INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

OCR for historical Kannada documents using clustering methods

P Ravi[1,2,3]*, C Naveena[1,3], Y H Sharathkumar[4,3]

1 Department of Computer Science and Engineering, SJB Institute of Technology, Bengaluru, 560060, Karnataka, India. Tel.: +919480507409
2 Department of Computer Science and Engineering, Vidyavardhaka College of Engineering, Mysuru, 570002, Karnataka, India
3 Affiliated to Visvesvaraya Technological University, Belagavi, 590018, Karnataka, India
4 Department of Information Science and Engineering, Maharaja Institute of Technology, Mysuru, 571438, Karnataka, India

OPEN ACCESS

Received: 31-07-2020
Accepted: 06-09-2020
Published: 03.10.2020

Editor: Dr. Natarajan Gajendran

Citation: Ravi P, Naveena C, Sharathkumar YH (2020) OCR for historical Kannada documents using clustering methods. Indian Journal of Science and Technology 13(35): 3652-3663. https://doi.org/10.17485/IJST/v13i35.1287

***Corresponding author**.

Tel: +919480507409
ravipbympr@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Motivation: In India, the Language Kannada is an ancient and official language in Karnataka State. The study of ancient Kannada scripts from stone carvings, leaf, metal, cloth, paper and other sources enhances our knowledge on the traditions and culture practiced in Karnataka. Due to Poor Quality, variability and the contrast, the Kannada ancient scripts become very challenging to extract the information or to recognize the characters. **Objectives:** To design a suitable Optical Character Recognition (OCR) technique to read ancient Kannada scripts. **Method**: Clustering by fast search and find of density peaks is a state-of-the-art density-based clustering algorithm that can effectively find clusters with arbitrary shapes. However, it requires to calculate the distances between all the points in a data set to determine the density and separation of each point. Consequently, its computational cost is extremely high in the case of large-scale data sets. In this work the given document is preprocessed. The features alike SIFT and SURF are extracted and clustered using K-Means clustering. The similarity is computed using different measures. **Findings**: The classification accuracy was studied under different clustering methods like K-means, Agglomerative, Density based clustering with distance based measures like Euclidean and Manhattan. To evaluate the performance of the proposed method, we created our own database of Ashok, Kadamba, Hoysala and Mysuru scripts and experiment was conducted in a database of 4 classes under 70, 50 and 30 different training models from each class. **Novelty:** We propose a K-means clustering using SIFT and SURF for Kannada ancient manuscript. Experiment was conducted in our own database to validate the performance of the presented system

Keywords: Historical Kannada; Karnataka; SIFT; SURF; KMeans

1 Introduction

Kannada is an olden language of the India and official language in Karnataka, originated in epigraphs or inscriptions before 230BC. Epigraphs or inscriptions are ancient scripts, written on stone carvings, palm leaf, metal, cloth and paper. The inscriptions used to enhance our knowledge of the astrological, cultural, philosophy and spiritual of the ancient people. Recognizing the inscription is not straightforward process because of its style of scripts, but we can recognize ancient scripts with the help of epigraphers, those who can easily identify ancient scripts. Identify them manually is a tiresome and time-consuming process. It could be good to develop the OCR to recognize Kannada ancient scripts automatically to overcome the disadvantage of the manual process system. In Kannada language, recognition of historical or ancient manuscripts is very tough task because of low quality, variableness in writing style, large character set and certain resemblances among the characters is shown in Figure 1. Features extraction and Clustering or classification is very significant stage to increase the performance of OCR system. The features are trained to proposed system by SIFT and SURF methods. When Kannada ancient manuscripts database is created, Identification becomes even more challenging as large variability is seen in the collected data samples. Hence, there is a need to organize and search for data inside the collections; so we used different clustering methods like K-means, Agglomerative, Density based clustering with distance based measures like Euclidean and Manhattan to inflation of the performance of OCR system. In the proposed method, we have done wide trials on many character dataset of Ashok Scripts, Kadamba, Scripts, Hoysala Scripts and Mysuru Scripts.

Fig 1. The sample character set of Kannada ancient script

An efficient system proposed by authors for character recognition by using LDA, PCA and k-means clustering, decreases the needless information in the training data and increases the performance of the system [(1)]. In [(2)] presented k-means clustering method for recognizing printed Kannada document, which presents a natural grade of font individuality and used to decrease the training dataset's size and got good accuracy. The Clustering technique with is used to extract features from handwritten signature images, here authors considered height-width, occupancy and distance ratio validate the signature with higher

accuracy [3]. In [4] proposed the recognition system for Hindi handwritten character by K-means clustering and SVM and result of proposed was better than Euclidean distance. The authors [5] presented clustering method for recognizing of handwritten character of Nandinagari is derived from Brahmi-based script and existing in India from 8th to 19th centuries. The SIFT and SURF approaches identify the interest points and derives feature descriptors and succeeding decent recognition accuracy. The clustering algorithm to recognizing Thai handwritten script. This clustered method is used as a rough classify method or global feature that reflects the structure significance of the characters [6]. [7] Proposed recognition system for Yi character using CNN with density-based clustering method and compared experiment results of different parameters, achieved good accuracy. The authors proposed [8] a system by using SIFT and HOG as feature extractor for recognizing handwritten character of Thai, Bengali and Latin and have succeeded good results. Finding text areas and decorative features in olden scripts and robust manner is encouraged by objects recognition system. SIFT descriptors are chosen to find interest regions, which is used for localization [9]. The authors [10] presented 2 feature extraction methods together with diagonal and transition features and conducted experiment on the Gurmakhi database and achieved good accuracy with different parameters. In OCR system feature extraction from overlapping character blocks is major problem and reduces the performance of the recognition phase. The authors [11] address this problem and proposed a system to escalation the performance of the recognition phase by the help of clustering system and Hamming Distance method and experimental show improvement in performance. In [12,13] proposed precise recognition system for recognition of Historical printed records by the combination of LSTM neural network and clustering and improve the recognition rate by combine the clustering and classification. [14] presented feature extraction method for printed Odia characters set by k- means and spectral clustering algorithm and comparison result has done between k- means and spectral clustering finally authors concluded K-means better than spectral clustering. The authors [20] presented OCR system with the help of connected component technique for recognized handwritten Kannada script with good accuracy. The complete process of OCR , Syntactical analysis and Ternary search tree of Kannada script is discussed [21]. Kannada OCR on the Android OS system for Kannada sign boards by the help of kohonen's procedure and finding the meaning of the word on the Internet [22]. A Complete Survey on Optical Character Recognition system to printed and handwritten Kannada language script is presented in [23]. The authors [24] designed a multipurpose OCR system for documents in any language printed in Kannada Script. The authors [25] proposed an accelerated algorithm of density clustering by k-means for several synthetic and real data sets. This algorithm involves additional computational costs. Although these algorithms improve the performance of the algorithms, they do not fully consider how to reduce redundant computations and accelerate the algorithms. Therefore, in this work, we use the different clustering methods like K-means, Agglomerative, Density based clustering with distance based measures like Euclidean and Manhattan. The algorithm determines the membership of a point to a cluster by considering not only the connectivity but also the separation of points. Thus, its performance is robust with respect to the radius of a neighborhood, compared to other density-based algorithms. However, as with other density-based algorithms, it requires to calculate the distances between all the points in a data set to determine the densities of the points and separations between the points. Consequently, its computational cost is extremely high in the case of large-scale data sets.

Here Section 2 explains the proposed system with feature extraction and Clustering methods. Further, section 3 discussed about experimentation and results. Final end with conclusion.

2 Proposed System

In this proposed work the given document is preprocessed. The features alike SIFT and SURF are extracted and clustered using K-Means clustering, which described following subsection.

2.1 SIFT (Scale Invariant Feature Transform)

The SIFT was introduced by David Lowe (2004) for finding distinct invariant features from images, which can be used to do reliable matching. This algorithm is used in recognizing handwritten character of Nandinagari and Thai [5,8] and Identifying text areas and decorative features in Ancient Scripts [9]. The extraction of SIFT features involves the following steps explained below.

Keypoint Detection

This stage involves finding points of interest known as keypoints, which are invariant to scale and orientation. The keypoints were identify through cascade filtering approach and for each of these keypoints the scale and location are determined and then gradient operators are used for orientation assignment. The identification of keypoints is summarized as follows.

Detection of scale space extrema

The 1st stage of keypoint detection is to detect the locations and scales, this will be repeatedly allocated with dissimilar views of the similar object. The positions that remain consistent to variations in scale are found by finding the steady features at altered

scales by continuous function of scale, it is specified by

$$L(x,y,\sigma) = G(x,y,\sigma)^{*}I(x,y) \tag{1}$$

Here σ is the width of Gaussian filter, I is an input image and * is the convolution operator . The Difference of Gaussian (DOG) image are calculated from 2 nearby scales that differ using constant multiplicative factor k.

$$D(x,y,\sigma) = L(x,y,k\sigma) - L(x,y,\sigma) \tag{2}$$

Keypoint Localization

The DOG images obtained in the above step are used to find the key points by the help of local minimum or maximum through dissimilar scales. Every pixel in the DOG image is matched with its 8 and 9 neighbors of the scale above and below correspondingly. Then pixel is selected to be the candidate key point if it is either a local minimum or maximum in $3 \times 3 \times 3$ regions at current and adjacent scales. The next step is to reject the key points that are associated with an edge or which has a low contrast since they are certainly corrupted by noise.

Orientation Assignment

Herein a consistent orientation was allocated to the keypoint, which makes the feature invariant to rotation when the descriptor for the keypoint was expressed in reference to orientation The scale of keypoint was used to select Gaussian smoothed image L. For each Gaussian smoothed image L(x, y), magnitude m(x, y) and orientation θ (x, y) are specified by

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \tag{3}$$

$$\theta(x,y) = \tan^{-1}((L(x,y+1) - L(x,y-1))/(L(x+1,y) - L(x-1,y))) \tag{4}$$

Keypoint Descriptor

After orientation selection, the feature description is calculated. This is done by calculating a set of orientation histograms in 4×4 pixel neighborhoods. The orientation histograms correspond to the keypoint orientation as shown in the (Figure 2).

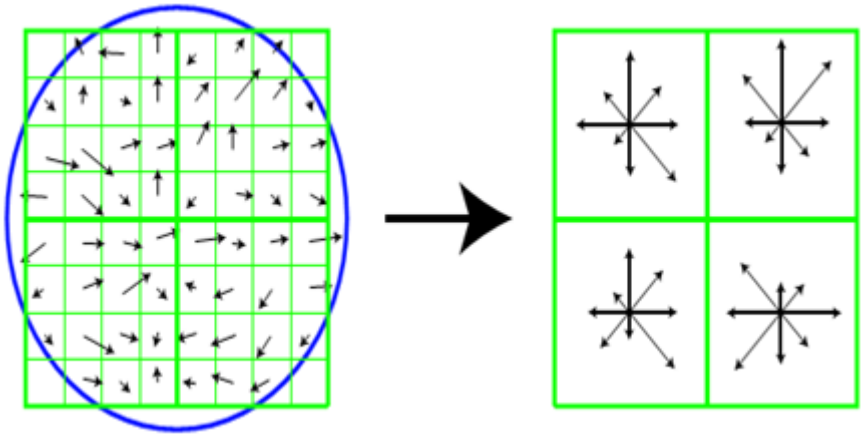

Fig 2. Image gradients and Keypoint descriptor

2.2 Speeded Up Robust Features (SURF)

It is a scale invariant and rotation invariant interest point detector and descriptor. This algorithm has been used in recognizing handwritten character of Nandinagari [5], Human recognition method by profile face and ear [17] This algorithm uses a keypoint detector and descriptor method which is explained as below.

Detecting Keypoints with Fast-Hessian

SURF makes use of Hessian matrix to detect the keypoints in the image. It is defined as

$$\mathrm{H}(\mathrm{P},\sigma) = \begin{bmatrix} Lxx(P,\sigma) & Lxy(P,\sigma) \\ Lyx(P,\sigma) & Lyy(P,\sigma) \end{bmatrix} \quad (5)$$

Extracting SURF Descriptor

The extraction of the SURF descriptor first involves orientation assignment around the keypoint and then the extraction of the descriptor with reference to the orientation.

Orientation Assignment

First a circular region is concede around the selected keypoints to calculate the dominant orientation based on the data in the circular region. The Haar wavelet reply in both vertical and horizontal directions is calculated and the main orientation is obtained by summing the wavelet responses and the maximum response yields the dominant orientation of the keypoint. The feature vector is then computed relative to the dominant orientation thus making it invariant to rotation.

Descriptor Components

A square section allied along the main orientation is measured around the nominated key point. This section is then allocated into 4×4 sub-sections and for each of these sub-sections the Haar wavelet reply is calculated. The sum of the wavelet replies d_x and d_y in the horizontal and vertical orders for each sub-section denotes the feature vector. Next the sum of the absolute value of the responses $|d_x|$ and $|d_y|$ are calculated which provides the data about the polarity of the changes in image intensity. Therefore, the feature vector V_j for the j^{th} sub-section is given as

$$\mathrm{V_j} = \left\{\sum \mathrm{d_x}, \sum \mathrm{d_y}, \sum |\mathrm{d_x}|, \sum |\mathrm{d_y}|\right\} \quad (6)$$

Concatenating the entire feature vector for the sixteen sub-regions surrounding the keypoint provides the descriptor vector of length 64(16×4).

2.3 K-Means clustering method

This method is the furthermost popular and easy clustering method for execution. It is a partition based clustering method that is used in different applications. Partition clustering attempts to split a group of N objects into K clusters, which means that the partitions optimize a certain standard function. Every cluster is denoted by the centroid of the cluster, e.g. k-means. Normally, K seeds are arbitrarily selected and then the relocation structure replicates the points between the clusters to optimize the clustering criteria [18].

This algorithm is executed in four steps:

1. Divide the data into K Clusters
2. Calculate the centroid for each k clusters
3. Allocate the data into group based on nearest point
4. Repeat the process by going back to Step 2 with no more data available

Agglomerative clustering method

It is also named as bottom-up process, starting with every object making its own set. Here we used complete linkage.

This algorithm is executed in five steps:

1. Starts the cluster set by treating each data object as a distinct cluster
2. Work out the resemblance among all pairs of clusters
3. Combine the 2 clusters that are similar
4. Revise the comparison matrix to replicate the pairwise comparison among the old clusters and the new clusters.
5. Steps 3 and 4 repeat till the cluster criteria is met

Density based clustering technique

This technique is unlike partitioning and hierarchical technique.

Density based Algorithm

1. An ε-neighborhood $N\varepsilon(x) = \{y\ X \mid dist(x, y) \leq \varepsilon\}$ of the point x,
2. Basic object, a point with a $|\ N\varepsilon\ (x)\ | \geq$ MinPts

3. Notion of a point y density-nearby from a basic object x
4. Definition of density-connectivity among two points x, y

Distance m easures

The similarity or dissimilarity is measures between two objects using Euclidean, Manhattan distance.

Euclidean distance

The distance is computed among 2 points by below equations.

$$d(i,j) = \sqrt{\sum_{k=1}^{n} \left[X_{ik} - X_{jk}\right]^2} \tag{7}$$

Manhattan distance

The distance is computed among two points by below equations.

$$d(i,j) = \sum_{k=1}^{n} \left|X_{ik} - X_{jk}\right| \tag{8}$$

Where i=($X_{i1,}X_{i2}$,……..X_{in}) and j=($X_{j1,}X_{j2}$,…….. $X_{jn)}$ are two n dimensional data objects.

3 Experimentation and Results

In order for experimentation, the dataset of Kannada's historical letters are shown in (Figure 1). So as to substantiate the proficiency of the proposed methodology, we completed broad trials on various Character dataset viz. Ashok, Kadamba, Hoysala and Mysuru Scripts. Each character dataset contains 25 pictures shown in (Figure 1). In this section, we aimed to study the performance of the proposed system under different clustering methods like K-means, Agglomerative, Density based clustering with distance based measures like Euclidean and Manhattan. We picked images randomly from the dataset and experiment is conducted in a database of 4 classes under 70, 50 and 30 different training models from each class**.** The accuracy of SIFT features with different clustering methods with Euclidean distance is shown in the (Tables 1, 2 and 3)**.** The accuracy of SIFT features with different clustering methods with Manhattan distance is shown in the (Tables 4, 5 and 6).The accuracy of SURF features with different clustering methods with Euclidean distance is shown in the (Tables 7, 8 and 9). In table shows the accuracy 70, 50 and 30 set for all the clusters. The accuracy of SURF features with different clustering methods with Manhattan distance is shown in the (Tables 10, 11 and 12). Figure 3**shows the comparison of SIFT and SURF in which SIFT achieves the maximum accuracy in all the cases. Figure 4 shows the comparison of clustering method with distance measures for both SIFT and SURF.** By tables, we analyze that the Density based clustering with SIFT features achieves maximum accuracy in all cases when compares to K-means and Agglomerative.

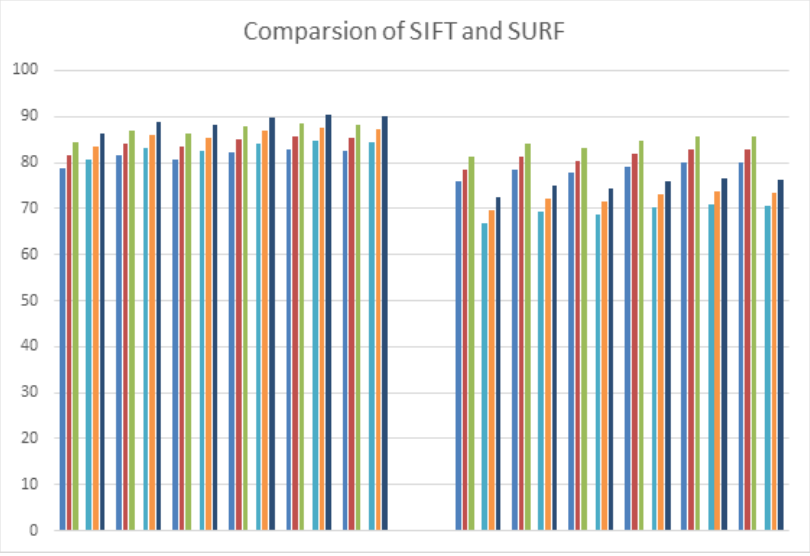

Fig 3. shows the comparison between SIFT (left side) and SURF (right side)

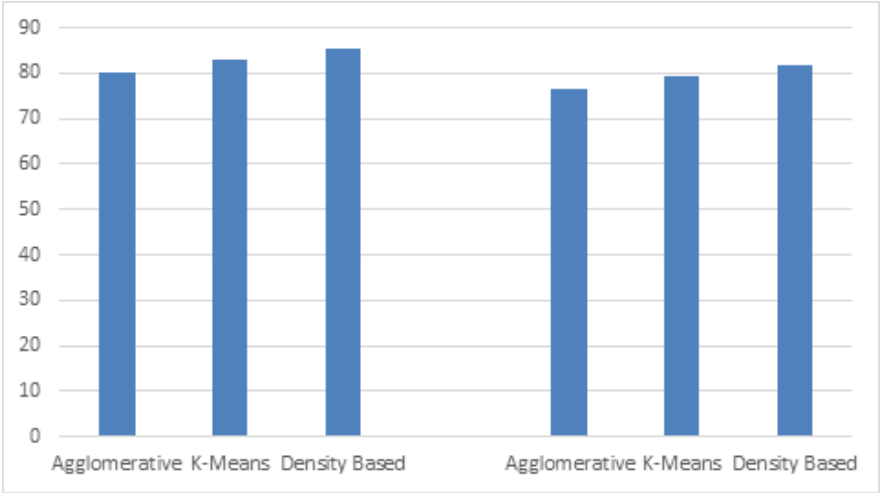

Fig 4. shows the comparison of clustering with distance measures between SIFT (leftside) and SURF (right side)

Table 1. The accuracy of SIFT features with 70 percent training with Euclidean distance

	K = 5	K = 10	K = 15	K = 20	K = 25	K = 30
Agglomerative	78.8	81.45	80.67	82.23	83	82.66
K-Means	81.6	84.25	83.47	85.03	85.8	85.46
Density Based	84.4	87.05	86.27	87.83	88.6	88.26
Agglomerative	77	79.65	78.87	80.43	81.2	80.86
K-Means	79.8	82.45	81.67	83.23	84	83.66
Density Based	82.6	85.25	84.47	86.03	86.8	86.46
Agglomerative	75.9	78.55	77.77	79.33	80.1	79.76
K-Means	78.7	81.35	80.57	82.13	82.9	82.56
Density Based	81.5	84.15	83.37	84.93	85.7	85.36

Table 2. The accuracy of SIFT features with 50 percent training with Euclidean distance

	K = 5	K = 10	K = 15	K = 20	K = 25	K = 30
Agglomerative	80.8	83.45	82.67	84.23	85	84.66
K-Means	83.6	86.25	85.47	87.03	87.8	87.46
Density Based	86.4	89.05	88.27	89.83	90.6	90.26
Agglomerative	79	81.65	80.87	82.43	83.2	82.86
K-Means	81.8	84.45	83.67	85.23	86	85.66
Density Based	84.6	87.25	86.47	88.03	88.8	88.46
Agglomerative	77.9	80.55	79.77	81.33	82.1	81.76
K-Means	80.7	83.35	82.57	84.13	84.9	84.56
Density Based	83.5	86.15	85.37	86.93	87.7	87.36

Table 3. The accuracy of SIFT features with 30 percent training with Euclidean distance

	K = 5	K = 10	K = 15	K = 20	K = 25	K = 30
Agglomerative	82.8	85.45	84.67	86.23	87	86.66
K-Means	85.6	88.25	87.47	89.03	89.8	89.46
Density Based	88.4	91.05	90.27	91.83	92.6	92.26
Agglomerative	81	83.65	82.87	84.43	85.2	84.86
K-Means	83.8	86.45	85.67	87.23	88	87.66
Density Based	86.6	89.25	88.47	90.03	90.8	90.46
Agglomerative	79.9	82.55	81.77	83.33	84.1	83.76
K-Means	82.7	85.35	84.57	86.13	86.9	86.56
Density Based	85.5	88.15	87.37	88.93	89.7	89.36

Table 4. The accuracy of SIFT features with 70 percent training with Manhattan distance

	K = 5	K = 10	K = 15	K = 20	K = 25	K = 30
Agglomerative	80.6	83.25	82.47	84.03	84.8	84.46
K-Means	83.4	86.05	85.27	86.83	87.6	87.26
Density Based	86.2	88.85	88.07	89.63	90.4	90.06
Agglomerative	78.8	81.45	80.67	82.23	83	82.66
K-Means	81.6	84.25	83.47	85.03	85.8	85.46
Density Based	84.4	87.05	86.27	87.83	88.6	88.26
Agglomerative	77.7	80.35	79.57	81.13	81.9	81.56
K-Means	80.5	83.15	82.37	83.93	84.7	84.36
Density Based	83.3	85.95	85.17	86.73	87.5	87.16

Table 5. The accuracy of SIFT features with 50 percent training with Manhattan distance

	K = 5	K = 10	K = 15	K = 20	K = 25	K = 30
Agglomerative	82.3	84.95	84.17	85.73	86.5	86.16
K-Means	85.1	87.75	86.97	88.53	89.3	88.96
Density Based	87.9	90.55	89.77	91.33	92.1	91.76
Agglomerative	80.5	83.15	82.37	83.93	84.7	84.36
K-Means	83.3	85.95	85.17	86.73	87.5	87.16
Density Based	86.1	88.75	87.97	89.53	90.3	89.96
Agglomerative	79.4	82.05	81.27	82.83	83.6	83.26
K-Means	82.2	84.85	84.07	85.63	86.4	86.06
Density Based	85	87.65	86.87	88.43	89.2	88.86

Table 6. The accuracy of SIFT features with 30 percent training with Manhattan distance

	K = 5	K = 10	K =15	K = 20	K = 25	K = 30
Agglomerative	83.5	86.15	85.37	86.93	87.7	87.36
K-Means	86.3	88.95	88.17	89.73	90.5	90.16
Density Based	89.1	91.75	90.97	92.53	93.3	92.96
Agglomerative	81.7	84.35	83.57	85.13	85.9	85.56
K-Means	84.5	87.15	86.37	87.93	88.7	88.36
Density Based	87.3	89.95	89.17	90.73	91.5	91.16
Agglomerative	80.6	83.25	82.47	84.03	84.8	84.46
K-Means	83.4	86.05	85.27	86.83	87.6	87.26
Density Based	86.2	88.85	88.07	89.63	90.4	90.06

Table 7. The accuracy of SURF features with 70 percent training with Euclidean distance

	K = 5	K = 10	K = 15	K = 20	K = 25	K = 30
Agglomerative	75.8	78.45	77.67	79.23	80	80.11
K-Means	78.6	81.25	80.47	82.03	82.8	82.91
Density Based	81.4	84.05	83.27	84.83	85.6	85.71
Agglomerative	74	76.65	75.87	77.43	78.2	78.31
K-Means	76.8	79.45	78.67	80.23	81	81.11
Density Based	79.6	82.25	81.47	83.03	83.8	83.91
Agglomerative	72.9	75.55	74.77	76.33	77.1	77.21
K-Means	75.7	78.35	77.57	79.13	79.9	80.01
Density Based	78.5	81.15	80.37	81.93	82.7	82.81

Table 8. The accuracy of SURF features with 50 percent training with Euclidean distance

	K = 5	K = 10	K = 15	K = 20	K = 25	K = 30
Agglomerative	77.8	80.45	79.67	81.23	82	82.11
K-Means	80.6	83.25	82.47	84.03	84.8	84.91
Density Based	83.4	86.05	85.27	86.83	87.6	87.71
Agglomerative	76	78.65	77.87	79.43	80.2	80.31
K-Means	78.8	81.45	80.67	82.23	83	83.11
Density Based	81.6	84.25	83.47	85.03	85.8	85.91
Agglomerative	74.9	77.55	76.77	78.33	79.1	79.21
K-Means	77.7	80.35	79.57	81.13	81.9	82.01
Density Based	80.5	83.15	82.37	83.93	84.7	84.81

Table 9. The accuracy of SURF features with 30 percent training with Euclidean distance

	K = 5	K = 10	K = 15	K = 20	K = 25	K = 30
Agglomerative	79.8	82.45	81.67	83.23	84	84.11
K-Means	82.6	85.25	84.47	86.03	86.8	86.91
Density Based	85.4	88.05	87.27	88.83	89.6	89.71
Agglomerative	78	80.65	79.87	81.43	82.2	82.31
K-Means	80.8	83.45	82.67	84.23	85	85.11
Density Based	83.6	86.25	85.47	87.03	87.8	87.91
Agglomerative	76.9	79.55	78.77	80.33	81.1	81.21
K-Means	79.7	82.35	81.57	83.13	83.9	84.01
Density Based	82.5	85.15	84.37	85.93	86.7	86.81

Table 10. The accuracy of SURF features with 70 percent training with Manhattan distance

	K = 5	K = 10	K = 15	K = 20	K = 25	K = 30
Agglomerative	66.8	69.45	68.67	70.23	71	70.66
K-Means	69.6	72.25	71.47	73.03	73.8	73.46
Density Based	72.4	75.05	74.27	75.83	76.6	76.26
Agglomerative	65	67.65	66.87	68.43	69.2	68.86
K-Means	67.8	70.45	69.67	71.23	72	71.66
Density Based	70.6	73.25	72.47	74.03	74.8	74.46
Agglomerative	63.9	66.55	65.77	67.33	68.1	67.76
K-Means	66.7	69.35	68.57	70.13	70.9	70.56
Density Based	69.5	72.15	71.37	72.93	73.7	73.36

Table 11. The accuracy of SURF features with 50 percent training with Manhattan distance

	K = 5	K = 10	K = 15	K = 20	K = 25	K = 30
Agglomerative	70.8	73.45	72.67	74.23	75	74.66
K-Means	73.6	76.25	75.47	77.03	77.8	77.46
Density Based	76.4	79.05	78.27	79.83	80.6	80.26
Agglomerative	69	71.65	70.87	72.43	73.2	72.86
K-Means	71.8	74.45	73.67	75.23	76	75.66
Density Based	74.6	77.25	76.47	78.03	78.8	78.46
Agglomerative	67.9	70.55	69.77	71.33	72.1	71.76
K-Means	70.7	73.35	72.57	74.13	74.9	74.56
Density Based	73.5	76.15	75.37	76.93	77.7	77.36

Table 12. The accuracy of SURF features with 30 percent training with Manhattan distance

	K = 5	K = 10	K = 15	K = 20	K = 25	K = 30
Agglomerative	74.8	77.45	76.67	78.23	79	78.66
K-Means	77.6	80.25	79.47	81.03	81.8	81.46
Density Based	80.4	83.05	82.27	83.83	84.6	84.26
Agglomerative	73	75.65	74.87	76.43	77.2	76.86
K-Means	75.8	78.45	77.67	79.23	80	79.66
Density Based	78.6	81.25	80.47	82.03	82.8	82.46
Agglomerative	71.9	74.55	73.77	75.33	76.1	75.76
K-Means	74.7	77.35	76.57	78.13	78.9	78.56
Density Based	77.5	80.15	79.37	80.93	81.7	81.36

4 Conclusion

Investigating ancient manuscript is not straightforward process because of poor quality, diversity, contrast and envelope of characters. In this analysis, the authors propose a K-means clustering using SIFT and SURF for Kannada ancient manuscript. In literature the algorithm involves additional computational costs. Although these algorithms improve the performance of the algorithms, they do not fully consider how to reduce redundant computations and accelerate the algorithms. Therefore, in this work we used Agglomerative Clustering, K-means clustering and Density based clustering methods with different distance measures like Manhattan Distance and Euclidean distance. Experimentation is conducted on our own dataset.

References

1) Pourmohammad S, Soosahabi R, Maida A. An efficient character recognition scheme based on k-means clustering. 2013. Available from: https://doi.org/10.1109/ICMSAO.2013.6552640.
2) Sheshadri K, Ambekar PKT, Prasad DP, Kumar RP. An OCR system for printed Kannada using k-means clustering. In: and others, editor. IEEE International Conference on Industrial Technology, Vina del Mar. 2010;p. 183–187. Available from: https://doi.org/10.1109/ICIT.2010.5472676.
3) Biswas S, Tai-Hoon K, Bhattacharyya D. Features extraction and verification of signature image using clustering technique. *International Journal of International Journal of International Journal of International Journal of Smart*. 2010;p. 43–55.
4) Gaur A, Yadav S. Handwritten Hindi character recognition using k-means clustering and SVM. In: 4th International Symposium on Emerging Trends and Technologies in Libraries and Information Services. 2015;p. 65–70. Available from: https://doi.org/10.1109/ETTLIS.2015.7048173.
5) Guruprasad P, Majumdar DJ. Optimal Clustering Technique for Handwritten Nandinagari Character Recognition. *International Journal of Computer Applications Technology and Research*. 2017;6(5):213–223. Available from: https://dx.doi.org/10.7753/ijcatr0605.1001.
6) Methasate I, Sae-Tang S. The clustering technique for Thai handwritten recognition. In: Ninth International Workshop on Frontiers in Handwriting Recognition. 2004;p. 450–454. Available from: https://doi.org/10.1109/IWFHR.2004.101.
7) Xiaodong J, Wendong G, Jie Y. Handwritten Yi Character Recognition with Density-Based Clustering Algorithm and Convolutional Neural Network. In: IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC). 2017;p. 337–341. Available from: https://doi.org/10.1109/CSE-EUC.2017.67.
8) Surinta O, Karaaba FM, Schomaker RBL, Wiering MA. Recognition of handwritten characters using local gradient feature descriptors. *Engineering Applications of Artificial Intelligence*. 2015;45:405–414. Available from: https://dx.doi.org/10.1016/j.engappai.2015.07.017.
9) Garz A, Diem M, Sablatnig R. Detecting Text Areas and Decorative Elements in Ancient Manuscripts. In: 12th International Conference on Frontiers in Handwriting Recognition. 2010;p. 176–181. Available from: https://doi.org10.1109/ICFHR.2010.35,2010.
10) Kumar M, Jindal M, Sharma R. k-nearest neighbor based offline hand-written Gurmukhi character recognition. *International Conference on Image Information Processing*. 2011;p. 1–4.
11) Ashir AM, Shehu GS. Adaptive clustering algorithm for optical character recognition. In: 7th International Conference on Electronics, Computers and Artificial Intelligence(ECAI). 2015;p. 13–16. Available from: https://doi.org/10.1109/ECAI.2015.7301192.
12) Soheili MR, Kabir E, Stricker D. Stricker, D, Sub-word image clustering in Farsi printed books. In: and others, editor. 7th International Conference on Machine Vision;vol. 9445. 2014.
13) Soheili MR, Yousefi MR, Kabir E, Stricker D. Merging clustering and classification results for whole book recognition. In: 10th Iranian Conference on Machine Vision and Image Processing (MVIP). 2017;p. 134–138. Available from: https://doi.org/10.1109/IranianMVIP.2017.8342338.
14) Panda, & Nayak S, & Nayak AM. Clustering of Odia Character Images Using K-Means Algorithm and Spectral Clustering Algorithm. 2020. doi:https://doi.org/10.1007/978-981-13-8461-5_7.
15) Chen J, Moon YS. Using SIFT features in palmprint authentication. In: 19th International Conference on Pattern Recognition. 2008;p. 1–4. Available from: https://doi.org/10.1109/ICPR.2008.4761867.
16) Badrinath GS, Gupta P. Palmprint Verification using SIFT features, First Workshops on Image Processing Theory, Tools and Applications. 2008;p. 1–8. Available from: https://doi.org/10.1109/IPTA.2008.4743763.

17) Rathore R, Prakash S, Gupta P. Efficient human recognition system using ear and profile face. In: IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS). 2013;p. 1–6. Available from: https://doi.org/10.1109/BTAS.2013.6712755.
18) Dhillon IS, Modha DS. *Machine Learning*. 2001;42:143–175. Available from: https://dx.doi.org/10.1023/a:1007612920971.
19) Manjunath MG, Devarajaswamy GK. Kannada Lipi Vikasa. 1st ed. Trust JSM, Matta SRS, Mantralaya, editors;Sri Raghavendra Swami Matta, Mantralaya. Jagadhguru Sri Madhvacharya Trust. 2004.
20) Belagali N, Shanmukhappa A, Angadi. OCR for Handwritten Kannada Language Script. *International Journal of Recent Trends in Engineering & Research (IJRTER)*. 2016;02(08). Available from: https://dopi.org/10.1109/ICPR.2008.4761867.
21) Sagar BM, Shobha G, P, Kumar R. Complete Kannada Optical Character Recognition with syntactical analysis of the script. In: International Conference on Computing, Communication and Networking. 2008;p. 1–4. Available from: https://doi.org/10.1109/ICCCNET.2008.4787744.
22) Manjunath AE, Sharath B. Implementing Kannada Optical Character Recognitionon theAndroid Operating System for Kannada Sign Boards. *International Journal of Advanced Research in Computer and Communication Engineering*. 2013;2(1).
23) Chandrakala HT, Thippeswamy D, G. A Comprehensive Survey onOCR Techniques for Kannada Script. *International Journal of Science and Research (IJSR)*. 2016;5(4).
24) Kumar HRS, Ramakrishnan AG. Lipi Gnani. *ACM Transactions on Asian and Low-Resource Language Information Processing*. 2020;19(4):1–23. Available from: https://dx.doi.org/10.1145/3387632.
25) Bai L, Cheng X, Liang J, Shen H, Guo Y. Fast density clustering strategies based on the k-means algorithm. *Pattern Recognition*. 2017;71:375–386. Available from: https://dx.doi.org/10.1016/j.patcog.2017.06.023.