal of Electrical and Computer Engineering (IJECE)
Vol. 10, No. 3, June 2020, pp. 2367~2374
ISSN: 2088-8708, DOI: 10.11591/ijece.v10i3.pp2367-2374

Combined cosine-linear regression model similarity with application to handwritten word spotting

Youssef Elfakir, Ghizlane Khaissidi, Mostafa Mrabti, Driss Chenouni, Manal Boualam
Laboratory of Computing and Interdisciplinary Physics, ENS, Sidi Mohamed Ben Abdellah University, Morocco

Article Info

Article history:

Received Apr 5, 2019
Revised Nov 9, 2019
Accepted Nov 25, 2019

Keywords:

Bag-of-visual word
Features extractions
Floating threshold
Handwritten Arabic documents
Similarity distance

ABSTRACT

The similarity or the distance measure have been used widely to calculate the similarity or dissimilarity between vector sequences, where the document images similarity is known as the domain that dealing with image information and both similarity/distance has been an important role for matching and pattern recognition. There are several types of similarity measure, we cover in this paper the survey of various distance measures used in the images matching and we explain the limitations associated with the existing distances. Then, we introduce the concept of the floating distance which describes the variation of the threshold's selection for each word in decision making process, based on a combination of Linear Regression and cosine distance. Experiments are carried out on a handwritten Arabic image documents of Gallica library. These experiments show that the proposed floating distance outperforms the traditional distance in word spotting system.

Copyright © 2020 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

Youssef Elfakir,
Laboratory of Computing and Interdisciplinary Physics,
National Superior School, Sidi Mohammed Ben Abdellah University,
Fes 30050, Morocco.
Email: youssef.elfakir1@usmba.ac.ma

1. INTRODUCTION

In the pattern recognition fields, the objects can be represented as sequence of features, where a similarity measure can be used as a tool to judge the similarity or dissimilarity between two sequences of features. From literature, similar is defined as looking or being near between two objects, but not the same. In this field, several similarity and distance measure have been used widely in various fields of the area: text similarity [1], document similarity [2], handwriting recognition [3-5], handwritten character [6, 7], speech recognition [8-10], video analysis [11]. So use a good similarity measure is fundamental step for many types of application and domain such as information retrieval and recognition, chemistry, clustering or classification.

The term similarity in the pattern recognition is mean a score that represents the strength of relationship between two sequences of data items. The pattern recognition or the word sporting is done based on a similarity measure to search a similar image that are looking or being near together. Mathematically, similarity distance is used to calculate how far between two sequences of data, is also named dissimilarity in other domains, or the concept is though the same and inversely [12]. Figure 1 shows the chronological overview of the similarity and distance measure that is commonly used in various fields, based on feature sequences of data [13, 14], which are divided into three different groups, one group distance based, the second and the third groups of similarity measure are respectively correlation and non-correlation based.

Similarity measure is an important topic been extensively applied in some fields such as decision making, pattern recognition, machine learning and market prediction based on distance function such as Euclidean, Manhattan, Minkowski and Cosine distance similarity, etc. Or, many limitations associated with distance measures which have been have mentioned in this paper and are aiming to overcome in our research.

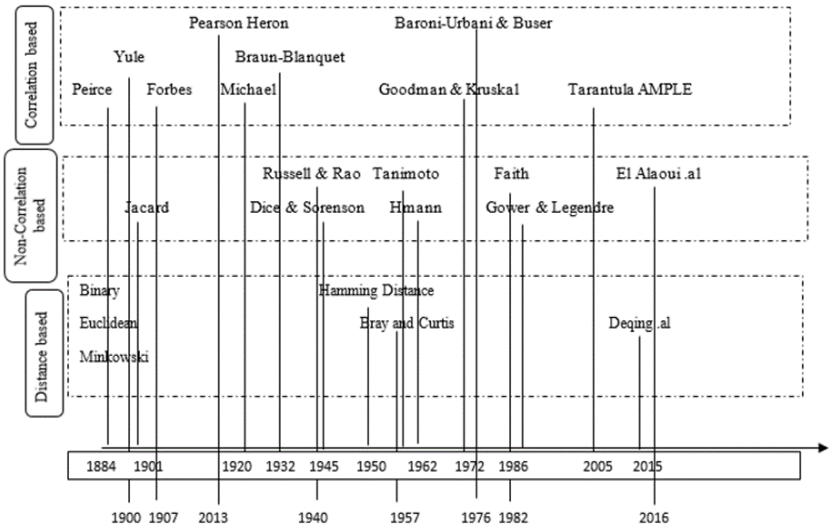

Figure 1. Chronological overview of the similarity and distance measures

The organization of this paper is as follows. First we describe on some of chronological overview of the similarity and distance measures used for similarity searching and retrieving systems. Then, we investigate the limitations of the existing distances in word spotting system in detail. Finally, we propose a floating distance based on the combination of Linear Regression and cosine distance and we apply them on a handwritten Arabic image documents of Gallica.

2. OVERWIEW OF SIMILARITY AND DISTANCE METRICS

For years, different distance measures are used to calculate the similarity between two data objects, the aim of these metrics is to found a specific distance function that allow a separation or classification between elements in a set of data [15, 16]. In fact, these distances operate directly in various fields as similarity searching and retrieval systems where the performance depends upon choosing function. Here, we present several similarity distances commonly used in pattern recognition and word spotting systems.

2.1. Euclidean distance

The Euclidean distance (1) is defined as the general and the standard metric for used in geometrical cases, the mathematical formula of this distance is defined as :

$$D_{xy} = \sqrt{\sum_{i=1}^{n}(X_i - Y_i)^2} \quad (1)$$

Which is a simple metric between two points of data (X, Y) that determines the root of square difference between them, is the default metric in the k-means algorithm and the most used in different clustering problems.

2.2. Manhattan distance

Manhattan or Taxicab distance (2) is a metric that calculates the absolute difference between two points of data, is also known as L1 distance, L1 distance or rectilinear distance, is defined as:

$$D_{xy} = \left|X_{ik} - X_{jk}\right| \quad (2)$$

2.3. Minkowski distance

The Minkowski distance (3) known as the generalized metric, can be used in case of data that are ordinal and quantitative. The distance of order p between two variables is defined as:

$$D_{xy} = \sqrt[p]{\sum_{i=1}^{n}(X_i - Y_i)^p} \quad (3)$$

This distance is similar to the Euclidean distance when p=2 and the Manhattan distance when p=1.

2.4. Cosine distance

Cosine distance (4) is the cosine of the angle between two vectors in an n-dimensional space given by the following formula:

$$D_{xy} = \frac{\text{X.Y}}{\|\text{X}\|*\|\text{Y}\|} = \frac{\sum_{i=1}^{n} X_i * Y_i}{\sqrt{\sum_{i=1}^{n} X_i^2} * \sqrt{\sum_{i=1}^{n} Y_i^2}} \quad (4)$$

This distance represents the dot product between two vectors divided by the product of the two vectors' lengths, and is often used in information retrieval and text mining.

2.5. Jaccard distance

Jaccard distance (coefficient), a term coined by Paul Jaccard, measures similarities of the two data items. The Jaccard distance (5) is measured by the following formula:

$$D_{xy} = \frac{|X \cap Y|}{|X \cup Y|} \quad (5)$$

The function is best used when calculating the similarity between small numbers of sets, and is often used to in the recommendation flied.

2.6. Chebyshev distance

Chebyshev distance (6), is also called Tchebychev or the maximum value distance. This metric calculates the absolute magnitude of the difference between two points of data by following:

$$D_{xy} = max_k\left(\left|X_{ik} - X_{jk}\right|\right) \quad (6)$$

3. COMPARISON OF DISTANCES METRIC IN WORD SPOTTING SYSTEMS

The aim of Query-by-example word spotting is search and localize they regions that are similar for a given query. Or to get the best performance still as a challenge, because several tasks influence on this system, as image processing, feature extractions, similarity measures, classification. While our work focuses on the influence of similarity measure distances in the context of a handwritten document images retrieval where many existing metrics can be adapted. We underline that the presented measure of similarity is applicable to any problem where the size of query is different one from the other.

In this part, we present a comparison of distance measures in word spotting system. For this, we use the Arabic handwritten document images form Gallica, which is the digital library of the National Library of France, in open access. First, we divide the document images into a set of local regions, densely sampled; these local regions are the basic structure used to spot the words in the document, and we extract all interests points from the images using SIFT [17] detector, and we use SIFT descriptor to represent each interest point in the images by their descriptor. Then, in the learning step, we select the 4 first images of document to group all descriptors to provide bag of visual descriptors and cluster centers, in this stage, the k-means algorithm is used with different number of centers (100,200,300,400). In the end, we encoding each region by a histogram using a bag-of-visual-descriptors [18, 19] instead to represent them by their descriptors. All regions are described with their histograms by assigning each descriptor in the region to the nearest visual descriptor in the codebook. So, each region is represented by a histogram of accumulates frequencies $H_{\text{j}} = H_{1,j}, H_{2,j}, H_{3,j}, H_{4,j}, H_{5,j}, \ldots\ldots H_{N,j}$ where N represents the number of the visual descriptor in the codebook and $H_{N,j}$ represents the cumulative frequency of k visual descriptor in the j$^{\text{eme}}$ region.

The process of the used system is presented in Figure 2. To evaluate the effect of similarity measure and determine the appropriate distance in the context of word spotting, we change this metric and we calculate the detection accuracy represented by F-score for different queries. The F-score is calculated as follows:

$$\text{F}_{\text{Score}} = 2.\frac{1}{\frac{1}{\text{recall}}+\frac{1}{\text{precision}}} = 2.\frac{\text{recall}+\text{precision}}{\text{recall}+\text{precision}} \quad (7)$$

Table 1 shows the F-score results for Gallica dataset, we observe that the F-score depending on the codebook size and also the metric distance used. For the cosine metric and code book size 300 we obtain F-score=0,78. While for the other metric we reported F-socre=0, 62 for Euclidean distance, F-socre = 0,52 for Chebyshev distance. So, the best result is obtained for the cosine metric and 300 clusters.

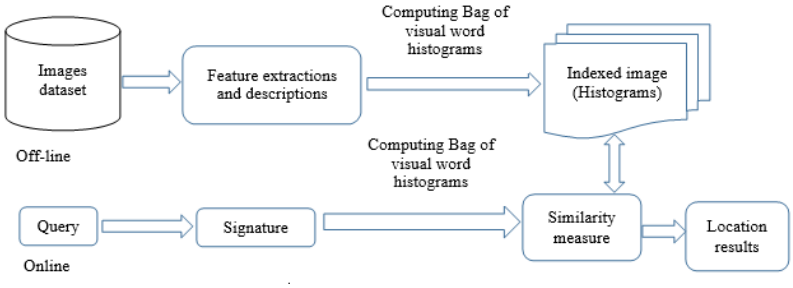

Figure 2. Process of the proposed system

Table 1. F-$_{score}$ for different codebook sizes and different metric distances

Codebook size	100	200	300	400
Euclidean distance	0.27	0.58	0.62	0.53
Manhattan distance	0.4	0.46	0.5	0.48
Minkowski distance	0.15	0.23	0.34	0.30
Cosine distance	0.627	0.76	0.78	0.629
Jaccard distance	0.27	0.36	0.5	0.41
Chebyshev distance	0.37	0.42	0.52	0.45

4. LINEAR REGRESSION MODEL SIMILARITY WITH APPLICATION TO HANDWRITTEN WORD SPOTTING

4.1. Analysis of cosine distance measure for word spotting

As shown in the previous part, the metric that gives the best result is the cosine distance. However, Table 1 shows that the size of the codebook influence directly on the performance. So we believe it is therefore necessarily to analyses diverse phenomena that appear when using codebook in high-dimensional spaces, this phenomena can be explain by the curse of dimensionality, the term was invented by Richard E.Bellman in [20, 21]. For this, the system presented in Figure 2 is therefore used to evaluate in detail different sizes of the codebook from 100 to 900 centers.

Then, to calculate the similarity between the histograms of each region in the image document and the query's histogram, we use the cosine similarity defined by:

$$S = 1 - \frac{\sum_{i=1}^{N} H_{i,j} R_i}{\sqrt{\sum_{i=1}^{N} H_{i,j}^2}\sqrt{\sum_{i=1}^{N} R_i^2}} \quad (8)$$

Where $H_{i,j}$ represents the occurrence of the ieme center of the codebook in the jeme region, and R_i represents the occurrence of the ieme center of the codebook. Or, to select them regions that are similar to the query, the cosine similarity "S" should be inferior to a certain threshold. For two regions perfectly similar $\frac{\sum_{i=1}^{N} H_{i,j} R_i}{\sqrt{\sum_{i=1}^{N} H_{i,j}^2}\sqrt{\sum_{i=1}^{N} R_i^2}} = 1$, so S=0. Therefore, as much as the threshold is less, the regions are more similar. For this, for each codebook size, we calculate the F-Score by choosing the best threshold of each size as shown in Figure 3. We observe that the size influence on the performance and best result is given for k=300 and decrease beyond this size.

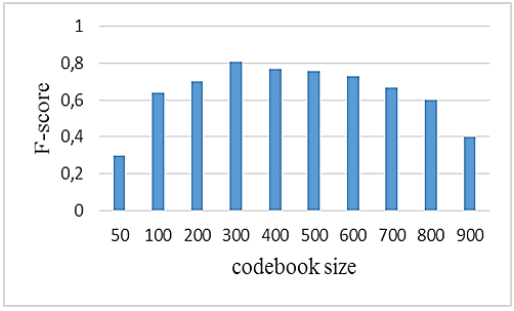

Figure 3. The F-$_{Score}$ curve

Now, we search the influence of the codebook size on the threshold. Form (2), when the size of the codebook increase (N), the probability to find zero descriptor in each cellule in the histogram for a given region increase as shown in Figure 3, that mean:

$\forall$ i≤N, When N increase, the probability $P(R_i)=0$ increase
When N decrease, the probability $P(R_i)=0$ decrease

Subsequently,

$\forall$ i≤N, When N increase, the probability $\sum_{i=1}^{N} P(H_{i,j})*P(R_i)$ decrease
When N decrease, the probability $\sum_{i=1}^{N} P(H_{i,j})*P(R_i)$ increase

So, when N increase, $\frac{\sum_{i=1}^{N} H_{i,j}R_i}{\sqrt{\sum_{i=1}^{N} H_{i,j}^2}\sqrt{\sum_{i=1}^{N} R_i^2}}$ *decrease and* $S = 1 - \frac{\sum_{i=1}^{N} H_{i,j}R_i}{\sqrt{\sum_{i=1}^{N} H_{i,j}^2}\sqrt{\sum_{i=1}^{N} R_i^2}}$ *increase*

Where Ri represents the histogram of the query, Hi,j represents the histogram of jeme region in the document and $\sum_{i=1}^{N} P(H_{i,j}) = n$ is the number of interest points in jeme region. So when the size of histogram (N) increase, the number of zero cellule increase too. Consequently, the threshold should take a count the size of the codebook (number of cluster). Figure 4 shows example of query and region histograms. For the experiments step, Figure 5 shows the curve of the best threshold for each codebook size. The threshold depends on the size of the codebook and can represented by : y=0.000265x+0.3993.

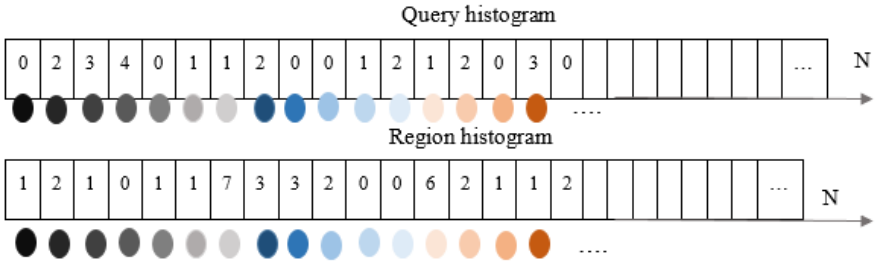

Figure 4. Example of query and region histograms

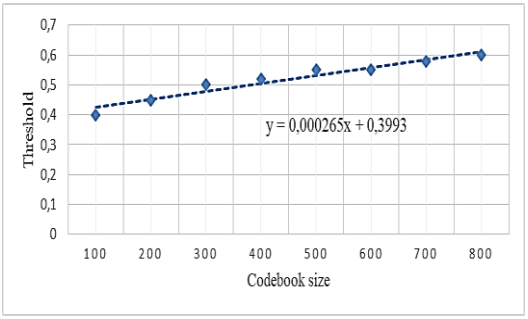

Figure 5. Best threshold for each codebook size

At this stage, and to perform the experiments, we should search the influence of the interest points number (number of descriptors n) on the threshold. Form (2), when the number of interest points n increase, the probability to find zero descriptor in each cellule in the histogram for a given region decrease as shown in Figure 3, that mean:

$\forall$ i≤N, When n increase, the probability $P(R_i)=0$ decrease
When n decrease, the probability $P(R_i)=0$ increase

Subsequently,

$$\forall\ i \leq N, \quad \begin{matrix} \text{When n increase, the probability } \sum_{i=1}^{N} P(H_{i,j}) * P(R_i) \text{ increase} \\ \text{When n decrease, the probability } \sum_{i=1}^{N} P(H_{i,j}) * P(R_i) \text{ decrease} \end{matrix}$$

So, when n increase, $\frac{\sum_{i=1}^{N} H_{i,j} R_i}{\sqrt{\sum_{i=1}^{N} H_{i,j}^2}\sqrt{\sum_{i=1}^{N} R_i^2}}$ $increase\ and\ S = 1 - \frac{\sum_{i=1}^{N} H_{i,j} R_i}{\sqrt{\sum_{i=1}^{N} H_{i,j}^2}\sqrt{\sum_{i=1}^{N} R_i^2}}$ $decrease$

In fact, when the number of interest points (n) increase, the number of zero cellule decrease too. Consequently, the threshold should take a count the number of interest points (number of descriptors), because this number change from region to other. Experimently, Figure 6 shows the threshold curve according to the number of descriptors where we can see that the threshold depend on the descriptors, this dependence can be represented by: y=-0.0009x+0.5219.

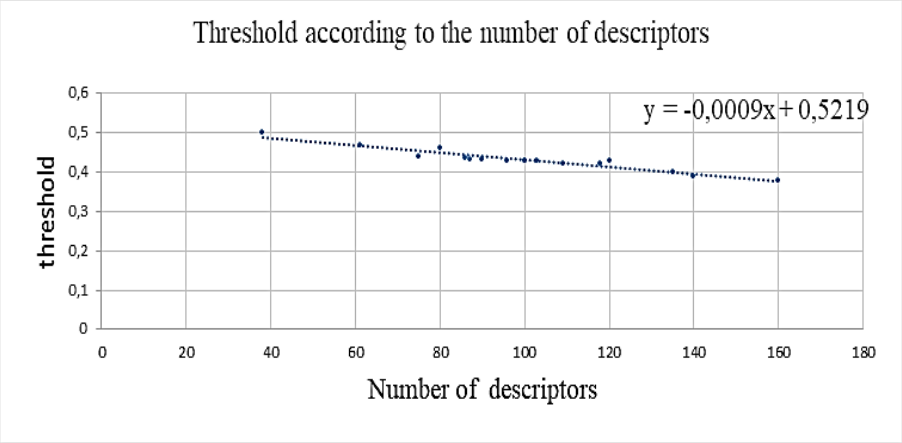

Figure 6. Threshold according to the number of descriptors

Given this condition, two regions are similar if the threshold S is less than a given value, or if two having even number of interest points will have same threshold, but this not mean that they are similar, because maybe they have different histograms and thereafter different distances of similarity. To conclude this part, each word/region in document has a certain number of interest points/descriptors, so a compromise between the number of descriptors and the threshold must be sought.

4.2. Combination of cosine-linear regression similarity

Regression focuses on the relationship between the outcomes and the inputs. It also provides a model that has some explanatory value, in our case, the inputs are size of codebook and number of interest points, the outcome is the threshold value that define if two regions are similar or not. Linear regression is a commonly used technique for modeling when the outcome variable is expressed as linear combination of the input variables, for a given set of input variables, the linear regression model provides the expected outcome value.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_{p-1} x_{p-1} + \varepsilon \quad (9)$$

where y is the threshold variable

x_j are the input variables, for j=1,2,…,p-1

β_0 is the value of y when each x_j equals zero {Offset}

β_j is the change in y based on a unit change in x_j {Coefficient}

$\varepsilon \sim N(0,\sigma^2)$ and the ε's are independent of each other

This random error (ε), denoted by ε, is assumed to be normally distributed with a mean of zero and a constant variance (σ^2). So, the used Equation is: T=-0, 0009 * n+0, 5219 for N=300.

However, the performance of regression analysis methods in practice depends on the type and form of the used data, and how it relates to the used regression method. As is shown in Figures 4 and 5, the threshold depends on the codebook size and the number of descriptors at each word, or the proposed method is a generic similarity measure that take a count these parameters and it does not make any specific condition to handwritten documents. The proposed similarity measure is evaluated in the context of word spotting task. The aim consists in querying Arabic of handwritten documents with query-by-example anagram and in selecting/reporting regions that similar to the query.

We report here some queries where the system yields automatically as shown in Figure 7, which are similar to the query and without chose the thresold , which is a problem in other system [22, 23]. Then, we use a filtering step to select one best result when confusion regions are returning, based on theirs similarity scores and theirs positions. Table 2 shows a comparison of the proposed method with state-of-the-art in term of Precision, tseted in the Gallica dataset. An improvement is observed with the proposed similarity distance over other methods.

Table 2. Performance of the proposed method and other works

Method	Precision
Almazon et al. [23]	68.4%
Rusinol et al. [24]	81%
Proposed method	83%
Howe et al. [25]	79%
Liang et al. [26]	67%
Fischer et al.[27]	62%
Terasawa et al. [28]	79%

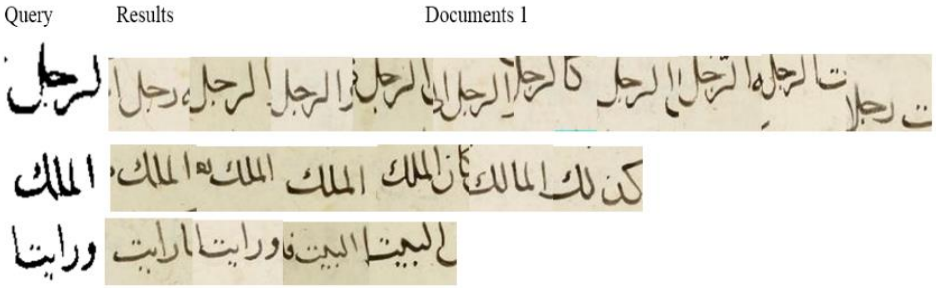

Figure 7. The retrieved regions for some queries

5. CONCLUSION

In this paper, we present a generic similarity measure for word spotting system that take a count the codebook size and the number of descriptors at query, and it does not make any specific condition to handwritten documents,a comparison of the proposed method with other methods shows that generic similarity measure gives an excellent result in term of Precision. We tested our method using experimental setup based on MATLAB code applied to Gallica database

REFERENCES

[1] Mihalcea R., Corley C. and Strapparava C., "Corpus-based and knowledge-based measures of text semantic similarity," In *AAAI*, vol. 6, pp. 775-780, 2006.
[2] Huang, Anna, "Similarity measures for text document clustering," In: *Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008)*, Christchurch, New Zealand, pp. 9-56, 2008.
[3] Marti, U. V., Bunke, H., "Using a statistical language model to improve the performance of an HMM-based cursive handwriting recognition system," In *Hidden Markov models: applications in computer vision*, pp. 65-90, 2001.
[4] H. Bunke, et. al., "Offline recognition of unconstrained handwritten texts using HMMs and statistical language models," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 709-720, 2004.
[5] C. Bahlmann and H. Burkhardt, "Measuring HMM similarity with the Bayes probability of error and its application to online handwriting recognition," *Proceedings of Sixth International Conference on Document Analysis and Recognition*, Seattle, WA, USA, pp. 406-411, 2001.
[6] J.A. R-Serrano, F. Perronnin, "A Model-based sequence similarity with application to handwritten word spotting," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2108-2120, Nov. 2012.

[7] Q. Wang, F. Yin and C. Liu, "Handwritten Chinese Text Recognition by Integrating Multiple Contexts," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 8, pp. 1469-1481, Aug. 2012.
[8] Singh, R., Raj, B., Stern, R. M., "Structured redefinition of sound units by merging and splitting for improved speech recognition," In *Sixth International Conference on Spoken Language Processing*, 2000.
[9] Huo, Q., Li, W., "A DTW-based dissimilarity measure for left-to-right hidden Markov models and its application to word confusability analysis," In *Ninth International Conference on Spoken Language Processing*, 2006.
[10] J. Hershey, P. Olsen, and S. Rennie, "Variational Kull back-Leibler Divergence for Hidden Markov Models," *Proc. Workshop Automatic Speech Recognition Understanding*, 2007.
[11] M. Brand and V. Kettnaker, "Discovery and segmentation of activities in video," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 844-851, Aug. 2000.
[12] Willett, P. *et al.*, "Grouping of coefficients for the calculation of inter-molecular similarity and dissimilarity using 2D fragment bit-strings," *Combinatorial chemistry & high throughput screening*, vol. 5, no. 2, pp. 155-166, 2002.
[13] Seung-Seok C, Sung-Hyuk C and Tappert C., "A Survey of Binary Similarity and Distance Measures, *Journal of Systemics, Cybernetics & Informatics*, vol. 8, chapter 1, pp. 43–48, 2010.
[14] Cha S., "Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions," *Int. J. Math. Model. Methods Appl. Sci*, vol. 1, chapter 4, pp. 300–307, 2007.
[15] Chang, W. L., Kanesan, J., Kulkarni, A. J., Ramiah, H., "Data clustering using seed disperser ant algorithm," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 25, no. 6, pp. 4522-4532, 2017.
[16] Ortega, J.P., Del, M., Rojas, R.B., Somodevilla, M.J., "Research issues on k-means algorithm: An experimental trial using matlab," In *CEUR Workshop Proceedings: Semantic Web and New Technologies*, pp. 83-96, Mar. 2009.
[17] Lowe, D. G., "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91-110, 2004.
[18] Marti, U-V., Bunke, H., "Using a statistical language model to improve the performance of an HMM-based cursive handwriting recognition system," *Int J Pattern Recognit Artif Intell*, vol. 15, no. 01, pp. 65-90, 2001.
[19] Nowak, E., Jurie, F., Triggs, B., "Sampling strategies for bag-of-features image classification," in *European Conference on Computer Vision, Lecture Notes in Computer Science(LNCS) 3954*, pp. 490-503, 2006.
[20] Ernest Bellman, R., "Dynamic programming," Princeton University Press, Rand Corporation, 1957,
[21] Ernest Bellman, R., "Adaptive control processes: a guided tour," Princeton University Press.
[22] Sankar, K.P., Manmatha, R., Jawahar, C.V., "Large-scale document image retrieval by automatic word annotation," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 17, no. 1, pp. 1-17, 2013.
[23] Almazán, J., Gordo, A., Fornés, A., *et al*., "Segmentation-free word spotting with exemplar SVMs," *Pattern Recognition*, vol. 47, no. 12, pp. 3967-3978, 2014.
[24] Rusiñol, M., Aldavert, D., Llados, R.J., "Efficient segmentation-free key word spotting in historical document collections," *Pattern Recognition*, vol. 48, pp. 545-555, 2015.
[25] Howe, H., Rath, T., Ranmatha, R., "Boosted decision trees for word recognition in handwritten document retrieval," in *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 377-383, 2005.
[26] Liang, Y., Fairhurst, M., Guest, R.., "A synthesized word approach to word retrieval in handwritten documents," *Pattern Recognit*, vol. 45, no. 12, pp. 4225-4236, 2012.
[27] Fischer, A., Keller, A., Frinken, V., *et al*., "Lexicon-free handwritten word spotting using character HMMs," *Pattern Recognit.Lett*, vol. 33, no. 7, pp. 934-942, 2012.
[28] Terasawa, K., Tanaka, y., "Slit style HOG feature for document image word spotting," in *Proceeding of the International Conference on Document Analysis and Recognition*, pp. 116-120, 2009.