The Imaging Science Journal

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/yims20

Squirrel Search Optimization-based near-duplicate image detection

Srinidhi Sundaram, Kamalakkannan Somasundaram & Sasikala Jayaraman

To cite this article: Srinidhi Sundaram, Kamalakkannan Somasundaram & Sasikala Jayaraman (26 Nov 2024): Squirrel Search Optimization-based near-duplicate image detection, The Imaging Science Journal, DOI: 10.1080/13682199.2024.2390793

To link to this article: https://doi.org/10.1080/13682199.2024.2390793

Published online: 26 Nov 2024.

THE IMAGING SCIENCE JOURNAL
https://doi.org/10.1080/13682199.2024.2390793

THE RPS ROYAL PHOTOGRAPHIC SOCIETY
Taylor & Francis Taylor & Francis Group

RESEARCH ARTICLE

Squirrel Search Optimization-based near-duplicate image detection

Srinidhi Sundaram[a], Kamalakkannan Somasundaram[b] and Sasikala Jayaraman[a]

[a]Faculty of Engineering and Technology, Department of Information Technology, Annamalai University, Annamalainagar, India; [b]Department of Information Technology, School of Computing Sciences, Vels Institute of Science, Technology & Advanced Studies, Chennai, India

ABSTRACT

Near duplicate (ND) image detection is a significant issue in a modern online environment with a wide range of applications like the detection of copyright violations and saving of storage space. Several existing ND detection techniques are perhaps not suitable for online applications due to the large computational burden, and may not successfully detect NDs containing large smooth and plain regions. In addition, the K-means algorithm used in most of the existing methods yield sub-optimal quantization of visual words. This article employs a robust algorithm of Squirrel Search Optimization (SSO) for quantization, fast-hessian matrix-based detector (FHMBD) and FAST Corner Detector (FCD) for the detection of KPs at both plain and non-smooth regions of all images, SURF for computing descriptors and Principal Component Analysis (PCA) for dimensionality reduction. The results of the developed method presented on five image databases. The proposed method offered 99.9% accuracy, 98.67% sensitivity, and 99.91% specificity respectively.

ARTICLE HISTORY
Received 2 April 2024
Accepted 3 August 2024

KEYWORDS
Near duplicate (ND) image detection; Squirrel Search Optimization (SSO); fast-hessian matrix-based detector (FHMBD); Principal Component Analysis (PCA); FAST Corner Detector (FCD)

Nomenclature

A_s	The body's surface area
CC^k	Climate constant at k-th iteration
$CM_{\min}$	Minimum value of climate constant
CNDDM	Classical ND Detection Method
CNN	Convolution Neural Network`
$Dist_{SQ}$	Distance glided by the squirrel
D_j	j-th descriptor
D'_j	j-th transformed descriptor
D^R_j	j-th reduced descriptor
$D^R_{j,i}$	i-th value of the j-th reduced descriptor
FCD	FAST Corner Detector
FD	False Duplicate
FHMBD	Fast-Hessian Matrix-Based Detector
FO	False Original
G_d and G_c	Random gliding distance and gliding constant, respectively
HL_{SQ}	Height loss after gliding
K_D	Drag coefficient
K_L	Lift coefficient
KP	Key Point
$LF(n)$	n-dimensional Levy Flight
m	The number of squirrels in the population
ND	Near Duplicate
nf	Number of feature vectors in the entire image database
nv	Length of reduced feature descriptor
PCA	Principal Component Analysis
P_{prob}	Predator probability constant
RANSAC	Random Sample Consensus
r_a and r_b	Random numbers (0,1)
SIFT	Scale Invariant Features Transform
SQ_p	Position of p-th squirrel
$SQ_p(0)$	Initial position of p-th squirrel
SQ_{low} & SQ_{up}	Lower and upper limits for squirrels, respectively
SQ^k_{at}	Squirrel presently sitting on acorn nut tree at k-th instant
SQ^k_{ht}	Squirrel sitting on hickory nut tree at k-th instant
SQ^k_{nt}	Squirrel sitting on a normal tree at k-th instant
SSO	Squirrel Search Optimization
SSONDM	SSO-based ND detection method
SURF	Speeded Up Robust Features
S_p	Speed
TD	True duplicate
TO	True original
w_k	k-th visual word
$w_{k,i}$	i-th value of k-th visual word
μ	Air density
θ	Gliding angle
β	A constant value

Introduction

Many of the images available on websites are duplicates and near-duplicate (ND) images of one another among the several trillions of images. This phenomenon is most noticeable in photos of well-known themes like celebrities, music albums, and art films. A duplicate is a digital image that is nearly identical to the original in terms of user perception, with the only differences being in scale, colour schemes, or storage format. In contrast, a near-duplicate (ND) has additional variations in terms of contract, luminance, rotation, translation, and a small alteration in the original document's background. The problem of comparing a pirated image to its original in huge image databases is known as ND image detection, which is

CONTACT Srinidhi Sundaram srinidhisristar@gmail.com Faculty of Engineering and Technology, Department of Information Technology, Annamalai University, Annamalainagar 608002, TN, India

one specific use of content-based similarity search, and is essential for visual search applications. It finds applications in many areas such as (a) reduction of storage space, (b) detection of copyright infringement and (c) understanding the public's behaviour and interests. ND detection of digital images is more arduous than detecting exact duplicates as the decomposition of an image still pertains to the domain of fiction and no grammar yet exists for images [1].

Several content-based ND detection methods have been suggested in recent decades [2]. Many of these techniques extract pertinent visual characteristics of photos from the database or websites, compare them with the attributes of query image (QI), and designate the images having properties similar to that of the QI as NDs [3]. Many of them identify key points (KPs) and employ Scale Invariant Features Transform (SIFT), Speeded Up Robust Features (SURF), etc. for extracting local descriptors at identified KPs. They then perform Training and Querying. During training, the extracted features are indexed through hashing or quantization into visual words (VW) [4]. In the querying phase, the descriptors of a QI are evaluated and searched in the indexed image database for nearest neighbours. The score of every database image that has such nearest neighbours is increased. Then the geometric consistency of the matched KPs is checked for true matching and the false matching is eliminated [5]. The identified images in the image database are reranked and the high-ranked digital images are treated as NDs. These approaches are still inefficient due to huge KPs and large-sized descriptors and may fail due to sparse KPs at smooth regions. Moreover, the K-means technique employed for training may yield local minima [6]. These drawbacks necessitate some kind of dimensionality reduction of descriptors before training and a robust optimization algorithm for realizing the global best solution during the quantization of descriptors.

Recently, a Squirrel Search Optimization (SSO), inspired by the gliding and dynamic movement behaviours of flying squirrels, has been presented for solving optimization problems [7], and exhibited to be better than those of other population-based approaches like particle swarm optimization, honeybee optimization, teaching, and learning-based optimization, and so on. In SSO, the flying squirrels indicating feasible solutions glide from one tree to another in search of acorns and hickory nuts and utilize the available food resources either for eating or for storing depending on their availability. The SSO was used for solving several practical optimization problems such as classification [8] and identification of non-line of Sight nodes in Vehicular Ad hoc Networks [9].

The prime goal of this article is to build an efficient and robust ND image detection method, suitable for large image databases, by obtaining fewer KPs using two different techniques, computing the descriptors by SURF, lowering the dimension of the descriptors by Principal Component Analysis (PCA) and applying SSO for quantization. The performances of the developed method on five image databases have been discussed in this article.

The article is tailored into five sections. Section 'Literature Survey' surveys the literature; Section 'Near-Duplicate Detection' explains the classical ND detection methods; Section 'Proposed SSO Based ND Detection' describes the developed ND detection method; Section 'Results and Discussions' discusses the performances; and Section 'Conclusion' concludes.

Literature survey

An efficient and robust methodology for image duplicate detection was developed for large databases. The method used a Fourier Mellin Transform-based compact feature vector having 12-values, thereby enabling fast indexing and querying in large databases [10]. A support vector machine-based ND detection technique was developed. The technique used point, region and probabilistic filters to avoid noisy attributes. The support vector machine matched the descriptors of all images in the database and detected the NDs [11]. An ND video detection scheme, involving both MPEG-7 and semantic descriptors, was outlined to make the scheme more robust to transformations [12].

A phase correlation technique employing polar band control was suggested for image duplicate identification. The method computed Fourier transform of overlapping blocks, and then adopted adaptive band control for obtaining a correlation matrix. Then seed filling was performed to show the duplicated parts [13]. An ND video detection algorithm, involving index and query signature-based colour and texture descriptors, was developed. The algorithm used a logical model by relating an N-gram sliding window with lattice-based structures [14]. A SIFT-based ND image detection was suggested for increasing the confidence and making it suitable for large image databases. The method adopted entropy-based filtering to eliminate ambiguous descriptors causing false positive matches and graph cut for query expansion to improve exploration quality. An attempt was made to lower the computational burden of the block-matching-based duplicate detection scheme by employing sequential hashing. The hashing lowered the problem space and drastically lowered computation time by eliminating block-matching operations [15].

Four robust features involving radon transform were developed and combined through a uniform sampling approach. Such composite features were superior to that of a single type of feature in respect of robustness, blurring and other geometrical transformations. Such composite features were applied to ND detection and exhibited to offer better performance over their

counterparts [16]. A scheme was developed for encoding an image as a binary vector . The encoding scheme converted the local regions of the image into texture descriptors, which were used to form a statistical histogram and then denoted by a binary vector. The scheme representing both local and global descriptors was used in ND image detection with the advantages of higher speed of computations and lower storage requirements [17].

An ND image detection method, involving multi-feature fusion, was suggested to avoid the spreading of false rumours by tracking the original images. The method used a bloom filter to improve the retrieval accuracy [18]. An ND image detection approach adopting a hierarchical neural network and sensitivity hashing was suggested. The neural network extracted descriptors, which were then used for efficiently constructing the locality sensitivity hashing-based index [19]. A pulse-linked neural network and correlation coefficient-based scheme was developed for detecting ND images. The neural network was used for extracting descriptors, and a correlation coefficient-based technique was used for feature matching. The study showcased that the scheme was efficient and accurate [20]. A morphological operator-based detection of duplicated regions in digital images was suggested. The study on 100 images portrayed that the method was efficient and robust to noise and transformations [21].

A block-based scheme involving contourlet transformation-based approximated band and singular value descriptors was presented for image forgery detection and was efficient [22]. A hybrid algorithm, adopting Fourier-Mellin and Zernike moments and requiring low computations with less complexity for detecting image forgeries, was outlined [23]. An unsupervised ND image detection employing deep learning-based descriptors was developed. The method fine-tuned deep convolutional networks to obtain the best performances [24].

A composite feature-matching technique integrating the SIFT and Convolution Neural Network (CNN) descriptors was outlined for ND image identification. The method initially created bag-of-visual-words, performed SIFT descriptor-based matching for detecting the duplicate pairs, and then matched the CNN descriptors to assess the image similarity [25]. A two-stage fast and accurate matching scheme adopting CNN descriptors was suggested for ND image detection. In the first stage, a sum-of-pooling was performed on convolutional descriptor maps for generating global descriptors, while in the second-stage, the local descriptors were obtained from the convolutional descriptor maps and the saliency map was obtained using the visual saliency detection technique. These descriptors were then matched between images for detecting NDs [26].

A laterally coupled pixel-based neural network was outlined for detecting multiple forgeries in a single doctored image. The network had feeding and linking channels, which receive the input and detect multiple forgeries, respectively. The study exhibited that the developed network was accurate and efficient in detecting multiple forgeries [27]. An ND detection approach that homogenously distributed the KPs over the entire image region involving a seagull algorithm was outlined. This approach was exhibited to be more reliable and efficient for large image databases [28]. A deep net-based robust ND detection scheme was proposed. It used Gabor filter-based descriptors and adopted a log-likelihood testing technique for checking duplication [29]. A CNN-based scheme was developed for detecting NDs. The scheme exploited the ResNet-50 tool for enhancing the acumen process and improving computational efficiency [30].

Near-duplicate detection

The process of ND image detection involves two phases: (1) the training phase for building a visual vocabulary for a given large image database, and (2) the querying phase for finding NDs for a QI as presented in Figure 1.

Training Phase: During training, the preprocessing block removes the noises by the Median filter and converts the colour images in the database into greyscale for the entire image database. The KP detection block applies the Difference of Gaussian or Fast-Hessian Matrix Based Detector (FHMBD) or FAST Corner Detector (FCD) for getting the KPs, while the subsequent block employs SURF or SIFT for evaluating the descriptor vectors. The last block applies K-means clustering for indexing the features and forming a visual vocabulary containing a bag of VWs.

The bag-of-visual-words model is a standard approach for representing images as a group of image descriptors [31]. As present-day images are generally high-resolution images containing an enormous number of KPs, the matching of KPs of the QI with those of the images in the database may be computationally inefficient requiring long execution times. This

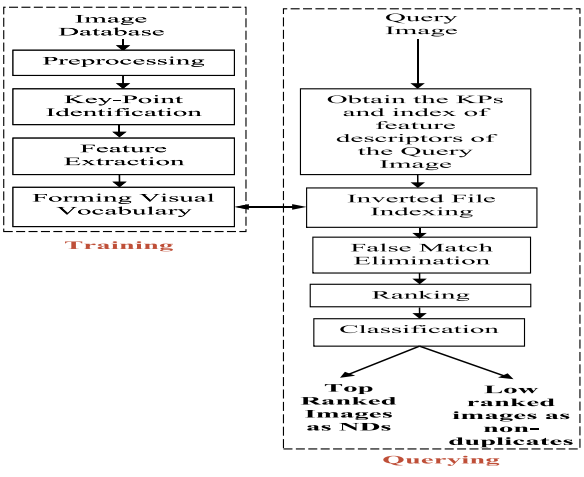


Figure 1. Schematic of ND detection.

issue can be overcome by generating a visual vocabulary for the entire image database through a clustering (K-means) technique [32]. Two KPs are said to be matched, if they belong to the same cluster or VW in the descriptor space. The VWs are also identified for the QI during the querying phase. Figure 2 illustrates how KPs representing tiny similar parts of different images can be linked to an equivalent VW. Figure 3 portrays the basics of clustering to form VWs and matching of QI. Figure 3(a) shows the KPs of the image database in feature space, Figure 3(b) exhibits the formation of VWs by clustering, and 3(c) displays the identification of VWs (descriptor) of QI.

An indexing table is then formed for the entire image database, which is analogous to 'index' in textbooks. The indexing table relates each VW with images having the same VW in the database. Figure 4(a) illustrates the formation of an inverted file indexing table for the entire image database. In each row, the first column denotes the VW number, and the second column lists the image numbers containing the corresponding VW.

Querying phase

During querying, the VWs of the QI are identified and searched in the inverted file index. That is, the VWs of the QI are directly searched in the indexed table to obtain the images that contain similar VWs. This indexing simplifies the querying process and avoids the comparison of descriptors of the entire image database, thereby searching faster than that of conventional matching [33]. Figure 4(b) shows how the VWs of the QI are directly searched in the indexed table to retrieve the images that contain those distinctive query words. As this procedure does not check the geometric layout of the matched VWs, a spatial verification step is typically performed both on the matched images and the QI using RANSAC to eliminate false matches [34].

At the end of querying, a histogram is formed based on the number of occurrences (matches) and then the images are ranked based on the number of occurrences. The top-ranked images are classified as NDs and the remaining images as non-duplicates. The classical ND detection methods (CNDDM) differ in the method of obtaining KPs, extracting features, obtaining visual vocabulary, and so on. In this article, two CNDDMs are used in the study for performance comparison, as given in Table 1.

Proposed SSO-based ND detection

The proposed SSO-based ND detection method (SSONDM) introduces (1) modification in KP identification, (2) PCA for Feature reduction, and (3) SSO-based clustering for forming visual vocabulary while following the conventional preprocessing, feature extraction and false-match elimination and classification, as given in Table 2.

KP identification

A few classical methods using a single KP detecting scheme are effective on images having smooth and plain regions (low gradients), and others are suitable for images having sharp corners and edges (high gradients). That is, they are not effective for images having both smooth and sharp edges. To overcome this issue, the proposed method adopts both FHMBD and FCD for detecting the KPs at both plain and non-smooth regions of all images in the image database and the QI as well. Among the identified KPs, half of the strongest KPs of each technique are used for evaluating the SURF descriptors. The descriptor at j-th KP is represented by a vector containing 64 values.

$$D_j = [D_{j1},\ D_{j2},\ \cdots\cdots,\ D_{j64}] \tag{1}$$

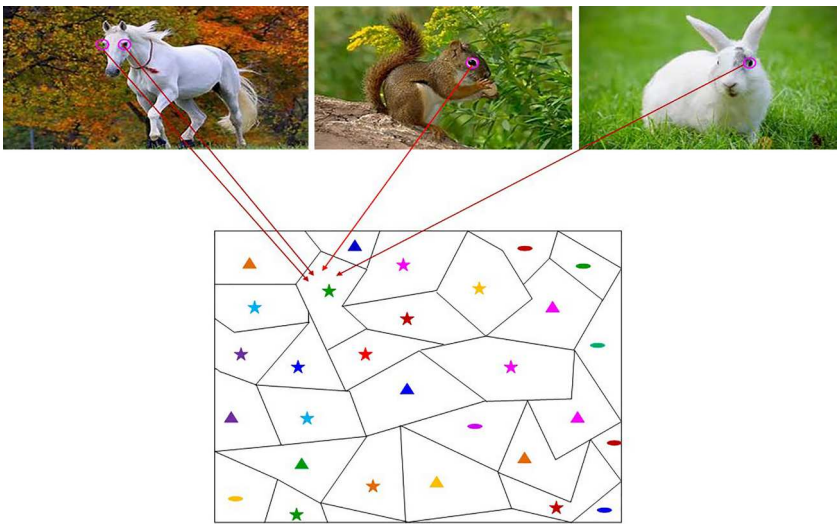


Figure 2. Concept of VWs.

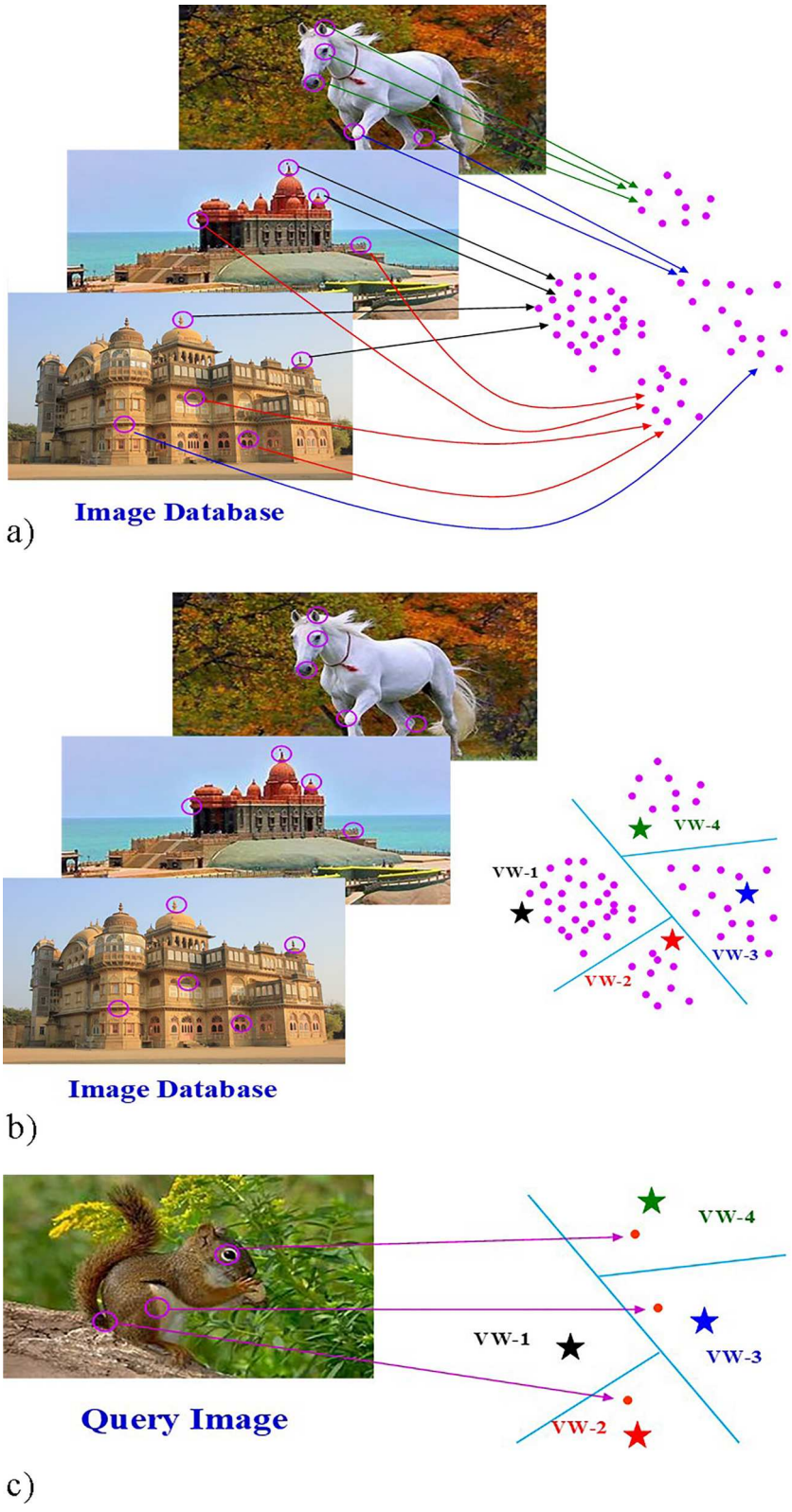
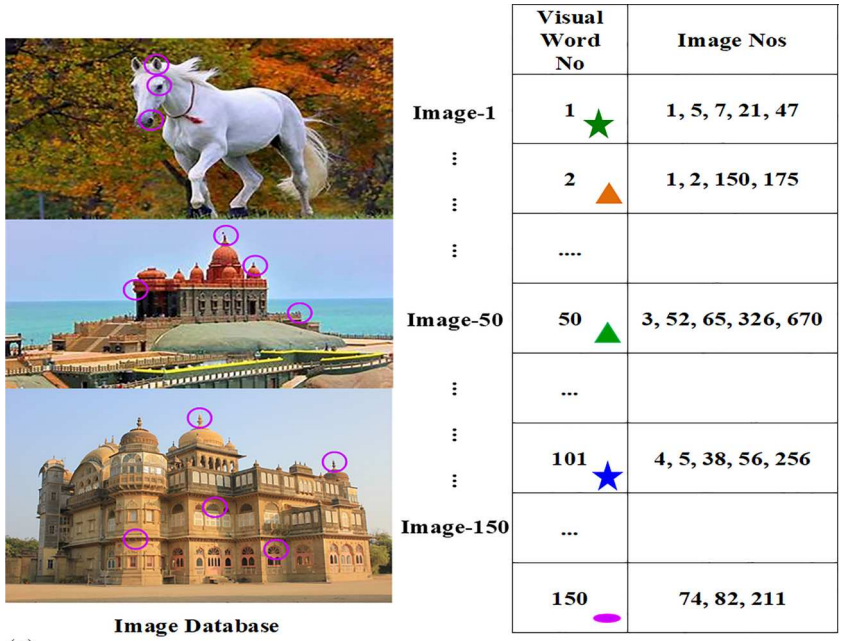


Figure 3. Construction of visual vocabulary and assignment of VW. (a) Pink circled KPs mapped in feature space; (b) Clustering of features to form visual vocabulary and (c) Identification of VWs (descriptor) of QI.

Feature reduction

The SURF descriptor, containing 64 values at each KP, makes the ND detection process uneconomical and inefficient, especially for large image databases. The PCA, a quantitatively rigorous method, employing orthogonal transformations to transform a set of correlated variables into a set of uncorrelated variables, is adopted for lowering the length of each SURF descriptor [35]. It can conveniently handle square matrices for reducing the dimension of the SURF-descriptor. So, the SURF-descriptor of Equation (1) is rearranged as a (8×8) matrix.



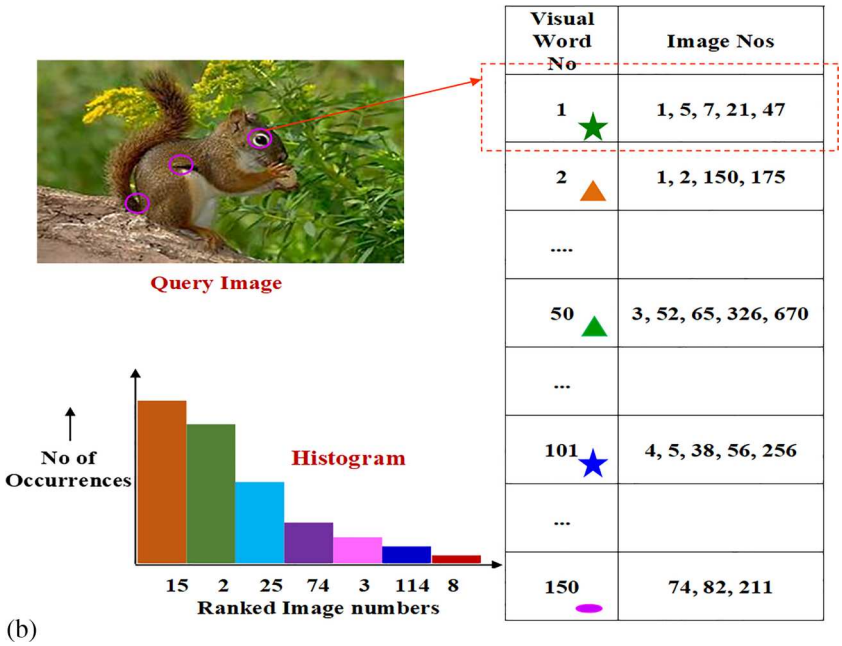


Figure 4. Principle of Inverted File Indexing. (a) Formation of inverted file index for entire image database; (b) Searching of VWs and formation of histogram.

Table 1. Composition of classical ND detection methods.

Modules	Technique	CNDDM-1	CNDDM-2
Preprocessing	Median Filter & RGB to Greyscale Conversion	✓	✓
Identification KPs	FHMBD	✓	
	FCD		✓
Evaluation of Feature Descriptors	SURF	✓	✓
Forming Visual Vocabulary	K-means Clustering	✓	✓
Elimination of False Matches	RANSAC	✓	✓
Classification	Considering the number of true matches.	✓	✓

$$D_j = \begin{bmatrix} D_{j1} & D_{j9} & D_{j17} & D_{j25} & D_{j33} & D_{j41} & D_{j49} & D_{j57} \\ D_{j2} & D_{j10} & D_{j18} & D_{j26} & D_{j34} & D_{j42} & D_{j50} & D_{j58} \\ D_{j3} & D_{j11} & D_{j19} & D_{j27} & D_{j35} & D_{j43} & D_{j51} & D_{j59} \\ D_{j4} & D_{j12} & D_{j20} & D_{j28} & D_{j36} & D_{j44} & D_{j52} & D_{j60} \\ D_{j5} & D_{j13} & D_{j21} & D_{j29} & D_{j37} & D_{j45} & D_{j53} & D_{j61} \\ D_{j6} & D_{j14} & D_{j22} & D_{j30} & D_{j38} & D_{j46} & D_{j54} & D_{j62} \\ D_{j7} & D_{j15} & D_{j23} & D_{j31} & D_{j39} & D_{j47} & D_{j55} & D_{j63} \\ D_{j8} & D_{j16} & D_{j24} & D_{j32} & D_{j40} & D_{j48} & D_{j56} & D_{j64} \end{bmatrix} \quad (2)$$

The PCA performs orthogonal transformations for the above square matrix and yields a transformed matrix with the same dimensions as in Equation (3).

Table 2. Composition of the proposed method.

Modules	Technique	SSONDM	Role / Operation
Preprocessing	Median Filter & RGB to Greyscale Conversion	✓	Performs preprocessing for all the images in the entire image database
KPs	FHMBD	✓	Adopts both techniques and selects 50% of the strongest KPs of each technique.
	FCD	✓	
Descriptors	SURF	✓	Evaluates feature descriptors at selected KPs using SURF. The dimension of each descriptor is 64.
Feature Reduction	PCA	✓	Applies PCA for reducing the dimension of each feature descriptor from 64 to 16.
Clustering	SSO	✓	Employs SSO for obtaining optimal VWs for the entire image database.
False Matches Elimination	RANSAC	✓	Checks the geometry of the matched KPs and eliminates the false matches.
Classification	Based on the number of true matches.	✓	The images are ranked based on the number of matched KPs, and are classed as duplicates or NDs if the number of matched KPs is greater than a threshold value

$$D'_j = \begin{bmatrix} D'_{j1} & D'_{j9} & D'_{j17} & D'_{j25} & D'_{j33} & D'_{j41} & D'_{j49} & D'_{j57} \\ D'_{j2} & D'_{j10} & D'_{j18} & D'_{j26} & D'_{j34} & D'_{j42} & D'_{j50} & D'_{j58} \\ D'_{j3} & D'_{j11} & D'_{j19} & D'_{j27} & D'_{j35} & D'_{j43} & D'_{j51} & D'_{j59} \\ D'_{j4} & D'_{j12} & D'_{j20} & D'_{j28} & D'_{j36} & D'_{j44} & D'_{j52} & D'_{j60} \\ D'_{j5} & D'_{j13} & D'_{j21} & D'_{j29} & D'_{j37} & D'_{j45} & D'_{j53} & D'_{j61} \\ D'_{j6} & D'_{j14} & D'_{j22} & D'_{j30} & D'_{j38} & D'_{j46} & D'_{j54} & D'_{j62} \\ D'_{j7} & D'_{j15} & D'_{j23} & D'_{j31} & D'_{j39} & D'_{j47} & D'_{j55} & D'_{j63} \\ D'_{j8} & D'_{j16} & D'_{j24} & D'_{j32} & D'_{j40} & D'_{j48} & D'_{j56} & D'_{j64} \end{bmatrix} \tag{3}$$

The first column indicates the first principal component that will span the direction of the dataset's greatest variation, the next column denotes the second principal component, and so on. The first few columns, may be one or two, are rearranged into a vector representing reduced features of the original SURF descriptor. For example, selecting the first two columns, the reduced feature descriptor with 16 values can be rearranged for j-th KP as follows:

$$D_j^R = \begin{bmatrix} D'_{j1} & D'_{j9} \\ D'_{j2} & D'_{j10} \\ D'_{j3} & D'_{j11} \\ D'_{j4} & D'_{j12} \\ D'_{j5} & D'_{j13} \\ D'_{j6} & D'_{j14} \\ D'_{j7} & D'_{j15} \\ D'_{j8} & D'_{j16} \end{bmatrix} \tag{4}$$

$$D_j^R = \begin{bmatrix} D'_{j1} & D'_{j2} & \cdots\cdots & D'_{j16} \end{bmatrix} \tag{5}$$

SSO-based clustering

Each reduced feature vector D_j^R is represented by a vector $(1 \times nv)$ as in Equation (5). All the reduced feature vectors of the whole image database are concatenated into a matrix $[D^R]$ of size $(nf \times nv)$. Here, nv & nf denote the length of the reduced descriptor and several reduced descriptor vectors for the whole image database, respectively. The reduced feature vectors of the entire image database are grouped into K-number of VWs $[w_1, w_2, \cdots\cdots, w_K]$ using a K-means clustering scheme [36,37], which minimizes Equation (6). The dimension of each VW is the same as that of the reduced feature descriptor $(1 \times nv)$.

$$Minimize \quad \Omega = \sum_{k=1}^{K} \sum_{j=1}^{nf} ||D_j^R - w_k||^2 \tag{6}$$

where

$$||D_j^R - w_k|| = \sqrt{\sum_{i=1}^{nv} (D_{j,i}^R - w_{k,i})^2} \tag{7}$$

As the K-means technique begins to search from a single random starting solution, and its nature of correcting the solution using gradients towards the nearby solution in the problem space, the technique may result in sub-optimal VWs. The proposed methodology, therefore, employs SSO for generating optimal VWs. The SSO is a population-based algorithm, which performs a search in the problem space with multiple starting solution points, and yields the global best solution [38]. This SSO is modelled from the clever foraging and gliding skills of flying squirrels, and requires the following processes for solving the clustering problem:

Initialization

The forest contains m-flying squirrels, each of which is modelled to represent the solution vector as follows:

$$\begin{aligned} SQ &= [SQ_1, SQ_2, \cdots\cdots, SQ_m] \\ &= [(w_{11}, w_{12}, \cdots, w_{1nv}), (w_{21}, w_{22}, \cdots, w_{2nv}), \\ &\cdots (w_{K1}, w_{K2}, \cdots, w_{Knv})] \end{aligned} \tag{8}$$

The initial position of i-th squirrel is randomly generated using Equation (9)

$$\begin{aligned} SQ_p(0) &= SQ_{low} + rand(0, 1) \times (SQ_{up} - SQ_{low}); \\ p &= 1, 2, \cdots, m \end{aligned} \tag{9}$$

Strength evaluation

The optimization process uses the squirrels' clever dynamic foraging behaviour of utilizing the available food resources for eating or storing depending on their availability. The hickory trees and acorn trees are considered to be the best and average respectively, while the normal trees have no food sources. The foraging behaviour of each squirrel in utilizing the food

sources can be estimated by the following strength function (S), formed from Equation (6), for each candidate solution.

$$Maximize \quad S = \frac{1}{1 + \sum_{k=1}^{K} \sum_{j=1}^{nf} ||D_j^R - w_k||^2} \tag{10}$$

Ranking

The strength values of all squirrels are sorted in descending order. The top-ranked squirrel is sitting on the hickory nut tree, and the subsequent three top-ranked squirrels occupy on acorn nuts trees, while the remaining squirrels sit on the ordinary trees. The three squirrels on the acorn nuts tree attempt to glide to the hickory nut tree. A few squirrels are deemed to have met their food requirements and are expected to move to the hickory nut tree. The other squirrels will seek out acorn nut trees to get their food requirements. However, the presence of predators has an impact on flying squirrels' foraging actions.

Locomotion

Three scenarios may arise during the foraging of squirrels. In each scenario, it is believed that the squirrel glides and explores effectively for its preferred food source when there is no predator. Nevertheless, the squirrel makes a small random move to find a hiding spot in the neighbourhood, if a predator is seen around. The foraging feature of squirrels is modelled under three scenarios:

Scenario 1: The squirrels on acorn nut trees may attempt to glide to the hickory nut tree by Equation (11).

$$SQ_{at}^{k+1} = \begin{cases} SQ_{at}^{k} + G_d \times G_c \times (SQ_{ht}^{k} - SQ_{at}^{k}) & rand(0, 1) \geq P_{prob} \\ Random\ Location & Otherwise \end{cases} \tag{11}$$

where, *rand*(0, 1) is a random number.

Scenario 2: The squirrels on normal trees attempt to glide to acorn nut trees by Equation (12).

$$SQ_{nt}^{k+1} = \begin{cases} SQ_{nt}^{k} + G_d \times G_c \times (SQ_{at}^{k} - SQ_{nt}^{k}) & rand(0, 1) \geq P_{prob} \\ Random\ Location & Otherwise \end{cases} \tag{12}$$

Scenario 3: The squirrels that are on normal trees possessing acorn nuts may glide to hickory nut tree with a view of storing hickory nuts to face the food shortage by Equation (13).

$$SQ_{nt}^{k+1} = \begin{cases} SQ_{nt}^{k} + G_d \times G_c \times (SQ_{ht}^{k} - SQ_{nt}^{k}) & rand(0, 1) \geq P_{prob} \\ Random\ Location & Otherwise \end{cases} \tag{13}$$

The gliding locomotion of squirrels by changing the lift and drag forces during the night is pictorially exhibited in Figure 5.

Aerodynamic gliding

The squirrel's gliding process can be modelled by equipoise glide in which the drag and lift forces form a combined force, which is just opposite to the direction of the gliding mass of the squirrel. The squirrel glides at an angle θ relating the lift and drag forces at constant speed as in Equation (14).

$$\frac{F_L}{F_D} = \frac{1}{\tan\theta} \tag{14}$$

The length of the squirrel's gliding route can be increased by decreasing the gliding angle (θ). The F_L to F_D ratio is likewise raised when gliding-path length increases. The F_D and the F_L created by downward deflections of air passing over the wings can be written by Equations (15) and (16), respectively.

$$F_D = \frac{1}{2\mu\ K_D\ S_p^2\ A_s} \tag{15}$$

$$F_L = \frac{1}{2\mu\ K_L\ S_p^2\ A_s} \tag{16}$$

At slow speeds, the drag element is prominently large, while at higher speeds, the drag element is lesser. The

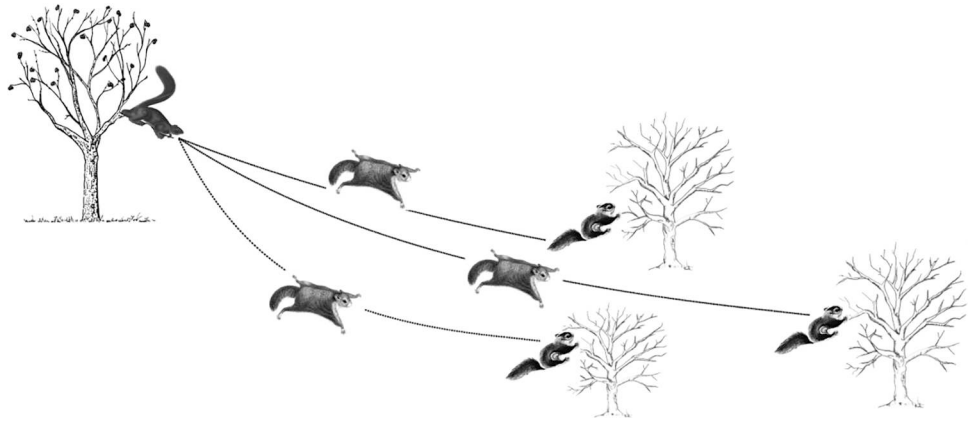


Figure 5. Gliding locomotion of Squirrels.

gliding element can be modelled at a steady state by Equation (17):

$$\theta = \arctan\left(\frac{F_D}{F_L}\right) \tag{17}$$

The distance ($Dist_{SQ}$) glided by the squirrel can be estimated by Equation (18).

$$Dist_{SQ} = \left(\frac{HL_{SQ}}{\tan\theta}\right) \tag{18}$$

A squirrel can vary the $Dist_{SQ}$ by altering the F_L to F_D ratio to reach the desired target location.

Seasonal changes

Squirrel's foraging behaviour is strongly affected by seasonal fluctuations. They lose a lot of energy to keep their body warm during winter, and their foraging activity is dangerous due to predators. They do not actively move during winter to keep themselves away from predators and keep themselves warm. The climate conditions therefore affect the behaviour of squirrels, which is modelled by three steps:

Step 1: Compute the climate constant (CC^k) using Equation (19).

$$CC^k = \sqrt{\sum_{j=1}^{3} (SQ_{at,j}^k - SQ_{ht}^k)^2} \tag{19}$$

Step 2: Examine the climate monitoring situation. That is, check whether $CC^k < CM_{\min}$ is satisfied. Here $CM_{\min}$ is the minimum value of the climate constant which is computed by

$$CM_{\min} = \frac{10e^{-6}}{365^{k/(MNI/2.5)}} \tag{20}$$

Step 3: The satisfaction of the climate monitoring indicates that the winter season is completed, and requires random relocation of squirrels for better exploration of the forest. So randomly relocate the squirrels.

After-winter relocation

After winter, the squirrels will become active. The squirrels that could not find hickory nuts trees will glide in several directions for finding a better food source. Such relocation can be represented by Equation (21).

$$SQ_{nt}^{new} = SQ_{low} + LF(n) \times (SQ_{up} - SQ_{low}) \tag{21}$$

The Levy Flight ($LF(n)$) is used for calculating the distance and is mathematically modelled as follows:

$$LF(n) = 0.01 \times \frac{r_a \times \xi}{|r_b|^{1/\beta}} \tag{22}$$

where ξ is computed by Equations (23) and (24).

$$\xi = \left[\frac{\Gamma(1+\beta) \times \sin\left(\frac{\pi\beta}{z}\right)}{\Gamma\left(\frac{1+\beta}{z} \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}\right)}\right]^{\frac{1}{\beta}} \tag{23}$$

$$\Gamma(z) = (z-1)! \tag{24}$$

The proposed SSO-based clustering commences with a population of squirrels that is randomly generated. During the solution process, the squirrels sit on the Hickory, Acorn and Normal trees depending on their strength. Considering the three scenarios, the squirrels fly to new positions using Equations (11)–(13). They are also relocated by Equation (21) depending on the climate condition. This process is repeated till convergence. The squirrel on the Hickory nut tree represents the optimal VWs. The steps of the SSO's

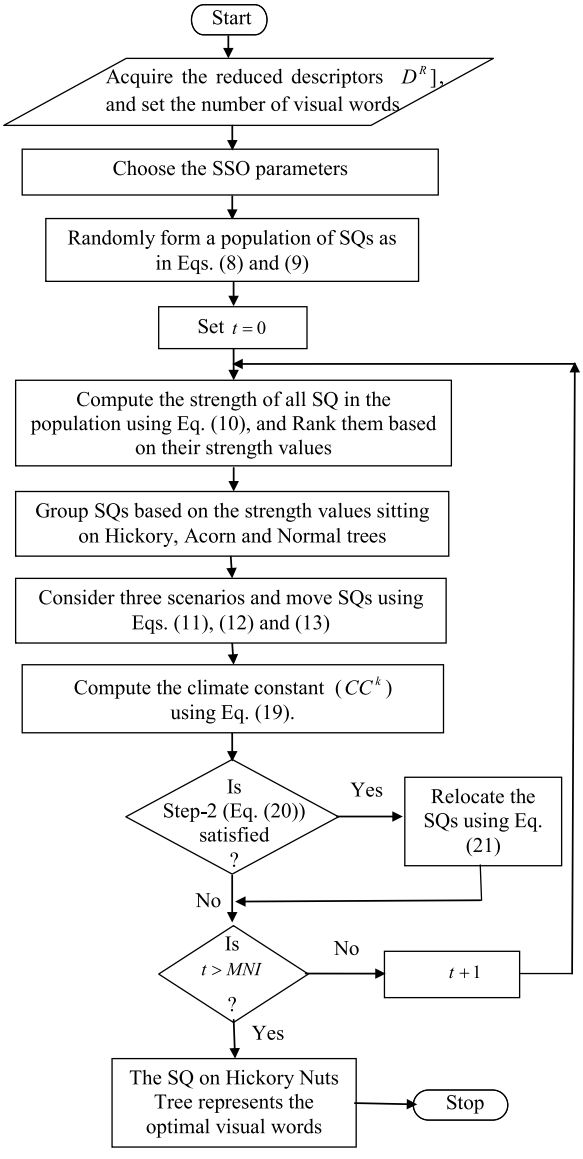


Figure 6. Flowchart of SSO.

iterative process are outlined in Figure 6. The methodology also employs the traditional modules of inverted file index, wrong match removal using RANSAC, identification of NDs based on match counts (ranked histogram) and classification. The top-scored images are classified as NDs.

Results and discussions

Test Data: 4970 digital RGB images were downloaded from several web sources. Among them, six images were selected and five NDs were fabricated by changing colour and intensity, cropping and rotation for each of the chosen six images, and added to the database. The database, therefore, contained 5000 images. The whole database is duplicated to form another four different-sized databases involving 10000, 15000, 20000 and 25000 digital images.

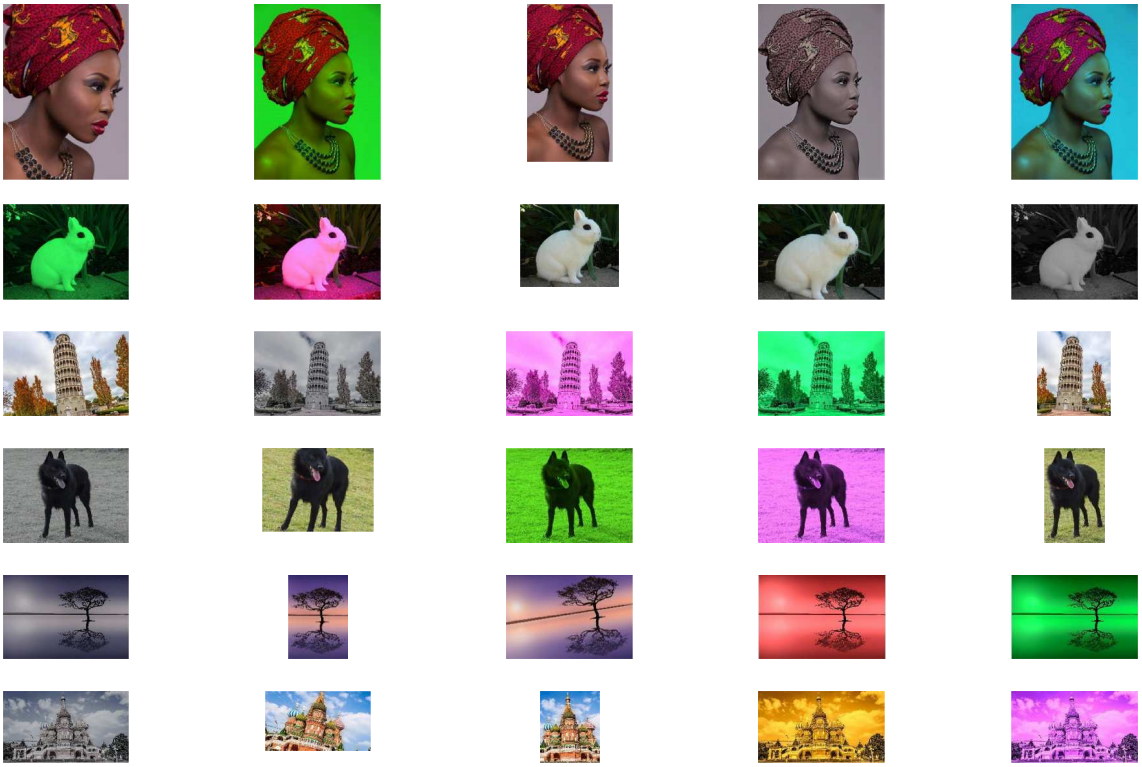
Performance Study: The SSONDM was tested on the above five image databases. In addition, for comparing the performances of the SSONDM with classical methods, two classical methods with different combinations of modules, as listed in Table 1, were developed.

Tables 3 and 4 present the six different-sized QIs (QI-1 to QI-6) and their NDs, respectively. The identified KPs by FHMBD and FCD modules were marked on the QIs and presented in Table 5. It is observed from the table that the FCD module found a greater number of KPs than that of FHMBD. The proposed SSONDM selected only 50% of the strongest KPs from each of the FHMBD and FCD modules and combined them into a set of KPs. It then evaluated the SURF descriptors at all these KPs. Such blending of KPs would cover both the low and high gradient parts of the images. During querying, the SSONDM compared the KPs of QI with all the KPs of all images in the database. The false matches were eliminated by the RANSAC module. The images were then ranked considering the count of matched pairs. The high-ranked images were taken as NDs. The matchings of the top-ranked images were marked by yellow lines between the matched KPs of QI-1 and its NDs before and after RANSAC are exhibited in

Table 3. Query images.

No	Image	No	Image
QI-1	 Size: (663*474*3)	QI-4	 Size: (598*800*3)
QI-2	 Size: (552*736*3)	QI-5	 Size: (640*960*3)
QI-3	 Size: (650*975*3)	QI-6	 Size: (720*1280*3)

Table 4. NDs in the image database.

ND1	ND2	ND3	ND4	ND5
				

Tables 6 and 7, respectively in addition to mentioning the number of matched pairs. It can be observed from the number of matched pairs that the RANSAC eliminated the wrong matches.

All the three methods successfully detected the ND images for QIs-1, -2 and -6. But both CNDDM-1 and CNDDM-2 failed to detect an ND for QI-3 and QI-5, and CNDDM-1 failed to detect an ND for QI-4, while the proposed SSONDM was able to successfully detect all the NDs for all the six QIs. The CNDDMs were unsuccessful in detecting a few of the rotated and cropped NDs. The success and failure of all three methods are marked with ✓ and ×, respectively in Table 8. The true and false classifications by the CNDDM-1, CNDDM-2 and SSONDM for the entire image database are denoted by

- true original (TO), the original image rightly categorized as original;
- true duplicate (TD), the ND image rightly detected as ND;
- false duplicate (FD), the authenticated image falsely identified as ND;
- false original (FO), the ND image falsely categorized as authentic.

The confusion matrices relating the actual and estimated classes for all five image databases are presented in Tables 9–13. In confusion matrices, the diagonal values, representing TO and TD, must equal the actual class numbers, while the off-diagonal values, denoting FD and FO, must be ideally zero. These off-diagonal values must be as small as possible. Comparing the off-diagonal values of the proposed SSONDM with CNDDM-1 and CNDDM-2 for all the five image databases, it is very clear that they are smaller than those of the CNDDM-1 and CNDDM-2, thereby portraying the better performances of the proposed SSONDM.

Five metrics given by Equations (19)-(23) were computed from the confusion matrices for all three methods for all the five image databases. Several performance values of the three methods for all the five databases may not convey anything precisely. Moreover, it was observed that there was a small variation in the performance values for different image databases, and therefore the average of these values was computed and presented in Figure 7. Comparing the performance values given in the figure, it is very clear that the SSONDM offered better performances than those of CNDDM-1 and CNDDM-2.

$$\text{Accuracy} = \frac{(TD + TO)}{(TD + TO + FD + FO)} \tag{25}$$

Table 5. QIs with identified KPs.

No	Image with FHMBD KPs	No. of KPs	Image with FCD KPs	No. of KPs
QI-1		442		675
QI-2		209		164
QI-3		2457		5549
QI-4		689		1100
QI-5		690		1685
QI-6		3763		8274

$$\text{Sensitivity} = \frac{TD}{(TD + FO)} \tag{26}$$

$$\text{Specificity} = \frac{TO}{(TO + FD)} \tag{27}$$

$$\text{Precision} = \frac{TD}{(TD + FD)} \tag{28}$$

$$F_1 = \frac{2 \times \text{Precision} \times \text{Sensitivity}}{\text{Precision} + \text{Sensitivity}} \tag{29}$$

The training time for forming the Bag of VWs by CNDDM-1, CNDDM-2 and SSONDM for all the five image databases and the querying time for detecting the NDs were measured, and presented in Tables 14 and 15, respectively. The training times were much larger compared to querying times as the methods had to process all the images in the entire database. It is seen from the tables that both the training and the querying times of the proposed SSONDM are lower than those of CNDDM-1 and CNDDM-2, thereby portraying the computational efficiency of the developed SSONDM.

Table 6. ND detection of SSONDM for QI-1 before RANSAC.

Querry and ND images with matched KPs	Querry and ND images with matched KPs	
		
No of matches: 463	No of matches: 422	No of matches: 18
		
No of matches: 333	No of matches: 243	

Table 7. Duplicate Detection of SSONDM for QI-1 after RANSAC.

Querry and ND images with matched KPs	Querry and ND images with matched KPs	No. of matched KPs
		
No of matches: 462	No of matches: 419	No of matches: 16
		
No of matches: 331	No of Matches: 241	

Conclusion

An efficient and robust ND image detection method was suggested to overcome the drawbacks of the existing classical methods. The method used FHMBD and FCD for identifying the most significant KPs both at low- and high-gradient regions. The PCA reduced the dimension of the SURF descriptor and lowered the subsequent computations during training. The SSO attempted to find the global best VWs than that of the existing K-means algorithm. The

Table 8. Success (✓) and Failure (×) in finding NDs.

		ND1	ND2	ND3	ND4	ND5
QI-1	CNDDM-1	✓	✓	✓	✓	✓
	CNDDM-2	✓	✓	✓	✓	✓
	SSONDM	✓	✓	✓	✓	✓
QI-2	CNDDM-1	✓	✓	✓	✓	✓
	CNDDM-2	✓	✓	✓	✓	✓
	SSONDM	✓	✓	✓	✓	✓
QI-3	CNDDM-1	×	✓	✓	✓	✓
	CNDDM-2	×	✓	✓	✓	✓
	SSONDM	✓	✓	✓	✓	✓
QI-4	CNDDM-1	✓	×	✓	✓	✓
	CNDDM-2	✓	✓	✓	✓	✓
	SSONDM	✓	✓	✓	✓	✓
QI-5	CNDDM-1	✓	✓	×	✓	✓
	CNDDM-2	✓	✓	×	✓	✓
	SSONDM	✓	✓	✓	✓	✓
QI-6	CNDDM-1	✓	✓	✓	✓	✓
	CNDDM-2	✓	✓	✓	✓	✓
	SSONDM	✓	✓	✓	✓	✓

method was tested on 5 image databases and found to offer better accuracy, sensitivity, specificity, precision and F1 values of 99.9, 98.67, 99.91, 87.11 and 92.52, respectively than those of classical methods. Moreover, the training time of the proposed SSSO method for the database with 25,000 images was 1485 s, which is lower than those of the CNDDM-1 and CNDDM-2 taking 1597 and 1830, s respectively. A similar reduction in the querying time of the proposed method was also realized. The improvement in the detection process may be studied with artificial neural networks, deep learning, neuro-fuzzy systems, other KP detection schemes, other hybrid descriptors, and evolutionary algorithms as future work.

Table 9. Confusion Matrix for the database containing 5000 images.

			Estimated Class	
			Original (4970)	ND (30)
Actual Class	CNDDM-1	Original (4970)	4965	5
		ND (30)	3	27
	CNDDM-2	Original (4970)	4957	13
		ND (30)	2	28
	SSONDM	Original (4970)	4967	3
		ND (30)	0	30

Table 10. Confusion Matrix for the database containing 10,000 images.

			Estimated Class	
			Original (9940)	ND (60)
Actual Class	CNDDM-1	Original (9940)	9924	16
		ND (60)	8	52
	CNDDM-2	Original (9940)	9906	34
		ND (60)	4	56
	SSONDM	Original (9940)	9932	8
		ND (60)	0	60

Table 11. Confusion Matrix for the database containing 15,000 images.

			Estimated Class	
			Original (14910)	ND (90)
Actual Class	CNDDM-1	Original (14910)	14883	27
		ND (90)	12	78
	CNDDM-2	Original (14910)	14853	57
		ND (90)	9	81
	SSONDM	Original (14910)	14898	12
		ND (90)	0	90

Table 12. Confusion Matrix for the database containing 20,000 images.

			Estimated Class	
			Original (19880)	ND (120)
Actual Class	CNDDM-1	Original (19880)	19840	40
		ND (120)	16	104
	CNDDM-2	Original (19880)	19800	80
		ND (120)	12	108
	SSONDM	Original (19880)	19860	20
		ND (120)	4	116

Table 13. Confusion Matrix for the database containing 25,000 images.

			Estimated Class	
			Original (24850)	ND (150)
Actual Class	CNDDM-1	Original (24850)	24790	60
		ND (150)	20	130
	CNDDM-2	Original (24850)	24745	105
		ND (150)	15	135
	SSONDM	Original (24850)	24820	30
		ND (150)	5	145

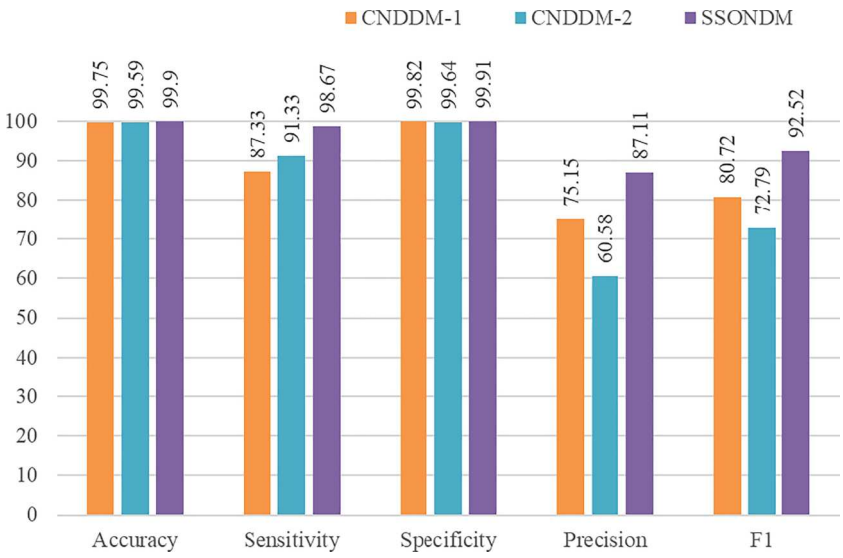


Figure 7. Comparison of average performance metrics.

Table 14. Training Time for Creating Bag of VWs (seconds).

Database	CNDDM-1	CNDDM-2	SSONDM
5000 Images	308	328	264
10,000 Images	559	655	538
15,000 Images	829	948	714
20,000 Images	1113	1293	1072
25,000 Images	1597	1830	1485

Table 15. Querying Time (seconds).

Database	CNDDM-1	CNDDM-2	SSONDM
5000 Images	2.0	2.3	1.8
10,000 Images	2.5	2.7	2.2
15,000 Images	3.2	2.9	2.5
20,000 Images	3.5	3.1	2.9
25,000 Images	4.0	3.5	3.2

Acknowledgements

The authors express their gratitude to the Annamalai University administrative authorities for providing the computer and internet resources needed to complete this work.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Notes on contributors

Srinidhi Sundaram is a Research Scholar, Department of Information Technology, Faculty of Engineering and Technology, Annamalai University, Annamalainagar-608002, Tamil Nadu, India. received the B.Sc Degree in Computer Science from Pandit Ravishankar Shukla University, M.C.A from Madurai Kamaraj University and M. Tech Degree in Information Technology from Sathya bama University, in 1999, 2003 and 2010 respectively. She is presently pursuing part-time Ph.D (Information Technology) at Annamalai University. She has 15 years of teaching experience and is specialized in the area of image forensics, soft computing and metaheuristic optimization.

Dr. Kamalakkannan Somasundaram is received his Ph.D. Degree in Computer Science from Vels Institute of Science, Technology & Advanced Studies (VISTAS) Tamil Nadu, India. He is currently working as Associate Professor, Department of Information Technology, School of Computing Sciences, Vels Institute of Science, Technology & Advanced Studies (VISTAS), Chennai, Tamil Nadu, India, which is a well-known university. He has 18 years of teaching experience in both UG and PG level. His research interest includes Data Mining, Big Data Analytics, Cloud Computing and Block Chain Technology. He has produced one Ph.D Research scholar. He has published more than 50 research articles in various international journals such as Sci, WOS, Scopus and UGC referred journals. He serves as an Examiner in various Universities and Colleges. He received Best Young Scientist award and Best Faculty award.

Dr. Sasikala Jayaraman is received the B.E. Degree in Electronics and Communication Engineering from Madras University, India in 1993, and the M.E and Ph. D degrees in Computer Science and Engineering from Annamalai University in 2005 and 2011 respectively. She has been working as an Associate Professor in the Department of Information Technology, Annamalai University, Tamil Nadu, India since 1999. Her research interests are in the area of optimization, evolutionary algorithms and image processing.

References

[1] Thaiyalnayaki S, Sasikala J, Ponraj R. Detecting near-duplicate images using segmented minhash algorithm. Int J Adv Intell Parad. 2019;12(1/2):192–206. doi:10.1504/IJAIP.2019.096963

[2] Thyagharajan KK, Kalaiarasi G. A review on near-duplicate detection of images using computer vision techniques. Arch Comput Methods Eng. 2021;28:897–916. doi:10.1007/s11831-020-09400-w

[3] Madhawa S., Balakrishnan P., Arumugam U. Employing invariants for anomaly detection in software defined networking based industrial internet of things. Journal of Intelligent and Fuzzy Systems. 2018;35(2):1267–1279. doi:10.3233/JIFS-169670

[4] Balakrishnan P, Somasundaram T. SLA enabled CARE resource broker. Future Generation Computer Systems. 2011;27:265–279. doi:10.1016/j.future.2010.09.006

[5] Madhawa S, Balakrishnan P, Arumugam U. Roll forward validation based decision tree classification for

detecting data integrity attacks in industrial internet of things. Journal of Intelligent and Fuzzy Systems. 2019;36(3):2355–2366. doi:10.3233/JIFS-169946

[6] Ezhilarasie R, Umamakeswari A, Reddy MS. Grefenstette Bias based genetic algorithm for multi-site offloading using docker container in edge computing. Journal of Intelligent and Fuzzy Systems. 2019;36(3):2419–2429. doi:10.3233/JIFS-169953

[7] Jain M, Singh V, Rani A. A novel nature-inspired algorithm for optimization: Squirrel Search Algorithm. Swarm Evol Comput. 2019;44:148–175. doi:10.1016/j.swevo.2018.02.013

[8] Jiang H, Yang Y, Ping W, et al. A novel hybrid classification method based on the opposition-based seagull optimization algorithm. IEEE Access. 2020;8:100778–90. doi:10.1109/ACCESS.2020.2997791

[9] Mani R, Jayaraman S, Ellappan M. Hybrid seagull and thermal exchange optimization algorithm-based NLOS nodes detection technique for enhancing reliability under data dissemination in VANETs. Int J Commun Syst. 2020;33(14):e4519. doi:10.1002/dac.4519

[10] Ghosh P, Gelasca ED, Ramakrishnan KR, et al. Duplicate image detection in large scale databases. Adv Intell Inf Process: Tools Appl 2008;2008:149–166.

[11] Chu WT, Lin CH. Consumer photo management and browsing facilitated by near-duplicate detection with feature filtering. J Vis Commun Image Represent. 2010;21(3):256–268. doi:10.1016/j.jvcir.2010.01.006

[12] Min HS, Choi JY, De Neve W, et al. Bimodal fusion of low-level visual features and high-level semantic features for near-duplicate video clip detection. Signal Process, Image Commun. 2011;26(10):612–627. doi:10.1016/j.image.2011.04.001

[13] Shao H, Yu T, Xu M, et al. Image region duplication detection based on circular window expansion and phase correlation. Forensic Sci Int 2012;222(1-3):71–82. doi:10.1016/j.forsciint.2012.05.002

[14] Belkhatir M, Tahayna B. Near-duplicate video detection featuring coupled temporal and perceptual visual structures and logical inference-based matching. Inf Process Manag. 2012;48(3):489–501. doi:10.1016/j.ipm.2011.03.003

[15] Sekeh MA, Maarof MA, Rohani MF, et al. Efficient image duplicated region detection model using sequential block clustering. Digital Investigation. 2013;10(1):73–84. doi:10.1016/j.diin.2013.02.007

[16] Lei Y, Zheng L, Huang J. Geometric invariant features in the Radon transform domain for near-duplicate image detection. Pattern Recognit. 2014;47(11):3630–3640. doi:10.1016/j.patcog.2014.05.009

[17] Nian F, Li T, Wu X, et al. Efficient near-duplicate image detection with a local-based binary representation. Multimed Tools Appl. 2016;75:2435–2452. doi:10.1007/s11042-015-2472-1

[18] Chen M, Li Y, Zhang Z, et al. Real-time, large-scale duplicate image detection method based on multi-feature fusion. J Real-Time Image Process. 2017;13:557–570. doi:10.1007/s11554-016-0632-9

[19] Hu W, Fan Y, Xing J, et al. Deep constrained siamese hash coding network and load-balanced locality-sensitive hashing for near duplicate image detection. IEEE Trans Image Process. 2018;27(9):4452–4464. doi:10.1109/TIP.2018.2839886

[20] Thyagharajan KK, Kalaiarasi G. Pulse coupled neural network based near-duplicate detection of images (PCNN - NDD). Adv Electric Comput Eng. 2018;18(3):87–96. doi:10.4316/AECE.2018.03012

[21] Hosny KM, Hamza HM, Lashin NA. Copy-move forgery detection of duplicated objects using accurate PCET moments and morphological operators. Imag Sci J. 2018;66(6):330–345. doi:10.1080/13682199.2018.1461345

[22] Sharma S, Ghanekar U. Dominating direction based an efficient copy–move image tampering detection technique. Imag Sci J. 2018;66(4):254–262. doi:10.1080/13682199.2017.1420021

[23] Rajkumar R, Roy S, Manglem Singh K. A robust and forensic transform for copy-move digital image forgery detection based on dense depth block matching. Imag Sci J. 2019;67(6):343–357. doi:10.1080/13682199.2019.1663069

[24] Morra L, Lamberti F. Benchmarking unsupervised near-duplicate image detection. Expert Syst Appl. 2019;135:313–326. doi:10.1016/j.eswa.2019.05.002

[25] Zhou Z, Wu QJ, Wan S, et al. Integrating SIFT and CNN feature matching for partial-duplicate image detection. IEEE Trans Emerg Top Comput Intell. 2020;4(5):593–604. doi:10.1109/TETCI.2019.2909936

[26] Zhou Z, Lin K, Cao Y, et al. Near-duplicate image detection system using coarse-to-fine matching scheme based on global and local CNN features. Mathematics. 2020;8(4):644. doi:10.3390/math8040644

[27] Thyagharajan KK, Nirmala G. Image manipulation detection through laterally linked pixels and kernel algorithms. Comput Syst Sci Eng. 2022;41(1):357–371. doi:10.32604/csse.2022.020258

[28] Sundaram S, Kamalakkannan S, Jayaraman S. Seagull optimization-based near-duplicate image detection in large image databases. The Imaging Science Journal. 2023;71(7):647–659. doi:10.1080/13682199.2023.2190944

[29] Yang H, Park H. An integrated approach to near-duplicate image detection. *2023 international Conference on Artificial Intelligence in Information and Communication (ICAIIC)*; 2023; Bali, Indonesia, pp. 425-428, doi:10.1109/ICAIIC57133.2023.10067005

[30] Naveena P, Rajana N, Golusu BVK, et al. Revolutionizing image duplicate detection: harnessing ResNet-50 with spatial transformers for unprecedented precision. Int J Novel Res Develop. 2024;9(4):521–530.

[31] Bampis L, Gasteratos A. Revisiting the Bag-of-visual-words model: a hierarchical localization architecture for mobile systems. Rob Auton Syst. 2019;113:104–119. doi:10.1016/j.robot.2019.01.004

[32] Kalaiarasi G, Thyagharajan KK. Visual content based clustering of near duplicate Web search images. *International Conference on Green Computing, Communication and Conservation of Energy*; December 12-14, 2013. 769–773.

[33] Hu Y, Zhang Q, Zhang Q, et al. A novel hashing-inverted index for secure content-based retrieval with massive encrypted speeches. Multimedia Syst. 2024;30(22).

[34] Sabry ES, Elagooz S, Abd El-Samie EF, et al. Matching evaluation based on image content discriminative features for different image types. Imag Sci J. 2024;72(1):23–51. doi:10.1080/13682199.2023.2171555

[35] Gopinath P, Shivakumar R. Hybrid feature extraction and LLTSA-based dimension reduction for vein pattern recognition. Imag Sci J. 2023: 1–21.

[36] Ikotun AM, Ezugwu AE, Abualigah L, et al. K-means clustering algorithms: a comprehensive review, variants

analysis, and advances in the era of big data. Inf Sci. 2023;622:178–210. doi:10.1016/j.ins.2022.11.139

[37] Kalaiarasi G, Thyagharajan KK. Clustering of near duplicate images using bundled features'. Cluster Comput J. 2019;22(5):11997–12007. doi:10.1007/s10586-017-1539-3

[38] Malathi SR, Vijay Kumar P. A high-performance low complex design and implementation of QRS detector using modified MaMeMi filter optimized with mayfly optimization algorithm. J Circuits Syst Comput. 2023;32(04):2350056. doi:10.1142/S0218126623500561