



A deep neural network and classical features based scheme for objects recognition: an application for machine inspection

Nazar Hussain¹ · Muhammad Attique Khan² · Muhammad Sharif¹ · Sajid Ali Khan³ · Abdulaziz A. Albeshier⁴ · Tanzila Saba⁵ · Ammar Armaghan⁶

Received: 4 November 2019 / Revised: 1 February 2020 / Accepted: 13 March 2020 /
Published online: 2 April 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

Computer Vision (CV) domain is widely used in the current era of automation and visual surveillance for the detection and classification of different objects in a diverse environment. The automatic machine inspection of different objects in the scenes is based on internal and external parameters like features that provide a huge amount of information related to the nature of an object in the scene. In this work, we propose a new automated method based on classical and deep learning feature selection. The proposed object classification method follows three steps. The data augmentation is performed in the first step to make the balance database. Later, Pyramid HOG (PHOG) and Central Symmetric LBP (CS-LBP) features are serially fused along with deep learning-based extracted features. The deep learning features are extracted from the pre-trained CNN model name Inception V3. In the third step, a new technique name Joint Entropy along with KNN (JEKNN) is employed to select the best features. The best-selected features are finally classified by well-known supervised learning methods and choose the best one based on higher accuracy. The proposed method is evaluated on Caltech101 balanced dataset and achieved maximum accuracy of 90.4% on Ensemble classifier which outperforms as compare to existing techniques.

Keywords Object classification · Augmentation · Classical features · CNN features · Feature selection

1 Introduction

Object classification has become a challenging task in the area of image processing and computer vision (CV) from the last few decades [6]. CV domain is widely used in the current

✉ Sajid Ali Khan
attique@ciitwah.edu.pk; sajidalibn@gmail.com

era of automation and visual surveillance for the detection and classification of different objects in a diverse environment [3, 19]. The diverse environments are pedestrian tracking, disease detection & classification, action recognition, gait recognition, video tracking, person re-identification, optical character recognition and automatic object detection in autonomous vehicles [9, 44, 50, 54, 56]. Enormous work has been done for object classification [58] but several challenges still exist such as different color schemes, orientation and complex background, and different sort of feature extraction [22].

Multiple techniques have been adopted to overcome these listed challenges and improves classification performance. Researchers introduced various methods for the classification of different objects under complex background. They mostly focused on reliable feature extraction techniques for classification [52]. Different features have been used in traditional approaches such as Histogram of Oriented Gradient (HoG) [10], Local Binary Patterns (LBP) [52] and Bag of Words (BoW) [12]. These features are classified through different kinds of Machine learning (ML) algorithms like Linear Support Vector Machine (L-SVM) [5], Quadratic SVM [29], Cubic SVM, Naïve Bayes [36], Bayesian model, and a few more.

Recently, a new component is introduced in machine learning known as deep learning (DL) and gain significant performance in many applications like object classification, video streaming, and many more [28, 25, 18]. In DL, the Convolutional Neural Network (CNN) is a famous approach for feature extraction. The systems which are based on CNN are showing better performance as compared to classical feature extraction techniques. Several pre-trained CNN models have been introduced which are freely available to access. These models are AlexNet [21], VGG16 [37], ResNet50 [15] and InceptionV3 [46]. These models are trained over a million of different object images. A simple CNN model includes few successive layers (i.e. Convolutional, pooling, fully connected) which are used to train the model.

The more recent, the fusion of multiple descriptors in one matrix shows much interest of CV researcher for several complex challenges such as gait recognition, object classification, action recognition, medical imaging, and many more [35, 38, 20]. The fusion process increases the information of an object under different aspects like shape, color, local points, etc. This process affects the system accuracy and reported by recent researchers, the almost average 5% accuracy is increased. However, this process has few drawbacks such as system time complexity and overall computational time which is double after this process. To handle the problem of system time complexity, the features selection techniques are presented by CV researchers. As we know that the machine learning deals with diverse datasets having different data type, dimensionality, noise, redundancy and irrelevant features. The challenges increase as the amount of data and features increased. The major aim of feature selection is to reduce the noise and redundancy of features to perform operations like classification and detection robustly with less computational time and high accuracy [1]. A few famous features selection methods are- Genetic Algorithm [26], firefly [49], Entropy-controlled [2], and Crow Search Algorithm [39].

The proposed method is inspired by Rashid et al. [33] presented a fusion technique along with the best features selection. At first, they have been extracted handcrafted feature using improved saliency method and applied entropy for robust feature selection. Later, the deep features have been extracted using pre-trained deep CNN models and fused their information along with first type extracted features. The entropy-controlled selection technique is employed for the selection of best-fitted features for final classification. To follows this work, we proposed a new automated system

that skips the segmentation part without losing any accuracy. Moreover, without segmentation step, our method shows better efficiency in terms of computational time.

2 Related work

Objects classification is gaining reputable status in the field of computer vision-based on famous application name intelligent machine inspection [7, 27, 34]. Many techniques are presented in the literature to tackle the problems of objects classification under the different environmental conditions like illumination, scale invariance, multifarious background, and many more. Godrati et al. [13] presented a deep learning model for 3D objects classification. The bags of features are extracted by employing spatial pyramid technique. Later, pre-trained CNN model is utilized to extract deep features which are combined and classified using Support vector machine (SVM). Weibel et al. [53] fused point features along with deep CNN features to provide better performance in rotation and noise inherited images. The presented method discriminate the objects in 3D indoor scenario. The Stanford dataset is used for evaluation of presented method and achieved better accuracy. Kaur et al. [16] implanted a computerized system to handle the problem of real-time object classification using convolutional feature map along with adaptive learning rate. The implemented model is trained on blurred and noisy data due to cluttering background. The offline model training is performed in Caltech256 dataset and achieved exceptional performance. Gill et al. [31] presented an object classification method under indoor and outdoor environment. In the implemented method, SIFT, SURF, and Tamura features which are combined and classified through SVM. This method is tested on MIT-Indoor dataset and achieved outstanding performance. Liu et al. [23] presented a deep learning-based system for objects classification. The pre-trained model is used and performs activation on middle for features extraction. The middle layers are fused along with latent features and classified through classifier. This fusion process gives exceptional performance on tested dataset.

Feature reduction and selection method introduced to select the most relevant features for classification. Many selection methods are introduced by researchers in literature which are work under specific problems [32, 41]. The selection of most important features from the original vector is a key challenge in the area of machine learning. A few famous feature selection methods are entropy-controlled, GA, Euclidean Distance (ED) based methods and a name few. Correlation and consistency based feature selection and reduction techniques performed better with supervised detection models [47]. In [24] developed a hybrid selection method by overcoming the limitation of Grey Wolf Optimizer (GWO) and Whale Optimization Algorithm (WOA) by reducing the immature convergence rate and improved selection of binary feature of objects. In the literature, the different sort of manual features has been used such as HOG, LBP, SURF, SIFT and few others. From the above studies, it can be easily summarize that sometimes, the fusion of these features provide better accuracy but it is not guaranteed for high accuracy therefore the selection process is performed to select the best subset of features to maintain the system efficiency and accuracy. The selection process also reduces the number of predictors which directly effects the classification time (Appendix Table 8).

2.1 Real-time object classification

Real-time object classification includes vehicle classification, number plate detection and object tracking. Deep learning model are used to perform real-time object detection and classification because of robustness and less computational time consumption as compared to machine learning. The implanted system is based on faster R-CNN to perform robustly in less time with higher rate of accuracy [51]. Bilal et al. [4] introduced a model to increase the speed of kernel classifier by applying soft cascade. The kernel detection and classification performance increased to detect a pedestrian from videos robustly by inclusion of corresponding features and rejection of irrelevant features. Zhi et al. [57] presented a modified CNN called LightNet to solve the 3D object detection in real-time environment. LightNet predict class and orientation labels of different 3D objects and shapes. Performance of introduced LightNet is robust on Shape Net Core55 dataset by adopting efficient training and validation techniques.

3 Problem statement and contributions

Various challenges are exist for objects classification in the static images which vitiates the system accuracy. These challenges are transparent and complex background, lighting conditions, and similarity among two or more than two objects. Several methods are presented in the literature but still they are failed to handle these challenges. Another major challenge of objects classification is an size of database which is used for training the models. As in this work, we utilized Caltech 101 dataset which includes 100 object classes. Therefore, an automated system performance is always depends on the number of selected features which are used for classification. In this work, a new automated method is proposed for objects classification using deep learning. Major contributions in this work are listed below:

- Data augmentation is performed by horizontal flip, vertical flip and transpose operation
- PHOG features are computed for the shape information of objects
- CNN based features are extracted using transfer learning and fused along with PHOG
- Select the best features by a new method name JEKNN
- Selected features are validated on various classifiers and best results are compared with recent techniques.

4 Proposed methodology

A new method is proposed for objects classification using deep learning and classical features selection. Three-step processes is performed as augmentation, CNN and classical features extraction and fusion, and selection of best features. In the CNN features, a pre-trained model is used along with transfer learning. A complete architecture of proposed method is shown in Fig. 1. In this figure, it is shown that the augmented database is utilized to extract CNN and classical features in a parallel processing and selects the best of them before fusion stage. At the end, by using classifier the labeled data is returns as an output.

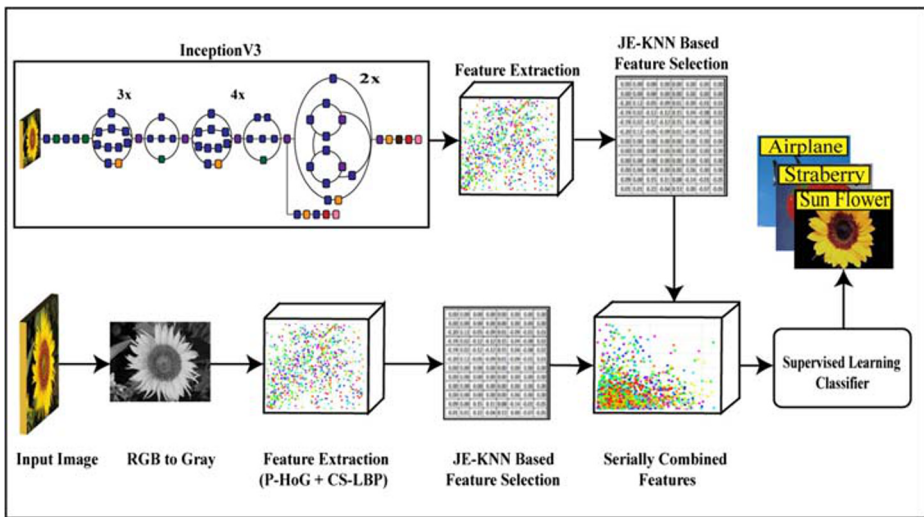


Fig. 1 Proposed architecture for objects classification using deep learning and classical features fusion

4.1 Database augmentation

In machine learning, especially in deep learning, data augmentation is a dominant data extension method that increases the training data. Increases in training data improve the performance of deep learning methods. In the image field, the augmentation includes flipping the images, translating image pixels, and few more [48]. Previously, the manual process is used for data augmentation which needs to be automated [8].

In this work, we utilized an automated technique for augmentation of the selected dataset. As in this work, we utilized the Caltech101 dataset [14] which includes 100 object classes where each class varies the number of images. Few object classes contain less than 100 images whereas few of them carry more than 800 images. The change in a number of images make the CNN training process more complex, therefore we perform image augmentation based on a higher number of an object class. In this dataset, the higher images are 800 in one class; therefore by following this class, we equalize the other classes with the same number of images by using flipping operations. Mathematically, the performed flip operations are defined as follows:

Let we have an input image matrix of dimension 256×256 denoted by $\tilde{M}_{i,j}$ as shown in Fig. 2 of i th rows and j th columns, where $\tilde{M}_{i,j} \in \mathcal{R}^{i \times j}$. The rows $i = \{1, 2, \dots, \tilde{m}\}$ and columns $j = \{1, 2, \dots, \tilde{n}\}$ where number of channels are 3. The nature of input image is RGB that utilized for three different flip operations for augmentation.

$$\tilde{M}^T = \tilde{M}_{j,i}. \tag{1}$$

Where, \tilde{M}^T denotes the transpose of original image. After this operation, the indices of original image are updated.

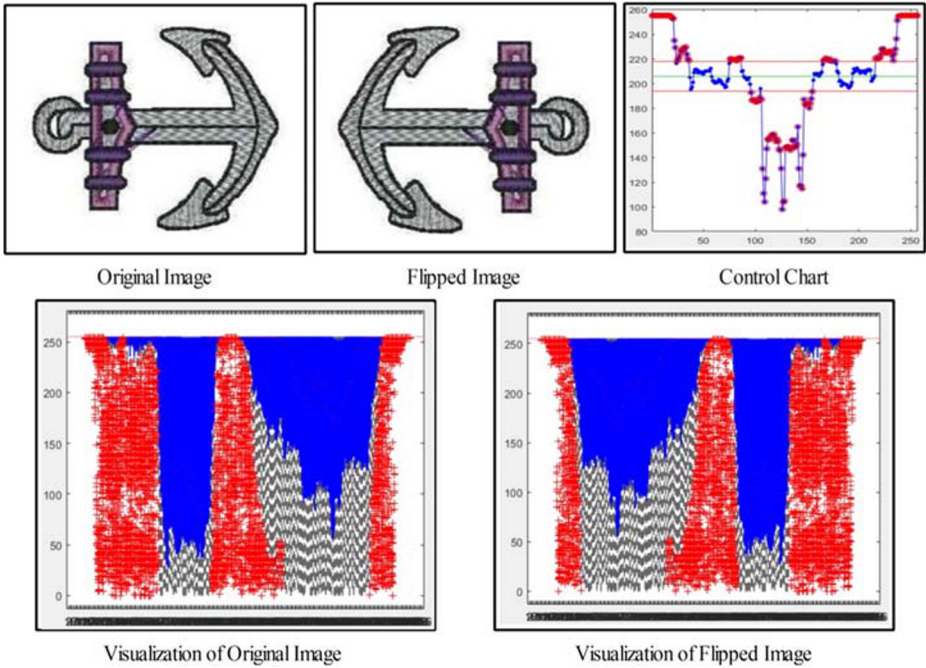


Fig. 2 An example of flipped operation on image $\tilde{M}_{i,j}$

$$\tilde{M}^H = \tilde{M}_i(\tilde{n} + 1 - j). \tag{2}$$

Where, \tilde{M}^H denotes the horizontal flip image.

$$\tilde{M}^V = \tilde{M}_{(\tilde{m} + 1 - i)}j. \tag{3}$$

Where, \tilde{M}^V denotes the vertical flip image. These three operations are performed until the lengths of images in each object class are equal to each other. An example of flipped image can be seen in Fig. 2. In this figure, it is shown that the image visualization is change after the flipped operation. However, it is also noticed that only places of i th and j th pixels are changed.

4.2 Features extraction

Feature extraction is a key step in pattern recognition for representation of an object in the image. The performance of any automated method is depends on the number of extracted features. The strong and relevant features give better accuracy but the redundant or noisy features vitiate the system performance. In this work, we extract two different type of features- Classical or well-known features and CNN based features. In the classical features, we computed Pyramid HOG [55] and Central symmetric LBP (CSLBP) [45] whereas using CNN, pre-trained model name inception V3 is utilized [46]. The detailed description of each feature type is defined below:

4.2.1 Classical features

Pyramid HOG features We have input image $\tilde{A}_{i,j}$ after data augmentation step where dimension of $\tilde{A}_{i,j}$ is $\tilde{m} \times \tilde{n}$ with i th rows and j th columns. The pixels range of image $\tilde{A}_{i,j}$ is 0 to 255. To compute the pyramids of input images, convert original image into gray and define three steps. In the first step, copy original image as shown in Fig. 3. In the second step, divide the original image into 2×2 layout, and in third step, further each layout of step 2 is divided into 2×2 , as shown in Fig. 3. This process gives total 21 layouts. From these 21 layouts, HOG features are extracted. HOG features are computed in five steps.

First of all, perform gamma correction to improve the contrast of image in terms of illumination and viewpoint change. The gamma correction is defined by the following expression.

$$\tilde{A}_{i,j} = \sqrt{\tilde{A}_{i,j}}. \tag{4}$$

Later, horizontal and vertical gradients are computed to further improve the weakened illumination properties by following mathematical expression:

$$\Delta_x(i, j) = \tilde{A}(i + 1, j) - \tilde{A}(i - 1, j). \tag{5}$$

$$\Delta_y(i, j) = \tilde{A}(i, j + 1) - \tilde{A}(i, j - 1). \tag{6}$$

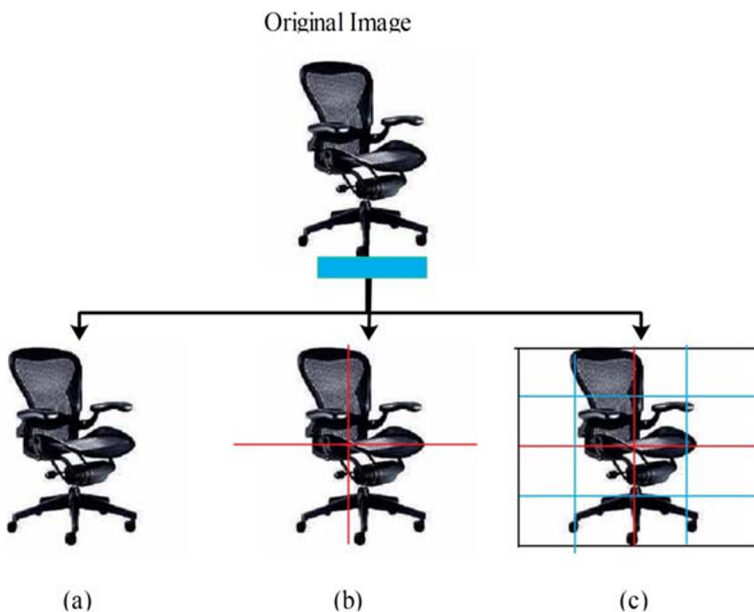


Fig. 3 Representation of PHOG features for object classification

By using these gradients, the magnitude and orientation is computed as:

$$\Delta(i, j) = \sqrt{\Delta_x(i, j)^2 + \Delta_y(i, j)^2}. \quad (7)$$

$$\theta(i, j) = \tan^{-1} \left(\frac{\Delta_y(i, j)}{\Delta_x(i, j)} \right). \quad (8)$$

By employing gradient and magnitude information, the supreme gradient value is selected. The selection of supreme gradient is defined through a following expression:

$$\overbrace{\Delta(i, j)} = \max_{c \in \{A\}} \{\Delta^c(i, j)\}. \quad (9)$$

Where, Δ^c represent the gradient magnitude from channel c . Later, cell quantization is performed based on neighborhood pixels where the size of number of neighbors is 8×8 . These cells are combined in the very next step which returns a feature vector. The resultant vector is normalized in the last step by L2-Norm.

$$\text{L2-Norm} : f = \frac{V}{\sqrt{\|V\|_2^2 + e^2}} \quad (10)$$

The resultant normalized PHOG vector is denoted by $\Delta_{PHOG}(N, f)$, where N denotes number of all testing images and f denotes extracted PHOG features.

Central symmetric LBP Secondly, central symmetric LBP features are extracted from gray images to handle the problem of illumination changes and simplify the complexity of original extracted LBP features. In original LBP features, the central pixel is compared with all other neighboring pixels whereas in CSLBP, only compare with equal spaced pixels. Mathematically, the original LBP features are computed as follows:

$$LBP_{r,n}(i, j) = \sum_{k=0}^{n-1} s(x_k - x_c) 2^k. \quad (11)$$

$$s(i) = \begin{cases} 1, & i \geq 0 \\ 0 & \text{Otherwise} \end{cases}. \quad (12)$$

Whereas, the CSLBP features are computed as:

$$CSLBP_{r,n}(i, j) = \sum_{k=0}^{(n/2)-1} s(x_k - x_{k+(n/2)}) 2^k. \quad (13)$$

$$s(i) = \begin{cases} 1, & i > T \\ 0 & \text{Otherwise} \end{cases}. \quad (14)$$

As compare to LBP, a CSLBP feature consumes less computational time. In LBP, 2^8 binary patterns are produces whereas in CSLBP, 2^4 binary patterns are produces for each window. The notation T works in this equation as a central pixel for generating binary patterns. Finally, the produced CSLBP and PHOG features are serially combined in one matrix as follows:

$$F_{Cls}(N, f) = \left(\frac{\Delta_{PHOG}(N, f)}{CSLBP_{r,n}(i, j)} \right)_{N \times \tilde{f}} \tag{15}$$

Where, \tilde{f} represent the length of combined classical features for each image and $F_{Cls}(N, f)$ depicts the fused vector.

4.2.2 CNN features

In CNN based feature extraction step, we utilized a pre-trained CNN model name Inception V3 [46]. Inception v3 has total 316 layers and 350 connection. Further, it includes total 94 convolutional layers. In this model several filters are applied on the same layer to extract deep features. Traditional CNN layers allow network to use certain size of filter for layers. Inception flexibility allows different size of filter and different number of parameters to be applied on same layer. In this model, a convolutional filter size is 1×1 to extract features. A simple Inception V3 model is shown in Fig. 4.

The Inception V3 model is initially trained in ImageNet database [11], therefore we copy the complete structure of Inception V3 by employing transfer learning concept and perform new training on modified augmented Caltech101 dataset. For this purpose, we divide the augmented dataset into 50:50 for training and testing. Later, train the Inception V3 on Caltech101 dataset using transfer learning. The cross-entropy loss function is utilized for feature extraction on avg_pool layer and obtained an resultant vector of dimension $N \times 2048$. After that these features are passed to JE-KNN selection method and best-selected features are fused along with handcrafted features as shown in Fig. 5. The detail of this Figure is given below section 4.3.

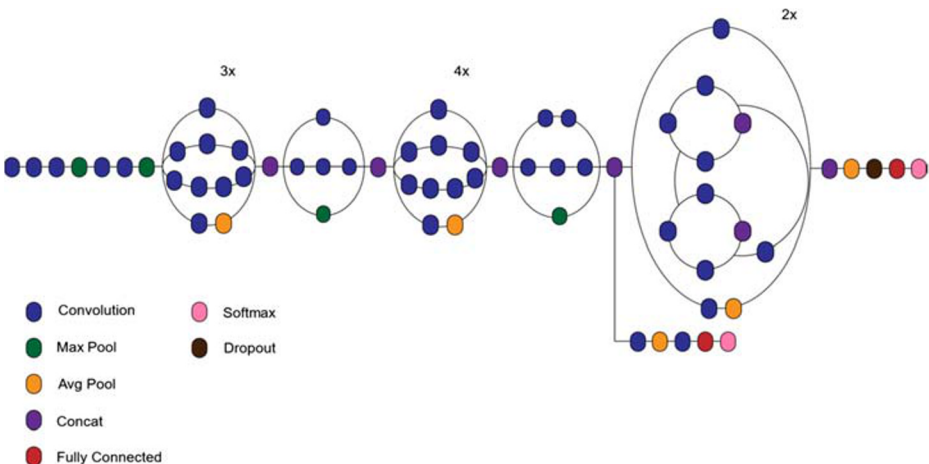


Fig. 4 Architecture of Inception V3 [46]

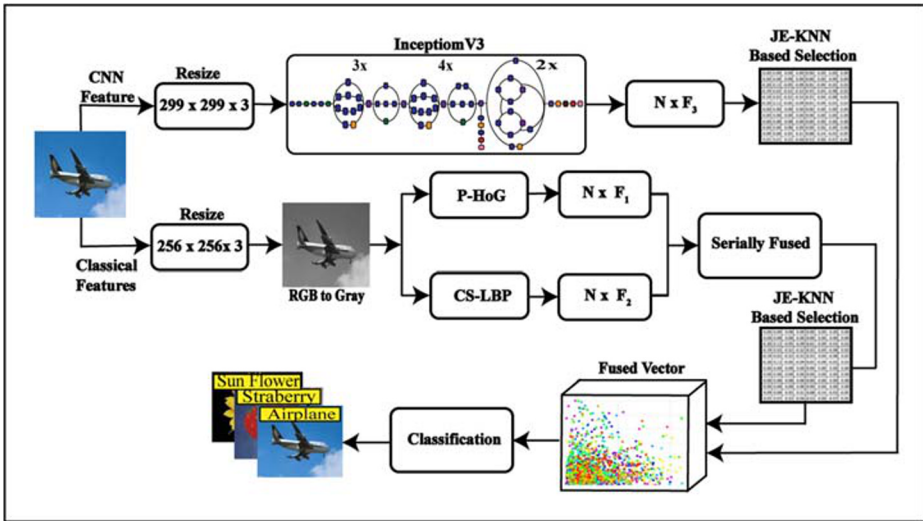


Fig. 5 Proposed classical and CNN feature fusion and reduction model for object classification

4.3 Feature selection

In the area of machine learning, feature selection is the process of obtaining the least number of strong features from original set with minimum data loss. Researchers try to find many algorithms that remove the problems of a huge amount of data into a few chunks. The high dimensional feature set increased algorithm memory, computational cost, and accommodation, significantly. For this purpose, an effective search algorithm is required that not only removes irrelevant feature but also handle the problem of redundant information. In this work, we presented a new selection method for irrelevant and redundant information reduction. A complete feature extraction and selection process is shown in Fig. 5. In this figure, the notation F_1 , F_2 , and F_3 denotes the extracted feature matrix of P-HOG descriptors, CS-LBP, and Inception V3 deep CNN. The notation N denotes the total number of images utilized for training and testing. Later on, P-HOG and CS-LBP features are serially combined and passed to JE-KNN based selection method. On the same time, the deep features extracted from Inception V3 are passed to JE-KNN. The features selected by this method are fused and perform classification. The detailed of each step is given below.

As we have two extracted feature vectors name classical vector denoted by $F_{Cls}(N, f)$ and CNN vector denoted by $F_{Cnn}(N, f)$ where N represent number of testing samples and f represent extracted number of features. The f is defined as- $f = (1, 2, \dots, nth)$. Then, we implement new feature selection method name Joint Entropy along with KNN (JE-KNN). Three-step process is follows in this method. In the very first step, the required weights are initialized where original input features are set as weights. In second step, Joint Entropy (JE) is implemented on original vector and produces a new vector which sorted into relevant and irrelevant features by employing a threshold function. In the last step, a threshold function is employed on JE obtained vector and provides to KNN classifier for loss calculation. This process is continues until, the required error rate is meet. Mathematical formulation of JE-KNN is expressed as follows:

Suppose we have extracted feature vector denoted by \tilde{f} and lengths of column in each feature vector denoted by \tilde{c} where the extracted vectors are $F_{Cls}(N, f)$ and $F_{Cm}(N, f)$, respectively. Then, the joint distribution among \tilde{f} and \tilde{c} is represent as $(\tilde{f}, \tilde{c}) \in (\tilde{f}_i, \tilde{c}_i)$ with probability distribution is $p(\tilde{f}_i, \tilde{c}_i)$. Hence, entropy $H(\tilde{f}, \tilde{c})$ is formulated as:

$$H(\tilde{f}, \tilde{c}) = \sum_{\tilde{f}_i, \tilde{c}_i} p(\tilde{f}_i, \tilde{c}_i) \log \frac{1}{p(\tilde{f}_i, \tilde{c}_i)}. \tag{16}$$

$$= \sum_{\tilde{f}_i, \tilde{c}_i} p(\tilde{f}_i) p(\tilde{c}_i | \tilde{f}_i) \log \frac{1}{p(\tilde{f}_i)} + \sum_{\tilde{f}_i, \tilde{c}_i} p(\tilde{f}_i) p(\tilde{c}_i | \tilde{f}_i) \log \frac{1}{p(\tilde{f}_i, \tilde{c}_i)}. \tag{17}$$

$$= \sum_{\tilde{f}_i} p(\tilde{f}_i) \log \frac{1}{p(\tilde{f}_i)} \sum_{\tilde{c}_i} p(\tilde{c}_i | \tilde{f}_i) + \sum_{\tilde{f}_i, \tilde{c}_i} p(\tilde{f}_i) p(\tilde{c}_i | \tilde{f}_i) \log \frac{1}{p(\tilde{c}_i | \tilde{f}_i)}. \tag{18}$$

$$= H(\tilde{f}) + \sum_{\tilde{f}_i} p(\tilde{f}_i) H(\tilde{c} | \tilde{f} = \tilde{f}_i). \tag{19}$$

$$= H(\tilde{f}) + {}^E [H(\tilde{c} | \tilde{f} = \tilde{f}_i)]. \tag{20}$$

$$H(\tilde{f}, \tilde{c}) = H(\tilde{f}) + H(\tilde{c} | \tilde{f}). \tag{21}$$

A threshold function is defined on $H(\tilde{f}, \tilde{c})$ based on average value of resultant JE matrix. Through this function, those features are selected that are equal or higher than average value feature. This process is continued 50 times iterations and each time, computes the performance using KNN classifier. After 50 times, select the best accuracy features as a final selection. A Matlab function name FitKNN [43] is utilized for this purpose along with 10 fold validation. In KNN, Euclidean Distance is employed which return accuracy and error rate. Based on the error rate, we decide the best-selected vector.

$$error = \sum_{i=1}^n \tilde{f}_i \tilde{c}_i \{ \tilde{a}_i \neq a_i \}. \tag{22}$$

Finally, the best accuracy and minimum error rate based selected vector is provides to multiple classifiers such as linear discriminant, SVM, Ensemble tree, and cosine KNN [40]. Best on the higher accuracy, a best classifier is selected. The proposed experimental results are presented in the detailed in below section.

5 Experimental setup and results

The proposed method is validated on publically available dataset name Caltech-101 [14]. This dataset consist of total 9144 RGB and gray images of 101 unique object classes, few of them are shown in Fig. 6. Each class consists of different number of images ranging from 31 to 800. Due to both RGB and gray images, makes it challenging and difficult to perform object classification. We utilized Intel Core i7 8th generation CPU equipped with 16 GB of RAM and 8 GB GPU. All simulations are performed on MATLAB 2018a.

In the experimental process, 50:50 approach is utilized along with 10 fold cross-validation. Later, we utilized multiple classifiers and select the best one based on the high-performance rate. The performance is calculated through following measures like accuracy, computational time and False Negative Rate (FNR).

5.1 Results

As mentioned above that Caltech101 dataset is utilized in this work for experimental process, therefore we split this dataset in four groups. In the first group, we select first 25 objects classes and perform classification, then 50, 75, and all. A brief description of this process is given in Table 1. In this table, it is described that 3 experiments are performed to analyze the performance of proposed system. The main reason behind these experiments is check the efficiency, scalability and change in accuracy of the proposed system after fusion and selection of feature process.

5.1.1 Experiment 1

In this experiment, the classical features such as PHOG and Central symmetric LBP (CSLBP) are fused and perform propose selection method. The results of this process are evaluated on different number of classes as presented in Table 2. In this table, multiple classifiers are utilized such as LDA, ESD etc. In this experiment, the best classification accuracy for first 25 classes is 47.3% along with error rate of 52.7% on ESD classifier whereas on other classifiers such as



Fig. 6 Sample images from Caltech-101 dataset

Table 1 Number of performed experiments for classification results on Caltech101 dataset

Experiment	Features	Number of classes	Cross-validation	Training/ Testing
Experiment 1	Fusion and selection of only classical features using proposed method	First 25 First 50 First 75 All (100)	10 Fold	50%/ 50%
Experiment 2	Fusion of CNN and classical features	First 25 First 50 First 75 All (100)	10 Fold	50%/ 50%
Experiment 3	Proposed approach	First 25 First 50 First 75 All (100)	10 Fold	50%/ 50%

LDA, L-SVM, and Co-KNN, accuracy is 41.4%, 41.2%, and 43.3%, respectively. On top 50 object classes, the best accuracy is 38% on ESD whereas the minimum is 28.4% on LDA. Further, increases in the object classes, the best accuracy is degraded and reached to 33.7% on LSVM. After all 100 classes, the best-noted accuracy is 30.2% on Co-KNN whereas the worst accuracy is 22.6% on LDA. From, the results, it is noted that, the accuracy on classical features is degraded when the number of object classes are increases. In addition, the testing classification time of each classifier against selected number of classes is also noted, shown in Fig. 7. In this figure, it is shown that the less number of object classes (25 numbers of classes) execute with minimum time whereas on all object classes, the computation time is high as compare to all others. Moreover, it is also noted that using classical features the accuracy of the system is decreases and time is increases when more number of classes are added such as 25 to 50 to 75 to all (100).

Table 2 Fusion and selection of only classical features using proposed method

Method	No. of Classes				Performance Measures		
	25	50	75	100	Accuracy (%)	FNR (%)	Time (S)
1 LDA	✓				41.4	58.6	39.621
		✓			28.4	71.6	51.149
			✓		26.0	74.0	58.905
				✓	22.6	77.4	75.822
2 L-SVM	✓				41.2	59.8	38.092
		✓			34.6	65.4	971.18
			✓		33.7	66.3	998.88
				✓	30.1	69.9	640.6
3 ESD	✓				47.3	52.7	123.09
		✓			38.0	62.0	342.47
			✓		24.4	76.4	807.09
				✓	23.8	76.2	1002.98
4 Co-KNN	✓				43.3	56.7	32.349
		✓			33.9	59.3	36.529
			✓		33.7	66.3	44.238
				✓	30.2	69.8	49.632

The bold values indicates improves results

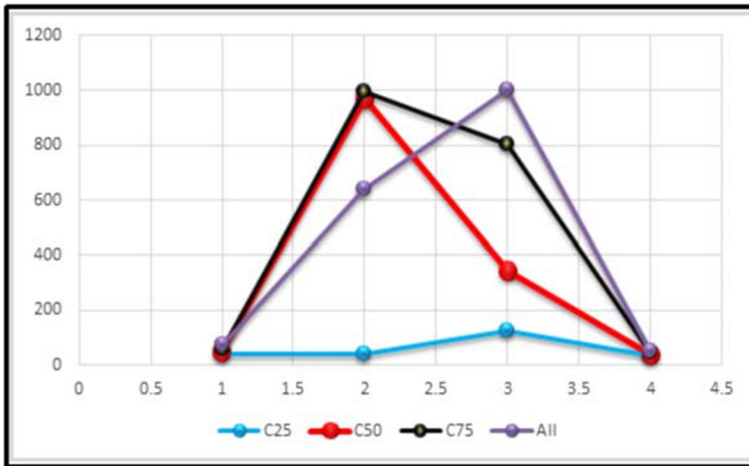


Fig. 7 Classification computation time of each classifier using classical features on different number of object classes

5.1.2 Experiment 2

In this experiment, the CNN features are fused along with classical features such as PHOG and Central symmetric LBP (CSLBP). The selection process is not performed on the fused vector in this experiment to analyze the effectiveness of proposed selection process. The results of this process are evaluated on different number of classes as presented in Table 3. In this table, multiple classifiers are utilized such as LDA, ESD etc. for classification results. The best-obtained classification accuracy for first 25 classes is 92.5% along with error rate of 7.5% on LDA classifier whereas on other classifiers such as L-SVM, ESD and Co-KNN, accuracy 88.5%, 92.2%, and 89.9%, respectively. On top 50 object classes, the best-obtained accuracy is 91.5% on LDA classifier whereas on other classifiers such as L-SVM, ESD and Co-KNN, the obtained accuracy is 88.3%, 91.4%, and 87.5%, respectively. It is noted that the fusion process maintains the accuracy after addition of more number of classes as compare to classical features. After all 100 classes, the best-noted accuracy is 87.3% on ESD classifier whereas the worst accuracy is 83.7% on Co-KNN. Overall, the results are improved after fusion process but on the end, the classification time is almost double. The classification time is also plotted in Fig. 8. In this figure, it is noted that after fusion process, the time is almost double as compared to classical features. Moreover, we also noted that the addition of more number of object classes little bit decreases the classification accuracy but on the other end, the classification time is high.

5.1.3 Proposed feature selection

In this experiment, the proposed feature selection method is employed on fused feature vector (CNN and Classical features). The best features are selected through Joint Entropy along with KNN fitness function. The selected features are classified through multiple classifiers and numerical results are presented in Table 4. In this table, the results are presented for different number of selected classes against each classifier which are listed in Table 4. The proposed selection process results are increased as compare to Tables 2 and 3. In this experiment, the

Table 3 Fusion of CNN and classical features

Method	No. of classes				Performance measures		
	25	50	75	100	Accuracy (%)	FNR (%)	Time (sec)
1- LDA	✓				92.5	7.5	139.06
		✓			91.5	8.5	175.25
			✓		88.3	11.7	83.381
				✓	86.3	13.7	571.30
2- L-SVM	✓				88.5	11.5	82.61
		✓			88.3	11.7	1606.00
			✓		87.0	13	2274.70
				✓	84.2	15.8	17,320
3- ESD	✓				92.2	7.8	444.29
		✓			91.4	8.6	1405.30
			✓		82.3	17.7	1045.70
				✓	87.3	12.7	2143.80
4- Co-KNN	✓				89.9	10.1	83.973
		✓			87.5	12.5	123.46
			✓		84.4	15.6	59.695
				✓	83.7	16.3	476.81

The bold values indicates improves results

best-achieved accuracy of 25 number of classes is 93.9% which is previously 92.5% (in Table 3). This best accuracy is obtained on LDA classifier with an error rate is 6.1%, can also be shown in Fig. 9 (confusion matrix). Secondly, the classification is performed on 50 numbers of object classes and achieved best accuracy of 92.6% which is previously 91.5% (in Table 3). This accuracy is obtained on LDA and also verified through Fig. 9 (50 classes). After that, classes are increases up to 75 and results are little bit diminish. The achieved accuracy on 75 numbers of objects classes is 90.4% with an error rate of 9.6%, can be validated through confusion matrix shown in Fig. 9 (75 Classes). This accuracy is higher as compare to previous achieve performance on fused vector 87.0% (Experiment 2). In the last, all object classes are consider for classification and achieve an accuracy of 90.1% which is best as compare to both

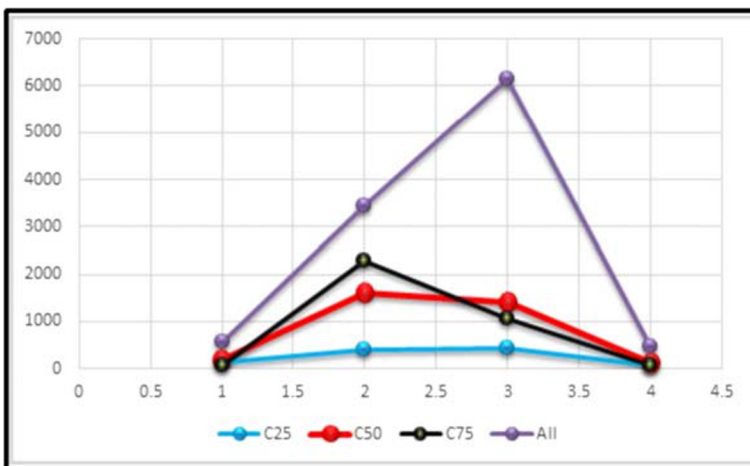


Fig. 8 Classification computation time of each classifier after fusion of CNN and classical features on different number of object classes

previous experiments. The achieved best accuracy is also verified through Fig. 9 (100 Classes). The classification time for all classifiers is also noted, in Fig. 10. In this figure, it is show that the proposed method is perform efficient on Caltech101 dataset. Moreover, the overall accuracy is also improved after employing proposed selection method.

5.2 Discussion

The brief description of proposed results, scalability of proposed method after addition of more number of object classes, effect of classification time due to number of features & object classes, and comparison with existing techniques based on accuracy are discussed in this section. As presented in Table 2-4 that three different experiments are performed based on Table 1. In the first experiment, only classical features are fused and achieved maximum accuracy of 30% on complete dataset. In the second experiment, the fused CNN and classical features without selection method and attain accuracy of 87.3% which is significantly improved after addition of CNN features. The fusion results show the worth of CNN for objects classification. In the last experiment, the proposed selection method is applied and achieved an accuracy of 90.1% with more efficiency. The proposed selection method increases the classification accuracy and reduced the computation time during the classification process.

Scalability is an important factor of any proposed algorithm. Our proposed method maintains the accuracy when more number of object classes are added which is clearly depicts from Table 2-4. But the classification time is increased due to addition of more number of classes. The classification time for each experiment is plotted in Figs. 7, 8, and 10 which show that the proposed selection method require less time for execution as compare to original classical and fused vector.

A detailed statistical analysis is also conducted and presented in Table 5. In this table, it is illustrated that the minimum, maximum, and average values are calculated for each classifier and then σ and CI are computed. Based on CI, the best results are achieved on LDA classifier of maximum accuracy of 90.1%, $\sigma = 0.3681$ and CI is 0.2125. The CI of ESD classifier is also

Table 4 Classification results of Caltech-101 using proposed method

Method	No. of Classes				Performance Measures		
	25	50	75	100	Accuracy (%)	FNR (%)	Time (sec)
1-LDA	✓				93.9	6.1	11.315
		✓			92.6	7.4	17.749
			✓		89.8	10.2	42.796
				✓	90.1	9.9	69.815
2-Linear SVM	✓				76.9	23.1	77.592
		✓			72.4	27.6	435.228
			✓		76.3	23.7	624.5
				✓	72.0	28	969.7
3-ESD	✓				89.2	10.8	77.546
		✓			91.5	8.5	335.6
			✓		90.4	9.6	581.7
				✓	89.4	11.6	877.8
4- Co-KNN	✓				76.6	23.4	19.813
		✓			73.6	26.4	10.721
			✓		70.1	29.9	17.231
				✓	69.1	29.9	26.686

The bold values indicates improves results

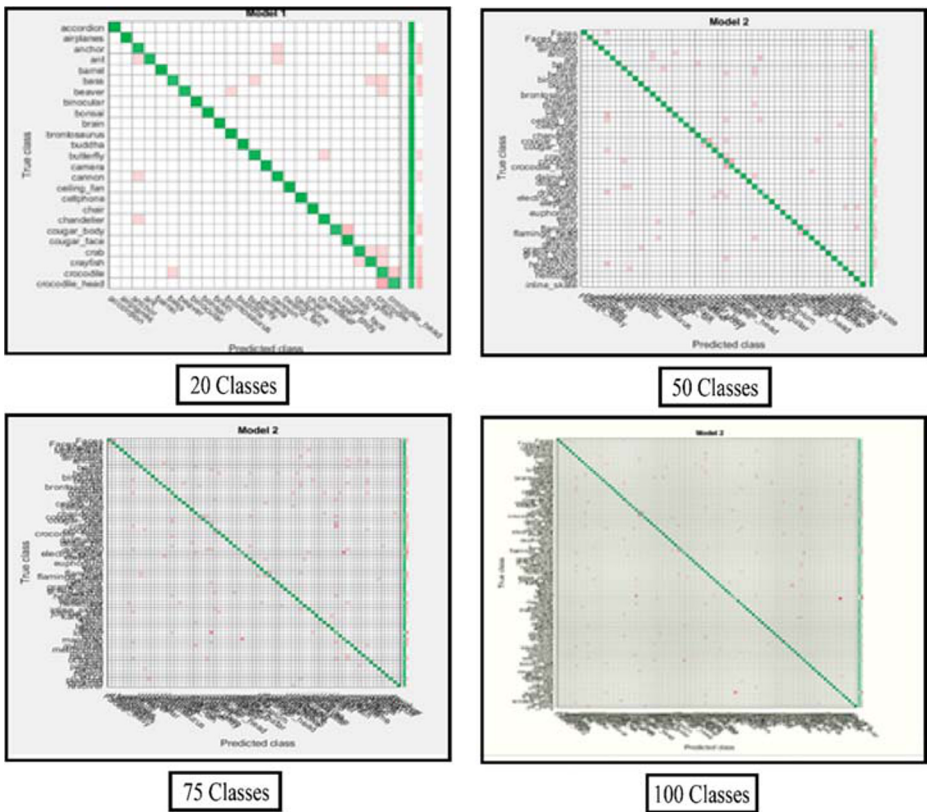


Fig. 9 Confusion of matrix for proposed method results

plotted in Fig. 11 which described that on 95% confidence level the CI is 89.633 ± 0.417 ($\pm 0.46\%$). The overall CI for LDA classifier is noted of 0.2125. From the results, it is clear that the proposed results are consistent after several numbers of iterations.

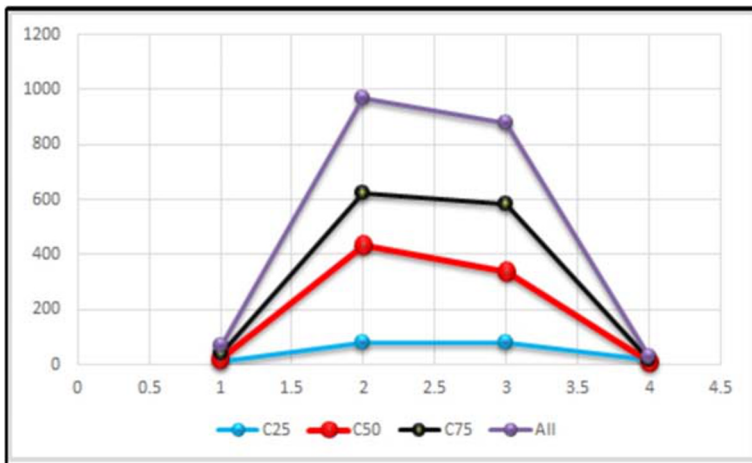


Fig. 10 Classification time after employing proposed method

Table 5 Statistical analysis of proposed feature selection method using Caltech 101 dataset

Method	Parameters				
	Min (%)	Avg (%)	Max (%)	Standard deviation	Confidence interval
LDA	89.2	89.6	90.1	0.3681	0.2125
LSVM	69.3	70.6	72.0	1.1025	0.6365
ESD	88.4	89.0	89.4	0.4109	0.2372
Co-KNN	66.7	67.9	69.1	0.9797	0.5656

The bold values indicates improves results

In the last, a fair comparison is conducted in term of accuracy and classification time with existing techniques, as presented in Table 5. In this, Song et al. [42] presented a PCA based feature selection technique along with SVM classifier and achieved an accuracy of 83.9% on Caltech-101 dataset. Li et al. [17] performed extreme learning and YCbCr color transformation for object classification and achieved 78% accuracy on Caltech-101 dataset. Pan et al. [30] employed K-Means clustering-based technique for feature reduction and achieved classification accuracy of 85.78%. The more recent, Rashid et al. [33] fused CNN and SIFT features and obtained classification accuracy of 89.7% on Caltech101 dataset. However, our method achieved accuracy of 93.9% on 25 objects classes, 92.6% on 50 objects classes, 90.4% for 75 classes and overall achieved 90% for complete Caltech101 dataset. Moreover, propose method is also outperforms in the form of computational time (Table 6).

5.3 Critical analysis

Based on the critical analysis of each step involves in the proposed method, it is described that a huge change is occurs in classification results. The augmentation is a key step in this regard and clearly shows the results in Table 7. In this table, it is observed that the classification accuracy is change after augmentation. Initially, we calculate the results on original Caltech101 dataset and attained accuracy of 80.40%. After horizontal flip for increase in data, the accuracy is increased more than 3%. Further, vertical and transpose operations are performed and accuracy is reached to 90.10%. It is clearly show that the increases in the images of each class train a good model that later gives improved classification accuracy. Further, we test the proposed method on different training/testing ratios; results can be seen in Fig. 12. In this









Confidence Level	Margin of Error	Error Bar
68.3%, $\sigma_{\bar{x}}$	89.6333 \pm 0.213 (\pm 0.24%)	
90%, $1.645\sigma_{\bar{x}}$	89.6333 \pm 0.35 (\pm 0.39%)	
95%, $1.960\sigma_{\bar{x}}$	89.6333 \pm 0.417 (\pm 0.46%)	
99%, $2.576\sigma_{\bar{x}}$	89.6333 \pm 0.548 (\pm 0.61%)	
99.9%, $3.291\sigma_{\bar{x}}$	89.6333 \pm 0.7 (\pm 0.78%)	
99.99%, $3.891\sigma_{\bar{x}}$	89.6333 \pm 0.827 (\pm 0.92%)	
99.999%, $4.417\sigma_{\bar{x}}$	89.6333 \pm 0.939 (\pm 1.05%)	
99.9999%, $4.892\sigma_{\bar{x}}$	89.6333 \pm 1.04 (\pm 1.16%)	

Fig. 11 Representation of confidence interval at different confidence levels for LDA classifier

Table 6 Proposed results comparison with existing techniques

Paper	Year	Method	Features	Accuracy (%)	Times (s)
Song et al. [42]	2018	PCA	SIFT	83.9	
Li et al. [17]	2018	EL + YCbCr	YCbCr	78	
Pan et al. [30]	2018	K-Mean Reduction	SIFT	85.78	
Rashid et al. [33]	2019	Entropy Based Selection and Deep fusion	Deep CNN and SIFT	89.7	302
Proposed		Joint Entropy along with KNN Fitness Function	Deep CNN and Handcrafted	25 classes: 93.9	11.35
				50 classes: 92.6	17.74
				75 classes: 90.4	581.7
				100 classes: 90.1	69.81

The bold values indicates improves results

Figure, it is show that the higher training ratio improves the proposed accuracy but it clashes the fair comparison. Hence we consider a ratio (50:50) in proposed work.

6 Conclusion

In conclusion, we propose an automated system for object classification using classical and deep features selection. Data augmentation is performed to handle the problem of sufficient training. Then classical features are computed from gray images for the cause of local properties of objects. Later CNN features are computed and combined along with classical features. In the next stage, we get the benefit of best-selected features that are obtained by JE-KNN based method and achieve tremendous accuracy. The best-selected feature method also gives support in reducing the overall computational time. Overall, the proposed method accomplished an accuracy of 90.1% on Caltech101 dataset. The comparison is conducted with recent techniques that show the authenticity of the presented method. However, during the analysis of proposed results we observed that proposed method increases the error rate for few classifiers. As compared to ESD classifier, the difference among accuracy of SVM and Co-KNN is almost 18% which is a huge difference and it is a main limitation of our work. This problem can be resolved through the selection of classifiers such as Softmax, ELM, and Naïve Bayes. In the future, deep reinforcement learning is employed to achieve better accuracy on this dataset. Moreover, a more efficient feature selection method will be proposed and apply to the same system. Furthermore, Caltech256 dataset will be used in the future studies related to object classification.

Table 7 Change in classification results after data augmentation step

Method	Flip Operations				Measures	
	Original Database	Horizontal Flip	Horizontal + Vertical Flip	Horizontal + Vertical +Transpose	Accuracy (%)	Error Rate (%)
LDA	✓				80.40	10.60
		✓			83.19	16.81
			✓		87.45	12.55
				✓	90.10	9.9

The bold values indicates improves results

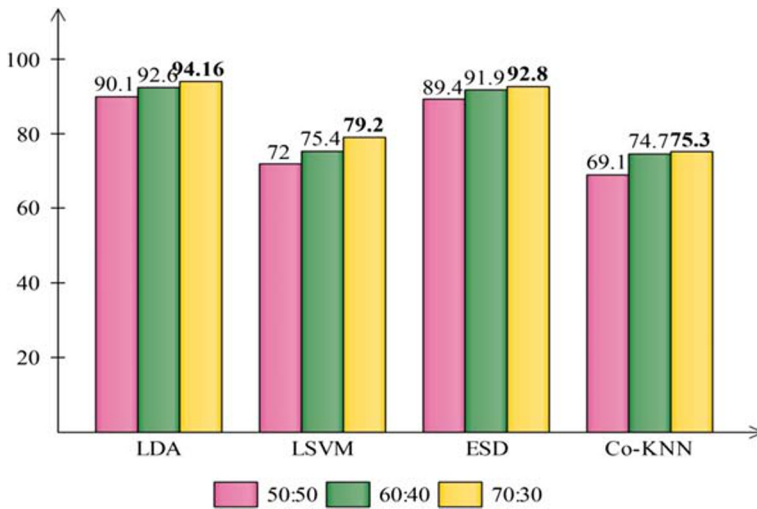


Fig. 12 Analysis of results on different training/testing ratios

Appendix

Table 8 Summary of existing techniques

Author	Feature	Method	Dataset
Hamza et al. [13]	Shape Aware bag of features	CNN	SHREC-2015
Weibel et al. [53]	Point Pair Features	Graph-based Convolution	Stanford
Kaur et al. [16]	Multimodal Blur features	Feature-based CNN	LabelMe, MIT-Indoor dataset
Gill et al. [31]	SIFT, SURF, and Tamura features	Supervised Learning	
Liu et al. [23]	Latent Features	Deep CNN	Caltech-101
Rani et al. [32]	Binary Features	PCA	Pascal VOC2012
Mafarja et al. [24]	Hog, Surf, Sift, Sift	GWO and WOA	Breast cancer

References

1. Adeel A, Khan MA, Sharif M, Azam F, Umer T, and Wan S (2019) Diagnosis and recognition of grape leaf diseases: an automated system based on a novel saliency approach and canonical correlation analysis based multiple features fusion. *Sustainable Computing: Informatics and Systems*
2. Arshad H, Khan MA, Sharif M, Yasmin M, Javed MYJIJOML, and Cybernetics (2019) Multi-level features fusion and selection for human gait recognition: an optimized framework of Bayesian model and binomial distribution," pp. 1–18
3. Arshad H, Khan MA, Sharif MI, Yasmin M, Tavares JMRS, Zhang Y-D and Satapathy SC (2020) "A multilevel paradigm for deep convolutional neural network features selection with an application to human gait recognition." *Expert Systems: e12541*
4. Bilal M and Hanif MSJJoSPS (2019) High performance real-time pedestrian detection using light weight features and fast cascaded kernel SVM classification, vol. 91, pp. 117–129

5. Cao X, Wu C, Yan P, and Li X (2011) Linear SVM classification using boosting HOG features for vehicle detection in low-altitude airborne videos, in 2011 18th IEEE International Conference on Image Processing, pp. 2421–2424.
6. Cao H, Du H, Zhang S, and Cai S (2020) InSphereNet: a concise representation and classification method for 3D object, in International Conference on Multimedia Modeling, pp. 327–339.
7. Chaudhuri DR, Chandra D, and Mittal A (2020) Indoor object classification using higher dimensional MPEG features," in Soft Computing for Problem Solving, ed: Springer, pp. 573–583.
8. Cubuk ED, Zoph B, Shlens J, and Le QV (2019) RandAugment: Practical data augmentation with no separate search," arXiv preprint arXiv:1909.13719
9. Cui Y, Xu H, Wu J, Sun Y, and Zhao JIIS (2019) Automatic vehicle tracking with roadside LiDAR data for the connected-vehicles system
10. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection
11. Deng J, Dong W, Socher R, Li L-J, Li K, and Fei-Fei L (2009) Imagenet: A large-scale hierarchical image database, in 2009 IEEE conference on computer vision and pattern recognition, pp. 248–255.
12. Fei-Fei L, Perona P (2005) A bayesian hierarchical model for learning natural scene categories, in 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), pp. 524–531.
13. Ghodrati H, Luciano L, Hamza ABJNPL (2019) Convolutional shape-aware representation for 3D object classification, vol. 49, pp. 797–817
14. Griffin G, Holub A, Perona P (2007) Caltech-256 object category dataset
15. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.
16. Kaur B, Bhattacharya JJESwA (2019) A convolutional feature map-based deep network targeted towards traffic detection and classification, vol. 124, pp. 119–129
17. Khan MA, Akram T, Sharif M, Awais M, Javed K, Ali H et al (2018) CCDF: automatic system for segmentation and recognition of fruit crops diseases based on correlation coefficient and deep CNN features. *Comput Electron Agric* 155:220–236
18. Khan MA, Khan MA, Ahmed F, Mittal M, Goyal LM, Hemanth DJ et al (2020) Gastrointestinal diseases segmentation and classification based on duo-deep architectures. *Pattern Recognition Letters* 131:193–204
19. Khan MA, Javed K, Khan SA, Saba T, Habib U, Khan JA et al (2020) "Human action recognition using fusion of multiview and deep features: an application to video surveillance." *Multimedia Tools and Applications* 1–27
20. Khan MA, Rubab S, Kashif A, Sharif MI, Muhammad N, Shah JH et al (2020) Lungs cancer classification from CT images: An integrated design of contrast based classical features fusion and selection. *Pattern Recognition Letters* 129:77–85
21. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks, in *Advances in neural information processing systems*, pp. 1097–1105.
22. Kumar B, Pandey G, Lohani B, Misra SCJJjop, and R. Sensing (2019) A multi-faceted CNN architecture for automatic classification of mobile LiDAR data and an algorithm to reproduce point cloud samples for enhanced training, vol. 147, pp. 80–89
23. Liu X, Zhang R, Meng Z, Hong R, and Liu GJWWW (2019) On fusing the latent deep CNN feature for image classification, vol. 22, pp. 423–436
24. Mafarja M, Qasem A, Heidari AA, Aljarah I, Faris H, and Mirjalili SJCC (2019) Efficient hybrid nature-inspired binary optimizers for feature selection, pp. 1–26
25. Majid A, Khan MA, Yasmin M, Rehman A, Yousafzai A and Tariq U (2020) Classification of stomach infections: A paradigm of convolutional neural network along with classical features fusion and selection. *Microscopy Research and Technique*
26. Mirjalili S (2019) "Genetic algorithm," in *Evolutionary Algorithms and Neural Networks*, ed: Springer, pp. 43–55.
27. Na B, Fox G (2019) Object classification by a super-resolution method and a convolutional neural networks. *International Journal of Data Mining Science* 1:16–23
28. Najafabadi MM, Villanustre F, Khoshgoftaar TM, Seliya N, Wald R, Muharemagic E (2015) Deep learning applications and challenges in big data analytics. *Journal of Big Data* 2:1
29. Neumann J, Schnörr C, Steidl G (2005) Combined SVM-based feature selection and classification. *Mach Learn* 61:129–150
30. Pan Y, Xia Y, Song Y, Cai WJMT, and Applications (2018) Locality constrained encoding of frequency and spatial information for image classification, vol. 77, pp. 24891–24907
31. Quattoni A, Torralba A (2009) Recognizing indoor scenes," in 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 413–420.
32. R. Rani, A. P. Singh, R. J. M. T. Kumar, and Applications (2019) Impact of reduction in descriptor size on object detection and classification," vol. 78, pp. 8965–8979

33. Rashid M, Khan MA, Sharif M, Raza M, Sarfraz MM, Afza FJMT, et al. (2019) Object detection and classification: a joint selection and fusion strategy of deep convolutional neural network and SIFT point features, vol. 78, pp. 15751–15777
34. Ravikumar S, Ramachandran K, Sugumaran V (2011) Machine learning approach for automated visual inspection of machine components. *Expert Syst Appl* 38:3260–3266
35. Rehman A, Khan MA, Mehmood Z, Saba T, Sardaraz M and Rashid M (2020) Microscopic melanoma detection and classification: A framework of pixel-based fusion and multilevel features reduction. *Microscopy Research and Technique* 83 (4):410–423
36. Rish I (2001) An empirical study of the naive Bayes classifier, in *IJCAI 2001 workshop on empirical methods in artificial intelligence*, pp. 41–46.
37. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. (2015) Imagenet large scale visual recognition challenge, vol. 115, pp. 211–252
38. Saba T, Khan MA, Rehman A, Marie-Sainte SL (2019) Region Extraction and Classification of Skin Cancer: A Heterogeneous framework of Deep CNN Features Fusion and Reduction. *Journal of Medical Systems* 43(9)
39. Sayed GI, Hassanien AE, Azar ATJNC, and Applications (2019) Feature selection via a novel chaotic crow search algorithm, vol. 31, pp. 171–188
40. Shaheen M, Zafar T, and Ali Khan S (2019) Decision tree classification: Ranking journals using IGIDI," *Journal of Information Science*, p. 0165551519837176
41. Sharif M, Khan MA, Tahir MZ, Yasmim M, Saba T and Tanik UJ (2020) "A Machine Learning Method with Threshold Based Parallel Feature Fusion and Feature Selection for Automated Gait Recognition." *Journal of Organizational and End User Computing (JOEUC)* 32(2):67–92
42. Song J , Yoon G, Cho H, Yoon SMJMT, and Applications (2018) Structure preserving dimensionality reduction for visual object recognition, vol. 77, pp. 23529–23545
43. Soucy P, Mineau GW (2001) A simple KNN algorithm for text categorization," in *Proceedings 2001 IEEE International Conference on Data Mining*, pp. 647–648.
44. Srivastava S, Priyadarshini J, Gopal S, Gupta S, and Dayal HS (2019) Optical character recognition on bank cheques using 2D convolution neural network, in *Applications of Artificial Intelligence Techniques in Engineering*, ed: Springer, pp. 589–596.
45. Sun H, Wang C, Wang B, El-Sheimy N (2011) Pyramid binary pattern features for real-time pedestrian detection from infrared videos. *Neurocomputing* 74:797–804
46. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2016) Rethinking the inception architecture for computer vision, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826.
47. Tallón-Ballesteros AJ, Caviqúe L, and Fong S (2019) Addressing low dimensionality feature subset selection: reliefF (–k) or extended correlation-based feature selection (eCFS)?, in *International Workshop on Soft Computing Models in Industrial and Environmental Applications*, pp. 251–260.
48. Tan B, Salakhutdinov R, Mitchell T, and Xing E (2019) Learning data manipulation for augmentation and weighting
49. Tilahun SL, Ngnotchouye JMT, Hamadneh NNJAIR (2019) Continuous versions of firefly algorithm: A review, vol. 51, pp. 445–492
50. Wang Y, Chen Y, Yang N, Zheng L, Dey N, Ashour AS et al (2019) Classification of mice hepatic granuloma microscopic images based on a deep convolutional neural network. *Appl Soft Comput* 74:40–50
51. Wang X, Zhang W, Wu X, Xiao L, Qian Y, and Fang ZJJor-TIP (2019) Real-time vehicle type classification with deep convolutional neural networks, vol. 16, pp. 5–14
52. Wei G, Cao H, Ma H, Qi S, Qian W, Ma ZJJoms (2018) Content-based image retrieval for lung nodule classification using texture features and learned distance metric, vol. 42, p. 13
53. Weibel J-B, Patten T, Vincze M (2019) Robust 3D object classification by combining point pair features and graph convolution, in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 7262–7268.
54. Wu K, Zhang D, Lu G, Guo ZJPR (2019) Joint learning for voice based disease detection vol. 87, pp. 130–139
55. Wu J, Shang Z, Wang K, Zhai J, Wang Y, Xia F, et al. (2019) Partially occluded head posture estimation for 2D images using pyramid HoG features, in *2019 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pp. 507–512
56. Xiong F, Xiao Y, Cao Z, Gong K, Fang Z, and Zhou JT (2019) Good practices on building effective CNN baseline model for person re-identification, in *Tenth International Conference on Graphics and Image Processing (ICGIP 2018)*, p. 110690I.
57. Zhi S, Liu Y, Li X, Guo YJC, and Graphics (2018) Toward real-time 3D object recognition: A lightweight volumetric CNN framework using multitask learning," vol. 71, pp. 199–207

58. Zhu Y, Shi J, Wu X, Liu X, Zeng G, Sun J et al (2020) Photon-limited non-imaging object detection and classification based on single-pixel imaging system. *Applied Physics B* 126:21

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

Affiliations

Nazar Hussain¹ · Muhammad Attique Khan² · Muhammad Sharif¹ · Sajid Ali Khan³ · Abdulaziz A. Albeshir⁴ · Tanzila Saba⁵ · Ammar Armaghan⁶

¹ Department of Computer Science, COMSATS University Islamabad, Wah Campus, Pakistan

² Department of Computer Science, HITEC University, Museum Road, Taxila, Pakistan

³ Department of Software engineering, Foundation University Islamabad, Islamabad, Pakistan

⁴ College of Computing and Informatics, Saudi Electronic University, Riyadh, Saudi Arabia

⁵ College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia

⁶ Department of Electrical Engineering, College of Engineering Jouf University, Sakaka, Saudi Arabia