IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 26, 2021, accepted March 26, 2021, date of publication April 5, 2021, date of current version April 14, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3071169

An Optimized Feature Selection Technique in Diversified Natural Scene Text for Classification Using Genetic Algorithm

GHULAM JILLANI ANSARI[1], JAMAL HUSSAIN SHAH[2], MYLÈNE C. Q. FARIAS[3], (Member, IEEE), MUHAMMAD SHARIF[1], (Senior Member, IEEE), NAUMAN QADEER[4], AND HABIB ULLAH KHAN[5]

[1]Department of Information Sciences, University of Education at Multan, Lahore 60000, Pakistan
[2]Department of Computer Science, COMSATS University Islamabad at Wah, Islamabad 45550, Pakistan
[3]Department of Electrical Engineering, University of Brasília, Brasilia 70910-900, Brazil
[4]Department of Computer Science, Federal Urdu University of Arts, Science and Technology at Islamabad, Islamabad 44000, Pakistan
[5]Department of Accounting and Information Systems, College of Business and Economics, Qatar University, Doha, Qatar

Corresponding authors: Jamal Hussain Shah (jamalhussainshah@gmail.com) and Habib Ullah Khan (habib.khan@qu.edu.qa)

This work was supported in part by the Qatar National Library, Doha, Qatar, and in part by Qatar University under Grant QUHI-CBE-21/22-1.

ABSTRACT Natural scene text classification is considered to be a challenging task because of diversified set of image contents, presence of degradations including noise, low contrast/resolution and the random appearance of foreground (font, style, sizes and orientations) and background properties. Above all, the high dimension of the input image's feature space is another major problem in such tasks. This work is aimed to tackle these problems and remove redundant and irrelevant features to improve the generalization properties of the classifier. In other words, the selection of a qualitative and discriminative set of features, aiming to reduce dimensionality that helps to achieve a successful pattern classification. In this work, we use a biologically inspired genetic algorithm because crossover employed in such algorithm significantly improve the quality of multimodal discriminative set of features and hence improve the classification accuracy for diversified natural scene text images. The Support Vector Machine (SVM) algorithm is used for classification and the average F-Score is used as fitness function and target condition. First after preprocessing input images, the whole feature space (population) is built using a multimodal feature representation technique. Second, a feature level fusion approach is used to combine the features. Third, to improve the average F-score of the classifier, we apply a meta-heuristic optimization technique using a GA for feature selection. The proposed algorithm is tested on five publically available datasets and the results are compared with various state-of-the-art methods. The obtained results proved that the proposed algorithm performs well while classifying textual and non-textual region with better accuracy than benchmark state-of-the-art algorithms.

INDEX TERMS Genetic algorithm, natural scene text, optimal feature selection, SFS, feature fusion, feature space dimensionality reduction.

I. INTRODUCTION

Features are discriminative elements that help to differentiate different types of objects in an image. It has been observed that pattern recognition classifiers have difficulties achieving a good performance when the feature space has high dimension [1]. Therefore, to design a better classifier and achieve a good accuracy, a possible strategy consists of reducing the complexity of the model by reducing the number of features, discarding non-informative and redundant features [2], [3] obtained from diverse set of images.

In statistical machine learning feature selection, also known as attribute selection, variable selection or variable subset selection is a method used to select a subset of optimal features that are considered more pertinent to the application [4]. There are multiple approaches to gather the best subset of features, including, principal component analysis (PCA) [5], [6], ant colony optimization [7], particle swarm optimization (PSO) [8]–[10], firefly [11] and genetic algorithm (GA) [4], [12]. It is worth pointing out that GAs are powerful stochastic biologically-inspired techniques that can be used in several image processing applications including image enhancement, image segmentation, image classification, and (naturally) feature selection.

The associate editor coordinating the review of this manuscript and approving it for publication was Qingli Li.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

In this work, our goal is to classify text and non-text regions in natural scene images using genetic algorithms. Most of the classification works available in literature consider text documents, rather than cropped scene texts. Moreover in different scenarios, images of business places and logos can also be classified as text areas [13]. More specifically, we used a GA technique to select the best appropriate features for the text classification problem. Before extracting the image features, we perform a preprocessing operation, which consists of performing histogram equalization for normal contrast images and a Fourier transform for contrast enhancement in low resolution images. Then, a feature space (population) is obtained by extracting Histogram of Oriented Gradients (HOG) [14], Local Binary Pattern (LBP) [15], color features [16] and contour features [17].

Finally, to select the best features, a GA framework is used to classify text and non-text regions in natural scene images using an integrated SVM [18]. The GA is robust and unbiased to a variety of texts, fonts, sizes, variations, angles, distortions, and skews [19]. Furthermore, using biological evolution concepts (e.g. survival of the fittest) in optimization problems has been proven to be promising [20].

In summary, to obtain the best features for the problem of text classification in natural scenes, an unbiased GA technique searches the whole feature space (population). The GA selects the best features among all individuals (features), taking into consideration a fitness function. The GA shrinks the length of the feature vector to reduce the running and training time of the classification system and attain an optimal accuracy, even in the presence of noise and other degrading factors [21]. Since GA acts as a powerful tool for feature optimization and classification, we selected it to find optimal and near-optimal solutions to our problem, considering the very large search space of the application and the limited amount of time.

This work has the following key contributions:

- The design of a multimodal feature system for optimization, which includes preprocessing, text localization and feature extraction stages of a diversified set of images with noise, low contrast/resolution and random appearance of foreground (font, style, sizes, orientations) and background properties.
- The design of a GA framework with an integrated/implicit SVM, which is able to reduce the dimension of the feature vector and classify natural text images as ''text'' or ''non-text'' areas.
- The use of the average F-score as target and fitness function to optimize the performance of the proposed classifier.

The rest of the paper is composed of the following sections. Section II highlights the related work, while Sections III describes in detail the proposed methodology. Section IV discusses the experimental results. In the last section, we provide our conclusions and discuss future works.

II. RELATED WORK

Nowadays, natural scene text classification is an important task that is gaining importance as a way to enhance model learning. The image feature extraction and optimization stages are very important parts of a good classifier. One of the best performing techniques is genetic algorithm (GA). GA is a feature optimization and classification method that is based on evolutionary theory. In this section, we describe the state-of-the art of GA techniques.

Vafaie and Jong [21] used a GA to pick the best features for a rule induction system. The feature selection policy described in their work has two main classes. The first class independently selects features for classification despite of their effect on performance. The second class selects a subset of the best optimized features (from the whole feature space), in such a way that it does not degrade the performance of the classifier system.

Raymer *et al*. [22] used a GA technique to extract and select features and train a classifier. Their method reduces the dimension of a weighted feature vector to scale the features, either in the linear or nonlinear way. The authors also employed a masking factor to act as a fitness function that helps performing the selection of the features.

Sun *et al*. [23] stated that GAs can select the best subset of features by encoding gender information (e.g. face length and width, mouth and eye size, angles, distances and areas). For this, they first used the principal component analysis (PCA) [24] to represent every image Eigen feature in a small dimensional space. A GA is then use to select the best features and reject irrelevant eigenvectors from gender information. After this, the selected features are fed to the Neural Network, SVM, Bayes Classifier, and Linear Discriminative Analysis stages for classification. The classifier fitness is evaluated using the accuracy (computed from the validation samples) and by varying the number of eigenvectors samples from 10 to 150.

Mohamad [25] used a GA technique to identify which feature combinations can be considered for classification using the classification accuracy as the fitness (or objective metric).

Uguz [5] analyzed the problem of text categorization and concluded that a large number of features increases the amount of noise in the process, misleading the classifier and reducing the accuracy and performance. The first stage of their algorithm uses an information gain (IG) measure to rate each feature within the document [26]. To reduce the dimensions, in the next stage of their algorithm, a combinatorial approach (a PCA [27]) and a GA are applied to the features, with the goal of ranking their order of importance. This way, to decrease the computational time and complexity of the classification process, features of less importance are eliminated. Other authors have used the K-nearest neighborhood [28], or the C.45 decision-tree [29] classifiers.

Jaberi and Madiafi [30] introduced a method to reduce the selection pressure, i.e. the tendency to only select the best members of the in-progress generation, that are later propagated to the next generation, and increase the genetic

diversity of the population of a GA using different selection procedures. As a result, using an elitist method that transferred the best individuals of in-progress generation to the next generation, they were able to reduce the complexity of the algorithm for different selection methods. This method also decreases the convergence time, but enhances the effect of selection pressure that often causes a convergence to a local minima.

Tsai *et al*. [31], proposed classification models for different domain datasets (including small-scale and large-scale) based on the selection of the best features followed by an instance selection (discarding faulty data) using a GA. The Bayesian network learning algorithm is used as fitness function and the chosen classifiers are the SVM and the K-nearest neighborhood (K-NN) techniques.

Catak [12], employed text datasets classification models with best feature selection using a GA technique. They introduced an objective function to maximize the sum of the feature ratio and of the F-score of the best chromosome. The authors used this objective function with three different classifiers (SVM, maximum entropy, and stochastic gradient descent), to check the efficiency of their proposed objective function.

Li *et al*. [32], proposed a classification method for electrocardiogram (ECG) signals using a GA and Back Propagation Neural (GA-BPNN) techniques. The features are extracted using a wavelet packet decomposition (WPD) transform. The best features are selected using a sum of square error (SSE) fitness function, with the help of the roulette wheel method.

Sharif *et al*. [33], used a GA to find out the appropriate features for the offline signature verification system. These selected features are then given to a SVM algorithm for verification, using an Artificial Neural Network (ANN) as a cost function. A few representatives classification methods [34]–[47] are also popular for scene character classification and further all these are useful for text/character recognition.

Similarly, the other evolutionary algorithms including PSO and firefly are also considered as eminent tool among researchers for selecting best classification features. In [9], multi-objective particle swarm optimization is used as a remedy for cost based feature selection. Recently, an extended work is presented by Song *et al.* [10] in which divide and conquer idea is employed to variable-sized cooperative co-evolutionary particle swarm optimization (VS-CCPSO) for feature selection. Finally, the work done in [11], [48], [49] are promising to read that reflect the challenge to deal with curse of dimensionality.

Seeing that, this literature reveals that the GA can be successfully used to select the relevant features for classification purposes. These techniques show promising results, since feature optimization is the key to successful model learning. In this paper, we propose a GA methodology for classifying text and non-text regions in natural images. Above all, GA can be used to reduce number of features into an optimal set of features.

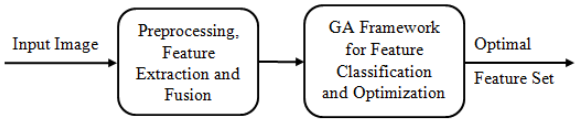


FIGURE 1. Block diagram for the proposed methodology.

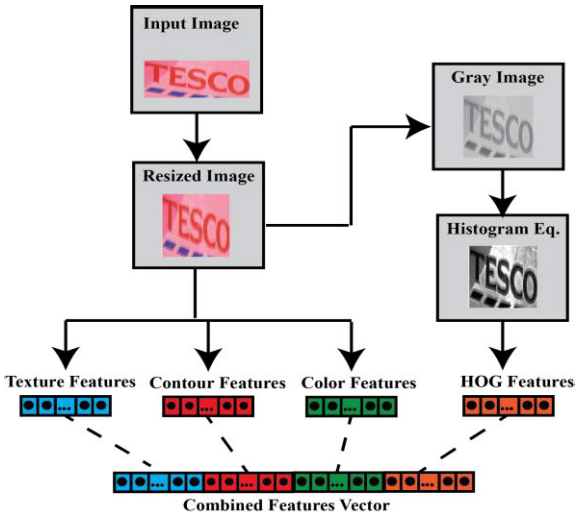


FIGURE 2. Preprocessing and feature extraction and fusion process to build combined feature vector.

Our proposed GA framework is specially designed to extract discriminative features from diversified set of images having different degrading factors and foreground/background properties. The input features are converted into binary chromosome to leverage and achieve more informative supervised information. This enables the framework to achieve an efficient robustness against a variety of foreground and background properties. Up to our knowledge, there is currently no such similar work in the literature. Our results show that the proposed algorithm performs better, when compared to state-of-the-art feature selection methods.

III. PROPOSED METHODOLOGY

In this methodology, the goal is to classify text and non-text regions in natural scene images using GA considering diversified set of images having noise, low contrast/resolution and random appearance of foreground (font, style, sizes, orientations) and background properties. The block diagram of the proposed methodology is shown in **Fig. 1**.

The major focus of this study is to study classification with reduced false positives along with a good score of precision, recall, F-Score and accuracy keeping the aforementioned diversity in mind. The diagram of the proposed methodology is depicted in two figures: **Fig. 2** shows image preprocessing for low/high contrast images, feature extraction using text localization and feature fusion, while **Fig. 3** shows the optimal feature vector extraction stage, which uses a SVM classification with an iterative GA framework

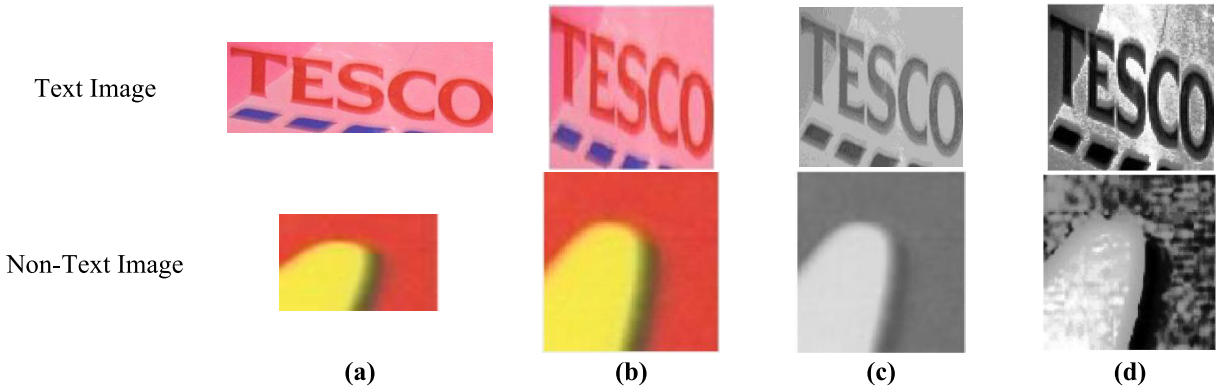


FIGURE 3. Example of text/-and non-text Images. (a) Original image. (b) Resized image. (c) Grayscale image. (d) Histogram equalized image.

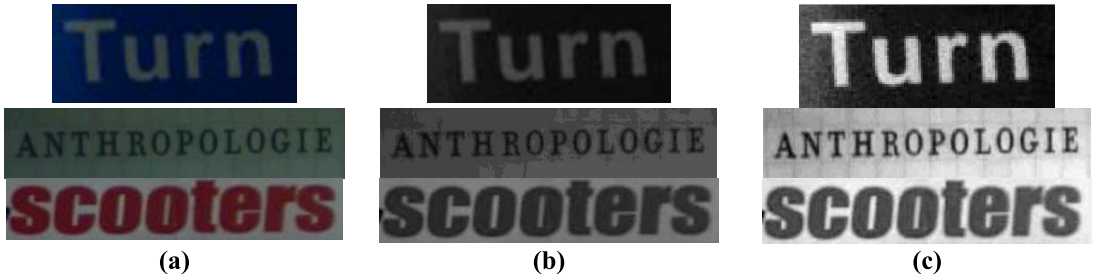


FIGURE 4. Text image enhancement for low contrast/resolution image. (a) Original image. (b) Grayscale image. (c) Contrast enhanced image using FFT-IFFT.

A. PREPROCESSING AND MULTIMODAL FEATURE PREPARATION FOR CLASSIFICATION OF NATURAL SCENE TEXT USING GA FRAMEWORK

The proposed diagram is shown in **Fig. 2**, which is employed to build multimodal feature vector after necessary preprocessing depending upon the image condition.

1) PREPROCESSING

The first step consists of resizing the input images into images to 100×100 pixels with the goal of maintaining the symmetry among all positive and negative text images. These samples are manually cropped using the benchmark datasets and illustrated in [40]. Next, we convert the color images into grayscale images. Then, we perform a histogram equalization to enhance the contrast of good resolution images, which helps to differentiating text and non-text images. Sample outputs of these steps are shown in **Fig. 3.**

Images having low resolution/contrast, the Fast Fourier Transform (FFT) and the inverse FFT (IFFT) as the inverse technique are applied. In this work, the logarithmic transformation act as a filtering function $H(x, y)$ that maps a tapered series of low input gray levels into a wider series of output values. The filtration function $H(x, y) = C^* \log(1 + g(i, j))$ is used to process image pixel $g(i, j)$, where C is the constant value and normally taken as 1 for enhancement. It is equally important that FFT must be completely reversible means to restore the image from frequency domain vector into spatial domain vector, so we used standard IFFT equation for this purpose.

Fig. 4 shows original low contrast and resolution image and their corresponding grayscale and contrast enhanced versions. The comparison of **Fig. 4(b)** and **Fig. 4(c)** shows significant improvement in contrast. This technique helps to achieve good results in the later sections.

2) FEATURE EXTRACTION AND FUSION

For a classification problem, we need to extract multiple features from an image to build a multimodal feature representation. The multimodal approach is generally able to represent most image properties so that we can obtain suitable supervised information for a classification problem. The obtained supervised information are susceptible to noise, non-universality, inter and intra-class variations, and spoofing attacks.

Popular image feature descriptors for multimodal representations include appearance features (HOG), color, contour and texture features (LBP). Common characteristics of text in images are: (1) the text is generally visible on different textures background; (2) the contour image reflects the salient edges from which geometric features can be detected; (3) Color is an important feature descriptor because generally text appears in different colors and on different color backgrounds to avoid readability issues.

Thus, a multimodal feature representation based on such characteristics can reveal significant content variations and

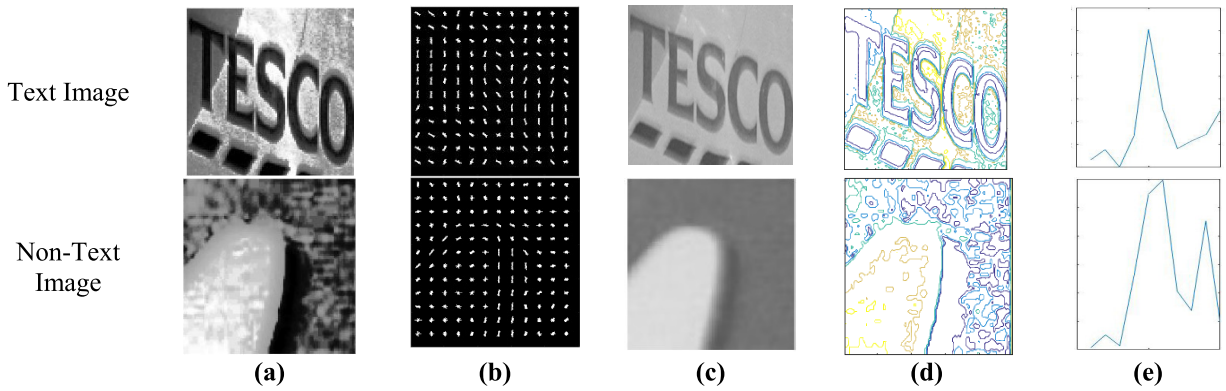


FIGURE 5. (a) Contrast-enhanced image using histogram equalization. (b) HOG visualization as obtained from (a). (c) Grayscale resized image. (d) and (e) Contour and LBP histogram Visualization based on (c).

effectively enhance the classification accuracy [50]. Given these features, it is clear that the text classification problem has a large dimension feature vector. Therefore, using a GA to obtain a discriminative optimal set of features enhances the learning capability of the classifier. The steps for feature extraction and fusion are shown in Algorithm I. Next, we described the set of features, which are extracted in a novel way to handle the diversity of different types of images used in this work.

a: APPEARANCE FEATURES

The Histogram of Oriented Gradient (HOG) operator was proposed by Dalal and Triggs [14]. In this work, we use the HOG operator to extract image appearance features resulting in a vector $A = [A_1, A_2, A_3, \ldots, A_n]$. In order to calculate the HOG of an image, we set fixed number of bins, and divide the image into blocks (arrays) of $A_x \times A_y$. Then, we compute the HOG for each image block i.e. each sub-image k. The HOG of an image block k is computed by calculating the image gradient ∇k using the difference schema:

$$\nabla k(p,q) = \frac{1}{2}(k(p+1,q) - k(p-1,q), \times k(p,q+1) - k(p,q-1)) \quad (1)$$

where p and q are coordinates, **Fig. 5(b)** shows examples of the HOG appearance features. The descriptor generates a collection of more than 100 features. To calculate gradient orientation **(1)** is applied to each training example. Subsequently, we end up with a feature vector A of length 1×3780.

b: CONTOUR FEATURES

Contours represent curves and edges that translate into meaningful geometric variations of the image resulting in a vector $B = [B_1, B_2, B_3, \ldots B_n]$. We extract open contouring in a novel way, which is robust manner to handle curves and edges and gives meaningful geometric variations. Open contours are very good features that can be used to localize text in images because they emphasizes the boundaries. **Fig. 5(d)** shows examples of contour features, which reflect open contours that link edge pixels into line fragments of the region boundary.

Assume that an image $G(x, y) : \varpi \rightarrow R$ has several objects in the background and foreground. Then, the curve and edges can be used to localize the boundaries of the Object of Interest (OoI) in images using the following steps [51], [52]:

- Choose an initial OoI
- Use some criteria to move forward and find stable curves and edges of OoI
- Stop when the stop criteria is met

Based on the abovementioned criteria the region based model is the best choice to obtain the open contours. This could be achieved using the popular partial differential equation for curve evaluation in open contours. Hence, this model partitions the image $G(x, y)$ into background and OoI on the basis of pixels intensity similarity adopting the following function given in **eq. (2)**, which needs to be minimized.

$$\min_{O,o_1,o_2} E(O,o_1,o_2) = \int_{inward(O)} (G(x,y) - o_1)^2 dxdy + \int_{outward(O)} (G(x,y) - o_2)^2 dxdy + \Gamma Length(O) \quad (2)$$

Assume the object for predicted curve O is parameterized as $O(l) \in [0, 1] \rightarrow O(l) = G(x, y)$ when $O(1) = O(0)$.

When $o_1 \neq o_2$, the OoI can be determined if and only if $G(x, y) \approx consto_1$ for region, whereas background is determined $G(x, y) \approx consto_2$. In **(2)**, the open contours are defined by first two terms followed by a regularization term which acts as penalty to define the length of the curve. Further, the **(2)** is extended for implicit representation and expressed:

$$\min_{\lambda,o_1,o_2} E(\lambda,o_1,o_2) = \int_{\varpi} (G(x,y) - o_1)^2 T(\lambda) dxdy + \int_{\varpi} (G(x,y) - o_2)^2 (1 - T(\lambda)) dxdy + \int_{\varpi} |\nabla W(\lambda)| dxdy \quad (3)$$

where T is the threshold value, which is in the range of [1]–[3], depending upon image quality for generating open/active contours, while ∇W is regularization term for handling the outliers so as to speed up the process. Further, the values of o_1 and o_2 in **(3)** can be computed by minimizing the function in the following equations, which are based on externality conditions.

$$o_1 = \frac{\int_{\varpi} G(x,y)T(\lambda)dxdy}{\int_{\varpi} T(\lambda)dxdy} \tag{4}$$

$$o_2 = \frac{\int_{\varpi} G(x,y)(1-T(\lambda))dxdy}{\int_{\varpi} (1-T(\lambda))dxdy} \tag{5}$$

whereas the λ is computed using gradient descent incorporating the loss function given as in **(6)** below, where t is the slope and δ is the intercept term and computed as $t = t + \Delta t$ and $\delta = \delta + \Delta\delta$ respectively.

$$\frac{\partial\lambda}{\partial t} = \delta(\lambda)\left[-(G(x,y)-o_1)^2+(G(x,y)-o_2)^2+\Gamma T(x,y)\right] \tag{6}$$

The above model localized text (OoI) on basis of intensity similarity and produce region boundary through edges and curves, which does not enclose the object. It can also handle random orientations and perform better with low contrast, unclear and complex background to determine stable regions. The final feature vector B has the length equal to 1×6.

c: TEXTURE FEATURES

Local Binary Patterns (LBP) is a popular feature descriptor proposed by Ojala *et al.* [15] to extract texture features in the image. It is computed for each pixel $p = (x, y)$ of the image. Consider λ_i is the level of intensity of a pixel in the 8-connected region of the pixel p and λ_c is the intensity of the central pixel p. The LBP [53], [54] is computed using the following equations:

$$lbp(x,y) = \sum_{i=0}^{7} 2^n(\lambda_i + \lambda_c) \tag{7}$$

The threshold for LBP that is $e(x)$ is calculated as:

$$e(x) = \begin{cases} 1, & x > 0 \\ 0, & \text{otherwise} \end{cases}$$

In this work, we consider a random arrangement of adjacent pixels. Hence, the LBP descriptor converts every pixel to an 8-connected region. As a result, the feature set $C = [C_1, C_2, C_3, \ldots, C_n]$ contains histogram values corresponding to each image region. Fig. 5(e) depicts encoded local texture features, which are very discriminative for the classification task. The feature vector C has a length of 1×9.

d: COLOR FEATURES

The color features $D = [D_1, D_2, D_3, \ldots, D_n]$ of an image can be extracted using a color-based distribution entropy operator. The Color based Distribution Entropy (CDE) method, which was introduced by Sun *et al.* [16], not only gives the information of the image colors, but also provides the spatial distribution of the different pixels in the image. The CDE descriptor uses a normalized spatial distribution histogram (NSDH) algorithm, which is built on the annular color histogram function [55]. Considering that A_i is the pixel set of an image, then $|A_{ij}|$ is the pixels count for color bin i within circle of radius r_i. Using the NSDH algorithm, the normalized color histogram of a particular color i is given by:

$$p_i = (p_{i1}, p_{i2}, \ldots, p_{iN}) \; where \; p_{ij} = |A_{ij}|/|A_i| \tag{8}$$

and the color distribution entropy can be determined using the following equation:

$$E_i(p_i) = -\sum_{j=1}^{N} p_{ij}\log_2(p_{ij}) \tag{9}$$

The length of the feature vector D at this stage is 1×256. **Fig. 5** shows the examples of all extracted features, which include that the gradient orientation, the region boundary and the LBP histogram.

e: FUSING EXTRACTED FEATURES

Extracted features from multiple sources can be pooled or fused at distinct levels, which can in the form of decision, score, and feature levels strategies. Among all, using a feature level fusion strategy is considered the most effective and powerful fusion strategy, because it imitates different information from the same data to provide better recognition results [56]. The proposed feature extraction and fusion steps are presented in **Algorithm I**.

In our case we have four feature spaces A, B, C and D which correspond to HOG, contour, LBP, and color features respectively. The feature vector A has n-dimensions, while the feature vectors B, C, and D have m-dimensions. To equalize the dimension of all vectors, the lower vector length is padded with zeroes. Let $\Phi = (A, B, C, D)^T$ be the image feature space. Then, for a random sample $\tau \in \Phi$, the feature vector X includes random samples i.e. $a \in A, b \in B, c \in C$ and $d \in D$. Therefore, all the selected feature vectors can be serially concatenated and defined as $X = (abcd)^T$. The output of **Algorithm I,** is the final feature vector X, which has a length 1×4051, which is a high dimension.

B. GA FRAMEWORK FOR FEATURE OPTIMIZATION AND CLASSIFICATION

Next, we conclude our methodology by employing GA framework to reduce the feature vector and to receive optimal feature set as an output.

1) GA-BASED FEATURE SELECTION AND CLASSIFICATION

The proposed framework is shown in **Fig. 6,** which shows the optimal feature vector extraction stage, that uses a SVM classification with an iterative GA framework. The parents in the population P are in the form of a binary chromosome. Each binary chromosome is a composition of the feature

Algorithm 1 Proposed Feature Extraction and Fusion

Begin

Input: Query image

Output: Fused feature vector

Step 1: Resize the input image to 100×100 dimension

Step 2: Convert input image to grayscale image

Step 3: Perform histogram equalization

Step 4: $A =$ Extract HOG features using eq. (1)

Step 5: $B =$ Extract contour features after Step 1 using (2-6)

Step 6: $C =$ Extract LBP features after Step 1 using (7)

Step 7: $D =$ Extract color features after Step 1 using (8-9)

Step 8: Extracted features are concatenated serially to build the following fused feature space

- $\Phi = (A, B, C, D)^T$

Step 9: Feature selection X is made based upon selected binary chromosome

- $X = (a, b, c, d)^T$, where $a \in A$, $b \in B$, $c \in C$, and $d \in D$

End

genes, where each gene g has a bit value 0 (not selected) or 1 (selected feature). In the beginning, the chromosomes from the population are selected randomly to initiate the process. Moreover, elitism is used to retain quite a few highest individuals for the next generation directly. **Table 1** depicts the parameters used for the proposed methodology obtained after a number of experiments.

Refer to **Fig. 6**, the important following steps are performed:

- An *Initial Chromosome (1)* is generated using random binary values from the *Combined Features Vector* to act as *Selected Features of Image*.
- The *Selected Feature Vector (2)* becomes the part of the *Training Set* (3), being updated on each generation of the *Training of SVM Classifier (5).*
- The *Testing Set (5)* is also obtained from the *Training Set* (for validation purposes).
- Both *Training Set (3)* and *Testing Set (5)* are fed into the *Training of SVM Classifier(5)* stage, to obtain a *Trained SVM Classifier (6)*, which iteratively enhances the model learning as training and testing samples are increased.
- During the training process, the *Fitness Function Evaluation (7)* operation is performed for calculating the *Classification Accuracy*.
- If the *Check Condition >= 92.0 (8)* is met, then the optimization process is finalized and the *Optimal Feature Vector* is obtained.

TABLE 1. Parameter values used in GA based feature selection for classification.

S. No.	Parameters	Values
1	Total generations	100
2	Population size	200
3	Crossover scheme	Two-Point crossover
4	Average cross over rate	0.7%
5	Average mutation rate	0.05%
6	Selection criteria	Roulette wheel
7	Elite count	2
8	Population type	Bit string chromosome
9	Fitness function	>= 92.0% Avg. F-score

- Otherwise, if termination condition is not satisfied then a *crossover (9)* is performed between the *1st parent (current chromosome)* and the *2nd parent (random binary values),* which is selected using the roulette wheel method.
- To finalize the process, a single iteration/generation *Mutation (10)* is performed to get a *new chromosome (after mutation)*. Yet again a new feature vector is selected and the process iterates between steps *(3)-to-(11)* until the target condition is achieved and the final *Optimal Feature Vector* is obtained.

The threshold values depicted in **Table 1** are obtained after number of (training and testing) experiments. Hence, to achieve this target goal, we have distributed the data samples into different (training-testing) ratios. These distribution sample ratios are (50-50), (60-40), (75-25), (80-20), (85-15) and (90-10). We found that the (80-20) distribution ratio does a better job in achieving the target condition. At the same time, average F-Score remain constant for (85-15) and (90-10) distribution samples, which shows that target condition 92.0% is optimal at (80-20).

The rest of the other parameters in the **Table 1** like mutation rate, crossover rate, population size and generation are also adjusted. They become optimal as the number of simulations reaches a target condition that is greater or equal to 92.0%. The **Fig. 7,** depicts a graph showing the average crossover, mutation and the F-Score values for the different distribution samples. From this graph, we are able to determine the optimal parameters that achieve the target condition and high accuracy levels for classification

To get an optimized feature subset, we set our chromosome to be comprised of bits corresponding of all features. The length of the binary chromosome is 1×16. **Fig. 8** shows the design of our chromosome.

Here, f_i is represents the bit value for the *ith* feature and n is the total number of features. If the bit value is 1 the feature is *selected*, while if the bit value is 0 the feature is *not selected* respectively. For example, if the F binary digit in a chromosome is given by $chromosome = 01000110$, the second, sixth, and seventh bits are selected as features and rest of the bits are not selected as a features. Hence, the chromosome is built using the following formulation:

$$chromosome = \underset{j=1}{n} \left\{ \vec{f_j} \mid \vec{f_j} \, \varepsilon[0, 1] \right\} \tag{10}$$

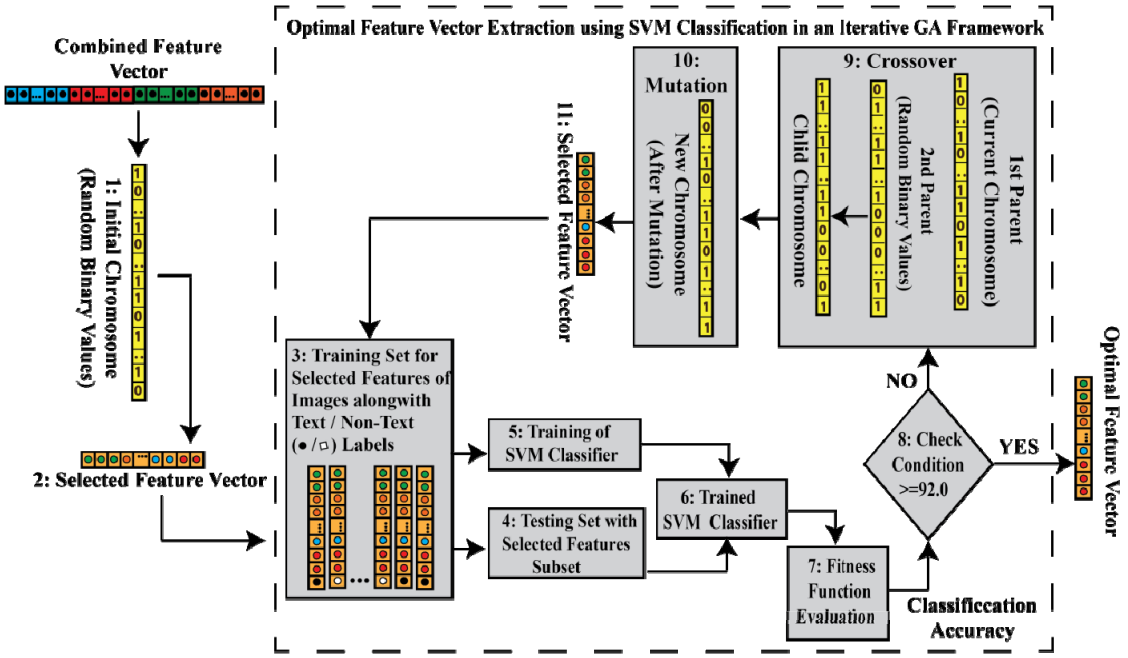


FIGURE 6. Illustration of GA framework to determine optimal feature vector. The process iteratively works until the target condition is met.

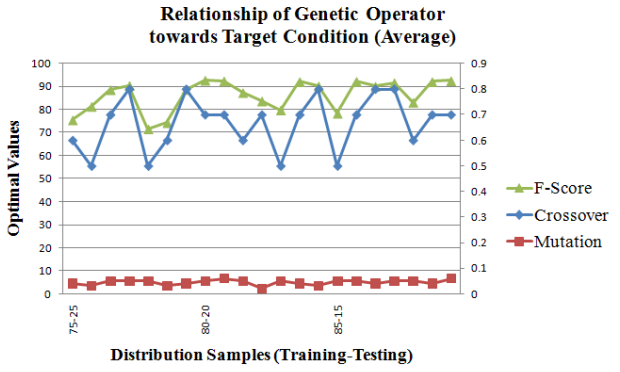


FIGURE 7. Performance illustration of genetic operators in achieving optimal values.

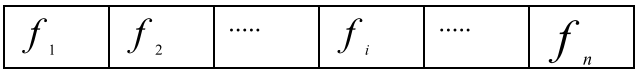


FIGURE 8. Design of chromosome for feature set.

The initial value of the chromosome is set by randomly choosing the bits. Then, the feature vectors for all images are trained and classified with text or non-text labels. Therefore, the process to generate an initial population is very simple and straight-forward, as shown below in Algorithm II. A random_initialize() function generates random bits i.e. [0, 1] that gradually initializes the chromosome [57]. Hence, the numbers of selected features are considered an arbitrary initial solution is f.

Algorithm 2 Population Initialization Process Based on Features.

```
Begin
i ← 1
while (i <= |P|) do
    for (every gene g value in ith chromosome) do
        if (random_initiaze () > f/F)g = 0;
        else
        end if
    end for
end while
End
```

After a population initialization, the input is trained using the SVM for binary classification. In this work, the F-score is not calculated separately. It is calculated for each and every generation. Thus, the average F-score is calculated through the following equation:

$$F = \sum_{i=1}^{n} F_i \Big/ \sum_{i=1}^{n}, \tag{11}$$

where $F_i = {}^{(2^* P_i^* R_i)}/_{(P_i + R_i)}$ and n is the generation count, f_i is the selected feature count, and F_i is the F-score from the previously calculated generations.

If the average score does not meet our predefined threshold, the next generation of the chromosomes is selected. The selection process ensures that the fitter chromosome has a higher probability of survival. Here in this study, our roulette wheel selection steps are given in Algorithm III. In **Algorithm III**, p_i calculates sum of all chromosome fitness in

Algorithm 3 Roulette wheel selection process

Begin

1: ***for*** $(i = 1 : n)$ ***do***

2: Compute joint probabilities by $prob_i = \sum_{j=1,i}^{n} P(j)$ and $p_0 = 0$

3: ***end for***

4: Find a random number *rvalue* in $[0, prob_n]$ where n is population and select the *ith* chromosome such that $prob_{i-1} < rvalue < prob_i$

End

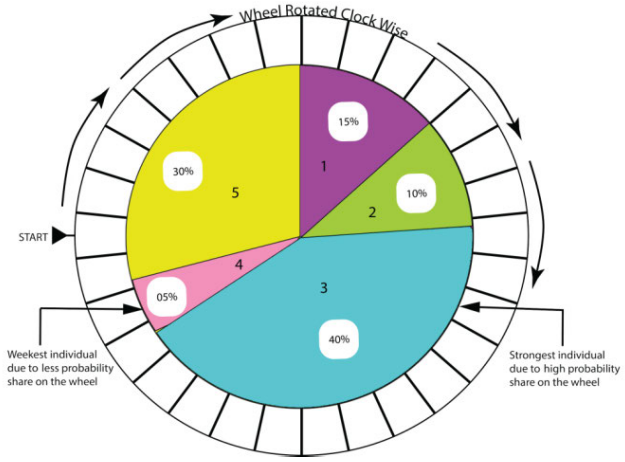


FIGURE 9. Snapshot demonstration of a probabilistic share of chromosomes in proposed methodology. GA in this paper uses a roulette wheel selection process.

the population, generates a random number r from interval $[0, p_n]$, and goes through the population n. It stops at the position when the sum of p_i is greater than r, which shows that it might be the chromosome with greater probability.

A snapshot of this process is shown in **Fig. 9**, where the clockwise roulette wheel shows the probability of five different chromosomes from the population. Chromosome 3 has the highest probability share (40%), while chromosome 5 has the second highest probability share (30%). Therefore, chromosome 3 covers most part of the wheel and is considered the fittest. As a consequence, it can be selected multiple times in the next operations.

Hence, Algorithm III helps to choose a second parent having the highest probability share for next generation crossover and mutation. The values for both operations are listed in Table 1. If the muted chromosome becomes powerful and superior to the parents, it substitutes the parents. If it is in between two parents then it substitutes the inferior parent, else the most inferior chromosome will be replaced from the population. The GA will be terminated when the generations count reaches to the maximum number or meet our target average F-Score.

When we achieve the target condition, then the evolution process stops. Otherwise, if the termination condition is not met then the next generation of the chromosome needs to be produced. The crossover operation is performed with a rate

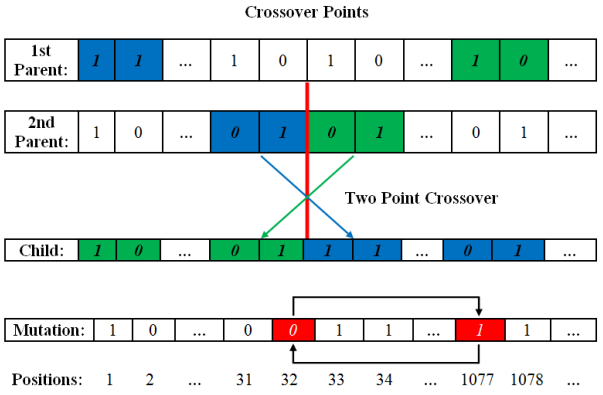


FIGURE 10. Demonstration of crossover and mutation adapted in the proposed method.

between the currently selected chromosome and another randomly initialized chromosome of 0.7%. Thereafter, the mutation process is applied with the rate of 0.05% on the child, which alters and interchanges the gene values in the child from 0-1 or from 1-0 with a goal of producing new chromosomes. Hereafter, all image features are chosen according to the genes of this new generation chromosome. The two-point crossover and mutation process is represented in **Fig. 10**. The process continues until we achieve the target condition.

In **Algorithm IV**, we check whether the average F-score (target condition) is achieved after the SVM training on selected features. If yes, then further evolution is stopped and the selected features are said to be optimal features set. But, if target condition is not met then the iterations are made among step 5-to-1 in **Algorithm IV**.

IV. EXPERIMENTAL EVALUATION

In this work the precision, recall and F-score are not separately evaluated. Instead, for every generation, we calculate the average values of precision and recall to determine average F-score, choosing the setting with the best target average f-score value [5]. The F-score is defined in **eq. (11)**, while precision, recall, and accuracy are computed using following equations:

$$P_i = \frac{TP_i}{TP_i + FP_i}$$

Thus the average precision (P) is defined as:

$$P = \sum_{i=1}^{n} P_i \Big/ \sum_{i=1}^{n} \tag{12}$$

The average recall (R) as:

$$R = \sum_{i=1}^{n} R_i \Big/ \sum_{i=1}^{n} \tag{13}$$

And the average accuracy as (A):

$$A = \sum_{i=1}^{n} A_i \Big/ \sum_{i=1}^{n} \tag{14}$$

where n is the number of generations. P_i, R_i and A_i calculated from the previous generation is used to compute P, R, and A for the current generation.

Algorithm 4 Proposed GA-Based Feature Selection and Classification

Input: Feature vector X
Output: Optimal feature set
Begin
Parameter Initialization:
- Generations/iterations size ← 100
- Size of population ← 200
- Type of population ← Bit string chromosome
- Target condition ←>= 92%
- Fitness function ← Compute **eq.**(11)
- Elite count ← 2

Train selected chromosome using SVM classifier
Testing selected feature subset
Apply fitness function Compute **eq.**(11)
if *(Target condition >= 92%)*
  Optimal feature vector
  Exit
  else
  a: ***while*** *(($j = 1$) <= Size of population)*
    - Keep finest solutions
    - Keep fitness function value
    - $j++$
    - ***while*** *(($i = 1$) <= Generations/iteration size)*
      - Compute **eqs.** (12-to-14)
      - Compute **eq.**(11)
      - $i++$

    end while
  end while
  b: Select new parents with ***Algorithm III***
  c: Perform two-point crossover at the rate of 0.7%
  d: Perform mutation at the rate of 0.05%
  e: Randomly select chromosomes
  f: Go to Step 2
end if
End

2) DATASETS DESCRIPTION

Before going into datasets description, it is important to discuss and resolve the imbalance class property within the datasets. It is equally necessary to avoid classifier biasness towards majority classes. Keeping this issue in mind the resampling technique is adopted, that help to lessen the discrepancy between the sizes of the classes [58]. Taking this idea further, we use SMOTE (Synthetic Minority Oversampling Technique) [59] with SVM and called it as SMOTE-SVM. The SVM is first train using linear kernel to generate support vectors (samples), and then these samples are oversampled with SMOTE. This idea enforces distribution between two classes (text and non-text) at border level instead of equalizing the number of samples for all the following datasets.

- ***ICDAR 2003:*** The ICDAR 2003 dataset was released for ICDAR 2003 Robust Reading Competition by Lucas *et al.* [60]. The dataset is a collection of 251 testing and 258 training character patches and word patches annotated by the bounded box and their text contents.
- ***SVT:*** The Street View Text dataset (SVT) [61] was specifically used for word spotting problems. The dataset is a collection of 647 words from which 250 testing images (video frames) with the availability of bounding box locations and ground truth labels along with 100 training images (video frames). Each image is taken from Google Street View.
- ***IIIT5K:*** The IIIT5K [62] is the largest and most challenging dataset reported to date due to variation is a font, color, layout, size and inclusion of noise, distortion, blur and varying illuminations. IIIT5K Word dataset is a collection of 5K words cropped from images found on the Internet, from which 3k and 2k words used for testing and training subset respectively.
- ***MSRA-TD500:*** This dataset [63] is a collection of 500 multi-oriented natural scene text (slanted and skewed). The dataset is split into 300/200 training /testing samples. Both English and Chinese texts of various orientations are part of this dataset.
- ***KAIST:*** The KAIST dataset [64] consist of 3000 indoor and outdoor natural scene images with text under unusual lighting conditions. This is also a benchmark multi-lingual dataset with English and Korean text. Thus very much challenging for classification purpose.

3) DISCUSSION

To test the proposed methodology, we compare it with benchmark wrapper-based feature selection approaches such as SFS, SBF, SFFS, SBFS and Plus-L-Minus-R. These methods have proven their success while searching best optimal feature set so as to improve the performance of the learning algorithm [65]. In wrapper methods, the selection of features is based on the performance of predictor. The predictor is wrapped on search algorithm to find a subset of features, which gives the highest predictor performance [66]. Therefore, wrapper methods structure work interactively and are much closer to the proposed methodology. In the proposed methodology the GA is also looped around the predictor until target condition is achieved in order to enhance the predictor performance. Wrapper methods are discussed in detail in [67].

Initially, the **Algorithm IV** parameters are set with a value as given in **Table 1**, which were obtained carefully after series of experimentation and also reflects best results. Primarily, the average F-score needs to be maximized up to 92.0% (target condition). The selected criteria for the tests are: (1) with or without preprocessing; (2) using benchmark wrapper approaches; (3) with or without GA; (5) scene character classification.

The performed tests are Test 1 *(Without preprocessing, feature fusion, and proposed methodology),* Test 2 *(With preprocessing and without proposed methodology),* Test 3 *(Without preprocessing and with benchmark wrapper feature selection*

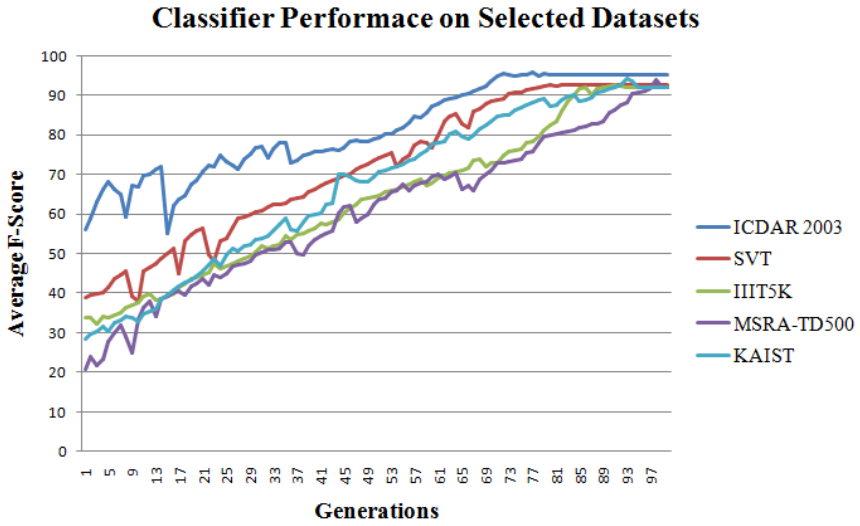


FIGURE 11. Average F-Score (%) improvement in SVM classifier after the implementation GA-based feature selection and optimization technique.

methods), Test 4 *(With preprocessing and with benchmark wrapper feature selection methods),* Test 5 *(Without preprocessing and with proposed methodology),* Test 6 *(With preprocessing and with proposed methodology),* and Test 7 *(Scene character classification).* Test 7 is basically applied to monitor the performance of the proposed method when considering a binary classifier with the existing benchmark techniques.

For performing the aforementioned tests (except Test 1) we used the parameters depicted in **Table 1**. In **Table 1**, the major target condition is the average F-Score, which is set to be to greater or equal to 92.0%. Also the fusion method in **Algorithm I** is employed to monitor the classifier performance, while SVM is trained implicitly rather than explicitly. The graphs in **Fig. 11** show the relationship between the average F-scores and the generations on the datasets including ICDAR 2003, SVT, IIIT5K, MSRA-TD500 and KAIST. The algorithm was implemented in MATLAB and tested using a *7-fold* cross-validation methodology for training and testing the classification algorithm. All tests were performed on a 3.4 GHz Processor, 8 GB of RAM PC machine, with a GTX 1070 GPU support.

Tables II depicts the results obtained for two (Test 1 and Test 2). These results are then compared with rest of the aforementioned tests to monitor the improvement in F-score and accuracy. Notice that, Test 1 is performed on the raw data without the feature fusion. In this test, the results were not satisfactory. As a result, the weak learning model is obtained with a poor performance for unseen data. Test 2 used a preprocessing step and incorporated the fused feature vector. When compared to the results of Test 1, the average F-score for ICDAR 2003 increased from 68.1% to 75.9%, for SVT from 64.4% to 72.6%, for IIIT5K from 64.8% to 70.5%, for MSRA-TD500 from 62.5% to 69.2% and for KAIST from 59.8% to 69.8%. It is worth pointing out that results in **Table 2** are used as benchmark for the rest of the aforementioned tests. Although Test 2 presented better results, the classification accuracy not yet acceptable for real applications.

Table 3 depicts the results for Test 3 and Test 4, incorporating the benchmark wrapper-based feature selection approaches. **Table 3**, shows the average F-Score, showing that SFFS performs better in Test 3 and Test 4 than the other wrapper methods. This result is probably due to the additional steps included in the SFFS backtracking technique. Notice that for Test 3, which is the performed on the raw data, we found an improvement when compared to Test 2. However, in Test 4, we observed a considerable improvement. When compared to Test 3 results, for SFFS, the average F-score for ICDAR 2003 increased from 77.1% to 77.3%, for SVT from 73.7% to 77.6%, for IIIT5K from 73.2% to 79.7%, for MSRA-TD500 from 72.9% to 77.8% and for KAIST from 73.8% to 78.3%. It is worth pointing out that values in **Table 3** are gradually enhanced for evaluation parameters for Test 3 and Test 4. Hence, **Table 3** shows better results when compared to **Table 2**. This confirms that the fused feature vector is playing a key role in enhancing the average F-Score and maximizing the model learning. Furthermore, the accuracy of the classifier is also getting gradually better. But, the performance figures are still not acceptable.

Table 4 finally shows the results of Test 5 and Test 6, which incorporate the proposed methodology. In Test 5, the proposed algorithm outperforms all benchmark feature selection techniques, but it is unable to reach the target condition defined in **Table 1**. We attribute this lower performance to the fact that the images were not preprocessed. Nevertheless, the performance of the classifier is acceptable, which can probably be attributed to the feature fusion approach. In Test 5, ICDAR 2003 achieves 88.2%, SVT 86.5%, IIIT5K 84.8%, MSRA-TD500 84.2%

TABLE 2. Classifier performance average (%) values of P, R, F, and A with raw and preprocessed data obtain after 100 iterations.

Classifier	ICDAR 2003				SVT				IIIT5K				MSRA-TD500				KAIST			
	P	R	F	A	P	R	F	A	P	R	F	A	P	R	F	A	P	R	F	A
Test 1																				
SVM	71.1	65.3	**68.1**	67.1	68.4	60.9	**64.4**	65.6	67.3	62.6	**64.8**	63.9	63.8	61.2	**62.5**	66.1	61.2	58.4	**59.8**	60.3
Test 2																				
SVM	78.7	73.2	**75.9**	77.5	73.9	71.3	**72.6**	73.4	71.2	69.9	**70.5**	72.1	70.2	68.2	**69.2**	70.6	69.7	70.0	**69.8**	71.7

TABLE 3. Classifier performance average (%) values of P, R, F, and A obtain after 100 iterations on benchmark wrapper feature selection methods.

Method	ICDAR 2003				SVT				IIIT5K				MSRA-TD500				KAIST			
	P	R	F	A	P	R	F	A	P	R	F	A	P	R	F	A	P	R	F	A
Test 3																				
SFS	76.1	70.4	**73.1**	79.2	74.6	70.7	**72.6**	76.3	71.7	69.6	**70.6**	74.4	73.1	71.0	**72.1**	73.9	71.6	66.5	**68.9**	72.6
SBF	74.9	69.7	**72.2**	78.6	72.7	69.4	**71.0**	74.1	70.9	68.4	**69.6**	70.3	70.2	68.3	**69.2**	70.6	72.4	69.1	**70.7**	69.1
SFFS	79.5	74.9	**77.1**	80.3	77.7	70.1	**73.7**	77.9	76.2	70.5	**73.2**	73.6	75.0	70.9	**72.9**	76.1	75.8	71.9	**73.8**	75.4
SFBS	77.1	73.2	**75.1**	76.2	74.9	69.8	**72.3**	75.7	75.4	70.2	**72.7**	71.9	73.8	70.0	**71.5**	74.3	73.2	68.6	**70.9**	70.8
PLMR	75.9	70.5	**73.1**	77.5	71.9	68.1	**69.9**	76.7	73.7	67.8	**70.6**	72.2	70.2	65.9	**67.9**	75.7	70.6	64.2	**67.2**	69.9
Test 4																				
SFS	80.1	74.5	**77.1**	80.7	78.4	73.1	**75.6**	81.7	79.7	75.2	**77.3**	78.9	77.2	75.6	**76.4**	78.7	77.4	74.3	**75.8**	77.8
SBF	77.3	71.2	**74.1**	80.9	77.3	72.7	**74.9**	81.6	74.3	71.6	**72.9**	82.7	75.3	72.4	**73.8**	77.7	76.3	70.8	**73.4**	79.9
SFFS	83.1	76.3	**77.3**	83.4	80.9	74.6	**77.6**	82.7	81.1	78.3	**79.7**	82.2	81.5	74.6	**77.8**	80.2	80.3	76.4	**78.3**	81.6
SFBS	80.2	74.1	**77.1**	79.2	78.1	71.9	**74.8**	81.9	80.4	77.9	**79.1**	80.6	76.9	71.1	**73.8**	79.3	79.9	74.6	**77.1**	80.2
PLMR	79.4	73.6	**76.4**	81.4	74.7	70.9	**72.7**	84.7	76.9	72.3	**74.5**	82.1	80.1	75.4	**77.6**	80.1	81.0	69.9	**75.0**	78.2

TABLE 4. Classifier performance average (%) values of P, R, F, and A obtain after 100 generations on proposed methodology.

Method	ICDAR 2003				SVT				IIIT5K				MSRA-TD500				KAIST			
	P	R	F	A	P	R	F	A	P	R	F	A	P	R	F	A	P	R	F	A
Test 5																				
SFS	84.4	79.5	81.9	82.2	83.6	80.1	81.8	83.7	81.9	78.4	80.1	79.7	80.2	78.5	79.3	82.4	82.7	77.7	80.1	81.0
SBF	83.9	80.3	82.1	81.3	81.9	78.6	80.2	82.5	80.6	76.7	78.6	83.5	79.8	74.2	76.9	81.6	80.0	74.6	77.2	84.2
SFFS	88.9	81.6	85.1	85.6	86.7	84.7	85.6	84.2	82.4	80.4	81.4	84.1	84.2	80.0	82.1	84.9	86.1	82.1	84.1	86.4
SFBS	87.3	82.4	84.7	82.9	85.4	82.1	83.7	83.1	83.7	77.2	80.3	84.7	82.4	79.9	81.1	80.3	85.4	81.6	83.3	83.7
PLMR	86.5	80.2	83.2	82.7	85.6	81.4	83.4	84.3	84.3	81.7	82.9	83.6	81.3	77.1	79.1	82.1	82.0	80.0	80.9	83.4
Ours	90.1	86.4	**88.2**	87.1	88.8	84.3	**86.5**	86.4	87.4	82.4	**84.8**	85.3	85.9	82.6	**84.2**	86.1	88.6	82.7	**85.5**	84.7
Test 6																				
SFS	87.8	84.6	86.2	85.1	85.3	81.6	83.4	84.6	84.6	81.5	83.0	82.4	83.5	82.1	82.8	85.4	84.5	80.1	82.2	85.6
SBF	88.1	81.4	84.6	86.3	84.6	80.2	82.3	84.1	83.5	80.3	81.8	86.7	82.4	80.1	81.2	84.9	83.2	78.9	80.9	87.1
SFFS	90.1	85.3	87.6	87.4	89.2	86.9	88.1	86.1	84.1	80.9	82.5	87.1	88.2	83.8	85.9	87.2	87.0	84.0	85.4	88.2
SFBS	89.9	83.1	86.3	84.8	88.7	85.1	86.8	87.7	85.7	82.7	84.2	86.4	86.3	82.9	84.5	85.4	88.3	82.3	85.2	85.8
PLMR	90.1	82.9	86.3	85.4	89.2	86.4	87.7	86.0	88.3	82.6	85.3	84.8	85.2	80.1	82.5	86.1	86.4	81.3	83.7	86.9
Ours	96.1	90.2	**95.1**	**91.8**	94.9	89.3	**92.6**	**89.5**	94.5	89.9	**92.1**	**88.1**	95.9	91.9	**93.9**	**90.1**	96.1	92.3	**94.1**	**89.9**

and KAIST 85.5% average F-Score. The average F-score and accuracy results are significantly better than the results in **Table 3**.

The average results for Test 6 also show that the proposed algorithm (with preprocessing and feature fusion) has a superior performance. We are able to achieve the target condition for all datasets along with better classifier accuracy. More especially, the accuracy increased from 87.1% to 91.8% for ICDAR 2003, form 86.4% to 89.5% for SVT, from 85.3% to 88.1% for IIIT5K, from 86.1% to 90.1 for MSRA-TD500 and from 84.7% to 89.9% for KAIST. The average accuracies for an optimal feature set show strong model learning. The dataset ICDAR 2003 attains target condition 95.1% in the 74th generation; SVT attains 92.6% in the 81st generation, IIIT5k attains 92.1% in the 90th generation, MSRA-TD500 attains 93.9% in the 98th while KAIST attains 94.1% in the 93rd generation respectively. Finally, **Table 5** depicts the results of Test 7, which shows a comparison with other state-of-the-art character classification techniques. In general, the proposed algorithm performs better with a classification accuracy of 91.8% for ICDAR-2003, 89.5% for SVT, 88.1% for IIIT5K, 90.1 for MSRA-TD500 and for 89.9% KAIST. Hence, results of **Table 4** are better than results in **Table 3**.

Here, it is also pertinent to mention that although all datasets reflects significant challenging characteristics, while MSRA-TD500 and KAIST are more challenging among the competitors. Since, these both datasets reflects different types of natural scenes (indoor and outdoor) specifically with random orientations and complex background, small distant text, low contrast/resolution images, sign boards, holdings, fences, different fonts, style and sizes. Acquiring certain

TABLE 5. Accuracy based (%) comparison of scene character classification on with existing technique.

Methods	Test 7					
	Datasets					
	Year	ICDAR 2003	SVT	IIIT5K	MSRA-TD500	KAIST
Goel *et al.* [36]	2013	89.7	77.3	-	-	-
Yi *et al.* [37] GHOG+SVM	2013	76.0	-	-	-	-
Yi *et al.* [37] LHOG+SVM	2013	75.0	-	-	-	-
Shi *et al.* [38] TSM	2013	78.0	-	-	-	-
Shi *et al.* [38] HOG+KNN	2013	66.0	-	-	-	-
Yao *et al.* [39]	2013	88.5	75.9	80.2	-	-
Lee *et al.* [34]	2014	81.0	-	-	-	-
Neumann *et al.* [47]	2015	-	68.1	-	-	-
Bai *et al.* [46]	2016	69.0	71.0	-	-	-
Tian *et al.* [45] Conv CoHOG	2016	81.7	77.2	78.8	-	-
Mishra *et al.* [68] CNN feat	2016	86.0	83.0	85.0	-	-
Bai *et al.* [43]	2017	89.2	-	-	-	-
Yin *et al.* [44] (n=3)	2017	84.5	76.5	81.6	-	-
Ansari *et al.* [40]	2018	84.1	81.3	82.9	-	-
Cheng *et al.* [41]	2018	91.5	82.2	87.0	-	-
Gao *et al.* [42]	2019	89.2	82.7	81.8	-	-
Ours		**91.8**	**89.5**	**88.1**	**90.1**	**89.9**

level of accuracy surely reflects the worth of the proposed methodology.

Further, we believe these results shows considerable improvements due to the following reasons: (1) the feature fusion strategy described in **Algorithm I**; (2) the evolutionary nature of the GA algorithm that selects and reduces feature space dimensions gradually at each generation; (3) the implicit use of the SVM, which enhances the accuracy rate.

Finally, **Fig. 11** shows the performance across datasets, where generations indices are shown in the x-axis and the average F-score is shown in the y-axis. From the graph, it is clear that the proposed method performs well for all selected datasets, reaching the target condition. Among all datasets, the results for ICDAR 2003 have the best performance for all selected parameters.

In Test 7, the proposed methodology is considered as a binary classification and then character recognition problem with 62 classes having 10 digit numbers and 52 English alphabets (both upper and lower case). **Table 5** shows that most character classification methods were tested on ICDAR 2003 dataset. In fact, only a few methods were tested on the SVT and IIIT5K datasets. We believe the reason for this is that the SVT and the IIIT5K have a diverse content, composed of outdoor scenes text images, which is very challenging because of the variations in font size, layout, color and the presence of distortions, such as noise, varying illumination, and blur. Notice from the results in **Table 5**, that the proposed method works better than various state-of-the-art methods that are based CNN like for example [40]–[44].

Similar case is with MSRA-TD500 and KAIST, because both datasets reflects diverse scene characteristics. Hence, to the best of our knowledge no such method reported using these datasets for scene character classification. Acquiring certain level of classification accuracy surely reflects the worth of the proposed methodology.

V. CONCLUSION AND FUTURE WORK

The main purpose of our proposed method is to design a text classification method using a feature selection procedure and an optimization algorithm. The proposed methodology achieves high classification accuracy when tested on natural scene text images. A novel average F-Score is defined as a threshold (up to >=92%) for the robust model, which increases the performance on unseen data. The proposed method is tested on five selected datasets. Experimental results have shown that the proposed methodology works well when compared to benchmark feature selection/optimization and existing methods in terms of binary classification. Up to our knowledge, this work is the first attempt to classify text and non-text regions in natural scene images considering diversity in all aspects.

In the future, we plan to incorporate instance selection as the basis for the feature selection or both in parallel (instance and feature selection) for natural scene text classification. We also plan is to implement a dynamic GA rather than a static version for high-quality model learning. This type of scheme tunes the GA parameters dynamically, including varying population size, gene arrangement in the chromosome, and genetic operators. Finally, we plan to test the use of GA with neural networks, which is known to considerably improve the classification of the learning model.

ACKNOWLEDGMENT

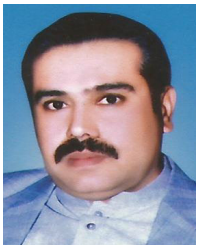
The findings achieved herein are solely the responsibility of the authors.

REFERENCES

[1] Z. M. Hira and D. F. Gillies, "A review of feature selection and feature extraction methods applied on microarray data," *Adv. Bioinf.*, vol. 2015, pp. 1–13, Jun. 2015.
[2] L. Zhao, Q. Hu, and W. Wang, "Heterogeneous feature selection with multi-modal deep neural networks and sparse group LASSO," *IEEE Trans. Multimedia*, vol. 17, no. 11, pp. 1936–1948, Nov. 2015.
[3] S. I. Cho and S.-J. Kang, "Gradient prior-aided CNN denoiser with separable convolution-based optimization of feature dimension," *IEEE Trans. Multimedia*, vol. 21, no. 2, pp. 484–493, Feb. 2019.

[4] O. H. Babatunde, L. Armstrong, J. Leng, and D. Diepeveen, "A genetic algorithm-based feature selection," *Int. J. Electron. Commun. Comput. Eng.*, vol. 5, no. 4, pp. 899–905, 2014.
[5] H. Uğuz, "A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm," *Knowl.-Based Syst.*, vol. 24, no. 7, pp. 1024–1032, Oct. 2011.
[6] C. Ding and D. Tao, "Robust face recognition via multimodal deep face representation," *IEEE Trans. Multimedia*, vol. 17, no. 11, pp. 2049–2058, Nov. 2015.
[7] M. H. Aghdam, N. Ghasem-Aghaee, and M. E. Basiri, "Text feature selection using ant colony optimization," *Expert Syst. Appl.*, vol. 36, no. 3, pp. 6843–6853, Apr. 2009.
[8] M. Nazir, A. Majid-Mirza, and S. Ali-Khan, "PSO-GA based optimized feature selection using facial and clothing information for gender classification," *J. Appl. Res. Technol.*, vol. 12, no. 1, pp. 145–152, Feb. 2014.
[9] Y. Zhang, D.-W. Gong, and J. Cheng, "Multi-objective particle swarm optimization approach for cost-based feature selection in classification," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 14, no. 1, pp. 64–75, Jan. 2017.
[10] X.-F. Song, Y. Zhang, Y.-N. Guo, X.-Y. Sun, and Y.-L. Wang, "Variable-size cooperative coevolutionary particle swarm optimization for feature selection on high-dimensional data," *IEEE Trans. Evol. Comput.*, vol. 24, no. 5, pp. 882–895, Oct. 2020.
[11] Y. Zhang, X.-F. Song, and D.-W. Gong, "A return-cost-based binary firefly algorithm for feature selection," *Inf. Sci.*, vols. 418–419, pp. 561–574, Dec. 2017.
[12] F. Catak, "Genetic algorithm based feature selection in high dimensional text dataset classification," *WSEAS Trans. Inf. Sci. Appl.*, vol. 12, no. 28, pp. 290–296, 2015.
[13] S. Karaoglu, R. Tao, T. Gevers, and A. W. M. Smeulders, "Words matter: Scene text for image classification and retrieval," *IEEE Trans. Multimedia*, vol. 19, no. 5, pp. 1063–1076, May 2017.
[14] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 886–893.
[15] T. Ojala, M. Pietikainen, and D. Harwood, "Performance evaluation of texture measures with classification based on kullback discrimination of distributions," in *Proc. 12th Int. Conf. Pattern Recognit.*, vol. 1, Oct. 1994, pp. 582–585.
[16] J. Sun, X. Zhang, J. Cui, and L. Zhou, "Image retrieval based on color distribution entropy," *Pattern Recognit. Lett.*, vol. 27, no. 10, pp. 1122–1126, Jul. 2006.
[17] W. Shen, X. Wang, Y. Wang, X. Bai, and Z. Zhang, "DeepContour: A deep convolutional feature learned by positive-sharing loss for contour detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3982–3991.
[18] J. Nalepa and M. Kawulok, "Selecting training sets for support vector machines: A review," *Artif. Intell. Rev.*, vol. 52, pp. 857–900, Jan. 2018.
[19] M. Gambhir and V. Gupta, "Recent automatic text summarization techniques: A survey," *Artif. Intell. Rev.*, vol. 47, no. 1, pp. 1–66, Jan. 2017.
[20] J. Liu, H. A. Abbass, and K. C. Tan, "Evolutionary computation," in *Evolutionary Computation and Complex Networks*. Cham, Switzerland: Springer, 2019, pp. 3–22.
[21] H. Vafaie and K. De Jong, "Genetic algorithms as a tool for feature selection in machine learning," in *Proc. 4th Int. Conf. Tools Artif. Intell. (TAI)*, 1992, pp. 200–203.
[22] M. L. Raymer, W. F. Punch, E. D. Goodman, L. A. Kuhn, and A. K. Jain, "Dimensionality reduction using genetic algorithms," *IEEE Trans. Evol. Comput.*, vol. 4, no. 2, pp. 164–171, Jul. 2000.
[23] Z. Sun, G. Bebis, X. Yuan, and S. J. Louis, "Genetic feature subset selection for gender classification: A comparison study," in *Proc. 6th IEEE Workshop Appl. Comput. Vis. (WACV)*, Dec. 2002, pp. 165–170.
[24] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cognit. Neurosci.*, vol. 3, no. 1, pp. 71–86, 1991.
[25] M. S. Mohamad, "Feature selection method using genetic algorithm for the classification of small and high dimension data," in *Proc. Int. Symp. Inf. Comput. Technol.*, 2004, pp. 13–16.
[26] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in *Proc. ICML*, 1997, pp. 412–420.
[27] I. T. Jolliffe, "Principal component analysis and factor analysis," in *Principal Component Analysis*. New York, NY, USA: Springer-Verlag, 1986, pp. 115–128.
[28] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, vol. 13, no. 1, pp. 21–27, Jan. 1967.
[29] J. R. Quinlan, "Induction of decision trees," *Mach. Learn.*, vol. 1, no. 1, pp. 81–106, 1986.
[30] K. Jebari and M. Madiafi, "Selection methods for genetic algorithms," *Int. J. Emerg. Sci.*, vol. 3, no. 4, pp. 333–344, Dec. 2013.
[31] C.-F. Tsai, W. Eberle, and C.-Y. Chu, "Genetic algorithms in feature and instance selection," *Knowl.-Based Syst.*, vol. 39, pp. 240–247, Feb. 2013.
[32] H. Li, D. Yuan, X. Ma, D. Cui, and L. Cao, "Genetic algorithm for the optimization of features and neural networks in ECG signals classification," *Sci. Rep.*, vol. 7, no. 1, p. 41011, Feb. 2017.
[33] M. Sharif, M. A. Khan, M. Faisal, M. Yasmin, and S. L. Fernandes, "A framework for offline signature verification system: Best features selection approach," *Pattern Recognit. Lett.*, vol. 139, pp. 50–59, Nov. 2020.
[34] C.-Y. Lee, A. Bhardwaj, W. Di, V. Jagadeesh, and R. Piramuthu, "Region-based discriminative feature pooling for scene text recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 4050–4057.
[35] T. Wang, D. J. Wu, A. Coates, and A. Y. Ng, "End-to-end text recognition with convolutional neural networks," in *Proc. 21st Int. Conf. Pattern Recognit. (ICPR)*, Nov. 2012, pp. 3304–3308.
[36] V. Goel, A. Mishra, K. Alahari, and C. V. Jawahar, "Whole is greater than sum of parts: Recognizing scene text words," in *Proc. 12th Int. Conf. Document Anal. Recognit.*, Aug. 2013, pp. 398–402.
[37] C. Yi, X. Yang, and Y. Tian, "Feature representations for scene text character recognition: A comparative study," in *Proc. 12th Int. Conf. Document Anal. Recognit.*, Aug. 2013, pp. 907–911.
[38] C. Shi, C. Wang, B. Xiao, Y. Zhang, S. Gao, and Z. Zhang, "Scene text recognition using part-based tree-structured character detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2961–2968.
[39] C. Yao, X. Bai, B. Shi, and W. Liu, "Strokelets: A learned multi-scale representation for scene text recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 4042–4049.
[40] G. J. Ansari, J. H. Shah, M. Yasmin, M. Sharif, and S. L. Fernandes, "A novel machine learning approach for scene text extraction," *Future Gener. Comput. Syst.*, vol. 87, pp. 328–340, Oct. 2018.
[41] Z. Cheng, Y. Xu, F. Bai, Y. Niu, S. Pu, and S. Zhou, "AON: Towards arbitrarily-oriented text recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5571–5579.
[42] Y. Gao, Y. Chen, J. Wang, M. Tang, and H. Lu, "Reading scene text with fully convolutional sequence modeling," *Neurocomputing*, vol. 339, pp. 161–170, Apr. 2019.
[43] X. Bai, B. Shi, C. Zhang, X. Cai, and L. Qi, "Text/non-text image classification in the wild with convolutional neural networks," *Pattern Recognit.*, vol. 66, pp. 437–446, Jun. 2017.
[44] F. Yin, Y.-C. Wu, X.-Y. Zhang, and C.-L. Liu, "Scene text recognition with sliding convolutional character models," 2017, *arXiv:1709.01727*. [Online]. Available: http://arxiv.org/abs/1709.01727
[45] S. Tian, U. Bhattacharya, S. Lu, B. Su, Q. Wang, X. Wei, Y. Lu, and C. L. Tan, "Multilingual scene character recognition with co-occurrence of histogram of oriented gradients," *Pattern Recognit.*, vol. 51, pp. 125–134, Mar. 2016.
[46] X. Bai, C. Yao, and W. Liu, "Strokelets: A learned multi-scale mid-level representation for scene text recognition," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2789–2802, Jun. 2016.
[47] L. Neumann and J. Matas, "Real-time lexicon-free scene text localization and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 9, pp. 1872–1885, Sep. 2016.
[48] Y. Zhang, D.-W. Gong, X.-Z. Gao, T. Tian, and X.-Y. Sun, "Binary differential evolution with self-learning for multi-objective feature selection," *Inf. Sci.*, vol. 507, pp. 67–85, Jan. 2020.
[49] Y. Zhang, Q. Wang, D.-W. Gong, and X.-F. Song, "Nonnegative Laplacian embedding guided subspace learning for unsupervised feature selection," *Pattern Recognit.*, vol. 93, pp. 337–352, Sep. 2019.
[50] I. Ul-Islam, "Feature fusion for pattern recognition," Ph.D. dissertation, Dept. Control Comput. Eng., Politecnico Di Torino, 2015.
[51] L. Mabood, H. Ali, N. Badshah, K. Chen, and G. A. Khan, "Active contours textural and inhomogeneous object extraction," *Pattern Recognit.*, vol. 55, pp. 87–99, Jul. 2016.
[52] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 266–277, Feb. 2001.
[53] F. Juefei-Xu, V. N. Boddeti, and M. Savvides, "Local binary convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 19–28.
[54] A. Alahmadi, M. Hussain, H. Aboalsamh, G. Muhammad, G. Bebis, and H. Mathkour, "Passive detection of image forgery using DCT and local binary pattern," *Signal, Image Video Process.*, vol. 11, no. 1, pp. 81–88, Jan. 2017.
[55] A. Rao, R. K. Srihari, and Z. Zhang, "Spatial color histograms for content-based image retrieval," in *Proc. 11th Int. Conf. Tools Artif. Intell.*, 1999, pp. 183–186.

[56] S. K. S. Modak and V. K. Jha, "Feature level fusion of face and hand for multibiometric based personal identification," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 6, pp. 237–242, 2017.
[57] I.-S. Oh, J.-S. Lee, and B.-R. Moon, "Hybrid genetic algorithms for feature selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 11, pp. 1424–1437, Nov. 2004.
[58] C. Padurariu and M. E. Breaban, "Dealing with data imbalance in text classification," *Procedia Comput. Sci.*, vol. 159, pp. 736–745, 2019.
[59] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, Jun. 2002.
[60] S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young, "ICDAR 2003 robust reading competitions," in *Proc. 7th Int. Conf. Document Anal. Recognit.*, 2003, pp. 682–687.
[61] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 1457–1464.
[62] A. Mishra, K. Alahari, and C. Jawahar, "Scene text recognition using higher order language priors," in *Proc. Brit. Mach. Vis. Conf.*, 2012, pp. 1–25.
[63] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu, "Detecting texts of arbitrary orientations in natural images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1083–1090.
[64] S. Lee, M. Cho, K. Jung, and J. H. Kim, "Scene text extraction with edge constraint and text collinearity," in *Proc. 20th Int. Conf. Pattern Recognit.*, Aug. 2010, pp. 3983–3986.
[65] T. Nowotny, A. Z. Berna, R. Binions, and S. Trowell, "Optimal feature selection for classifying a large set of chemicals using metal oxide sensors," *Sens. Actuators B, Chem.*, vol. 187, pp. 471–480, Oct. 2013.
[66] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Comput. Elect. Eng.*, vol. 40, no. 1, pp. 16–28, Jan. 2014.
[67] B. Xue, M. Zhang, W. N. Browne, and X. Yao, "A survey on evolutionary computation approaches to feature selection," *IEEE Trans. Evol. Comput.*, vol. 20, no. 4, pp. 606–626, Aug. 2016.
[68] A. Mishra, K. Alahari, and C. V. Jawahar, "Enhancing energy minimization framework for scene text recognition with top-down cues," *Comput. Vis. Image Understand.*, vol. 145, pp. 30–42, Apr. 2016.



GHULAM JILLANI ANSARI received the Ph.D. degree in computer science from COMSATS University at Wah, Islamabad. Since 2006, he has been serving in the field of education. He is currently an Assistant Professor with the University of Education at Multan, Lahore. His Ph.D. research belongs to computer visualization and graphics. His research interests include deep learning, data mining, artificial intelligence, machine learning, image processing, and computer vision.

JAMAL HUSSAIN SHAH received the Ph.D. degree in pattern recognition from the University of Science and Technology China, Hefei, China. Since 2008, he has been in the education field. He is currently an Assistant Professor with COMSATS University Islamabad, Wah Cantt, Pakistan. His areas of specialization are automation and pattern recognition. He has 21 publications in IF, SCI and ISI journals and in national and international conferences. His research interests include deep learning, algorithms design and analysis, machine learning, image processing, and big data.

MYLÈNE C. Q. FARIAS (Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of California Santa Barbara (UCSB), USA, in 2004, for work in no-reference video quality metrics. She has worked as a Research Engineer at CPqD, Brazil, in video quality assessment and validation of video quality metrics. She has also worked as an Intern with the Philips Research Laboratories, The Netherlands, in video quality assessment of sharpness algorithms and with Intel Corporation. She is currently an Associate Professor with the Department of Electrical Engineering, University of Brasília (UnB). She is a Researcher with GPDS. Her current interests include video quality metrics, video processing, multimedia signal processing, watermarking, and visual attention. She is a member of the IEEE Signal Processing Society, ACM, and SPIE.ing no-reference video quality metrics.

MUHAMMAD SHARIF (Senior Member, IEEE) received the Ph.D. degree in image processing from the Institute of IT, COMSATS, Islamabad, in 2013. He is currently an Associate Professor with COMSATS University Islamabad, Wah Cantt, Pakistan. He has more than 150 research publications in IF, SCI, and ISI journals and in national and international conferences and received 100+ impact factor. His research interests include medical imaging, biometrics, computer vision, machine learning, and agriculture/plants imaging. He is also currently serving as an Associate Editor for IEEE Access Journal, a Guest Editor for four journal special issues, and a reviewer for many well reputed journals.

NAUMAN QADEER received the B.Sc. degree and the M.Sc. degree in computer science from Bahauddin Zakariya University, Multan, Pakistan, in 1998 and 2001, respectively, and the M.S. degree in computer science from the University of Agriculture, Faisalabad, Pakistan, in 2004. From 2002 to 2005, he worked as a Lecturer with the University of Agriculture. Since 2005, he has been serving as a Lecturer for Federal Urdu University for Arts, Science and Technology, Islamabad, Pakistan. His research interests include multi-agent systems, robotics, image processing, and deep neural networks.

HABIB ULLAH KHAN received the Ph.D. degree in management information systems from Leeds Beckett University, U.K. He is currently an Associate Professor of MIS with the Department of Accounting and Information Systems, College of Business and Economics, Qatar University, Qatar. He has more than 20 years of industry, teaching, and research experience. He is an Active Researcher and his research work published in leading journals of the MIS field. His research interests include the area of IT security, online behaviour, IT adoption in supply chain management, internet addiction, mobile commerce, computer mediated communication, IT outsourcing, big data, cloud computing, and E-learning. He is a member of leading professional organizations, such as IEEE, DSI, SWDSI, ABIS, FBD, and EFMD. He is a reviewer of leading journals of his field and also working as an Editor for some journals.