Comparative Study of Distance Measurements in Texture Image Clustering

Nang Nandar Htun
University of Computer Studies (Taunggyi)
yinnoblewadi@*gamil.com*

Abstract

Image clustering is a crucial but challenging task in machine learning and computer vision. Existing distance measure methods for vectors are also essential and important in image clustering. The structure of the feature forms as the vectors and then different types of distance measures are applied in image clustering to get successful image groups for information management. To highlight the different type of distance measure methods, we propose to perform comparative analysis of distance measurements in image clustering. In this comparative study, Gray Level Co-occurrence Matrix (GLCM) Feature and K-means Clustering algorithms with different distance measure methods are used to compare the clustering results according to different distance measurements. The main output of proposed system is the number of image clusters. In experiment, clustering results are measured by means of clustering accuracy or purity and performed on the Brodatz texture images dataset to show the properties of different distance measurements for image clustering.

Keyword: Image Clustering; GLCM feature; K-means clustering; distance measure methods, Brodatz texture image dataset;

1. Introduction

Image clustering is a fundamental machine learning and vision data analysis tool. Different applications such as image annotation on content and image retrieval can be interpreted as different image clustering examples. Technically, the image clustering process is the aggregation of images into clusters such that images in the same clusters are distinct from one another.

According to literature, the use of spatial contextual information in the image data is proposed using a spatial fuzzy clustering algorithm. A new dissimilarity index takes into account the effect of the neighboring pixels on the middle pixel of a 3-3 window is used as the main feature of their process. For non-homogeneous areas of the image, the algorithm adapts to the image material so that its effect on the next pixels is minimized. An overlapping cluster scheme is proposed that blends two clusters depending on their similarity and overlap. This integration scheme allows the automated identification of an optimal number of clusters as iteration progresses. Experimental tests with simulated and actual photos revealed that the proposed algorithm is more sounds-tolerant than the traditional c-means algorithm [2], which addresses uncertainty in classification and dealing with various cluster types and sizes.

To construct image classes an image clustering technique is built based on the Particle Swarm Optimizer (PSO). The algorithm distinguishes the centers of a particular consumer cluster with a comparable number of clusters. The suggested grader of images was used with biological, MRI and satellite images to demonstrate its wide range of applicability. Experimental tests indicate that the PSO classification picture fits better than the most sophisticated categories of photography: Fuzzy C-means, K-means, Genetic Algorithms and K-harmonic means. It also shows the effect of various PSO control parameter values on output [4].

For clustering by local discriminating and regional integration (LDMGI), a new image clustering algorithm is provided. For every data point, we create a local clique containing this data point and its corresponding data points in order to deal with the data points collected from a non-linear multiplier. Driven by the Fisher criteria, a local discriminant model is used to determine the grouping efficiency of samples within the local clique for each local clique. We also suggest a single purpose functionality to incorporate functional versions with all functional cliques globally to achieve the clustering outcome. LDMGI is comparable to other clustering approaches in experiments and LDMGI is much better than NCut according to algorithmic parameters [1].

There has been a lot of literature work on the clustering of images [3, 5, 6, 7, and 8]. Various strategies for clustering, including K-means, agglomerative clustering etc. were historically studied. Although they have succeeded in data clustering, conventional methods rely on fixed distance metrics, for example, auto encoding [4] and auto-encrypting variance bays, which are hard to find unattended feature-learning methods for studying images that are included in cluster images. Technically, they follow a multi-stage pipeline which initially trains deep neural networks using unregulated methods and utilizes conventional clustering methods for images as post processing. Moreover distance measure methods are also fundamental in image clustering and influent in output results. According to these concepts in image clustering, this paper motivated to highlight the important of distance measure methods in image clustering by using GLCM features with K-means clustering for image clustering.

1.1 Motivation

High volume image data has been created with the development of the image capture system. Gathering images into appropriate group to disclose relevant details is a difficult and significant problem in image analysis. Mathematical distance measurements for vectors play a very important role in image processing and development image clustering. Texture image clustering is a well-established technique for evaluating the mechanical and physical properties of distance measure and texture feature. By performing texture image clustering, we can highlight the important of distance measure and properties of texture feature in our research work. Our research work develop to proposed comparative study of different distance measurements in texture image clustering by using Gray Level Co-occurrence Matrix (GLCM) Feature and K-means Clustering algorithm to highlight the important of different distance measurements in image processing techniques.

In this paper, there are six sections. Introduction and system overview are presented in section I and section II respectively. The description of GLCM and K-means clustering are present in section III. The different types of distance measure in this analysis are shown in section IV. The dataset description and experimental results are presented in section V. Finally, conclusion is presented in section VI.

2. System Overview

Our research aims to use GLCM and K-means with various measurements to test the properties of these measures and to apply them to content-based image retrieval. Figure 1 displays the flow diagram for the image clustering. The system is composed of two main parts: extraction and clustering of images.

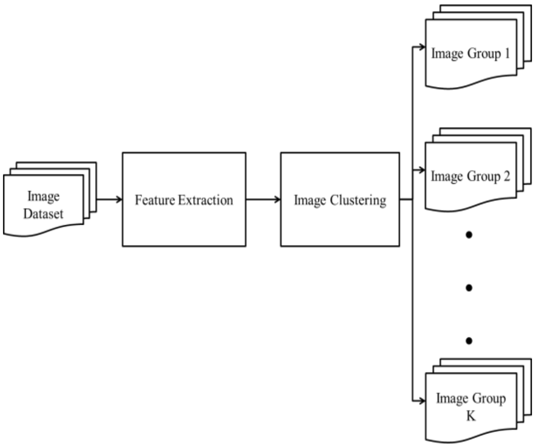


Figure 1. Flow of System Overview

In Figure 1, the image dataset consists of all images in the Brodatz texture images array, and from each image is extracted the GLCM function. For each image the extracted feature is clustered using K-means clustering algorithm in the image clustering stage. The final image clustering system output is image groups which have similar images within a group.

3. Feature Extraction and Image Clustering

This section presents the feature extraction and image clustering stages of overview system. Gray Level Co-occurrence Matrix (GLCM) is used in feature extraction stage while K-means clustering algorithm is used in image clustering stage. GLCM is the pixel co-occurrence based texture feature and is widely used in texture analysis.

Feature Extraction helps to reduce the number of features in a dataset by generating new features from the current images and discarding the original images afterwards. Then these extracted features should be able to sum up much of the details found in the original image package. In this way it is possible to construct a condensed version of images called features from a combination of the original image properties.

Under Unsupervised Machine Learning K-means is a clustering algorithm. It is used to classify a group of data points into clusters where they are identical in points within one cluster. K-Means for Image Clustering is one of the common Unsupervised Image Processing Algorithms. K-means h performs better than other density-based, expectation-maximization clustering algorithms. It is one of the robustness methods in image processing methods, particularly for image clustering, segmentation and image annotation.

3.1. Gray Level Co-occurrence Matrix (GLCM)

GLCM is measured for the distance and angle of a chosen pair. The reason of choosing the Gray Level Co-occurrence Matrix (GLCM) Feature is that GLCM is the best texture feature in texture analysis. Since our texture analysis is performed on Brodatz texture images dataset, GLCM Feature is chosen to use in our comparative study for different distance measure in texture image clustering.

The relative frequencies of a pair of each reference pixel and its neighboring pixels are determined for a certain pair of size, d and angleθ. In order to get a normalized matrix, the matrix thus obtained is divided by sum of all the frequencies [9]. For different offsets multiple GLCMs can also be measured. These offsets describe pixel relationships with respect to specific direction and distance between a reference pixel and its neighboring pixels. In these eight angle values the angle value can be defined: 0, 45, 90, 135, 180 ... 315 and the distance value can be defined as any positive integer: 1,2, ... and so on. Similar distance and angle values perform as separate offsets. Also, the various offsets may produce various GLCMs for an image. Figure 2 displays the different GLCMs for a texture image. Four different types of offsets (d, θ) are included in this Figure 2, (1,0), (1,45), (1,90), and (1,135).

In Figure 2, 8 gray scale levels is used to get the standard image (SI) and then GLCMs are created form this standard image (SI) according to offsets (d,θ). In this system, since we use 4 offsets, four GLCMs are obtained. After getting GLCMs, we derive statistics form each GLCM called contrast, correlation, energy and homogeneity. The mathematical formulas of these statistics values are [10]:

$$\mathrm{contrast} = \sum_{m=1}^{M}\sum_{n=1}^{M}(m-n)^2 G(m,n) \quad (1)$$

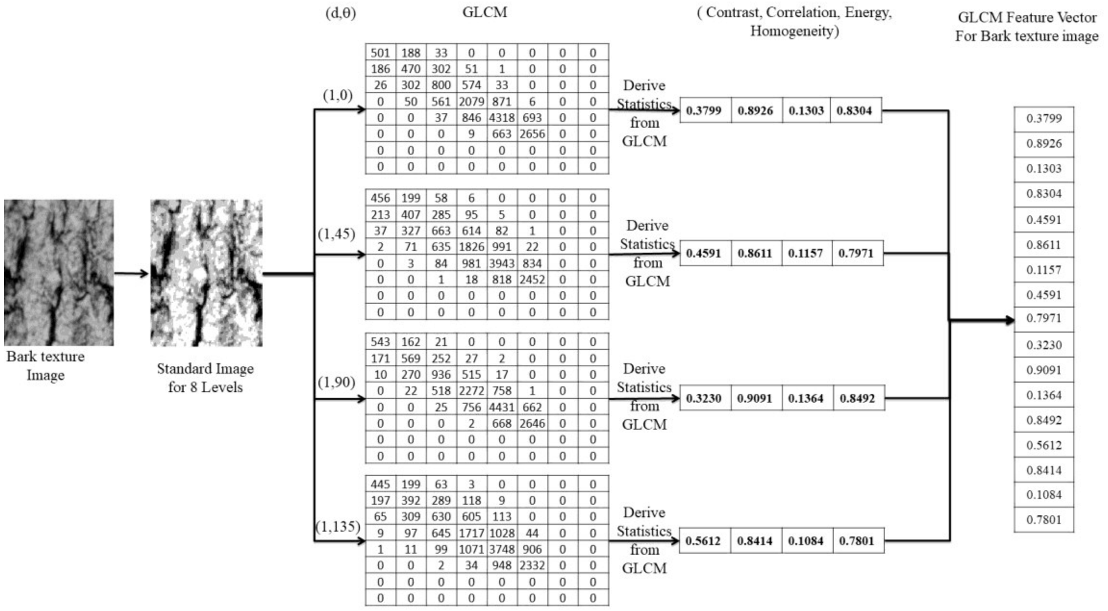
$$\mathrm{correlation} = \sum_{m=1}^{M}\sum_{n=1}^{M}\frac{(m-\mu_m)(n-\mu_n)}{\sigma_m . \sigma_n} \quad (2)$$

$$\mathrm{energy} = \sum_{m=1}^{M}\sum_{n=1}^{M} G(m,n)^2 \quad (3)$$

$$\mathrm{homogeneity} = \sum_{m=1}^{M}\sum_{n=1}^{M}\frac{G(m,n)}{1+(m-n)^2} \quad (4)$$

where G is the GLCM with co-occurrence of 8 levels with m=n= (1, 2,8) and G (m, n) means the count of co-occur of level m and level n in GLCM matrix, μ_m is the mean value of co-occurrence count of level m, μ_n is the mean value of co-occurrence count of level n, σ_m is the variance value of co-occurrence count of level m and σ_n is the mean value of co-occurrence count of level n. Figure 3 shows the calculation of μ_m, μ_n, σ_m and σ_n of a GLCM.

In this system, 4 offsets are used to generate 4 GLCMs. There are 4 statistical values for each GLCMs and hence, the length of GLCM feature for an image is 16 (4 x 4). The extracted features are used in K-means clustering to produces image clusters for image grouping.


(d,θ)
GLCM
(Contrast, Correlation, Energy, Homogeneity)
GLCM Feature Vector For Bark texture image
Bark texture Image
Standard Image for 8 Levels
(1,0)
(1,45)
(1,90)
(1,135)
Derive Statistics from GLCM

(1,0)

501	188	33	0	0	0	0	0
186	470	302	51	1	0	0	0
26	302	800	574	33	0	0	0
0	50	561	2079	871	6	0	0
0	0	37	846	4318	693	0	0
0	0	0	9	663	2656	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

0.3799	0.8926	0.1303	0.8304

(1,45)

456	199	58	6	0	0	0	0
213	407	285	95	5	0	0	0
37	327	663	614	82	1	0	0
2	71	635	1826	991	22	0	0
0	3	84	981	3943	834	0	0
0	0	1	18	818	2452	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

0.4591	0.8611	0.1157	0.7971

(1,90)

543	162	21	0	0	0	0	0
171	569	252	27	2	0	0	0
10	270	936	515	17	0	0	0
0	22	518	2272	758	1	0	0
0	0	25	756	4431	662	0	0
0	0	0	2	668	2646	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

0.3230	0.9091	0.1364	0.8492

(1,135)

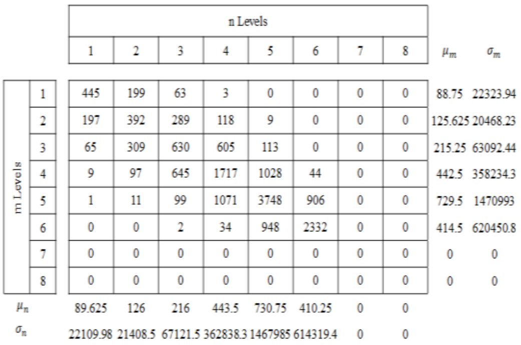
445	199	63	3	0	0	0	0
197	392	289	118	9	0	0	0
65	309	630	605	113	0	0	0
9	97	645	1717	1028	44	0	0
1	11	99	1071	3748	906	0	0
0	0	2	34	948	2332	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

0.5612	0.8414	0.1084	0.7801

GLCM Feature Vector For Bark texture image

0.3799
0.8926
0.1303
0.8304
0.4591
0.8611
0.1157
0.4591
0.7971
0.3230
0.9091
0.1364
0.8492
0.5612
0.8414
0.1084
0.7801

Figure 2. Extraction of GLCM feature from Bark texture image



		n Levels									
		1	2	3	4	5	6	7	8	μ_m	σ_m
m Levels	1	445	199	63	3	0	0	0	0	88.75	22323.94
	2	197	392	289	118	9	0	0	0	125.625	20468.23
	3	65	309	630	605	113	0	0	0	215.25	63092.44
	4	9	97	645	1717	1028	44	0	0	442.5	358234.3
	5	1	11	99	1071	3748	906	0	0	729.5	1470993
	6	0	0	2	34	948	2332	0	0	414.5	620450.8
	7	0	0	0	0	0	0	0	0	0	0
	8	0	0	0	0	0	0	0	0	0	0
μ_n		89.625	126	216	443.5	730.75	410.25	0	0		
σ_n		22109.98	21408.5	67121.5	362838.3	1467985	614319.4	0	0		

Figure 3. GLCM matrix G (m,n) and calculation of mean and variance values for m and n

3.2. K-Means Clustering Algorithm

Texture analysis is the tasks that perform analysis of texture feature by using supervise or unsupervised machine learning algorithms. Texture analysis includes: texture image classification, texture image clustering and texture image segmentation. Texture image clustering is the challenging task in image processing that are widely used in many image processing based application such as image retrieval, duplicate image detection and similar image grouping [7, 8]. The reason of choosing the Gray Level Co-occurrence Matrix (GLCM) Feature and K-means Clustering algorithms with different distance measure methods in our research work is to highlight the important of mathematical based distance measure in image processing of texture analysis.

K-means algorithm is an iterative algorithm that tries to divide the dataset into predetermined K non-overlapping (cluster) sub-groups where every dataset is a pure group. It seeks as close as possible to the data points of the inter-clusters, while holding the clusters as distinct as possible. Info points are allocated to a cluster, such that the distance between the data points and the middle of the cluster (arithmetic means of all data points in this cluster) is minimized as much as possible. The fewer cluster differences, the more (like) uniform in the same cluster the data points. The flow of K-means algorithm is as follow [12]:

(i). Pre-defined the value of K for the number of clusters.

(ii). Randomly choose the initial centroid for each cluster.

(iii). Set the objects to the closest clusters according to distance value.

(iv). Recalculate new cluster centroid (cluster center) to represent the cluster.

(v). If the cluster centroid is change, go to step 2 and create clusters and recalculate new cluster centroids again. Otherwise, the clusters from the steps are the final output clusters for the K-means algorithms.

The flow of K-means algorithm is shown in Figure 4.

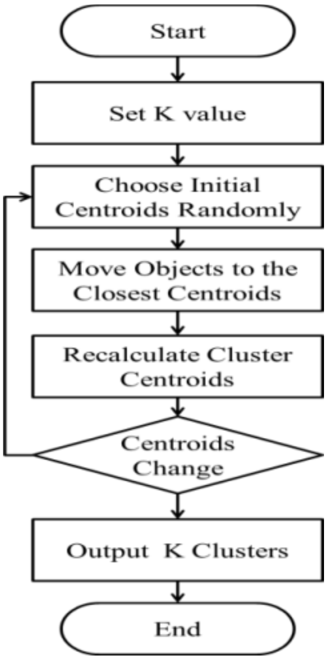

Start
Set K value
Choose Initial Centroids Randomly
Move Objects to the Closest Centroids
Recalculate Cluster Centroids
Centroids Change
Output K Clusters
End

Figure 4. Flow of K-means Algorithm

In Figure 4, step 3 used distance measure or similarity measure to set the objects to the closest clusters. In step 3, all of the objects need to calculate the distance between centroids. For example, the value of K is 5 and there are 5 centroids values that are randomly choose. Each object from dataset needs to calculate distance between each centroid.

If the distance between object and centroid 2 has the minimum value, this object moves to cluster 2 (this object becomes the member of group 2). Typically, Euclidean distance measure use in K-means algorithms to cluster images. Our main focus is to use different distance and similarity measure methods in step 3 of K-means algorithms to show the properties of these measure methods in image clustering.

4. Different Distance measurements in Image Clustering

Choosing distance measures is an important step for the clustering of images. It determines how the similitude of two elements (x, y) is determined and will affect cluster form. Euclidean distance, Manhattan distance, Minkowski distance, cosine similarity and humming distance etc. are the classical methods for distance measurements. Euclidean distance, Pearson correlation distance, and cosine similarity are used in our comparative analysis with GLCM feature in K-means algorithms. The mathematical formula of Euclidean Distance measure is as follow [11]:

$$dist_E(i,j) = \sqrt{\sum_{f=1}^{F}(i_f - j_f)^2} \qquad (1)$$

where F is the length of feature vector i and j. Example calculation of Euclidean distance are shown in Figure 5. In this example, the distance between vector x and three centroids are calculate to set the vector x to the closest cluster. After calculation of Euclidean distances, the vector x move to the cluster 1 because the distance between vector x and c1 has the minimum distance value.

The mathematical formula of Pearson Correlation Distance is as follow [13]:

$$dist_P(i,j) = 1 - \frac{\sum_{f=1}^{F}(i_f-\bar{i})(j_f-\bar{j})}{\sqrt{\sum_{f=1}^{F}(i_f-\bar{i})^2} \cdot \sqrt{\sum_{f=1}^{F}(j_f-\bar{j})^2}} \qquad (2)$$

where ¯i and ¯j are the mean value of vector i and j. Example calculation of Pearson correlation distance are shown in Figure 5. After calculation of Pearson Correlation distances, the vector x move to the cluster 1 because the distance between vector x and c1 has the minimum distance value.

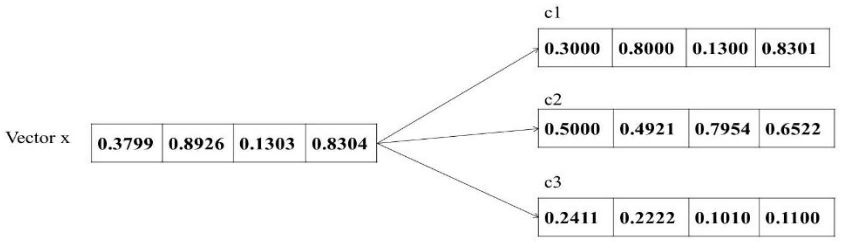
The mathematical formula of Cosine Similarity is as follow [14]:

$$sim_{cos}(i,j) = \frac{\sum_{f=1}^{F}(i_f \cdot j_f)}{\sqrt{\sum_{f=1}^{F}(i_f)^2} \cdot \sqrt{\sum_{f=1}^{F}(j_f)^2}} \qquad (3)$$

After calculation of Cosine similarities, the vector x move to the cluster 1 because the distance between vector x and c1 has the highest similarity value as shown in Figure 5.

5. Dataset and Experimental Results

This section describes the description of texture dataset used in this analysis and structure of experiment results according to clustering accuracy.



Vector x: 0.3799 | 0.8926 | 0.1303 | 0.8304

c1: 0.3000 | 0.8000 | 0.1300 | 0.8301

c2: 0.5000 | 0.4921 | 0.7954 | 0.6522

c3: 0.2411 | 0.2222 | 0.1010 | 0.1100

$$dist_E(x,c1) = \sqrt{\sum_{f=1}^{4}(x_f - c1_f)^2} = \sqrt{(0.3799-0.3000)^2+(0.8926-0.8000)^2+(0.1303-0.1300)^2+(0.8304-0.8301)^2} = 0.122307$$

$$dist_E(x,c2) = \sqrt{\sum_{f=1}^{4}(x_f - c1_f)^2} = \sqrt{(0.3799-0.5000)^2+(0.8926-0.4921)^2+(0.1303-0.7954)^2+(0.8304-0.6522)^2} = 0.805567$$

$$dist_E(x,c3) = \sqrt{\sum_{f=1}^{4}(x_f - c1_f)^2} = \sqrt{(0.3799-0.2411)^2+(0.8926-0.2222)^2+(0.1303-0.1010)^2+(0.8304-0.1100)^2} = 0.994252$$

$$dist_P(x,c1) = 1 - \frac{\sum_{f=1}^{4}(x_f-\overline{x})(c1_f-\overline{c1})}{\sqrt{\sum_{f-1}^{4}(x_f-\overline{x})^2} \cdot \sqrt{\sum_{f-1}^{4}(c1_f-\overline{c1})^2}} = 1 - 0.990923 = 0.009077$$

$$dist_P(x,c2) = 1 - \frac{\sum_{f=1}^{4}(x_f-\overline{x})(c2_f-\overline{c2})}{\sqrt{\sum_{f=1}^{4}(x_f-\overline{x})^2} \cdot \sqrt{\sum_{f=1}^{4}(c2_f-\overline{c2})^2}} = 1 - (-0.55538) = 1.55538$$

$$dist_p(x,c3) = 1 - \frac{\sum_{f=1}^{4}(x_f-\overline{x})(c3_f-\overline{c3})}{\sqrt{\sum_{f=1}^{4}(x_f-\overline{x})^2} \cdot \sqrt{\sum_{f=1}^{4}(c3_f-\overline{c3})^2}} = 1 - 0.2235 = 0.7765$$

$$dist_P(x,c1) = \frac{\sum_{f=1}^{4}(x_f-c1_f)}{\sqrt{\sum_{f=1}^{4}(x_f)^2} \cdot \sqrt{\sum_{f=1}^{4}(c1_f)^2}} = 0.997501$$

$$dist_P(x,c2) = \frac{\sum_{f=1}^{4}(x_f-c2_f)}{\sqrt{\sum_{f=1}^{4}(x_f)^2} \cdot \sqrt{\sum_{f=1}^{4}(c2_f)^2}} = 0.797437$$

$$dist_p(x,c3) = \frac{\sum_{f=1}^{4}(x_f-c3_f)}{\sqrt{\sum_{f=1}^{4}(x_f)^2} \cdot \sqrt{\sum_{f=1}^{4}(c3_f)^2}} = 0.852915$$

Figure 5. Example Calculation of Euclidean distance, Pearson correlation distance and Cosine similarity

5.1 Brodatz Texture Images Dataset

Brodatz Texture Images Dataset is the standard dataset and purposed for texture analysis including [15]: texture image classification, texture image clustering and texture image retrieval. In this system, there are 13 types of texture image clusters and each cluster contains 96 images. All of the images in each cluster have zero rotation degree and size is 128 x 128 in gray scale image with .tiff format. The 13 types of clusters are:

water, bark, weave, wood, wool, raffia, sand, straw, brick, pigskin, bubbles, grass and leather. The sample images from Brodatz dataset are shown in Figure 6.

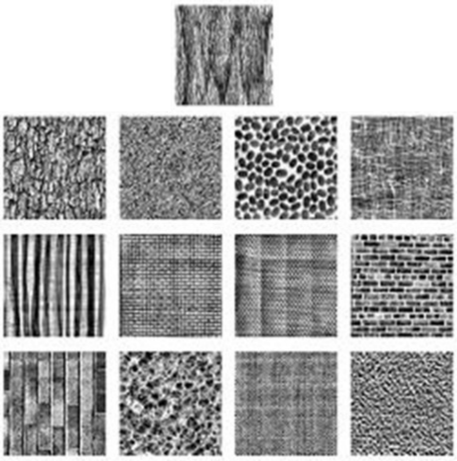


Figure 6. Sample images from Brodatz Texture Image Dataset

5.2 Experimental Results

Clustering accuracy or Purity is measured to show the performance of clustering. The clustering are performed on the Brodatz dataset with GLCM feature and K-means clustering, and different type of distance measurements are used in K-means clustering to highlight the important of distance measure in image clustering.

Purity is measure to show how each cluster has miss clustering (non-members) objects and error in clustering. The highest value of purity value means this clustering has a little small error in clustering and this cluster is good cluster. The value of Purity is between 0 and 1 and the best cluster has the highest value of purity closed to 1. The mathematical formula of Purity is [16]:

$$\text{Purity} = \frac{\sum_{i=1}^{C} \text{Max}\,(c_i \cap g_i)}{N} \qquad (4)$$

Table 1: Purity for different distance measures in K-mean clustering with K=3

Measurement	C1	C2	C3	G1	G2	G3	Purity
Euclidean	94	98	96	96	96	96	0.9930
Pearson	93	97	98				0.9548
Cosine	94	94	100				0.9583

Table 2: Purity for different distance measures in K-mean clustering with K=4

Measurement	C1	C2	C3	C4	G1	G2	G3	G5	Purity
Euclidean	94	98	82	96	96	96	96	96	0.9583
Pearson	84	97	95	108					0.9661
Cosine	104	85	95	100					0.9687

In Table 1 and Table 2, C1, C2, C3 and C4 are the resulted cluster from K-means clustering according to different distance measurements. With the value of K is 3, Euclidean distance measure has the highest purity value while Person correlation distance has the highest purity value with K=4.

Table 4: Purity for Euclidean distance measures in K-mean clustering with K=5, 6… 13

Measurement	K	Purity
Euclidean	5	0.858333
	6	0.8667
	7	0.8956
	8	0.8794
	9	0.8921
	10	0.8852
	11	0.8804
	12	0.8502
	13	0.85441
Average		0.873594

Table 5: Purity for Pearson Correlation distance measures in K-mean clustering with K=5, 6… 13

Measurement	K	Purity
Pearson Correlation	5	0.9744
	6	0.9737
	7	0.9745
	8	0.9621
	9	0.96112
	10	0.9221
	11	0.9211
	12	0.9097
	13	0.8921
Average		0.903424

Table 6: Purity for Pearson Correlation distance measures in K-mean clustering with K=5, 6… 13

Measurement	K	Purity
Cosine	0.9677	0.9784
	0.9564	0.9237
	0.9471	0.9145
	0.954	0.9021
	0.9322	0.90112
	0.9221	0.8921
	0.9164	0.8911
	0.9038	0.8897
	0.9013	0.8521
Average		0.90498

In Table 3, the purity values of Euclidean distance measure according to the value of K from 5 to 13. The purity values of Pearson correlation and Cosine similarity are also shown in Table 4 and Table 5.

5.3.Results and Discussion

In experiment, the number of cluster for each experiment are 3, 4, 5, …, 13 and the main significant difference are number of images between clusters (C1, C2, C3, …) and Groups (G1, G2, G3, …). Clusters are generated by K-means clustering algorithm. Groups are ground truth clusters already defined in Dataset. According to experimental results in Table 4 to Table 6, the highest purity value of Euclidean distance is 0.858333, and Pearson Correlation and cosine similarity are 0.9744 and 0.9784 when the value of K is 5. Hence, the small clustering has the good purity value. Pearson correlation distance measure has the good clustering results because it has highest average purity values over different values of K. Since Person Correlation distance considers the mean values each vector, it has the better measurement in distance measure. According to our comparative study of these there distance measure methods, K-means clustering with Pearson Correlation distance can lead to better clustering results in image clustering.

6. Conclusion

To highlight the important of mathematical based distance measure in image processing of texture analysis, the comparative study of distance measures in texture image clustering using GLCM and K-means clustering algorithm are performed. Both the GLCM function and the K-means clustering algorithm are useful when clustering texture images. With four separate offset, the GLCM features are extracted from an image to reflect the surface texture state of an image in numerical representation. Our key focus in this paper is to conduct a comparative analysis on the Brodatz Texture Images Dataset for image clustering of various distance measurements. K-means clustering efficiency for various distance approaches is calculated by the purity or clustering accuracy measurements. Our studies show that Pearson Correlation outperforms other methods of distance measuring (Euclidean and Cosine similarity) and that Pearson Correlation is even more resilient to K value compared to two other methods. In the future, we will explore how to extract more selective image features to further boost the clustering efficiency as well as research different forms of clustering algorithms for image clustering in Data Mining.

References

[1]. Yang, Y., Xu, D., Nie, F., Yan, S. and Zhuang, Y., 2010. Image clustering using local discriminant models and global integration. IEEE Transactions on Image Processing, 19(10), pp.2761-2773.

[2]. Liew, A.W.C., Leung, S.H. and Lau, W.H., 2000. Fuzzy image clustering incorporating spatial continuity. IEE Proceedings-Vision, Image and Signal Processing, 147(2), pp.185-192.

[3]. Chang, Jianlong, Lingfeng Wang, Gaofeng Meng, Shiming Xiang, and Chunhong Pan, Deep adaptive image clustering. In Proceedings of the IEEE international conference on computer vision, pp. 5879-5887. 2017.

[4]. Omran, M., Engelbrecht, A.P. and Salman, A., 2005. Particle swarm optimization method for image clustering. International Journal of Pattern Recognition and Artificial Intelligence, 19(03), pp.297-321.

[5]. Zheng, X., Cai, D., He, X., Ma, W.Y. and Lin, X., 2004, October. Locality preserving clustering for image database. In Proceedings of the 12th annual ACM international conference on Multimedia (pp. 885-891).

[6]. Gao, B., Liu, T.Y., Qin, T., Zheng, X., Cheng, Q.S. and Ma, W.Y., 2005, November. Web image clustering by consistent utilization of visual features and surrounding texts. In Proceedings of the 13th annual ACM international conference on Multimedia (pp. 112-121).

[7]. Tolias, Y.A. and Panas, S.M., 1998. On applying spatial constraints in fuzzy image clustering using a fuzzy rule-based system. IEEE Signal Processing Letters, 5(10), pp.245-247.

[8]. Silakari, S., Motwani, M., & Maheshwari, M. (2009). Color image clustering using block truncation algorithm. arXiv preprint arXiv:0910.1849.

[9]. Mohanaiah, P., Sathyanarayana, P. and GuruKumar, L., 2013. Image texture feature extraction using GLCM approach. International journal of scientific and research publications, 3(5), pp.1-5.

[10]. Haralick, R.M., Shanmugam, K. and Dinstein, I.H., 1973. Textural features for image classification. IEEE Transactions on systems, man, and cybernetics, (6), pp.610-621.

[11]. Milligan, G.W. and Cooper, M.C., 1988. A study of standardization of variables in cluster analysis. Journal of classification, 5(2), pp.181-204.

[12]. Hartigan, J.A. and Wong, M.A., 1979. Algorithm AS 136: A k-means clustering algorithm. Journal of the Royal Statistical Society. Series C (Applied Statistics), 28(1), pp.100-108.

[13]. Sturn, A., Quackenbush, J. and Trajanoski, Z., 2002. Genesis: cluster analysis of microarray data. Bioinformatics, 18(1), pp.207-208.

[14]. Muflikhah, L. and Baharudin, B., 2009, November. Document clustering using concept space and cosine similarity measurement. In 2009 International Conference on Computer Technology and Development (Vol. 1, pp. 58-62). IEEE.

[15]. Valkealahti K & Oja E (1998). Reduced multidimensional cooccurrence histograms in texture classification. IEEE Transactions on Pattern Analysis and Machine Intelligence 20:90-94.

[16]. Sripada, S.C. and Rao, M.S., 2011. Comparison of purity and entropy of k-means clustering and fuzzy c means clustering. Indian journal of computer science and engineering, 2(3), pp.343-346.