



2D object recognition: a comparative analysis of SIFT, SURF and ORB feature descriptors

Monika Bansal¹ · Munish Kumar²  · Manish Kumar³

Received: 24 June 2020 / Revised: 24 October 2020 / Accepted: 4 February 2021 /
Published online: 20 February 2021

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC part of Springer Nature 2021

Abstract

Object recognition is a key research area in the field of image processing and computer vision, which recognizes the object in an image and provides a proper label. In the paper, three popular feature descriptor algorithms that are Scale Invariant Feature Transform (SIFT), Speeded Up Robust Feature (SURF) and Oriented Fast and Rotated BRIEF (ORB) are used for experimental work of an object recognition system. A comparison among these three descriptors is exhibited in the paper by determining them individually and with different combinations of these three methodologies. The amount of the features extracted using these feature extraction methods are further reduced using a feature selection (k-means clustering) and a dimensionality reduction method (Locality Preserving Projection). Various classifiers i.e. K-Nearest Neighbor, Naïve Bayes, Decision Tree, and Random Forest are used to classify objects based on their similarity. The focus of this article is to present a study of the performance comparison among these three feature extraction methods, particularly when their combination derives in recognizing the object more efficiently. In this paper, the authors have presented a comparative analysis view among various feature descriptors algorithms and classification models for 2D object recognition. The Caltech-101 public dataset is considered in this article for experimental work. The experiment reveals that a hybridization of SIFT, SURF and ORB method with Random Forest classification model accomplishes the best results as compared to other state-of-the-art work. The comparative analysis has been presented in terms of recognition accuracy, True Positive Rate (TPR), False Positive Rate (FPR), and Area Under Curve (AUC) parameters.

Keywords ORB · SIFT · SURF · K-means · LPP · PCA

✉ Munish Kumar
munishcse@gmail.com

1 Introduction

2D-Object Recognition System is getting more attention in the field of computer vision. It is widely used in various real life applications such as content based image retrieval, medical imaging, object detection, security surveillance system etc. It is a process that helps to identify the object in an image and associates the label to the object. It is very easy for a human to recognize the object in the real world without knowing its various parameters. He/She can easily identify an object despite of different viewpoints and differences. But, a machine need to accomplish this task by collecting a large number of images and storing labels of them. Here, it becomes desirable to use learning in order to map between the image (input) and the label of the object (output). The machine gets trained by extracting various features of the objects and storing them in memory. The overall efficiency of the object recognition system depends on the type of extracting features. The authors have experimented various feature extraction and classification algorithms to develop an efficient object recognition system. An object recognition system works in many phases, 1) Initially, a training dataset is prepared by extracting features from the image database by using various feature extraction and reduction algorithms. 2) Then this training dataset is classified into various categories by using image classification algorithms. Classification can be done by using a supervised or unsupervised learning algorithm. 3) In the last phase, the testing of data from the input images are matched with the stored dataset of the training model and proper class is assigned to the object recognized. The outcome of this recognition system entirely depends on the quality of various features extracted from the image database. So, feature extraction method plays a very crucial role as it improves the overall performance of an object recognition system. An Object recognition is a very cumbersome task for a machine as it requires more computational efforts to recognize the object from the input images. To reduce the computations, K-means clustering is used as a salient feature selection in the experiment. There are various feature extraction methods like SIFT (Lowe [22]), SURF (Bay et al. [4]), ORB (Rublee et al. [26]), FAST (Rosten et al. [25]), BRISK (Leutenegger et al. [20]), BRIEF (Calonder et al. [7]), etc., which helps to recognize an object.

Three main feature descriptor algorithms, i.e. SIFT, SURF, and ORB are considered during the comparative study. By using these three algorithms, feature vector has been generated that takes high memory space for storing the feature vector. So, k -means clustering algorithm and Locality Preserving Projection (LPP) are used to reduce the feature vector size. K-means clustering algorithm is used to cluster the calculated descriptors into k clusters and Locality Preserving Projection (LPP) is a reduction algorithm used to reduce the size of the feature vector. Then various classification algorithms, i.e. K-Nearest Neighbor (KNN), Naïve Bayes, Decision Tree, and Random forest are used to classify the extracted feature vector. The authors have implemented their experiments on a public dataset, namely, Caltech-101 image dataset. This dataset consists of more than 9000 images and these images are categorized into 102 categories. Dataset partitioning strategy is using 80% per category as a training dataset and the remaining 20% per category is taken as a testing dataset. In this work, the authors have observed that a hybrid of these three feature extraction algorithms is achieving more accurate results as compared to others. The Experimental results are discussed in Section 4 of this article. This paper is divided into four sections. In Section 1, it gives a brief introduction to the problem and the proposed system. In section 2, a literature survey is presented. In section 3, the details of the proposed system for object recognition and of various techniques used in this work have been presented. Section 4 discusses the experimental results in a view of the proposed system. Finally, section 5 gives the conclusion of the proposed work and presents the relevant future directions to the work.

2 Related work

Bosch et al. [6] presented performance results of random forests/ferns classifier and multi-SVM. They represented images as spatial pyramid using appearance and shape features. Then these features were classified using multi-SVM, Random Forest/Ferns classifiers. This experiment was done on Caltech-101 and Caltech-256 image database. The paper showed an improvement over other state-of-art classifiers when using 15 and 30 training images. As for Caltech-101, M-SVM classifier showed 81.3% accuracy when using 30 training images and Random Forests/Ferns showed approximately 70% accuracy when using 15 training images and approximately 80% accuracy when using 30 training images. For Caltech-256, Random Forest/Ferns classifier showed approximately 38% (using 15 training images) and approximately 45% (using 30 training images). Wang et al. [34] proposed Locality-constrained Linear Coding (LLC) scheme for image classification problem. This proposed scheme is implemented on the most widely used datasets: Caltech-101, Caltech-256 and Pascal VOC2007. The authors reported that the proposed scheme is giving more accurate results than others' research. Kulkarni et al. [19] presented the use of the ORB feature descriptor for object recognition on field-programmable gate arrays (FPGA). An FPGA is a form of highly configured hardware. They compared the performance of ORB over SIFT and SURF and proved ORB as providing more accurate results at high speed.

Bayraktar et al. [5] presented a detailed performance analysis among various feature detectors and descriptors. They proved SIFT-SURF combination as the most accurate, based on correct classification rate and ORB-BRIEF combination as the fastest algorithm compared to others. Jayech et al. [13] proposed a combination of SURF descriptor and dynamic random forest classifier for object recognition. This has been experimented on CIFAR-10 and STL-10 database and outcomes more accurate results as compared to the standard random forest. Karami et al. [16] described the comparative view among SIFT, SURF and ORB. They experimented with these techniques on distorted images of fisheyes. In their paper, they have compared the performance of SIFT, SURF, and ORB against rotation, scaling, shearing, intensity, and noise. All these techniques showed different results for different image transformations. Yang et al. [35] proposed a parallel key SIFT analysis for image recognition. They proved more accurate results on the Caltech-256 dataset. Loussaief et al. [21] implemented a Bag of Features approach for image classification. They experimented SURF algorithm against the global color feature for feature extraction. They experimented with various machine learning framework techniques on Caltech 101 images for image classification. They implemented Linear SVM, Quadratic SVM, Fine Gaussian SVM, Cubic SVM, Fine KNN, Weighted KNN, Boosted Tree, and Bagged Tree classifier on four datasets of the Caltech-101 database. After making a comparison chart, they proved that Cubic SVM was outperforming compared to others. Tareen et al. [33] presented a comprehensive comparison among SIFT, SURF, ORB, BRISK, KAZE, and AKAZE feature detectors and descriptors. They experimented with these techniques for image matching problem using MATLAB and OPENCV. Two different Datasets are used for the experiment. They proved that different techniques are working efficiently for different transformations. Chien et al. [8] presented a comparison among SIFT, SURF, ORB, and A-Kaze feature descriptors in terms of accuracy and computational. The paper helps to select the descriptors with respect to various generic invariance properties, i.e. rotation, scaling, features extracted, time of execution, and memory requirement to store the data. Gupta et al. [18] proposed a combination of Local Binary

Patterns (LBP) and Local Directional Peak Valley Binary Pattern (LDPVBP) for content based image retrieval. The proposed method has shown promising results with a shorter feature vector of size 56. Kim et al. [10] developed an efficient shape descriptor for content based image retrieval. The proposed system has performed excellently just by using angles, directions and locations from the part of the object. Gupta et al. [9] have presented an efficient object recognition system using SIFT and ORB feature detector. They have achieved a precision rate of 69.8% and 76.9% using ORB and SIFT feature descriptors, respectively. Using a combination of ORB and SIFT feature descriptors, they have achieved a precision rate of 85.6%. Bansal et al. [2] have presented a comprehensive study in the field of 2D object recognition. They have discussed various feature extraction techniques and classification algorithms which are required for object recognition. They have also presented an extreme gradient based technique for object recognition [3].

3 Methodology and experimental design of the proposed system

The experiment is demonstrated on various feature extraction, feature learning and image classification algorithms for a Caltech-101 dataset. Caltech-101 is a challenging dataset that includes 102 categories and 40–800 images per category. The size of each image is roughly 300×200 pixels. A comparative study among these algorithms based on recognition accuracy, True Positive Rate (TPR), False Positive Rate (FPR), and Area Under Curve (AUC) parameters is presented. Further, a 5-Fold cross-validation dataset partitioning is applied to improve the accuracy results based on various performance evaluation parameters. The study reveals that a combination of SIFT, SURF, and ORB with Random Forest classifier is showing more accurate results as compared to other state-of-the-art techniques, especially when used with 5-Fold cross-validation dataset partitioning strategy. The proposed system is implemented in the following stages (see Fig. 1).

- Collecting the image database.
- Generating feature vectors using three feature descriptors SIFT, SURF and ORB individually and in different combinations.
- Making clusters using k-means clustering algorithm from the unlabeled input dataset.
- Reducing the size of the feature vector by applying an LPP feature reduction algorithm.
- Classifying the training dataset into several classes using various classification algorithms, i.e. KNN, Naïve Bayes, Decision Tree, Random Forest.
- Using testing data that is evaluated from the input image.
- Matching of testing data with the training dataset to predict the label for the input image.

3.1 Feature extraction techniques

An object may be identified based on its color, texture, blob, shape or any other feature. The performance of an object recognition system mainly depends on the meaningful features extracted from the image database. Three feature extraction algorithms, namely, SIFT, SURF, and ORB have been considered in the experiment.

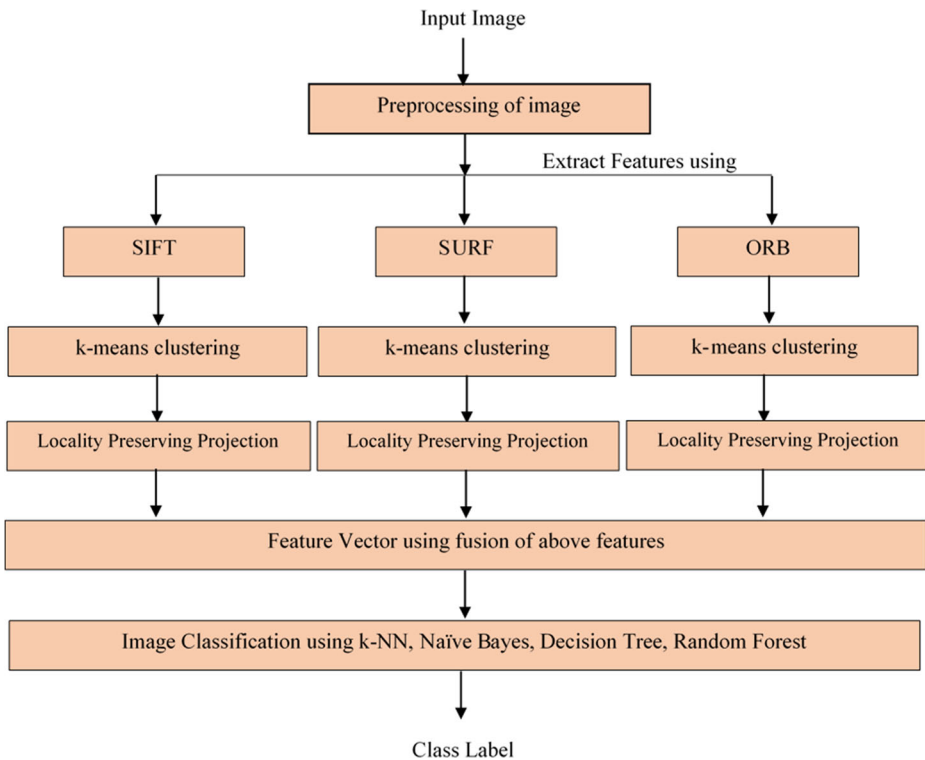


Fig. 1 Block Diagram of the proposed approach for object recognition system

3.1.1 Scale Invariant Feature Transform (SIFT)

Scale Invariant Feature Transform (SIFT) is a local keypoint detector and descriptor that was proposed by David Lowe in 1999 [22]. This algorithm extracts the features of an object considering different scale, rotation, illumination, and geometric transformations. SIFT has been proved as the most widely used algorithm in an object recognition. It obtains the image as an input and produces a set of features as output. SIFT output's feature vector of size $(kp \times 128)$ where kp is used for the number of keypoints detected and 128 for KeyPoint descriptors.

SIFT works in four phases

1. Scale-space Extrema Detection
2. Keypoint Localization
3. Orientation Assignment
4. Keypoint Descriptor

SIFT builds a multi-resolution pyramid over the input image. First, Difference of Gaussian (DoG) is applied to detect local extrema in space scale. Preferred extrema are considered as keypoints. Second, more accurate locations of keypoints are determined using threshold value. Third stage assigned the orientation to describe the keypoints with invariance to image rotation. Finally, in the last stage, a set of 128 keypoint descriptors is computed that derive 16 blocks of 4×4 size and each block creates 8 bin orientation histograms. Shivakanth et al.

[29] presented the functioning of SIFT descriptor for object recognition in their paper. Sivic et al. [31] and Kim et al. [17] created a bag of words using SIFT descriptor and outperformed the results of the Caltech-101 dataset. Jeong et al. [14] presented a new weighted keypoint matching algorithm via SIFT and produces more accurate results than others.

Pros

- SIFT produces good results for rotated, scaled images.
- It is invariant to lighting, and viewpoints.
- It derives more accurate results than other descriptors.

Cons

- It is mathematically complicated.
- It takes more time to compute the feature vector.
- Sometimes it does not give accurate results on blurring images.
- It does not show accurate results for distorted images.

3.1.2 Speeded up robust feature (SURF)

Speeded Up Robust Feature (SURF) introduced by Bay et al. [4]. It is a speedy version of SIFT. SURF is also a local keypoint detector and descriptor. It outputs 64 or 128 keypoint descriptors. SURF also works in four stages like SIFT, but with some changes. For feature detection, SURF uses LOG with Box-filter algorithm instead of LOG with DOG algorithm used for corner detection. Moreover, for feature descriptor, SURF used wavelet responses in vertical and horizontal directions.

Pros

- SURF produces better results in rotation, blur, and illumination as compared to SIFT.
- SURF is faster than SIFT.
- The length of feature vector is shorter than the features extracted through SIFT.

Cons

- It does not show good results in the scale area as compared to SIFT.

3.1.3 Oriented Fast and rotated BRIEF (ORB)

Oriented Fast and Rotated BRIEF (ORB) was introduced by Rublee et al. [26] and it was developed in OpenCV lab. ORB is much faster than SIFT and SURF. It performs feature extraction using the FAST keypoint detector, Harris corner detector, and BRIEF keypoint descriptor. Some modifications are further introduced in the method to enhance the performance.

Pros

- ORB extracts less (i.e. 32 feature descriptors) but more meaningful features of the image.

- It also takes less computational cost comparative to SIFT and SURF algorithms.
- Moreover, ORB is licensed free as compared to the patented SIFT and SURF algorithms.

Cons

- It is less scale invariant as compared to SURF.
- It does not work on blurry, and distorted images.

3.2 K-means clustering algorithm

K-means clustering algorithm is used to partition the dataset into k clusters according to some defined distance measure. Euclidean distance method or max-min measurement is used to find the distance between the centroid of the cluster and the object.

$$args = mindist(c_i, x)^2$$

Where c_i is the centroid of i^{th} centroid and x is the data point.

k-means produces two outputs – first is the centroids of k -clusters and second is the labels for the training data. The centroid is the collection of features that defines a group.

3.3 Locality preserving projection (LPP)

This is a dimensionality reduction algorithm. LPP is a linear approximation of nonlinear Laplacian Eigenmap by He et al. [11]. It is used to preprocess the low-level data in terms of large dimensions before using any classification technique. LPP follows Laplacian of a graph that maps the large dataset into a subset. It preserves the local information while mapping. The basic purpose of LPP is to reduce the computational cost of classification. There are two popular dimensionality reduction methods - Principal Component Analysis (PCA) and Locality Preserving Projection (LPP). Abdelmajed et al. [1] presented a comparative view between these two algorithms and proved as LPP gives high accuracy as compared to PCA. Shermina [28] analyses the results of the LPP over PCA on AT & T face database.

Locality Preserving Projection (LPP) works in three steps-

- Constructing the adjacency graph
- Choosing weight
- Create Eigenmap by computing eigenvectors and eigenvalues.

In our experiment, we have reduced the size of various features extracted from each technique to 8 components by applying LPP feature reduction algorithm.

3.4 Classification algorithms

Classification helps in object recognition by providing a class to the image. Image dataset is classified according to the extracted features of images and then a class is assigned to that similar group. Features that are extracted from testing images are compared with the training dataset and then object recognition system provides a class name for testing images. We have

used various machine learning techniques like Naïve Bayes classifier, KNN, Decision Tree, Random Forest for classification in our experiment.

3.4.1 *K-Nearest Neighbor (K-NN)*

k-NN classifier is the simplest image classifier. This classifier simply finds the distance between feature vectors and chooses the class of data points nearest to the data points of testing data. The K-NN algorithm works in four steps:

- First, it calculates the distance between the data point of testing data with each data point of training data.
- Then it sorts the distances calculated in step 1.
- It picks the k-minimum distances from the list of sorted distances. Here, K is used to picking the number of data points on which a decision would be made.
- Then, at last, it applies the majority voting that returns the class having the maximum value of that testing data.

For example, if two categories of images are taken, then k-NN algorithm returns the class which has the minimum distance from the data points of the testing image to each data-points of the training image dataset. In other words, when testing data is applied to this algorithm, it assigns the class name to test data which is nearest to the data points of testing data. To find the distance in a K-NN classifier, some distance metric or similarity function is used like Euclidean distance or Manhattan block distance.

Pros

- It performs well in multi-model classes.
- This classifier is easy to understand and implement.
- It requires less memory to store the feature vector for training data.
- The cost of the learning process being zero and complex concepts can be learned by local approximation using simple procedures.

Cons

- It can lead to classification error when there is only a small subset of features.
- It is computationally being expensive to find the K nearest neighbours when the dataset is very large.
- The performance of the classifier depends on the number of dimensions of the feature vector and therefore it is not a scalable approach.

Zhang et al. [36] proposed a hybrid of SVM and KNN classifiers for visual category recognition and they achieved more accurate results compared to NN and DAGSVM classifiers. The experiment was implemented on various datasets like MNIST, USPS, CUREt, and Caltech-101. Kim et al. [17] described and made a comparison between K-NN and SVM classifiers. They used SIFT feature descriptor and k-means clustering algorithm to extract the features of Caltech-4 cropped image categories. They experimented K-NN and SVM classifiers on these feature vectors.

Table 1 Feature extraction technique wise recognition accuracy for object recognition

Feature Extraction Techniques	Classifier Wise Recognition Accuracy (Training Set: 80% and Testing Set: 20%)			
	k-NN	Naïve Bayes	Decision Tree	Random Forest
SIFT (16)	63.61%	63.37%	62.62%	69.03%
SURF (16)	57.26%	51.32%	52.88%	55.19%
ORB (16)	46.30%	47.80%	47.23%	64.59%
SIFT+SURF	71.74%	72.31%	72.60%	76.29%
SIFT+ORB	66.43%	73.47%	72.89%	80.10%
SURF+ORB	68.51%	69.83%	69.78%	75.95%
SIFT+SURF+ORB	75.54%	79.29%	79.23%	83.27%

3.4.2 Naïve Bayes

Naïve Bayes is a probabilistic classifier that follows Bayes’ theorem. In Bayes’ theorem, the probability of an event is described that is based on prior knowledge of the condition that is related to some event. In object recognition, Naïve Bayes classifier helps to assign the class label to an object based on some conditional probability. Here, the conditional probability is computed on the extracted features of an object over the extracted features of other objects. The term naïve is defined because the features of one object are not dependent on the features of other objects.

For object recognition system, Naïve Bayes first constructs a probability model in class *c* from the extracted features of training data using eq. (1).

$$P(x_1, x_2, \dots, x_m | y = c) = \prod_{k=1}^m P(x_k | y = c) \tag{1}$$

Then, a conditional posterior probability distribution is computed for the feature vector belonging to a class using eq. (2).

$$P(y = c | x_1, x_2, \dots, x_m) = \frac{P(y = c) \prod_{k=1}^m P(x_k | y = c)}{P(x_1, x_2, \dots, x_m)} \tag{2}$$

Table 2 Feature extraction technique wise true positive rate for object recognition

Feature Extraction Techniques	Classifier Wise True Positive Rate (Training Set: 80% and Testing Set: 20%)			
	k-NN	Naïve Bayes	Decision Tree	Random Forest
SIFT	63.60%	63.40%	62.60%	69.00%
SURF	57.30%	51.30%	52.90%	55.20%
ORB	46.30%	47.80%	47.20%	64.60%
SIFT+SURF	71.70%	72.30%	72.60%	76.30%
SIFT+ORB	66.40%	73.50%	72.90%	80.10%
SURF+ORB	68.50%	69.80%	69.80%	76.00%
SIFT+SURF+ORB	75.50%	79.30%	79.20%	83.30%

Table 3 Feature extraction technique wise based on false positive rate for object recognition

Feature Extraction Techniques	Classifier Wise False Positive Rate (Training Set: 80% and Testing Set: 20%)			
	k-NN	Naïve Bayes	Decision Tree	Random Forest
SIFT	0.90%	0.80%	0.80%	0.50%
SURF	1.20%	1.70%	1.60%	1.30%
ORB	1.10%	0.60%	0.70%	0.40%
SIFT+SURF	0.80%	0.50%	0.40%	0.40%
SIFT+ORB	0.90%	0.30%	0.30%	0.20%
SURF+ORB	0.60%	0.30%	0.30%	0.20%
SIFT+SURF+ORB	0.50%	0.10%	0.10%	0.10%

For new feature vector $x' = (x'_1, x'_2, \dots, x'_m)$ during testing data, a class Y' is predicted using eq. (3).

$$Y' = \operatorname{argmax}_{c = 1, 2, \dots, m} P(y = c) \prod_{k=1}^m P(x'_k | y = c) \quad (3)$$

Pros

- It is easy to understand and develop
- Image dataset can be trained easily.
- It takes less time for computations.

Cons

- It is a probabilistic approach.
- It is very confident for independent features that it makes.
- It computes less classification accuracy as compared to other classifiers.

Park [23] proposed Naïve Bayes classifier for image classification. He experimented that the Naïve Bayes classifier is producing more efficient results in terms of speed and accuracy on Caltech dataset. Initially, he experimented with four features, i.e. Discrete Cosine

Table 4 Quantitative comparison of different feature extraction techniques based on area under curve for object recognition

Feature Extraction Techniques	Classifier Wise Area Under Curve (Training Set: 80% and Testing Set: 20%)			
	k-NN	Naïve Bayes	Decision Tree	Random Forest
SIFT	81.40%	94.90%	93.10%	95.00%
SURF	78.00%	93.80%	92.60%	92.20%
ORB	72.60%	87.60%	86.60%	94.20%
SIFT+SURF	85.50%	97.20%	97.20%	97.70%
SIFT+ORB	82.80%	96.30%	96.00%	97.60%
SURF+ORB	83.90%	95.80%	95.30%	96.40%
SIFT+SURF+ORB	87.50%	97.80%	97.70%	97.70%

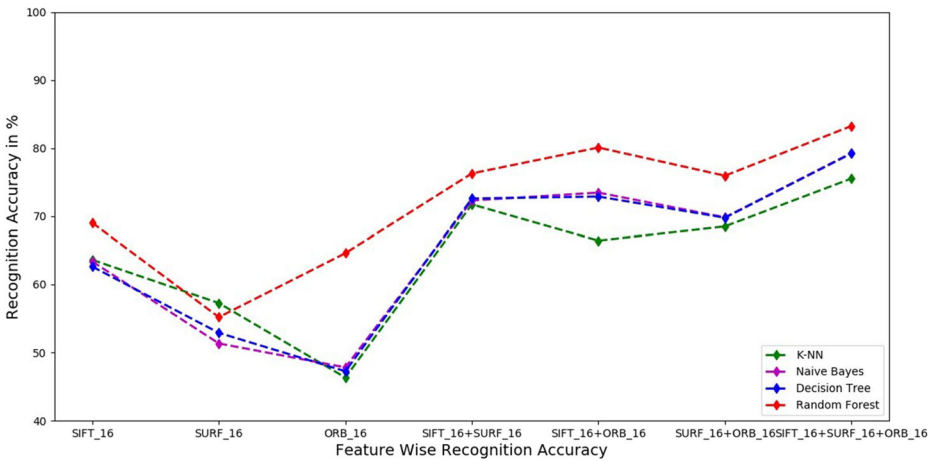


Fig. 2 Feature wise recognition accuracy achieved for Caltech-101 dataset

Transformation (DCT), Local Binary Pattern (LBP), Covariance Descriptor (CD) and Wavelet Packet Transform (WPT) and calculates the average classification accuracy with CNN classifier. He observed DCT feature as DCT is giving more accurate results compared to others. In his second experiment, he applied CNN, FM, MLPNN and NB classifiers over DCT features and proved that Naïve Bayes classifier outperforms other classifiers with 77.2% accuracy.

3.4.3 Decision tree

A decision tree algorithm is a classifier used to partition the dataset into various subsets and these subsets are further partitioned till no further partitioning of the dataset is possible. The leaves of the decision tree represent the class label of an object. The partitioning of a dataset is performed based on the features of the image. Each node represents some feature of the object and the leaf node represents the name of the class. Through decision tree, it is easy to represent

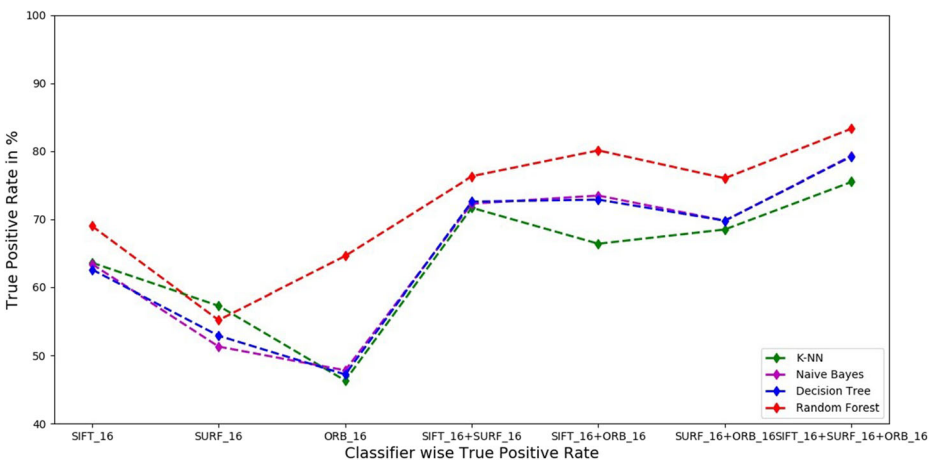


Fig. 3 Feature wise True Positive Rate (TPR) achieved for Caltech-101 dataset

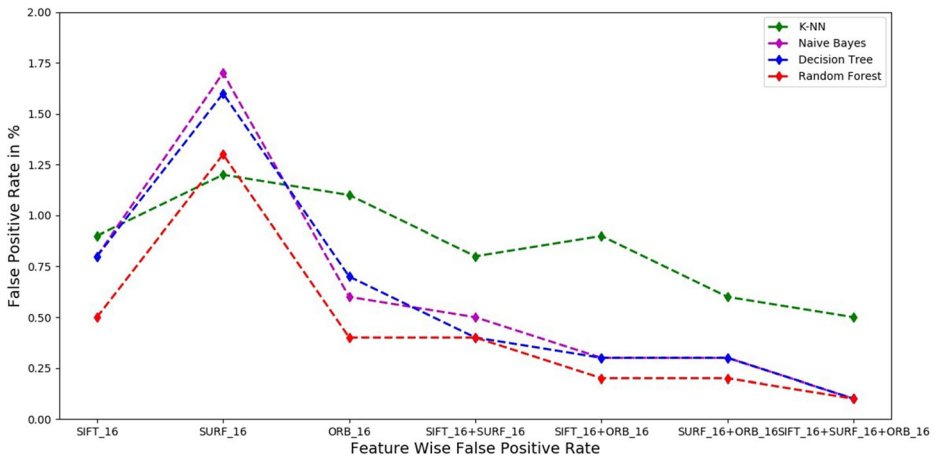


Fig. 4 Feature wise False Positive Rate (FPR) achieved for Caltech-101 dataset

the various features of an object and it makes the interpretation easy for testing data. This technique basically helps in a Bag of Words and an approach used for object recognition.

Pros

- It is easy to classify the objects in hierarchical way using their similar properties.
- It can handle both numerical and categorical dependent/independent features.
- It is easy to identify the object by traversing the tree using top-down approach.

Cons

- It is considered as weak classifier due to improper identification of some objects.
- It may favor variables that have more categories.
- It is generally less preferable than other statistical classifiers when higher precision and recall are required.
- It needs a boosting algorithm to improve the recognition results.

Table 5 Feature extraction techniques based on recognition accuracy for object recognition using 5-fold cross validation dataset partitioning

Feature Extraction Techniques	Classifier Wise Recognition Accuracy (5-Fold Cross Validation Dataset Partitioning)			
	k-NN	Naïve Bayes	Decision Tree	Random Forest
SIFT	63.19%	62.18%	62.04%	69.06%
SURF	58.50%	52.66%	54.45%	57.17%
ORB	48.65%	48.57%	47.50%	66.15%
SIFT+SURF	72.54%	72.74%	73.75%	76.71%
SIFT+ORB	67.86%	73.23%	72.70%	81.59%
SURF+ORB	70.52%	72.21%	72.06%	77.99%
SIFT+SURF+ORB	77.04%	80.16%	80.12%	84.55%

Table 6 Feature extraction techniques based on true positive rate for object recognition using 5-fold cross validation dataset partitioning

Feature Extraction Techniques	Classifier Wise True Positive Rate (5-Fold Cross Validation Dataset Partitioning)			
	k-NN	Naïve Bayes	Decision Tree	Random Forest
SIFT	48.65%	48.57%	47.50%	66.15%
SURF	58.50%	52.70%	54.50%	57.20%
ORB	48.70%	48.60%	47.50%	66.20%
SIFT+SURF	72.50%	72.70%	73.80%	76.70%
SIFT+ORB	67.90%	73.20%	72.70%	81.60%
SURF+ORB	70.50%	72.20%	72.10%	78.00%
SIFT+SURF+ORB	77.00%	80.20%	80.10%	84.60%

Patil et al. [24] experimented with global color, local color feature and texture features of Corel images. 1000 Corel images were used for the experimental work out of which 500 images were used for training and the remaining 500 were used for testing. For image annotation, they proposed a decision tree classifier and rough set classifier. They experimented that the results of the rough set classifier were giving more accurate results as compared to a decision tree.

3.4.4 Random Forest

Random forest multi-class classifier is a collection of randomized decision trees that work very efficiently in a classification task. This classifier is used when the image is split into sub images to create a codebook or a bag of words for an object recognition. Random forest function is working in two phases. In the first phase, it trains the image dataset by injecting the randomness into the training of the trees and then combines the outputs of each trained tree into a single classifier. In the second phase, the test image is passed down through each tree until it reaches the leaf node. Then the average of the posteriors across all trees is taken as the classification of the input image. Schroff et al. [27] solved the object class segmentation problem using random forest classifier.

Table 7 Feature extraction technique wise false positive rate for object recognition using 5-fold cross validation dataset partitioning

Feature Extraction Techniques	Classifier Wise False Positive Rate (5-Fold Cross Validation Dataset Partitioning)			
	k-NN	Naïve Bayes	Decision Tree	Random Forest
SIFT	0.90%	0.80%	0.80%	0.60%
SURF	1.40%	1.70%	1.60%	1.40%
ORB	1.00%	0.70%	0.80%	0.40%
SIFT+SURF	0.70%	0.50%	0.40%	0.40%
SIFT+ORB	0.80%	0.30%	0.30%	0.20%
SURF+ORB	0.60%	0.30%	0.30%	0.20%
SIFT+SURF+ORB	0.50%	0.20%	0.20%	0.10%

Table 8 Feature extraction technique wise area under curve for object recognition using 5-fold cross validation dataset partitioning

Feature Extraction Techniques	Classifier Wise Area Under Curve (5-Fold Cross Validation Dataset Partitioning)			
	k-NN	Naïve Bayes	Decision Tree	Random Forest
SIFT	81.20%	94.60%	93.00%	94.90%
SURF	78.50%	94.20%	93.10%	92.70%
ORB	73.80%	87.60%	86.60%	94.80%
SIFT+SURF	85.90%	97.30%	97.20%	96.50%
SIFT+ORB	83.50%	96.30%	95.90%	97.60%
SURF+ORB	84.90%	96.30%	95.80%	96.90%
SIFT+SURF+ORB	88.30%	97.90%	97.90%	98.00%

Pros

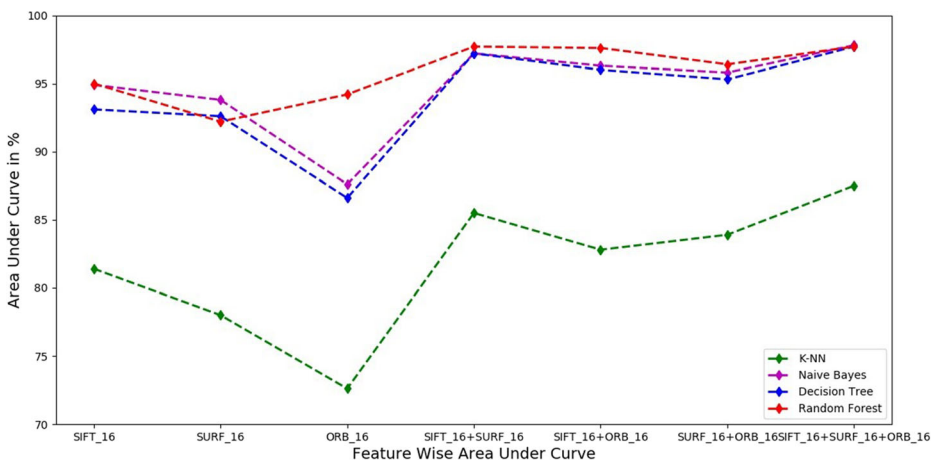
- It is a strong classifier as it consists of the properties of decision tree and boosting which has improved the results of object recognition.
- It runs efficiently on large database.
- It is able to handle both numerical and categorical data without scaling and transformation.
- It is faster than many other boosting classifiers such as XGBoosting Classifier.

Cons

- A large number of decision trees can slow the task of predicting the class.
- It has not shown sufficient results on unbalanced data.

1. Performance Analysis and Experimental Validation

In this section, a comparative analysis of various feature extraction algorithms with various classifiers is presented. The analysis is measured based on four performance evaluation

**Fig. 5** Feature wise Area Under Curve (AUC) achieved for Caltech-101 dataset

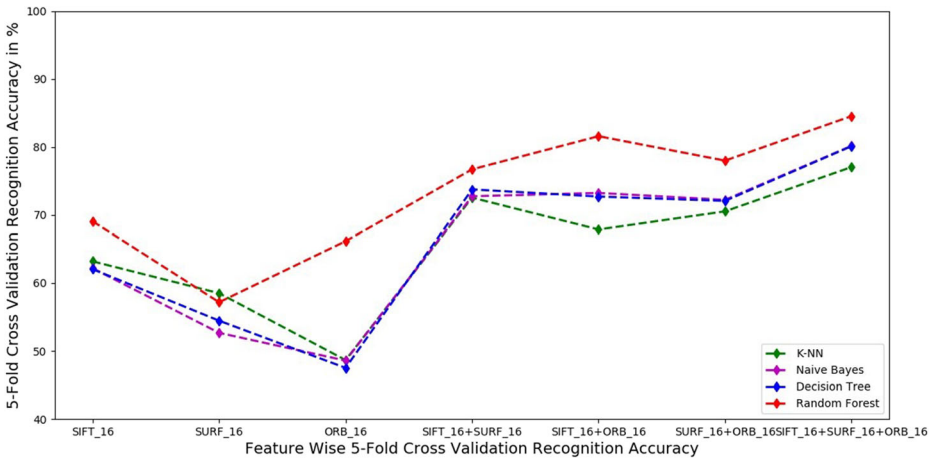


Fig. 6 Feature wise 5-fold cross validation recognition accuracy for Caltech-101 dataset

parameters, i.e. recognition accuracy, True Positive Rate (TPR), False Positive Rate (FPR) and Area Under Curve (AUC). Because these four parameters are essential to test the effectiveness of the system. A challenging image dataset Caltech101 is considered for the experimental work. Caltech101 dataset contains 101 categories of the images where 80% of all categories are used for training data and the remaining 20% of all categories are used for testing data. Furthermore, the results achieved using dataset partitioning strategy are analyzed using a 5-Fold cross-validation dataset partitioning algorithm. The algorithms are implemented on Intel Core i7 processor having Windows 10 operating system and 8GB RAM. Programming language Python 3.6 and OpenCV image processing library is used for coding. Table 1 depicts the classifier wise recognition accuracy and experimental reports describing that a hybrid of SIFT, SURF, and ORB feature extraction algorithms has achieved high recognition accuracy i.e. 83.27% when Random Forest classifier is used. Table 2 represents classifier wise true positive rate and reports that a hybrid of these three feature extraction techniques achieved a high true positive rate i.e. 83.20% when classified with a Random Forest classifier. Table 3 depicts the classifier wise false positive rate

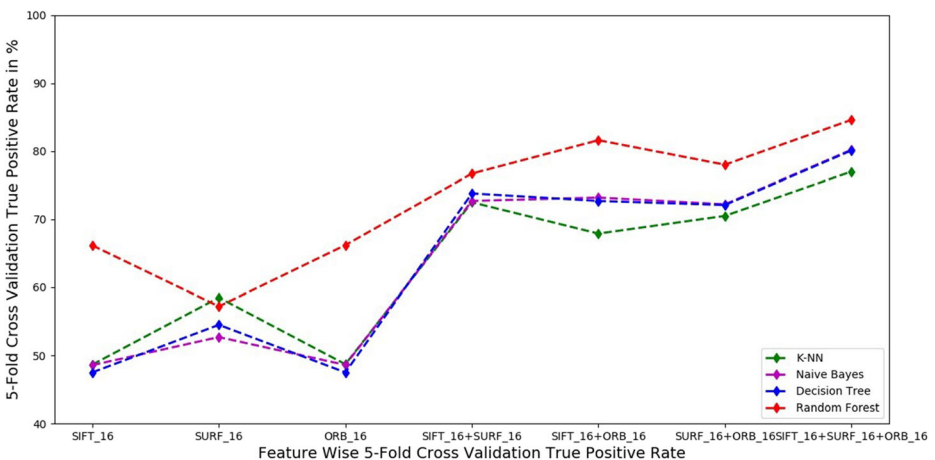


Fig. 7 Feature wise 5-fold cross validation True Positive Rate (TPR) for Caltech-101 dataset

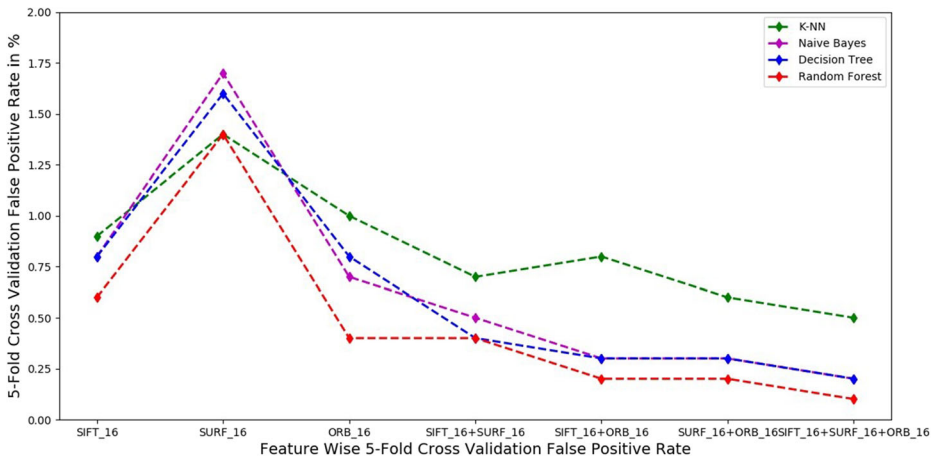


Fig. 8 Feature wise 5-fold cross validation False Positive Rate (TPR) for Caltech-101 dataset

and reports that the hybrid is showing a low false rate, i.e. 0.10% when classified with a Random Forest classifier. Table 4 reports that the hybrid achieved a higher area under curve, i.e. 97.80% when classified with a Naïve Bayes classifier. These experimental results are graphically depicted in Figs. 1, 2, 3 and 4. Furthermore, the authors applied a 5-Fold cross-validation dataset partitioning strategy to improve our classification results done on various parameters and get some improvement in recognition accuracy as shown in Table 5 and the true positive rate shown in Table 6. But there is no improvement in the false positive rate (refer Table 7). In classifier wise area under curve (refer Table 8), a major improvement (98%) using Random Forest classifier is achieved. Experimental results based on 5-fold cross validation technique are also graphically depicted in Figs. 5, 6, 7, 8 and 9.

3.5 Comparison with other techniques proposed by other authors

This subsection covers a comparative analysis among various techniques proposed by many authors on Caltech-101 dataset (Table 9). This includes a comparison based on fusion of

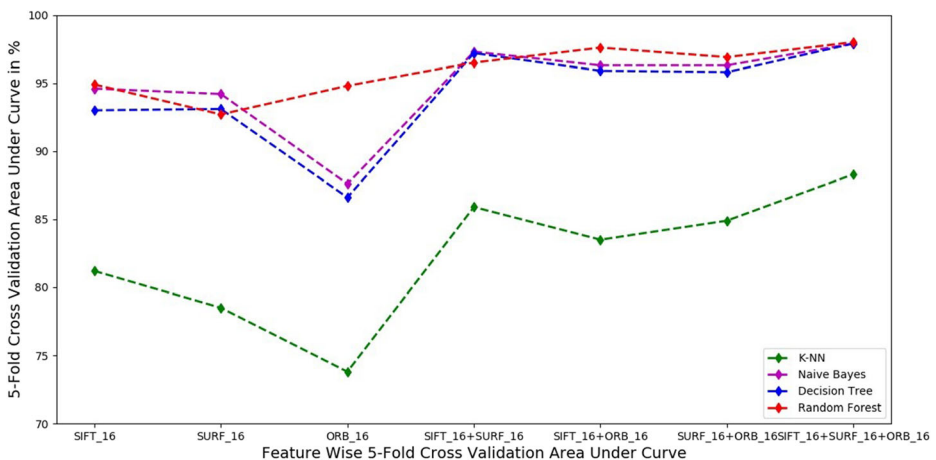


Fig. 9 Feature wise 5-fold cross validation Area Under Curve (AUC) for Caltech-101

Table 9 Comparison with existing methods on Caltech-101 dataset

Authors	Feature Extraction method	Classification algorithm	Accuracy
Huang et al. [12]	LVS, SIFT	Classification pipeline	75.7%
Srivastava et al. [32]	SURF, LBP	Support Vector Machine	79.0%
Kabbai et al. [15]	SURF, Wavelet Transform		79.9%
Proposed System	SIFT, SURF, ORB	Random Forest	83.3%

conventional feature extraction methods. Our approach outperforms the existing methods for the same setup.

3.6 Comparison with other techniques based on deep features

Nowadays, deep learning is in big demand for object recognition as it automatically performs both feature extraction and classification. Although, deep convolutional neural networks have shown tremendous results for object recognition, but still there are various issues due to which machine learning is still in demand. Some of the major issues are as deep model requires a large memory to store the large collection of features extracted, and they also take more time for computations than conventional methods. In this subsection, a comparison is made among features using our proposed approach and some popular deep learning model (i.e. VGG19 (Simonyan et al. [30]) and ResNet50 (He et al. [12]) for object recognition with Random Forest (RF) classifier. Table 10 shows the comparative analysis based on accuracy, TPR, FPR, and Area Under Curve (AUC). Here, it is proved that our approach outperforms over deep features. In future, a combination of various conventional and deep models can be experimented that may help to improve the recognition accuracy.

4 Conclusion and future directions

In this section, the authors have presented concluding results of the present work. The paper presents object recognition results using SIFT, SURF, ORB feature descriptors and various combinations of these feature descriptors using k-NN, Naïve Bayes, Decision Tree, and Random Forest classification models. The experimental work was conducted using a public dataset, namely, Caltech-101. In this article, performance of various techniques is evaluated based on four performance evaluation parameters, i.e. recognition accuracy, True Positive Rate (TPR), False Positive Rate (FPR) and Area Under Curve (AUC). And the experimental work describes that Random Forest classifier has shown better performance in all cases when a hybrid of SIFT, SURF, and the ORB feature descriptors is used for object recognition system. Further, 5-Fold cross validation dataset partitioning is also applied to assess the effectiveness

Table 10 Comparison with existing CNN features based state-of-art work

Technique	Accuracy	TPR	FPR	AUC
Proposed features + RF	83.27%	83.30%	0.1%	97.70%
VGG19+RF	76.73%	77.58%	0.22%	50.44%
ResNet50+RF	77.99%	77.94%	0.19%	50.28%

of all feature descriptors and classification techniques considered in this work. In future work, these feature descriptors and combinations of these feature descriptors can be used to improve the recognition results of documents images, medical images, spectral images, etc. It's an experimental study where the performance of three feature descriptors, namely, SIFT, SURF and ORB are considered with several classification models in terms of recognition accuracy, True Positive Rate (TPR), False Positive Rate (FPR) and Area Under Curve (AUC). The classification models used in the experimental work are Naive Bayes, K-NN, Decision Tree and Random Forest. The experiment concluded that the Random Forest classifier is attaining more accurate results as compared to Naive Bayes, K-NN, and Decision Tree classifiers. The primary aim of this work was to present the effectiveness of SIFT, SURF and ORB feature detectors for object recognition of a public database named Caltech-101.

Declarations The authors declare that they have no conflict of interest in this work. The authors have considered a public dataset, namely, Caltech-101 for experimental work.

References

1. Abdelmajed AKA (2016) A comparative study of locality preserving projection and principle component analysis on classification performance using logistic regression. *J Data Anal Inform Process* 4(2):55–63
2. Bansal M, Kumar M, Kumar M (2020a) 2D object recognition techniques: state-of-the-art work. *Arch Computation Methods Eng*. <https://doi.org/10.1007/s11831-020-09409-1>
3. Bansal M, Kumar M, Kumar M (2020b). XGBoost: object recognition using shape descriptors and extreme gradient boosting classifier. DOI: https://doi.org/10.1007/978-981-15-6876-3_16.
4. Bay H, Ess A, Tuytelaars T, Van GL (2008) Speeded-up robust features (SURF). *Comput Vis Image Underst* 110(3):346–359
5. Bayraktar E, Boyraz P 2017 Analysis of feature detector and descriptor combinations with a localization experiment for various performance metrics. arXiv preprint arXiv:1710.06232
6. Bosch A, Zisserman A, Munoz X 2007 Image classification using random forests and ferns. *Proceedings of the IEEE 11th International Conference on Computer Vision (ICCV)*, 1–8.
7. Calonder M, Lepetit V, Strecha C, Fua P 2010 BRIEF: binary robust independent elementary features. *Proceedings of the European Conference on Computer Vision*, 778–792.
8. Chien HJ, Chuang CC, Chen CY, Klette R (2016) When to use what feature? SIFT, SURF, ORB, or A-KAZE features for monocular visual odometry. In: *Proceedings of the International Conference on Image and Vision Computing*. IVCNZ, New Zealand, pp 1–6
9. Gupta S, Kumar M, Garg A (2019) Improved Object Recognition Results using SIFT and ORB Feature Detector. *Multimed Tools Appl* 78:34157–34171
10. Gupta S, Roy PP, Dogra DP, Kim BG (2020) Retrieval of colour and texture images using local directional peak valley binary pattern. *Pattern Anal Applic* 23:1569–1585. <https://doi.org/10.1007/s10044-020-00879-4>
11. He X, Niyogi P (2004) Locality preserving projections. *Adv Neural Inform Process Syst*:153–160
12. He K, Zhang X, Ren S and Sun J 2016 Deep residual learning for image recognition. In *Proc IEEE Conf Comput Vis Pattern Recognition*, 770–778.
13. Jayech K, Mahjoub MA 2017 Object recognition based on dynamic random forests and SURF descriptor. *Proceedings of the International Conference on Intelligent Data Engineering and Automated Learning*, 355–364.
14. Jeong D M, Kim J H, Lee Y W and Kim B G 2018 Robust weighted Keypoint matching algorithm for image retrieval. *Proc Second Int Conf Video, Image Processing*:145–149.
15. Kabbai L, Abdellaoui M, Douik A (2019) Image classification by combining local and global features. *Vis Comput* 35(5):679–693
16. Karami E, Prasad S and Shehata M 2017 Image matching using SIFT, SURF, BRIEF and ORB: performance comparison for distorted images. arXiv preprint arXiv:1710.02726
17. Kim J, Kim BS, Savarese S 2012 Comparing image classification methods: K-nearest-neighbor and support-vector-machines. *Appl Mathematics Electric Comput Eng* 133–138.
18. Kim B, Hong G, Psannis KE (2017) Design of efficient shape feature for object-based watermarking technology. *Multimed Tools Appl* 76:22741–22759. <https://doi.org/10.1007/s11042-017-4344-3>

19. Kulkarni AV, Jagtap JS, Harpale VK (2013) Object recognition with ORB and its implementation on FPGA. *Int J Adv Comput Res* 3(3):164
20. Leutenegger S, Chli M, Siegwart R (2011) BRISK: binary robust invariant scalable Keypoints. *Proceedings of the International Conference on Computer Vision (ICCV)*, 2548–2555.
21. Loussaief S, Abdelkrim A (2018) Deep learning vs. bag of features in machine learning for image classification. *Proc Int Conf Adv Syst Electric Technol*, 6–10
22. Lowe DG (2004) Distinctive image features from scale-invariant Keypoints. *Int J Comput Vis* 60(2):91–110
23. Park DC (2016) Image classification using Naïve Bayes classifier. *Int J Comput Sci Electronics Engineering (IJCSSEE)* 4(3):135–139
24. Patil MP, Kolhe SR (2014) Automatic image annotation using decision trees and rough sets. *Int J Comput Sci Appl (IJCSA)* 11(2):38–49
25. Rosten E and Drummond T (2006) Machine learning for high speed corner detection. *Proc 9th Eur Conf Comput Vision*, 1:430–443.
26. Rublee E, Rabaut V, Konolige K, Bradski G (2011) ORB: an efficient alternative to SIFT or SURF. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 1–8
27. Schroff F, Criminisi A, Zisserman A (2008) Object class segmentation using random forests. *Proc British Machine Vision Assoc Conference (BMVC)*, 1–10.
28. Shermira J (2010) Application of locality preserving projections in face recognition. *Int J Adv Comput Sci Appl* 1(3):82–85
29. Shivakanth AM (2014) Object recognition using SIFT. *Int J Innovat Sci Eng Technol (IJSET)* 1(4):378–381
30. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
31. Sivic J, Russell BC, Efros AA, Zisserman A, Freeman WT (2005) Discovering objects and their location in images. *Proc Tenth IEEE Int Conf Computer Vision* 1:370–377
32. Srivastava D, Bakhthula R, Agarwal S (2019) Image classification using SURF and bag of LBP features constructed by clustering with fixed centers. *Multimed Tools Appl* 78(11):14129–14153
33. Tareen SAK, Saleem Z (2018) A comparative analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK. *Proceedings of the International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, 1–10
34. Wang J, Yang J, Yu K, Lv F, Huang T, Gong Y (2010) Locality-constrained linear coding for image classification. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3360–3367
35. Yang F, XIE M (2017) Codebook learning for image recognition based on parallel key SIFT analysis. *IEICE Trans Inf Syst* 100(4):927–930
36. Zhang H, Berg AC, Maire M, Malik J (2006) SVM-KNN: discriminative nearest neighbor classification for visual category recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 2:2126–2136

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

Monika Bansal¹ · Munish Kumar² · Manish Kumar³

¹ Department of Computer Science, Punjabi University, Patiala, India

² Department of Computational Sciences, Maharaja Ranjit Singh Punjab Technical University, Bathinda, Punjab, India

³ Department of Computer Science, Baba Farid College, Bathinda, Punjab, India