Received: February 18, 2020. Revised: April 2, 2020.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Distance Matrix Based Keypoint Selection for Bag-of-Visual Words Model on Image Classification

Catur Supriyanto[1,2] **Hanung Adi Nugroho[1]** **Teguh Bharata Adji[1]***

[1]*Department of Electrical and Information Engineering, Faculty of Engineering, Universitas Gadjah Mada, Yogyakarta, 55281, Indonesia*
[2]*Department of Informatics Engineering, Faculty of Computer Science, Universitas Dian Nuswantoro, Semarang, 50131, Indonesia*
* Corresponding author's Email: adji@ugm.ac.id

Abstract: Bag-of-visual words (BoVW) becomes the most popular approach for local features extraction in image classification. A large number of keypoints lead to high computational costs of visual words generation. Iterative keypoint selection (IKS) as the baseline method has been proposed to reduce the number of keypoints by selecting the representative keypoints. However, random initial keypoint of IKS leads to non-repeatable results. Thus, this paper proposes a distance matrix based keypoint selection (DMKS) algorithm to reduce the number of keypoints. The novelty of this algorithm is the number of representative keypoints can be adjusted to obtain a high classification accuracy and the algorithm does not need random initial keypoints. The performance of proposed algorithm is then compared with that of IKS1 and IKS2. Support vector machine (SVM), K-Nearest Neighbor (KNN), and deep learning (DL H2O) classifiers are used to evaluate the algorithms on the public datasets. On Coil-100 and Caltech-101 datasets, DMKS achieves classification accuracy of 90.22% and 41.99%, respectively. Although the accuracy of DMKS is slightly increase, the processing time of DMKS is faster than the baseline methods. DMKS also produces a smaller number of representative keypoints, therefore DMKS can reduce the time-consuming of visual words generation more effectively in BoVW model.

Keywords: Distance matrix, Keypoint selection, Bag-of-visual words, Image classification.

1. Introduction

There are two major fields in the computer vision, i.e., object and scene classifications. Both can use local or global image features to classify the individual image [1]. Bag-of-visual words (BoVW) is a popular approach for local feature extraction in image retrieval. BoVW has been implemented in many studies, such as object classification [2], scene classification [3], signature verification [4], and pornographic image detection [5].

In the image classification based on BoVW model, features (called visual words) are generated in several stages. The first is keypoint extraction. The number of keypoints vary from hundreds or thousands in each image. Then, these keypoints are grouped by using clustering algorithms. K-means is the most widely used clustering algorithm in BoVW model [6]. The centroids of each cluster are used as visual words. Therefore, the number of visual words depend on the number of clusters.

A problem arises when many keypoints are extracted in each image. A large number of keypoints lead to high computational time of visual words generation. To handle this problem, W.-C. Lin, C.-F. Tsai, Z.-Y. Chen, and S.-W. Ke [7] proposed two keypoint selection algorithms, i.e., iterative keypoint selection (IKS1 and IKS2). The algorithms aim to select a subset of keypoints from an image for generating the visual words. The more discriminative of a keypoint the more appropriate to be selected as a representative keypoint. The discriminative of the keypoints have the large disimilarity to the other keypoints. Unfortunately, IKS1 and IKS2 select the

Algorithm 1 Iterative Keypoints Selection 1 (IKS1)

Require: An image that contains m keypoints ($Reduced\_Keypoints$)
Ensure: Selected Keypoints (SK)
Initialize threshold value T for the distance parameter
Randomly pick a keypoint from $Reduced\_Keypoints$ as the representative keypoint (RK) and put it in SK
for each keypoint in $Reduced\_Keypoints$
    Compute the distance between each keypoint and RK using Eq. (1)
    if the distance $\leq T$
        Remove the keypoint from $Reduced\_Keypoints$
    else
        Keep the keypoints in $Reduced\_Keypoints$
    end if
end for
Stop if there is no keypoint in $Reduced\_Keypoints$. Otherwise, go to step 2
Return SK

Algorithm 2 Iterative Keypoints Selection 2 (IKS 2)

Require: An image that contains m keypoints ($Reduced\_Keypoints$)
Ensure: Selected Keypoints (SK)
Initialize threshold value T for the distance parameter
Initialize number of clusters k (the parameter of performing k-means)
[Cluster_ID, centroid]=kmeans ($Reduced\_Keypoints, k$)
for each cluster $i \in (1, \dots, k)$
    Find the centroid of cluster i
    Find the keypoint as the representative keypoint (RK) in the cluster i with the minimum distance to the centroid and put it in SK
    for each keypoint in the cluster i
        Compute the distance between each keypoint and RK using Eq. (1)
        if the distance $\leq T$
            Remove the keypoint from $Reduced\_Keypoints$
        else
            Keep the keypoints in $Reduced\_Keypoints$
        end if
    end for
end for
Stop if the size of $Reduced\_Keypoints < k$ or there is no keypoint with a distance $\leq T$. Otherwise, go to step 3
Return SK

initial keypoint randomly, which lead to different results in each execution.

The problem of random initialization also happened in other algorithms, such as k-means clustering, k-modes clustering, and backpropagation neural network (BPNN). K-means and k-modes use a random choice to select initial centers and BPNN generates a random initial weights in the first iteration. Random initialization of the algorithms leads to non-repeatable and undesirable results in the experiments [8].

This study proposes a distance matrix based keypoint selection (DMKS) algorithm. Similar to IKS1 and IKS2, DMKS is also simple to be applied. DMKS uses distance matrix of keypoints to avoid the initial centroid. Based on the distance matrix, we select the discriminative keypoints. The proposed keypoint selection has two advantages: there is no random keypoint initialization and the top representative keypoints can be selected for visual words generation. In our perspective, we proposed keypoint selection based on distance approach, W.-C. Lin, C.-F. Tsai, Z.-Y. Chen, and S.-W. Ke [7] is the most similar with our study. Our proposed method is also to tackle the problem of IKS1 and IKS2.

The rest of the paper is organized as follows. Section 2 describes the theory of BoVW and keypoint selection. Section 3 describes materials and methods. Section 4 represents the experimental results and provides some discussions. Section 5 draws the conclusions.

Algorithm 3 Distance Matrix Based Keypoint Selection (DMKS)

Require: An image that contains m keypoints
Ensure: Selected Keypoints (SK)
Initialize threshold value T for the distance parameter
Initialize number of selected keypoints $n\%$
Generates $m \times m$ distance matrix of keypoints
for i from 1 to m do
  for j from 1 to m do
    if i is not equal j then
      Find the distance between i-th and j-th keypoints
      if distance $> T$ then
        $weight_{keypoint_i} \leftarrow weight_{keypoint_i} + 1$
      end if
    end if
  end for
end for
Sorted the weight of keypoints in descending order
Put the top $n\%$ keypoints into SK
Return SK

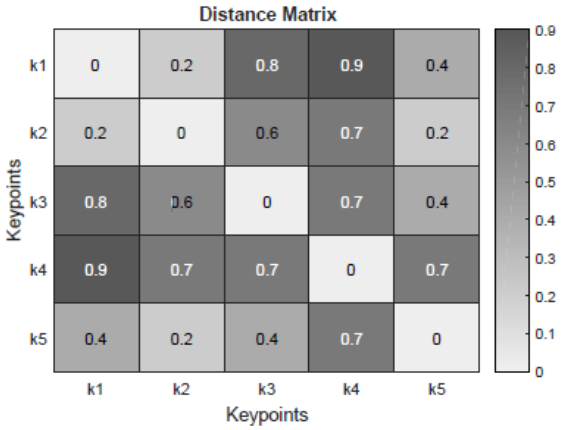


Figure. 1 DMKS illustration of 5 keypoints

2. Bag of visual words model and keypoint selection

BoVW model consists of several stages to generate the features, i.e., keypoint extraction, visual words generation, and histogram of visual words construction. Some methods have been applied for keypoint extraction, including Scale Invariant Features Transform (SIFT) [9] and Speeded-Up Robust Features (SURF) [10]. The extracted keypoints of SIFT are robust to changes in viewpoint, illumination and affine distortion [11]. The superiority of SIFT over the other keypoint extraction methods has been shown by Mikolajczyk and Schmid [12] by comparing ten different keypoint extraction methods and the result shows that SIFT performs better.

BoVW deals with the problem of computational cost in the visual words generation. Many studies use k-means algorithm to generate the visual words. Another approach, T. Urruty, S. Gbehounou, H. T. Le, J. Martinet, and C. Fernandez [13] proposed an iterative random visual words selection. They chose randomly the keypoints to be the candidate visual words without clustering algorithm. The candidate visual words are weighted by using information gain (IG). The visual words with the highest IG values will be used to construct the histogram of visual words.

To reduce the number of keypoints, W.-C. Lin, C.-F. Tsai, Z.-Y. Chen, and S.-W. Ke [7] proposed IKS1 and IKS2 that is based on the iterative process. The pseudocode of IKS1 and IKS are presented in Algorithm 1 and Algorithm 2. The output of the algorithms is selected keypoint (SK) that will be used for visual words generation. In IKS1, m keypoint ($Reduced\_Keypoints$) are extracted from an image. A keypoint is selected randomly and this keypoint is regarded as representative keypoint (RK) and put it in

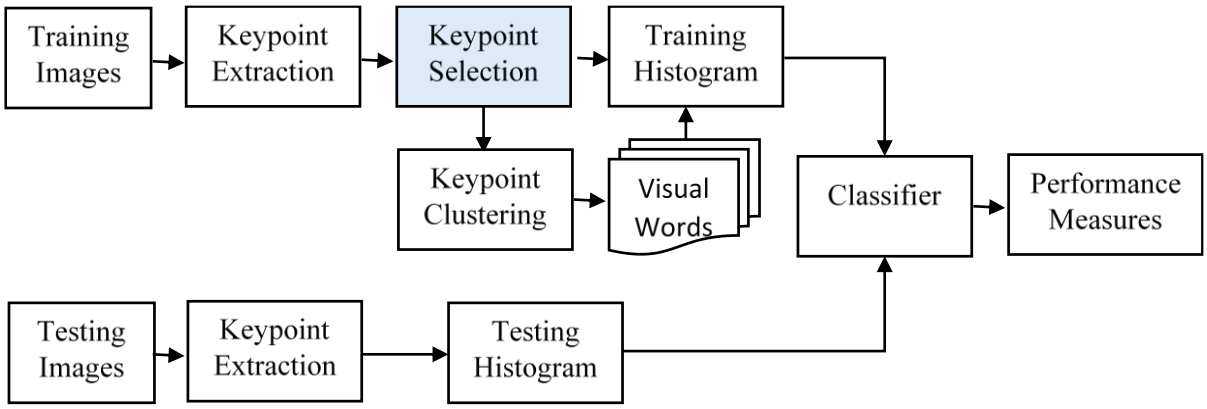


Figure. 2 The Process of BoVW image classification with keypoint selection

Algorithm 4 Initial Centroid Algorithm

Require: m number of keypoints
Ensure: k number of initial centroid
1: For each keypoint in image I calculate the distance from the origin (zero vector), e.g., data point (0,0) for 2D.
2: Sort the distances obtained in the previous step. In accordance with these distances sort the original data points.
3: Divide the sorted data points into k number of equal partitions.
4: In each partition, calculate the mean of the data points. These mean values will be taken as the initial centroids to be used in the k-means algorithm.

SK. Then, the Euclidean distance between each keypoint and RK is defined as

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

where x and y are two keypoints. The keypoint is a vector of length $n = 128$. The keypoint will be removed when the distance is below than predefined threshold T. The iteration of IKS1 stops until no longer keypoints exist in $Reduced\_Keypoints$.

IKS2 implements k-means clustering to group the keypoints. IKS2 is also based on the iterative process. In each iteration, the keypoints in $Reduced\_Keypoints$ are grouped into k number of clusters. In each generated cluster, the distance of a keypoint to its centroid is measured. The closest keypoint to the centroid are regarded as RK and put it in SK. The keypoint is removed when the distance is less than or equal to the predefined threshold, and the remaining keypoints will be used for the next iteration. The iteration stops when the number of remaining keypoints is smaller than k or the remaining keypoints in $Reduced\_Keypoints$ cannot be removed (distance $> T$).

3. Material and methods

3.1 Proposed method

The proposed keypoint selection method is based on distance matrix. The method builds $m \times m$ distance matrix from m keypoints of an image. A distance matrix is a table that show the distance between pairs of keypoints. The distance is calculated using the Euclidean distance as shown in Eq. (1).

From each row of the matrix, the method calculates the number of keypoints (number of column) that have the distance greater than the predefined threshold. Different from IKS1 and IKS2, the predefined threshold of DMKS does not reduce the number of keypoints. DMKS uses the predefined threshold to calculate the weight of keypoints ($weight_{keypoint_i}$). For example, Fig. 1 shows the distance matrix of five keypoints. If the predefined distance threshold is 0.5, then the first iteration (first row or k_1), results in two keypoints (k_3 and k_4) where the distance is higher than the predefined threshold. Therefore, the weight of k_1 is 2. In the second iteration (second row or k_2), results in two keypoints (k_3 and k_4) where the distance is higher than the predefined threshold. Therefore, the weight of k_2 is 2. The process repeats until the last row. Finally, in the descending order, the weight of k_4,

Table 1. Number of total images, training images, and extracted keypoints of training images

	No. Images	No. Training Images	No. Keypoints
Coil-100	7,200	1,000	60,029
Caltech-101	8,677	1,010	465,138

k_3, k_2, k_1, and k_5 are 4, 3, 2, 2, and 1, respectively. The result shows that k_4 is the most discriminative keypoint, since k_4 has the highest weight. In the next process, we can use k_4 and k_3 as the top 20% for visual words generation. Algorithm 3 shows the pseudocode of DMKS.

3.2 Data collection

The golden datasets for this research are Coil-100 and Caltech-101 datasets. These datasets are commonly used for object classification. Coil-100 dataset contains 100 object categories. Totally 7,200 images in Coil-100 dataset. The second dataset is Caltech-101 which contains 101 object categories. There are 8,677 images in Caltech-101 dataset. In this study, 10 images of each category are used to generate the visual words. The detailed number of images, training images and extracted keypoints of both datasets are shown in Table 1.

3.3 Experimental steps and methods

The process of image classification with keypoint selection is shown in Fig. 2. Generally, the process is divided into several steps.

3.3.1. Keypoint extraction

The images are split into training and testing images. Keypoints are extracted by using Scale Invariant Feature Transform (SIFT) for each training and testing image. The keypoints of SIFT have 128 dimensional vectors. In the developed BoVW model, the visual words is constructed from the training images, and not from the entire images, which is similar to the training process by L. Zhuo, Z. Geng, J. Zhang, and X. g. Li [14]. These histograms are able to be used as input for the classifier.

3.3.2. Keypoint selection

In the training process, the number of extracted keypoints are reduced by the keypoints selection method. This study compares the proposed keypoint selection DMKS with the baseline methods IKS1 and IKS2. In order to get the optimal performance, some parameters can be setup. The three methods need to adjust the distance threshold. IKS1 and IKS2 reduce the keypoints by adjusting the distance threshold to reduce the number of keypoints. Meanwhile, DMKS adjusts the distance threshold only for weighting the number of keypoints. The distance threshold ranges from 0.1 to 0.9 with an interval of 0.1.

Additional adjustment is required for IKS2 and DMKS. IKS2 needs to adjust the number of clusters in k-means. W.-C. Lin, C.-F. Tsai, Z.-Y. Chen, and S.-W. Ke [7] chose $k = 3$ because the number produces good classification accuracy in Caltech-101 dataset. Whereas, DMKS also needs to adjust the percentage of number of keypoints to reduce the number of keypoints. The percentage of number of keypoints ranges from 10% to 90% with an interval of 10%.

3.3.3. Keypoint clustering

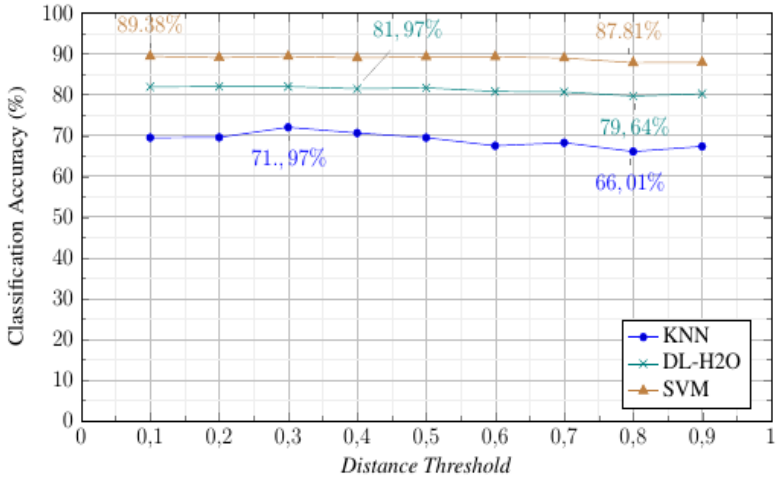
K-means algorithm is used to group the selected keypoints and to generate a set of centroids. To overcome the random problem of initial centroid, this study uses a method from Goyal and Kumar [15] to select the initial centroid of k-means (see Algorithm 4). These centroids are used as visual words to construct the histogram or feature vector of each image. Therefore, the number of visual words is similar to the number of clusters or centroids. The experiment uses 200 visual words. In the similar research and dataset, the number produces better classification accuracy than 100 visual words [7].

3.3.4. Histogram generation

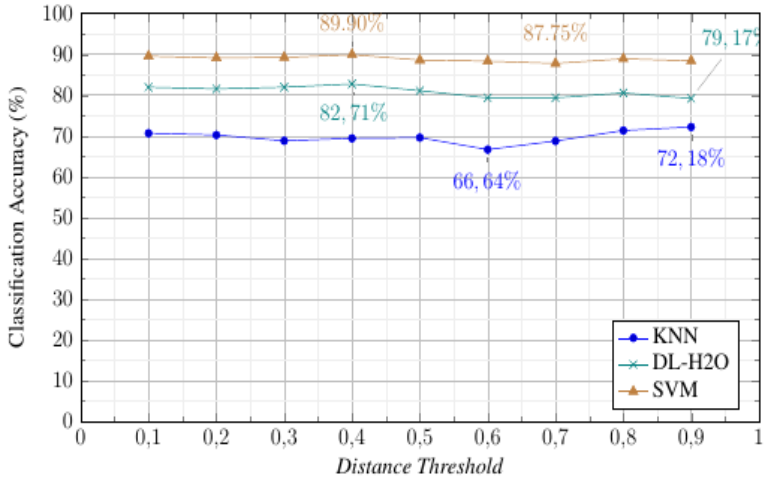
The distance between each keypoint and each visual word was then measured. The keypoint which has the minimum distance with the visual word is assigned. The frequency of keypoints in each visual word is counted to construct the histogram of visual words. The concept of intra-class and inter-class term distributions proposed by H. Zhou, J. Guo, and Y. Wang [16] is adopted into our weighting scheme. The weighting scheme has been proven successful in image classification [17]. The weight of each visual words is computed by the following equations:

$$s(t_i)^2 = \frac{1}{K}\sum_{k=1}^{K}\left(\overline{tf_{kl}} - \overline{tf_l}\right)^2 \quad (2)$$

$$s(t_{ki})^2 = \frac{1}{|C_k|}\sum_{j\in C_k}^{K}\left(tf_{ij} - \overline{tf_{kl}}\right)^2 \quad (3)$$



(a)



(b)

Figure. 3 Classification accuracy of: (a) IKS1 and (b) IKS2 with different distance threshold and different number of selected keypoints over Coil-100 dataset

$$F(t_{ki}) = \frac{s(t_i)^2}{s(t_{ki})^2 + 1} \times \frac{\overline{tf_{kl}}}{\overline{tf_l}} \quad (4)$$

$$\lambda_2 = \frac{K!}{(K-2)!.2!} \quad (5)$$

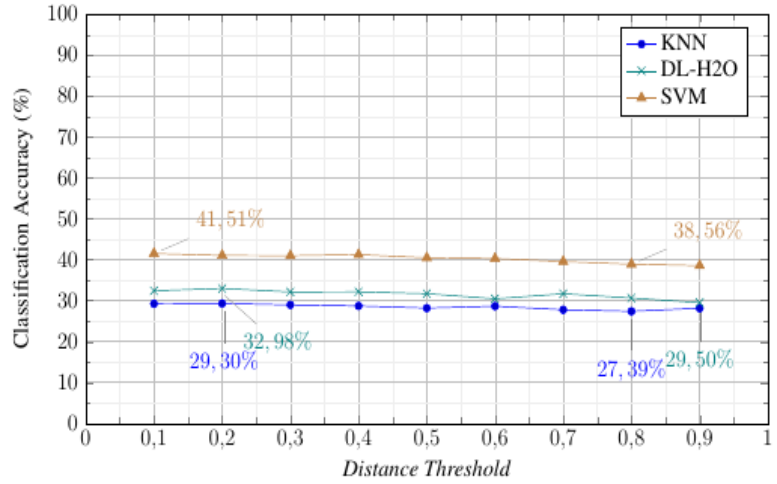
$$G(t_i) = \frac{1}{\lambda_2} \sum_{1 \le q < r \le K} \left| F(t_{q,i}) - F(t_{r,i}) \right| \quad (6)$$

$$w(t_i, d) = tf_{id} \times G(t_i) \quad (7)$$

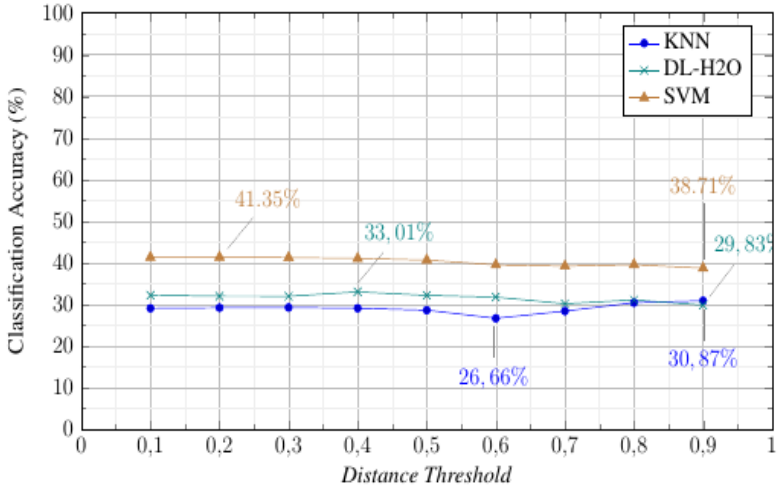
Here, f_{ij} is term frequency of term t_i in document j, $\overline{tf_l}$ is the average term frequency of term $t_i$in the collection of documents, $\overline{tf_{lk}}$ is the average term frequency of term t_i in the category k, $|C_k|$ is the document frequency of term t_i in the category k, and K is the number of categories. The weight of each visual word is calculated by Eq. (7).

3.3.1 Classification

In this study, support vector machine (SVM), k-nearest neighbor (KNN), and deep learning (DL-H2O) are used as the classifiers. SVM is effective in high dimensional data with small number of training dataset [1]. SVM is a binary classifier which classifies an object belongs to two distinct classes. SVM can be extended into multiclass classifier by using some approaches, such as one-vs-one, one-vs-all, binary tree, and error-correcting output codes (ECOC). We use ECOC with one-vs-one coding design [18] in our approach. In terms of accuracy and speed, ECOC provides better performance than other multiclass SVM approaches [19, 20]. SVM with Radial Basis Function (RBF) kernel is used in the



(a)



(b)

Figure. 4 Classification accuracy of: (a) IKS1 and (b) IKS2 with different distance threshold and different number of selected keypoints over Caltech-101 dataset

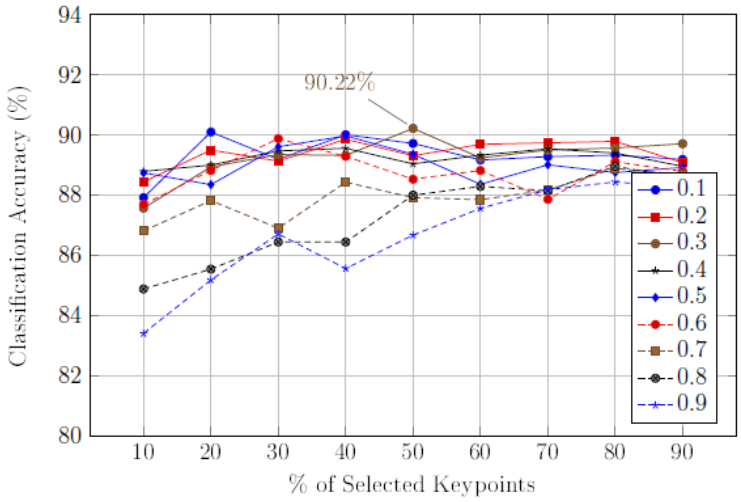
experiment. KNN is the simplest classification algorithm. KNN supports multiclass classification. KNN starts by computing the distance between testing data and each training data. The label of testing data is determined by the major label of its *k* nearest neighbor. Deep learning (DL) can be categorized into two type, such as deep neural networks (DNNs) and convolutional neural networks (CNNs) [21]. CNN is more complex which consists of feature extraction layers and classification layers. DNN is simpler which only have several classification layers. In the BoVW model, DNN is more suitable in comparison with other classifiers. This study use DL-H2O as DNN frameworks. The three classifiers are trained and tested in 10-fold cross validation. All of the experiments were conducted on MATLAB, except KNN and DL-H2O classifiers, which were implemented in Rapidminer.

3.4 Evaluation

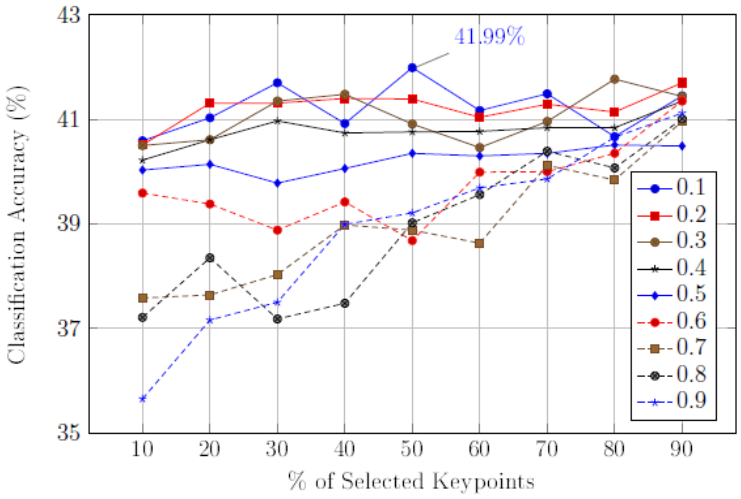
In this study, high classification accuracy, low processing time of the keypoint selection, and small number of generated keypoints are used to show that the keypoint selection method gives good performance. All the experiments are performed using a system with CPU Intel core i7 and a physical memory (RAM) of 8 GB.

Table 2. Difference in highest and lowest accuracy of IKS1 and IKS2 (in %)

Dataset	Algorithm	KNN	DL-H2O	SVM
Coil-100	IKS1	5.96	2.33	1.57
	IKS2	5.54	3.54	2.15
Caltech-101	IKS1	1.91	3.48	2.95
	IKS2	4.21	3.18	2.64



(a)



(b)

Figure. 5 Classification accuracy of the proposed method with different distance threshold and different number of selected keypoints over: (a) Coil-100 dataset and (b) Caltech-101 dataset

4. Experimental results

4.1 Results of iterarive keypoint selection algorithm

We first investigate the classification accuracy of the previous algorithms on different classifiers, i.e. SVM, KNN, and DL-H2O. Fig. 3 shows the classification accuracy of IKS1 and IKS2 on Coil-100 dataset and Fig. 4 shows the classification accuracy of IKS1 and IKS2 on Caltech-101 dataset. In the evaluations, both algorithms achieve the highest accuracy on SVM classifier than the other classifier algorithms. SVM is becoming a successful classifier on both Coil-100 dan Caltech-101 datasets. Based on the figures, evaluation of different distance threshold in IKS1 and IKS2 does not affect significantly classification accuracy, as can seen in Table 2, the different between the lowest accuracy and highest accuracy in each classifier is 1% to 5%.

4.2 Results of Distance Matrix based Keypoint Selection

In the evaluation of the proposed method, we use SVM classifier, since SVM can successfully achieve high classification accuracy in comparison with KNN and DL-H2O. Fig. 5 shows the classification accuracy of the proposed method using Coil-100 and Caltech-101 datasets. In both datasets, the classification rates have a similar pattern. A small number of keypoints lead to a decrease in classification rate. A significant decrease occurs in the distance thresholds 0.7, 0.8, and 0.9. In the Coil-100 dataset, the proposed method gets the highest

Table 3. Processing time and number of selected keypoints of IKS1 and IKS2 on Coil-100 and Caltech-101 datasets

	Time (min)				No. Keypoints			
	Coil-100		Caltech-101		Coil-100		Caltech-101	
Distance Threshold	IKS1	IKS2	IKS1	IKS2	IKS1	IKS2	IKS1	IKS2
0.1	6.54	8.86	270.97	146.69	59,987	59,052	464,606	464,118
0.2	6.01	8.69	223.82	137.18	59,343	59,049	458,373	463,593
0.3	6.62	8.09	251.24	140.78	57,987	58,944	443,415	461,475
0.4	6.34	8.69	251.72	165.39	56,423	58,212	421,245	453,513
0.5	6.15	8.39	238.83	168.56	53,731	54,354	387,793	424,194
0.6	5.82	6.64	170.11	67.04	48,731	36,561	326,330	295,359
0.7	5.22	4.40	51.59	4.91	39,461	8,802	206,543	29,409
0.8	4.62	4.19	14.93	3.43	26,574	4,350	90,010	6,621
0.9	4.15	4.02	6.36	3.01	14,748	3,006	33,235	3,246

Table 4. Processing time of DMKS on Coil-100 and Caltech-101 datasets (in minutes)

% of selected keypoints	Coil-100		Caltech-101	
	No. Keypoints	Time (min)	No. Keypoints	Time (min)
10	4.08	6,037	18.94	46,573
20	5.06	12,002	14.00	93,016
30	4.21	18,055	13.45	139,603
40	7.61	24,028	15.59	186,058
50	5.82	30,279	13.19	232,814
60	7.85	36,001	18.00	279,080
70	6.82	42,049	15.11	325,625
80	6.19	48,027	18.97	372,122
90	4.46	54,090	16.42	418,680

Table 5. Comparison results of the proposed method and baseline methods

	IKS1			IKS2			DMKS		
	Acc	Time (min)	Num Keypoints	Acc	Time (min)	Num Keypoints	Acc	Time (min)	Num Keypoints
Coil-100	89.38%	6.54	59,987	89.90%	8.69	58,212	90.22%	5.82	30,279
Caltech-101	41.51%	270.97	464,606	41.35%	137.18	463,593	41.99%	13.19	232,814

classification accuracy of 90.22% on 50% selected keypoints when distance threshold 0.3 (the brown line with circle) is applied. Meanwhile, the highest accuracy of 41.99% in the Caltech-101 dataset is achieved on 50% selected keypoints when distance threshold 0.1 (the blue line with circle) is applied.

Table 3 shows the processing time and number of keypoints of IKS1 and IKS2 over both datasets. The higher number of distance threshold lead to produce small number of keypoints with less processing time. Meanwhile, Table 4 shows the processing time and number of selected keypoints of DMKS on both datasets. Compared to IKS1 and IKS2, adjusting the parameter of DMKS does not affect the processing time too much, since the process of removal is performed after iterations.

Table 5 shows the comparison results of DMKS and the baseline methods IKS1 and IKS2 in terms of classification accuracy, processing time, and number of selected keypoints. The performance in Table 5 obtained when the three methods yield highest classification accuracy in different parameter settings.

In Coil-100 dataset, IKS1 and IKS2 obtain the best accuracy of 89.38% and 89.90% when the distance thresholds are 0.1 and 0.4, respectively. Meanwhile, in Caltech-101 dataset, IKS1 and IKS2 achieve the best accuracy of 41.51% and 41.35% when the distance thresholds are 0.1 and 0.2, respectively. Compared to IKS1 and IKS2, DMKS produces a slightly higher classification accuracy in both datasets. In term of processing time, the performance of the three methods does not differ too much in Coil-100. However, a significant result is shown in Caltech-101 where DMKS improves the keypoint selection speed about 20 times faster than IKS1 and 10 times faster than IKS2. DMKS also successfully reduces the large number of keypoints in both datasets. In BoVW model, the small number of keypoints can reduce the processing time of keypoint clustering and therefore can speed up the visual words generation.

5. Conclusions and Future Works

This study proposes keypoint selection method based on distance matrix (DMKS). There are two following features in the proposed method. First, DMKS adopts a distance matrix to avoid the problem of randomly selecting initial keypoints and thus gives the fixed classification results. Second, DMKS is able to select the number of representative keypoints which produce high classification accuracy. Through comparative studies on two public datasets (i.e. Coil-100 and Caltech-101), the classification accuracy of the proposed method is slightly higher than that of the baseline methods. In the Coil-100, the proposed DMKS method, IKS1 method, and IKS2 method produce an accuracy of 90.22%, 89.38%, and 89.90%, respectively. Meanwhile, in the Caltech-101, the proposed DKMS, IKS1, and IKS2 achieve an accuracy 41.99%, 41.51%, and 41.35%, respectively. The proposed DKMS method successfully demonstrates lower computational cost and produces smaller number of representative keypoints significantly in each dataset. The small number of keypoints can effectively reduce the computational cost of keypoint clustering in generating visual words. For future work, reducing computational cost of the proposed method and avoiding randomize initial keypoint using another approach are still challenging.

References

[1] D. Zhang, M. M. Islam, and G. Lu, "A review on automatic image annotation techniques", *Pattern Recognition*, Vol. 45, No. 1, pp. 346-362, 2012.

[2] M. A. E.-D. Aly, M. Munich, and P. Perona, "Bag of words for large scale object recognition - properties and benchmark", In: *Proc. of the Sixth International Conference on Computer*, 2011.

[3] J. Zou, W. Li, C. Chen, and Q. Du, "Scene classication using local and global features with collaborative representation fusion", *Information Sciences*, Vol. 348, pp. 209-226, 2016.

[4] M. Okawa, "Offline signature verification based on bag-of-visual words model using kaze features and weighting schemes", In: *Proc. of IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016.

[5] C. C. Yan, Y. Liu, H. Xie, Z. Liao, and J. Yin, "Extracting salient region for pornographic image detection", *Journal of Visual Communication and Image Representation*, Vol. 25, pp. 1130-1135, 2014.

[6] C.-F. Tsai, "Bag-of-words representation in image annotation: A review", International Scholarly Research Network ISRN Artificial Intelligence, pp. 1-19, 2012.

[7] W.-C. Lin, C.-F. Tsai, Z.-Y. Chen, and S.-W. Ke, "Keypoint selection for efficient bag-of-words feature generation and effective image classification", *Information Sciences*, Vol. 329, pp. 33-51, 2016.

[8] S. S. Khan and A. Ahmad, "Cluster center initialization algorithm for K-modes clustering", *Expert Systems with Applications*, Vol. 40, pp. 7444-7456, 2013.

[9] D. G. Lowe, "Distinctive image features from scale-invariant keypoints", *International Journal of Computer Vision*, Vol. 60, pp. 91-110, 2004.

[10] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features", In: *Proc. of the ninth European Conference on Computer Vision*, 2006.

[11] Y. Xie, S. Jiang, and Q. Huang, "Weighted visual vocabulary to balance the descriptive ability on general dataset", *Neurocomputing*, Vol. 119, pp. 478-488, 2013.

[12] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors", *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 110, pp. 1615-1630, 2005.

[13] T. Urruty, S. Gbehounou, H. T. Le, J. Martinet, and C. Fernandez, "Iterative random visual word selection", In: *Proc. of International Conference on Multimedia Retrieval*, 2014.

[14] L. Zhuo, Z. Geng, J. Zhang, and X. g. Li, "ORB feature based web pornographic image recognition", *Neurocomputing*, Vol. 173, pp. 511-517, 2016.

[15] M. Goyal and S. Kumar, "Improving the initial centroids of k-means clustering algorithm to generalize its applicability", *J. Inst. Eng. India Ser. B,* Vol. 95, pp. 345-350, 2014.

[16] H. Zhou, J. Guo, and Y. Wang, "A feature selection approach based on term distributions", *SpringerPlus*, Vol. 5, pp. 245-260, 2016.

[17] C. Supriyanto, H. A. Nugroho, and T. B. Adji, "A global weighting scheme based on intra-class and inter-class term distributions in bag-of-visual words image classification", *IAENG International Journal of Computer Science*, Vol. 45, pp. 228-236, 2018.

[18] T. G. Dietterich and G. Bakiri, "Solving multiclass learning problems via error-correcting output codes", *Journal of Artificial Intelligence Research*, Vol. 2, pp. 263-286, 1995.

[19] F. Deng, S. Guo, R. Zhou, and J. Chen, "Sensor multifault diagnosis with improved support vector machines", *IEEE Transactions on Automation Science and Engineering,* Vol. 14, pp. 1053-1063, 2017.

[20] A. Ayodeji and Y. k. Liu, "Support vector ensemble for incipient fault diagnosis in nuclear plant components", *Nuclear Engineering and Technology*, Vol. 50, pp. 1306-1313, 2018.

[21] G. Nguyen, S. Dlugolinsky, M. Bobák, V. Tran, Á. L. García, I. Heredia, P. Malík, and L. Hluchý, "Machine Learning and Deep Learning frameworks and libraries for large-scale data mining: a survey", *Artificial Intelligence Review*, Vol. 52, pp. 77–124, 2019.