RESEARCH ARTICLE

Anticancer Peptides Classification Using Kernel Sparse Representation Classifier

EHTISHAM FAZAL[1], MUHAMMAD SOHAIL IBRAHIM[2], SEONGYONG PARK[3,4], IMRAN NASEEM[5,6,7], AND ABDUL WAHAB[8]

[1]Lambda Theta, Karachi 74270, Pakistan
[2]College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China
[3]Cancer Data Science Laboratory, Center for Cancer Research, National Cancer Institute, National Institutes of Health (NIH), Bethesda, MD 20894, USA
[4]Department of Anesthesiology and Pain Medicine, Asan Medical Center, College of Medicine, University of Ulsan, Songpa-gu, Seoul 05505, South Korea
[5]School of Electrical, Electronic and Computer Engineering, The University of Western Australia, Crawley, WA 6009, Australia
[6]College of Engineering, Karachi Institute of Economics and Technology, Korangi Creek, Karachi 75190, Pakistan
[7]Research and Development, Love for Data, Karachi 75600, Pakistan
[8]Department of Mathematics, Nazarbayev University, Astana 010000, Kazakhstan

Corresponding author: Abdul Wahab (abdul.wahab@nu.edu.kz)

ABSTRACT Cancer is one of the most challenging diseases because of its complexity, variability, and diversity of causes. It has been one of the major research topics over the past decades, yet it is still poorly understood. To this end, multifaceted therapeutic frameworks are indispensable. *Anticancer peptides* (ACPs) are the most promising treatment option, but their large-scale identification and synthesis require reliable prediction methods, which is still a problem. In this paper, we present an intuitive classification strategy that differs from the traditional *black-box* method and is based on the well-known statistical theory of *sparse-representation classification* (SRC). Specifically, we create over-complete dictionary matrices by embedding the *composition of the K-spaced amino acid pairs* (CKSAAP). Unlike the traditional SRC frameworks, we use an efficient *matching pursuit* solver instead of the computationally expensive *basis pursuit* solver in this strategy. Furthermore, the *kernel principal component analysis* (KPCA) is employed to cope with non-linearity and dimension reduction of the feature space whereas the *synthetic minority oversampling technique* (SMOTE) is used to balance the dictionary. The proposed method is evaluated on two benchmark datasets for well-known statistical parameters and is found to outperform the existing methods. The results show the highest sensitivity with the most balanced accuracy, which might be beneficial in understanding structural and chemical aspects and developing new ACPs. The Google-Colab implementation of the proposed method is available on the GitHub page (https://github.com/ehtisham-Fazal/ACP-Kernel-SRC).

INDEX TERMS Amino acid composition (AAC), anticancer peptide (ACP), composition of the K-spaced amino acid pairs (CKSAAP), kernel sparse reconstruction classification (KSRC) matching pursuit (MP), over-complete dictionary (OCD), sample-specific classification.

I. INTRODUCTION

According to the *global cancer statistics 2020*, [1], cancer is one of the leading causes of mortality worldwide. It is a diversified group of numerous complicated diseases, rather

The associate editor coordinating the review of this manuscript and approving it for publication was Kin Fong Lei.

than a single one, marked by uncontrolled cell growth and a propensity to rapidly spread or infiltrate other body parts. Cancer's inherent complexity and heterogeneity have proven to be significant barriers to the development of effective anti-cancer therapies [2]. Cancer can be treated with conventional clinical methods such as surgery, radiation, and chemother-apy, but these methods have drawbacks that can be painful

for patients [3]. Although the aforementioned conventional methods deliver positive outcomes, they can also have some substantial adverse effects, including myelosuppression, cardiac toxicity, and gastrointestinal damage [4].

The discovery of anticancer peptides (ACPs) has transformed the paradigm for treating cancer. The ACPs can interact with the anionic cellular components of cancer cells and repair them selectively without harming normal or healthy cells in the body. This amazing feature of the ACPs is vital for therapeutic strategies. The ACPs are typically composed of 5 to 50 amino acids that are often synthesized using antimicrobial peptides (AMPs), many of which have cationic characterists. These features have resulted in the development of novel alternative cancer therapies.

The biggest challenge with the ACPs is distinguishing them from other synthetic or natural peptides [5]. Researchers employ a variety of approaches to identify the ACPs [6]. Although the experimental procedures are gold-standard methods, they are costly and time-consuming, and hence, unsuitable for large-scale searches for prospective ACP candidates. As a result, alternative methodologies for identifying APCs are desired.

Technical advances in artificial intelligence (AI) have substantiated that it is a powerful tool for dealing with incredibly complex situations [7]. Many studies have used machine learning models to predict proteins and classify peptide sequences; see, for instance, [8], [9], [10], [11], [12], and [13]. Even for the ACPs alone, there are several *in silico* approaches for identifying new ACPs. For instance, Tyagi et al. have proposed a *support vector machine* (SVM)-based classification algorithm in [6]. Another study, [14], employed Chou's *pseudo amino acid composition* to predict the ACPs and tested their mutagenicity using the *Ames test*. Generalized chaos game representation methods [15], deep learning-based short-term memory models [5], ensemble learning models [16], augmentation strategies for improved classification performance [17], and *ETree classifiers*-based on *amino acid composition* (AAC) [18] are examples of alternative approaches. Other studies include classical and deep learning-based approaches such as evolutionary genetic algorithm-based ensemble classification of anticancer peptides (iACP-GAEnsC) [19] and deep neural network and skip-gram-based word embedding model (cACP-DeepGram) [20]. Akbar and co-workers have also proposed a variety of other techniques that show comparable performance; see, for instance, cACP [21] and cACP-2LFS [22].

Recently, some state-of-the-art deep learning methods, such as ACP-MHCNN [23], ACP-2DCNN [24], and MLACP 2.0 [25], have been developed. ACP-MHCNN technique uses a multi-headed *convolutional neural network* (CNN) to predict ACP peptides, ACP-2DCNN uses a two-dimensional CNN, and MLACP 2.0 is a combination of seven different classifiers and multiple feature encoding techniques. Some other approaches use neural networks and multitask learning using hybrid sequencing information; see, for instance, ENNAACT [26], XDeep-AcPEP [27], and ACPNet [28].

Although existing machine learning techniques have some advantages for ACP prediction, there is still a need for improvement. For instance, deep learning models provide cutting-edge performance, but their *black-box* nature obscures the classification judgment. A relatively simple model, on the other hand, may not provide appropriate classification accuracy. To that aim, the *sparse-representation classification* (SRC) method provides a great balance, where constrained optimization is a proven method for explainable sparse modeling [29], [30], [31], [32]. In the SRC, a test sample may be reconstructed using a linear combination of dictionary items with sparse weights under the basic principle [33], [34], [35]. The SRC is a non-parametric learning approach in which the magnitude of a sparse vector corresponds to the contribution of the dictionary atoms [36], [37].

Various sparse vector combinations can be used to tackle optimization and ill-posed problems. *Basis-pursuit* (BP) [38], *orthogonal-matching-pursuit* (OMP) [39], and *matching-pursuit* (MP) [40] are some prominent methods for the SRC. These strategies employ l_1-norm regularization to relax l_0-norm rigid sparsity constraint, allowing gradient estimation from continuous error surfaces [41], [42]. The BP furnishes the sparsest solution, but its computing cost grows exponentially. The MP is faster than the BP and OMP, although its sparsity is not guaranteed.

Aside from the optimization approach, the efficacy of the *over-complete dictionary* (OCD) is the most important feature for the construction of an SRC model. In this regard, Zhang et al. [43] proposed a kernel SRC. The kernel mapping converts the nonlinear relationship between different atoms (samples in the OCD) to a linear relationship, allowing the classification of even more complex patterns [7], [43], [44]. Furthermore, a *composition of K-spaced amino-acid pairs* (CKSAAP) is employed to capture a diverse range of peptide sequences, yielding a comprehensive feature vector.

Motivated by the success of the SRC and the *kernel trick*, in this work, we propose to combine polynomial kernel-based *principal component analysis* (PCA) embedding to reduce the feature space dimensions and *synthetic minority oversampling technique* (SMOTE) using *K*-Means [45] to balance the sample space dimension for the construction of the *kernel SRC* model. Details of the proposed approach, including datasets, feature encoding techniques, and classification methods, are furnished in Section II. The experimental analysis and discussion of results are provided in Section III. The paper is concluded in Section IV.

II. PROPOSED APPROACH

We propose a *kernel sparse representation classification* (KSRC) method in this section which includes feature encoding, dimension reduction for *OCD matrix* (ODM) design, and *n*-fold cross-validation for model evaluation. Fig. 1 shows the complete block diagram describing the overall classification process. Individual steps are described in detail in the following subsections.
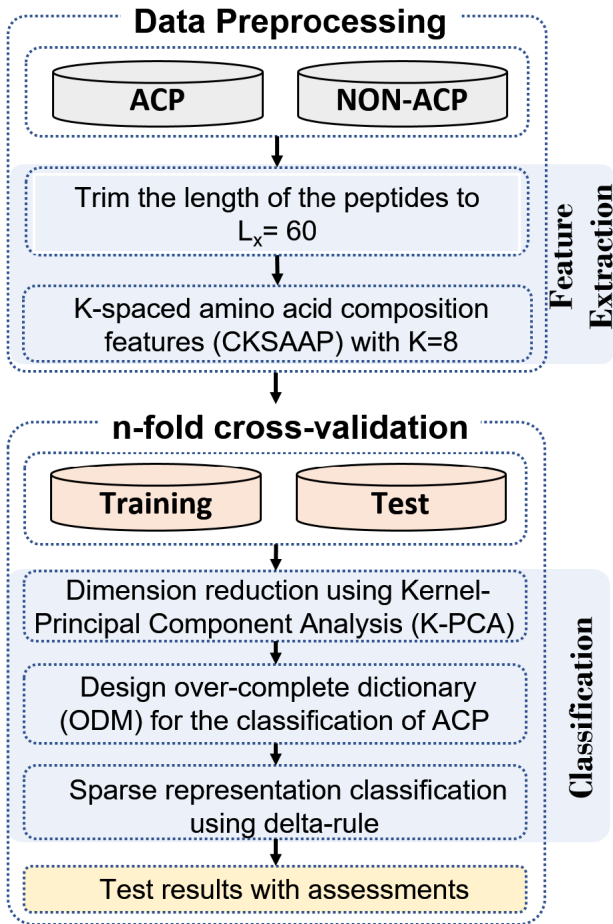
Data Preprocessing

ACP   NON-ACP

↓

Trim the length of the peptides to
$L_x = 60$

↓

K-spaced amino acid composition
features (CKSAAP) with K=8

Feature Extraction

↓

n-fold cross-validation

Training   Test

↓

Dimension reduction using Kernel-
Principal Component Analysis (K-PCA)

↓

Design over-complete dictionary
(ODM) for the classification of ACP

↓

Sparse representation classification
using delta-rule

↓

Test results with assessments

Classification

FIGURE 1. Overview of the proposed ACPs classification strategy.

A. DATASET

There are many datasets available, including those in [14], [46], and [47]. Three benchmark datasets are used in this work to design and evaluate the ACP classification strategy. The first dataset, *ACP344*, was obtained from [14] and contains 344 peptide sequences, 138 of which are ACPs and the remaining 206 are non-ACP samples. The second dataset, *ACP740*, was obtained from previous studies by Chen et al. [46] and Wei et al. [47]). It contains 740 peptide sequences, 376 of which are ACP samples and 364 are non-ACP samples. A filtered and curated version of the *ACP740* dataset can be found in [5]. Different classifiers are designed and evaluated for each dataset according to the protocols reported in the literature. Specifically, 5-fold and 10-fold cross-validation protocols are used for *ACP344*, whereas only the 5-fold cross-validation protocol is used for *ACP740*. In all experiments, the minimum, average, and maximum lengths of the peptide sequences in *ACP344* were set to 11, 26.11, and 97, and in *ACP740* were set to 10, 26.35, and 60, respectively. In the third dataset, two ACP samples were chosen at random from the *ACP740* [5] dataset, and different mutations were developed for mutation sensitivity analysis. It is worth noting that this independent mutant dataset is solely utilized for mutation analysis and is not included in the design of the ODM.

B. FEATURES ENCODING

Protein or peptide sequences are often recorded and stored in *FastA* format, with each amino acid represented by an alphabetic symbol; see, for instance, [48]. These variable-length alphabetic sequences are processed using a variety of sequence encoding techniques, such as AAC, *di-peptide AAC* (DAAC), etc., to extract numerically meaningful features. The AAC is the most basic feature encoding approach, providing a feature vector containing the frequency count of essential amino acids; hence, the overall AAC feature vector length equals the total number of amino acids, i.e., 20. Similarly, the DAAC is the frequency of peptide pairings, with the total length of the feature vector equal to the number of possible combinations of 20 amino acid pairs (i.e., $20 \times 20 = 400$). The DAAC feature vector containing the frequencies of 0-spaced amino acid pairs (i.e., the DAAC of amino acid pairs separated by $K = 0$ residues) is given mathematically by

$$\boldsymbol{\psi}_0 := \left[\frac{\psi_{AA}}{N_0} \quad \frac{\psi_{AC}}{N_0} \quad \frac{\psi_{AD}}{N_0} \quad \cdots \quad \frac{\psi_{YY}}{N_0} \right]^T \in \mathbb{R}^{400}.$$

Here, ψ_{string} is the DAAC descriptor furnishing the frequency of the peptide pairing described by the input *string* and $N_k := L_x - (k+1)$ is the number of local sequence windows defined in terms of the protein sequence length L_x and the number of residues k with $0 \le k \le K$.

Both the AAC and DAAC have widely used sequence encoding methods and have been successfully used to design classifiers for various protein and peptide sequences [13]. However, these techniques are limited in their representation as they do not cover the diverse patterns of the amino-acids pairs. To improve the pattern capture in DAAC, a modified version is proposed in [13] by concatenating the DAAC feature vectors of at most K-spaced amino acid pairs. For example, for $K = 2$, we need to calculate $\boldsymbol{\psi}_k$, for $k = 0, 1, 2$, and the final CKSAAP feature vector, Ψ_K, will be a concatenated version of $\boldsymbol{\psi}_0$, $\boldsymbol{\psi}_1$, and $\boldsymbol{\psi}_2$. Here,

$$\boldsymbol{\psi}_1 := \left[\frac{\psi_{AxA}}{N_1} \quad \frac{\psi_{AxC}}{N_1} \quad \frac{\psi_{AxD}}{N_1} \quad \cdots \quad \frac{\psi_{YxY}}{N_1} \right]^T \in \mathbb{R}^{400},$$

$$\boldsymbol{\psi}_2 := \left[\frac{\psi_{AxxA}}{N_2} \quad \frac{\psi_{AxxC}}{N_2} \quad \frac{\psi_{AxxD}}{N_2} \quad \cdots \quad \frac{\psi_{YxxY}}{N_2} \right]^T \in \mathbb{R}^{400},$$

$$\Psi_K := \left[\boldsymbol{\psi}_0^T \quad \boldsymbol{\psi}_1^T \quad \cdots \quad \boldsymbol{\psi}_K^T \right]^T \in \mathbb{R}^{400(K+1)}.$$

and k represents a gap value used for the calculation of kth DAAC feature vector ($\boldsymbol{\psi}_k$) whereas K is the largest possible gap for which the CKSAAP feature vector Ψ_K is calculated. Fig. 2 shows an example of the $\boldsymbol{\psi}_1$ calculation. Therefore, even when the shortest peptide sequence consists of only 10 amino acids, with CKSAAP of $K = 8$, we can cover both the small and large gap patterns.

C. DIMENSIONALITY REDUCTION USING KERNEL PCA

In machine learning, a large amount of data is often considered useful. A *curse of dimensionality* is nonetheless created when there are few measurements or samples but more attributes (i.e., $A > M$ with A and M being the number
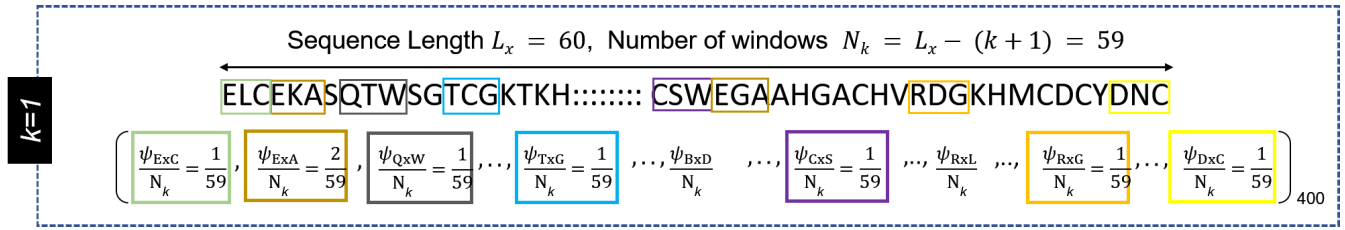
FIGURE 2. Illustration of K-spaced DAAC ψ_k descriptor calculation for $k = 1$. Extracted from [9].

of attributes and measurements, respectively). In this study, our dataset has a small number of samples (less than $1,000$) but the description from the CKSAAP with $K = 8$ contains $3,600$ attributes. This curse of dimensionality not only makes our classification problem ill-posed mathematically but is also very crucial for the design of an OCD for sparse representation with $A < M$. To that end, we suggest using principal component analysis to prone out the least informative dimensions $f < M = N + P$ from the original feature space of $A = 3600$, allowing us to design an ODM of size $f \times M$. Here, N and P are, respectively, the numbers of negative and positive samples in the dictionary.

We employ linear and non-linear projection methods to provide a comparison of the SRC and KSRC methods. A comparison of the eigenvalue spread is presented in Fig. 3 for the *ACP740* dataset. Specifically, from one out of 5-fold cross-validation results (i.e., $n = 5$), the feature set of $A = 3600$ attributes and $M = 740 \times (n-1)/n = 592$ measurements (samples) is projected in the linear and kernel eigenspace and top 600 eigenvalues are plotted. It can be observed that the KPCA compresses the feature dimension more effectively; hence the relative eigenvalues in the KPCA are smaller. It is also worth noting that the actual number of samples used for the design of the OCD is limited by the training samples in n-fold cross-validation. To avoid data leakage, no test sample is used for the estimation of the kernel or for the design of the OCD in our proposed method. First, a PCA projection is learned on training samples, and later the test samples are projected onto the same space using already learned projection.

D. OVER-COMPLETE DICTIONARY MATRIX FOR SPARSE REPRESENTATION CLASSIFICATION

The ODM represents the matrix consisting of feature vectors of ACPs and non-ACPs and is composed of atoms (i.e., training sample vectors). The ODM is used for the SRC in which all ACPs and non-ACPs are characterized using class indices $l = 1$ and $l = 2$, respectively.

In this section, we take f, N, and P as the number of features, training samples with negative classes, and training samples with positive classes, respectively. If $\mathbf{d}_i^{(l)} \in \mathbb{R}^f$ represents the ith training sample from lth class label then the ODM, $\mathbf{D} \in \mathbb{R}^{f \times (N+P)}$, is formed as

$$\mathbf{D} := \begin{bmatrix} \mathbf{d}_1^{(1)} & \mathbf{d}_2^{(1)} & \cdots & \mathbf{d}_N^{(1)} & \mathbf{d}_1^{(2)} & \mathbf{d}_2^{(2)} & \cdots & \mathbf{d}_P^{(2)} \end{bmatrix}.$$
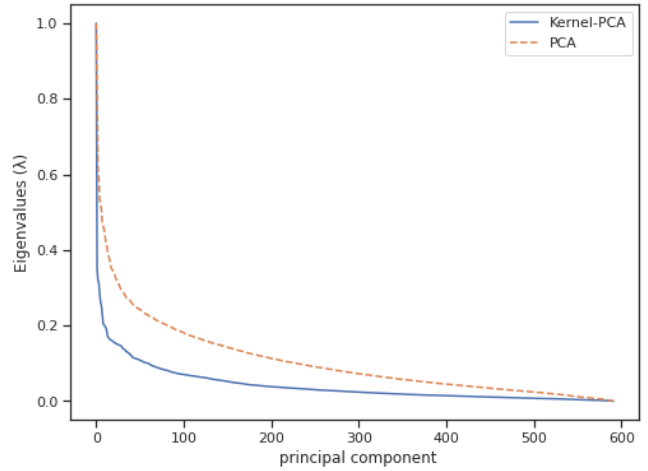

FIGURE 3. Eigenvalues of the top 500 principal components.

A test sample vector $\mathbf{t} \in \mathbb{R}^f$ can be represented as

$$\mathbf{t} = \mathbf{D}\boldsymbol{\gamma},$$

where the coefficient vector $\boldsymbol{\gamma} \in \mathbb{R}^{N+P}$ is defined by

$$\boldsymbol{\gamma} := \begin{bmatrix} \gamma_1^{(1)} & \gamma_2^{(1)} & \cdots & \gamma_N^{(1)} & \gamma_1^{(2)} & \gamma_2^{(2)} & \cdots & \gamma_P^{(2)} \end{bmatrix}^T.$$

If the true class label of the test sample $\mathbf{t}$ is $l = 1$ then all entries $\gamma_1^{(2)}, \gamma_2^{(2)}, \cdots, \gamma_P^{(2)}$ should be zero. Similarly, if the true class label is $l = 2$ then all entries $\gamma_1^{(1)}, \gamma_2^{(1)}, \cdots, \gamma_N^{(1)}$ should be zero. According to *sparse reconstruction theory*, if the dictionary $\mathbf{D}$ is given then the sparse vector $\boldsymbol{\gamma}$ can be recovered [34], [49]. In principle, the sparsest $\boldsymbol{\gamma}$ can be sought as the solution to the optimization problem

$$\arg \min_{\boldsymbol{\gamma}} \|\boldsymbol{\gamma}\|_0 \quad \text{subject to} \quad \mathbf{t} = \mathbf{D}\boldsymbol{\gamma}, \tag{1}$$

where $\|\cdot\|_0$ is the l_0-norm that counts the number of non-zeros entries in the vector.

The constrained optimization problem (1) is non-convex, which makes it difficult to find the optimal vector $\boldsymbol{\gamma}$. Several algorithms for recovering the sparse vector $\boldsymbol{\gamma}$ by solving a *convex relaxation* of the constrained optimization problem (1) have been proposed in the literature. To that end, these algorithms make use of the l_1-norm to solve the *relaxed* optimization problem

$$\arg \min_{\boldsymbol{\gamma}} \|\boldsymbol{\gamma}\|_1 \quad \text{subject to} \quad \mathbf{t} = \mathbf{D}\boldsymbol{\gamma}. \tag{2}$$

Some notable techniques for solving the optimization problem (2) are the BP [38], OMP [39], and MP [40]. Among these algorithms, the BP is considered the most robust method as it furnishes the sparsest solution, but its computing cost grows exponentially. The OMP technique can provide a reasonable trade-off between sparsity and computational complexity, but the latter is also very high. On the other hand, MP is faster than the BP and OMP techniques, although its optimality in terms of sparsity is not guaranteed. In the proposed approach, we use the MP as the l_1-minimization algorithm because of its suitability for the task at hand.

It should be noted that $\boldsymbol{\gamma}$ is expected to contain high-value entries corresponding to the columns of $\mathbf{D}$ that are relevant to the class label of the probe $\mathbf{t}$. This embedded information about the class label of $\mathbf{t}$ can be used to identify $\mathbf{t}$. Let

$$e_l(\mathbf{t}) := \|\mathbf{t} - \mathbf{D}\boldsymbol{\theta}_l(\boldsymbol{\gamma})\|_2, \quad l = 1, 2,$$

where the vector $\boldsymbol{\theta}_l(\boldsymbol{\gamma})$ has all zero entries except at the locations corresponding to class l where the value is one. The decision can be ruled in favor of the class using minimum reconstruction error, i.e.,

$$\text{class-label}(\mathbf{t}) = \arg\min_l \left(e_l(\mathbf{t})\right).$$

E. EVALUATION PROTOCOL

The proposed algorithm has been evaluated for *true positive rate* (TPR) or sensitivity (S_n), *true negative rate* (TNR) or specificity (S_p), *prediction accuracy* (Acc), *Matthew's correlation coefficient* (MCC), *balanced accuracy* (Bal.Acc.), *Youden's index* (YI), and *F1 Score* defined as

$$S_n := \frac{TP}{TP + FN},$$
$$S_p := \frac{TN}{TN + FP},$$
$$\text{Acc.} := \frac{TP + TN}{TP + TN + FP + FN},$$
$$\text{MCC} := \frac{TPTN - FPFN}{\sqrt{\Delta}},$$
$$\text{Bal. Acc.} := \frac{S_n + S_p}{2},$$
$$\text{YI} := S_n + S_p - 1,$$
$$\text{F1 Score} := 2 * \frac{\text{Precision} * S_n}{\text{Precision} + S_n},$$

where TP, FP, TN, and FN indicate the true positive, false positive, true negative, and false negative, respectively. Here,

$$\text{Precision} := \frac{TP}{TP + FP},$$
$$\Delta := (TP + FP)(TN + FN)(TP + FN)(TN + FP).$$

III. EXPERIMENTAL RESULTS

In this section, we perform different experiments to validate our methodology, supporting the selection of various hyper-parameters, solver approaches, embedding strategies, the number of principal components, etc.

A. COMPARISON OF DICTIONARY MATRICES

The robustness and effectiveness of the ODM are the most critical elements of a sparse representation classifier. We employ principal component embedding of the CKSAAP features to create a useful dictionary. In particular, the two most frequently used approaches, polynomial-kernel projection and linear projection, are compared. Three comparison criteria are used: 1) the compactness and compressing power of the embedding method, 2) the linear separability of ACPs and non-ACPs in the embedding space, and 3) the classification performance.

As shown in Fig. 3, the kernel PCA requires fewer components to represent the same amount of information as linear PCA. In Fig. 4, we examine the *area-under-the-receiver-operator-characteristic* (AUROC) curve for the classification of the ACPs from the *ACP344* dataset to further substantiate this assertion. The linear PCA-based SRC and polynomial PCA-based KSRC were tested specifically using dictionaries consisting of the first 10 principal components. The findings show that the KSRC can do better classification with fewer features.
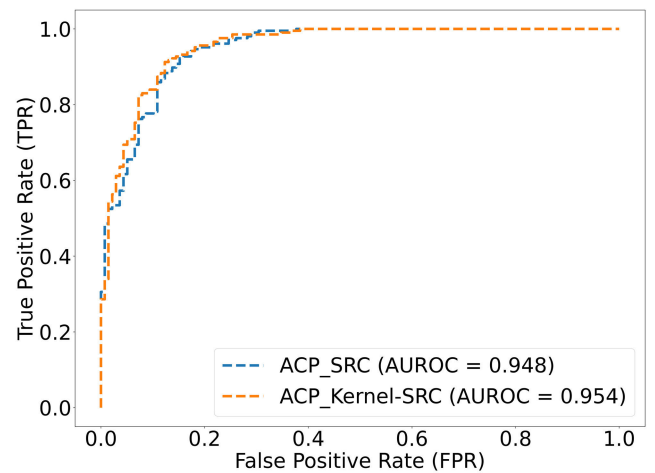

FIGURE 4. Comparison of the AUROC curves of the SRC and KSRC on *ACP344* dataset for same configurations using 10 principal components.

Although compactness is important for sparse representation, the linear separability of class distributions in embedding space is also an important condition. To that end, the *t-distributed stochastic neighbor embedding* (TSNE) [50] plots of linear and kernel PCA embeddings of CKSAAP features of *ACP344* dataset are compared in Fig. 5. Again, the kernel PCA demonstrates superior linear separability between ACPs and non-ACPs samples.

In Fig. 6, we compare the variants of the *ACP344* dataset to further validate the robustness of the OCD method. In particular, the TSNE plots of the kernel PCA embedding of the CKSAAP features from the original *ACP344* dataset and mutants of 138 ACPs from the *ACP344* dataset are compared. The objective of this experiment is to assess the sensitivity of OCD against random mutation. The results show that the separability in the empirical distributions of the ACPs and
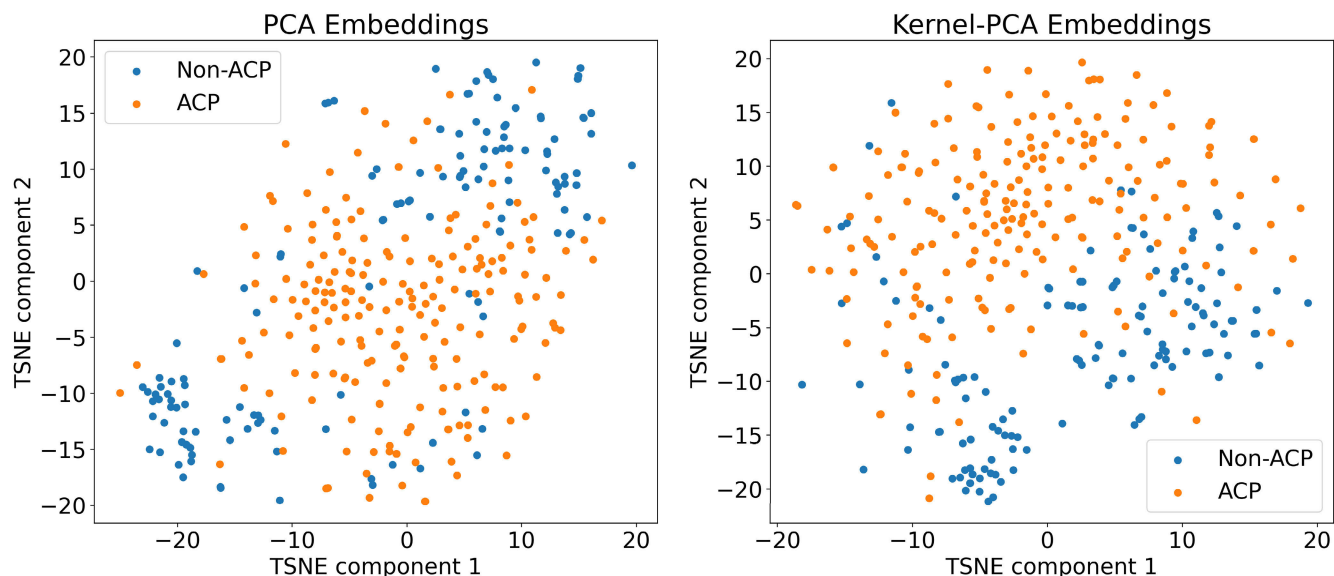
FIGURE 5. Comparison of the linear-PCA and kernel-PCA embeddings of the *ACP344* dataset using the TSNE and 300 principal components.

non-ACPs decreases with the mutation rate. Precisely, when more amino acids in ACPs are mutated, the likelihood of their having anticancer capabilities decreases. A number of statistical methods are available to quantify this distribution separability, ranging from the *strictly standardized mean difference* (SSMD) [51] to distribution overlap [52]. However, we are interested in improving classification performance in our experiments, and accordingly, it is the most important aspect of our analysis.

For the *ACP344* dataset on 10-fold cross-validation, the kernel PCA-based KSRC method achieves the maximum mean MCC of 0.8590 with only 80 principal components. In contrast, the maximum mean MCC of 0.848 was achieved in the linear PCA-based SRC with 300 principal components, which is 1.3% lower than the KSRC performance. In the aforementioned experiments, the MP solver was employed for both the SRC and KSRC methods, while all other settings were unchanged.

B. COMPARISON OF THE OPTIMIZATION ALGORITHMS

1) CLASSIFICATION PERFORMANCE

One major challenge in sparse representation classification is to obtain a suitable solution for the optimization problem (2). The efficiency of the solution depends on a variety of factors, ranging from the quality of the ODM to the robustness of the solver. There are many algorithms available to deal with ill-posed problems like the one given in (2). The popular l_1-minimization algorithms include BP, OMP, and MP. As previously stated, there is a trade-off between the robustness of these algorithms in providing the sparsest solution and their computational efficiency. In the proposed framework, we have adopted the MP algorithm, which is an efficient yet effective l_1-minimization algorithm for the task at hand.

To deal with non-linearity and dimension reduction, the polynomial kernel PCA method [53] is used, while the K-means SMOTE [45] is employed to balance the dictionary. The performance of the proposed method is assessed on the benchmark *ACP344* dataset for a varying number of principal components. The findings in Fig. 7 show that the performance of the proposed MP-based KSRC is similar compared to state-of-the-art BP-based KSRC and OMP-based KSRC methods. Specifically, the proposed MP-based ACP-KSRC achieved a mean 10-fold MCC of 0.8590 with only 80 principal components, while the BP and OMP-based approaches achieved a mean 10-fold MCC of 0.8550 and 0.8419 with 40 and 175 principal components, respectively. This clearly demonstrates the sparse solution recovery of the BP method. However, due to the nature of our investigation, the sparsity of the solution is not the key aspect. Rather, we are concerned with classification performance, which is superior in the case of MP.

2) TIME COMPLEXITY

If we overlook the minor performance gain in MP and examine the run of principal components, we can see that the MP is utilizing more features, which may increase the computational cost. Therefore, we compare the temporal complexity of the preceding experiments for a varying number of principal components. In particular, the *box plots* of all three solvers are shown in Fig. 8 on a semi-log scale, displaying the median and quartile values for total sample reconstruction time. Interestingly, the time complexity of the MP solver is linearly proportional to the number of principal components, but it is exponential in BP and OMP. This means that even with double the number of principal components, the classification time in the MP is still lower than the BP and
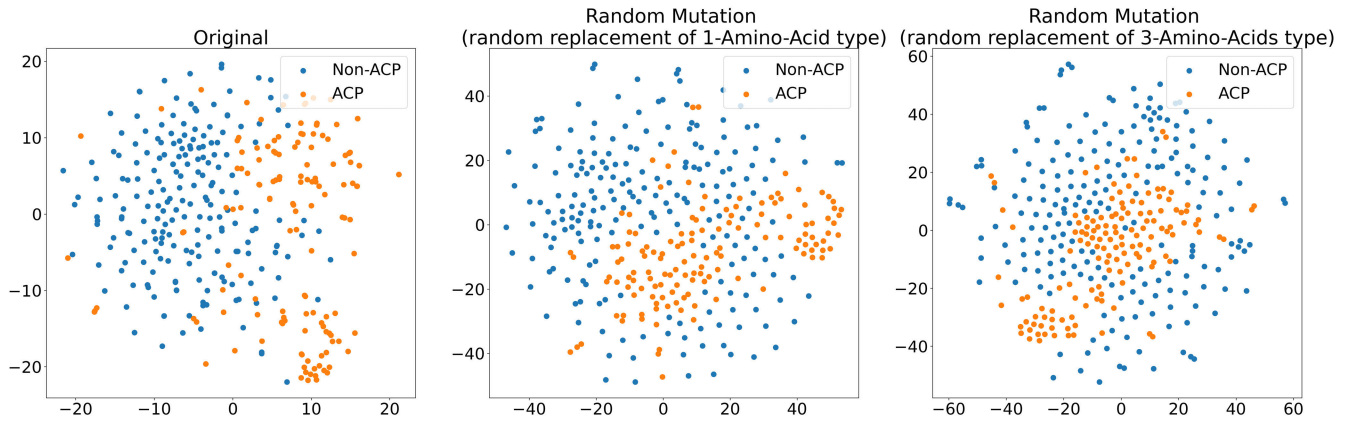
FIGURE 6. Mutation rate and its effect on the feature space. Scatter plot of 2-components of the TSNE of kernel PCA embedded original and mutant ACPs CKSAAP composition features.
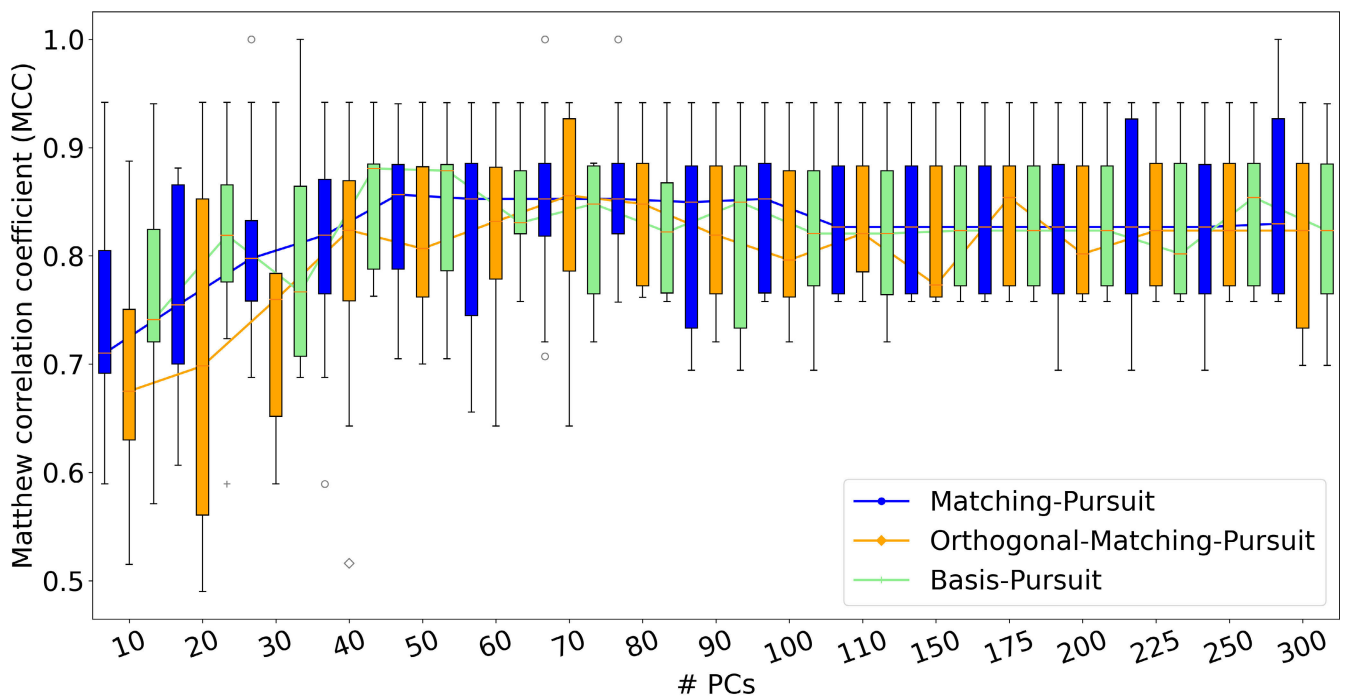

FIGURE 7. Classification performance for different optimization solvers with a varying number of principal components for the *ACP344* dataset.

equivalent to the OMP with 40 principal components. The best performance configuration in the OMP is attained with 175 principal components, the processing time required in the OMP and BP is roughly ten times that of the MP. All experiments were carried out on a freely available *Colab-Notebook* equipped with *Intel Xeon CPU* @2.20 GHz, and 13 GB RAM.

C. COMPARISON WITH STATE-OF-THE-ART ACP CLASSIFICATION APPROACHES

In this section, we compare the performance of the proposed ACP-KSRC method with that of the current state-of-the-art ACP classification algorithms on the *ACP344* [14] and *ACP740* [5] datasets. It should be noted that the proposed ACP-KSRC does not have a training phase, and the training data is only used to construct the dictionary matrix. However, to make a fair comparison, the training and testing samples in all methods were kept consistent as described in previously published research. For instance, the *ACP344* dataset is assessed using 5 and 10-fold cross-validation protocols, whereas the *ACP740* dataset is evaluated using 5-fold cross-validation only. The summary of hyper-parameters used in the study is given in Table 1. Other important details about the individual experiment are given in the relevant subsections.
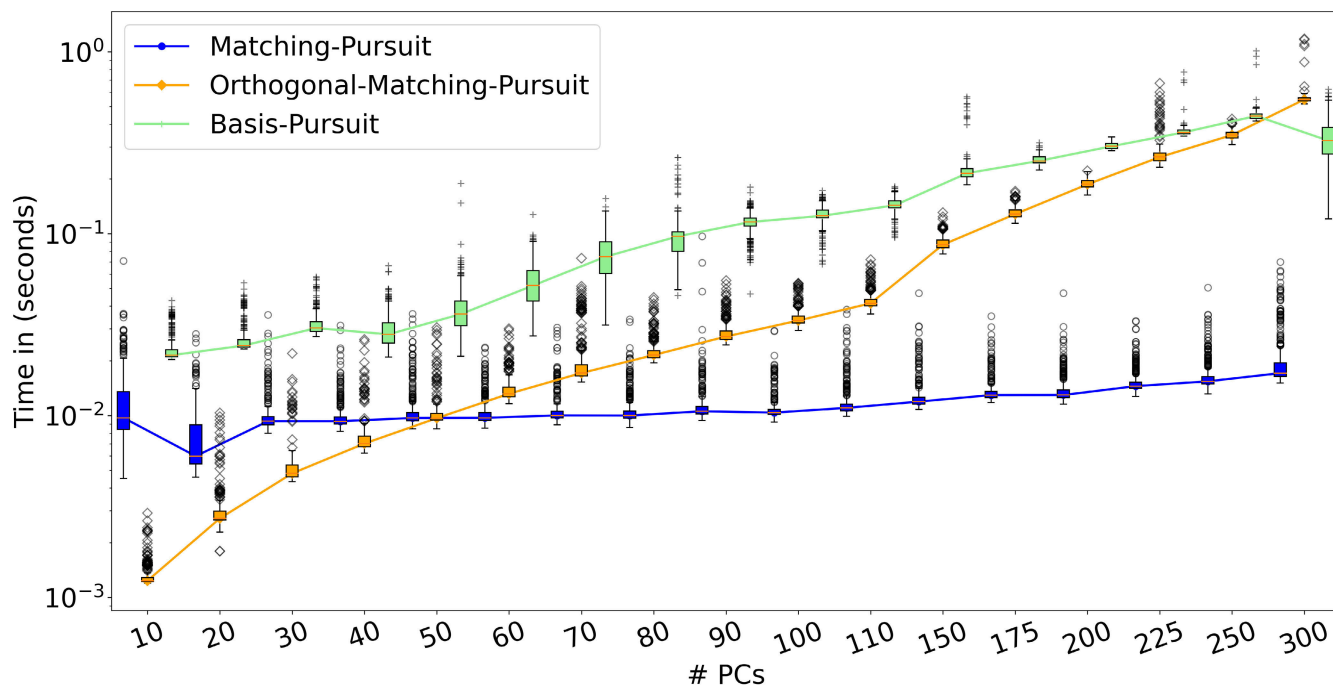
FIGURE 8. Reconstruction time for different configurations and a varying number of principal components for the *ACP344* dataset.

TABLE 1. Summary of hyper-parameters used in the study. Here, *n* is the number of folds in cross-validation, *K* is the maximum space between CKSAAP, *f* is the number of features (attributes), Tol, N_{iter}, and B_{iter} are the tolerance, total number of iterations, and backtracking iterations, respectively.

Dataset	n	K	Kernel	f	Tol	N_{iter}	B_{iter}	Solver
ACP-344	5	8	Poly	80	0.0004	32	100	MP [40]
ACP-344	10	8	Poly	80	0.0004	32	100	MP [40]
ACP-740	5	8	Poly	100	0.0004	32	100	MP [40]

1) ACP344 DATASET

For a fair comparison, the proposed method is evaluated on *ACP344* dataset using two widely used evaluation protocols reported in the literature. In Table 2 and Table 3, we compare the performance statistics of different algorithms on the *ACP344* dataset for 5 and 10-folds cross-validation. For both protocols, the number of principal components for the dictionary matrix is set at $f = 80$ in this experiment. Since the dataset is unbalanced, the conventional accuracy metric is not a suitable representative of the overall performance. To that end, class-specific evaluation parameters such as the MCC and Youden's index are used to indicate the overall classification ability of the classifier.

TABLE 2. Performance comparison of the ACP-KSRC with contemporary methods on 5-fold cross-validation on *ACP344* dataset.

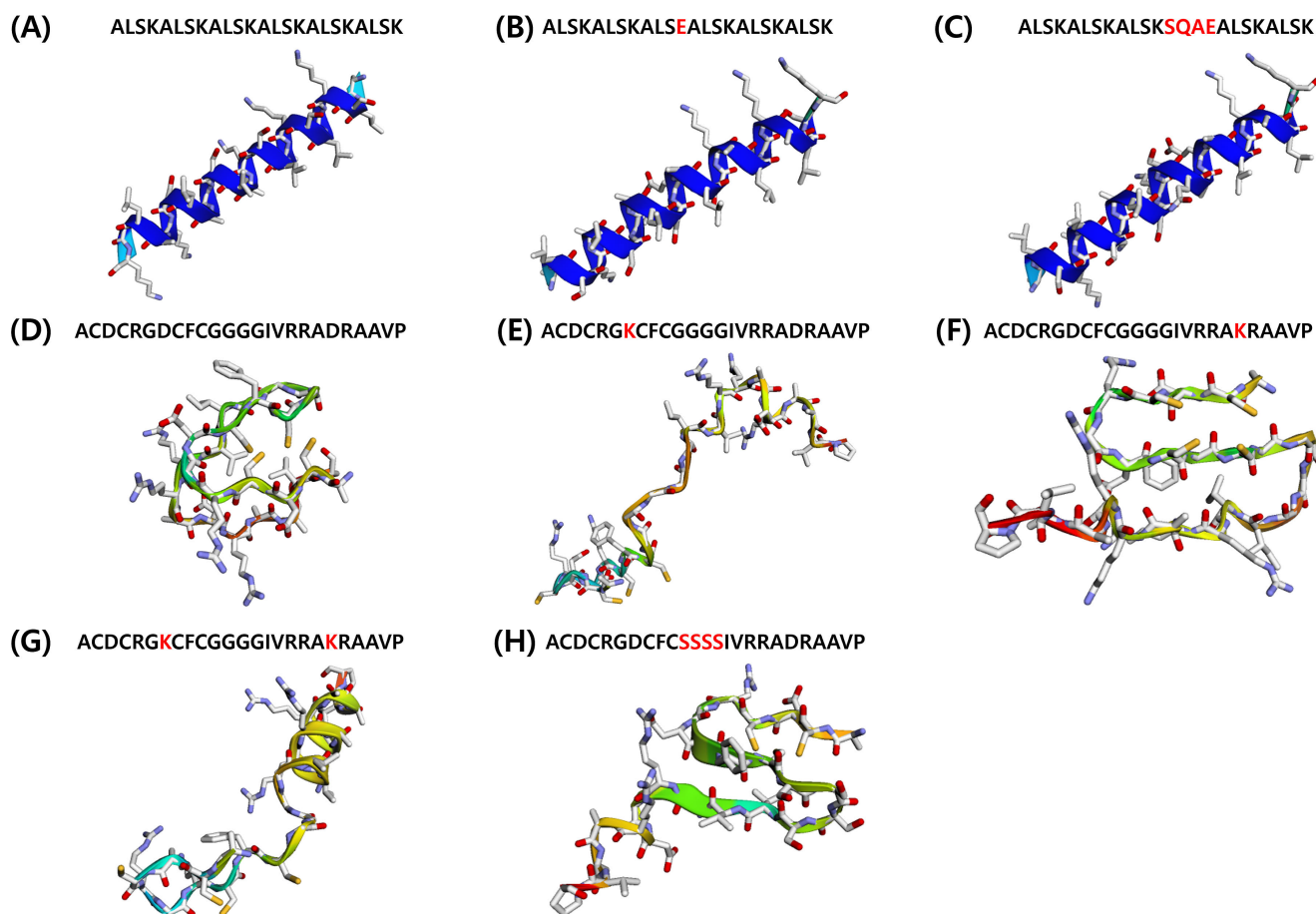| Methods	S_n	S_p	Acc.	Bal. Acc.	MCC	YI	F1-score	Year
ZH method [14], [16]	85.23%	92.73%	89.76%	88.00%	0.78	0.77	0.89	2014
Li method [16], [46]	90.60%	96.70%	94.25%	93.05%	0.87	0.87	0.93	2016
IACP [16], [54]	88.00%	89.10%	94.80%	93.05%	0.89	0.87	0.91	2016
EnACP [16]	92.20%	98.10%	95.41%	95.00%	0.91	0.90	0.94	2020
AntiCP (Model2) [55]	81.32%	90.02%	86.90%	85.07%	0.71	0.71	0.82	2021
ACP-KSRC	96.07%	82.97%	90.02%	89.10%	0.81	0.79	0.92	2023

For 5-fold cross-validation, the proposed method yields the third-best F1-score, demonstrating its ability to effectively differentiate the features of the ACPs. In particular, the ACP-KSRC achieved the MCC value of 0.81 which is 10.99%, 8.99% and 6.90% lower than the EnACP [16], IACP [16], [54], and Li method [16], [46], respectively. It is noteworthy to point out that the *ACP344* dataset is highly skewed, and the number of training samples available in 5-fold cross-validation for the construction of the dictionary matrix is very small, i.e., 110 ACPs and 165 non-ACPs. Since there is no training in the SRC, the performance is highly sensitive to the dictionary samples. This property of the SRC is a double-edged sword; on the one hand, it allows explainability from the given known dictionary atoms, but on the other hand, the diversity of the discriminating feature is also limited when there are fewer atoms. Due to the small number of samples in the *ACP344* dataset, it is suggested to use 10-fold cross-validation.

TABLE 3. Performance comparison of the ACP-KSRC with contemporary methods on 10-fold cross-validation on *ACP344* dataset.

Methods	S_n	S_p	Acc.	Bal. Acc.	MCC	YI	F1-score	Year
SAP [56]	86.23%	95.63%	91.86%	90.93%	0.83	0.81	0.89	2020
ACP-LDF [57]	87.70%	96.10%	92.73%	91.90%	0.84	0.83	0.92	2020
ACP-LDF [57]	85.50%	96.10%	92.15%	91.05%	0.83	0.82	0.92	2020
ACP-LDF [57]	86.20%	97.10%	92.70%	91.65%	0.84	0.83	0.92	2020
ACP-DL [5]	75.82%	86.32%	82.16%	81.07%	0.62	0.62	0.77	2022
ACP-KSRC	97.07%	86.97%	93.02%	91.89%	0.85	0.84	0.94	2023

Notably, the proposed method yields the best results for 10-fold cross-validation, demonstrating its ability to effectively differentiate the features of the ACPs. In particular, the ACP-KSRC achieved the highest MCC value of 0.85 which

(A) ALSKALSKALSKALSKALSKALSK

(B) ALSKALSKALS**E**ALSKALSKALSK

(C) ALSKALSKALSK**SQAE**ALSKALSK

(D) ACDCRGDCFCGGGGIVRRADRAAVP

(E) ACDCRG**K**CFCGGGGIVRRADRAAVP

(F) ACDCRGDCFCGGGGIVRRA**K**RAAVP

(G) ACDCRG**K**CFCGGGGIVRRA**K**RAAVP

(H) ACDCRGDCFC**SSSS**IVRRADRAAVP

FIGURE 9. *AlphaFold2* predicted 3D structures of original and mutant peptide sequences. (A) and (D) structure of original sequences in Table 5. (B), (E) and (F) structures after point mutation. (G) structure after double-point-mutation. (C) and (H) structures after loop mutation.

is 27.06% higher than the ACP-DL, 1.18% higher than the ACP-LDF with the RF and SVM classifiers, 2.35% higher than the ACP-LDF with *LibD3C* classifier, and 4.71% higher than the SAP with the SVM classifier [16]. This supports the claim that the proposed method can be used to predict new ACPs or ACP-like peptides. Other assessment metrics mirror this efficacy, indicating the distinct potential between the ACPs and non-ACPs.

2) ACP740 DATASET

We compare our *ACP740* dataset results with the ACP-DL [5], ACP-DA [17] and ACP-MHCNN [23] in Table 4, as these are the only algorithms that used the *ACP740* dataset. The proposed method outperforms both the ACP-DL [5] and ACP-DA [17] algorithms in terms of the class-specific evaluation parameter MCC for $f = 100$ principal components. In particular, the ACP-KSRC achieved the highest MCC value of 0.67 which is 6% and 4.48% higher than the ACP-DL and ACP-DA, respectively. This efficacy is also reflected in other evaluation metrics, indicating the ability of the ACP-KSRC to discriminate between ACPs and non-ACPs. This suggests that the proposed method can be used to predict ACPs or ACP-like peptides.

TABLE 4. Performance comparison of ACP-KSRC and contemporary methods on 5-fold cross-validation on *ACP740* dataset.

Methods	S_n	S_p	Acc.	Bal. Acc.	MCC	YI	$F1$-score	Year
ACP-DL [5]	82.61%	80.59%	83.48%	83.30%	0.63	0.62	0.71	2019
ACP-DA [17]	86.98%	83.26%	82.03%	85.12%	0.64	0.70	0.85	2021
ACP-MHCNN [23]	88.90%	83.10%	86.00%	86.00%	0.72	0.71	0.86	2021
ACP-KSRC	86.23%	81.62%	83.91%	83.94%	0.67	0.67	0.84	2023

Although the performance of the proposed method is inferior to the state-of-the-art deep learning-based method ACP-MHCNN [23] which provides cutting-edge performance, the proposed method provides a great balance between explainability and prediction performance. In contrast, the *black-box* nature of the ACP-MHCNN obscures the classification judgment. To showcase the explainability of SRC, in the subsequent section, a mutation analysis is performed.

D. MUTATION ANALYSIS

For mutation sensitivity analysis, we have used two ACP samples randomly selected from the *ACP740* dataset. Two separate peptides were chosen to have distinct 3D structures, i.e., one had more redundant amino acids than the other. Different mutants of the sequences were constructed

for sensitivity analysis. Table 5 lists the original and mutant sequences.

TABLE 5. Mutation effect and classification-score sensitivity of ACP-KSRC.

Mutation / (Seq-ID)	Sequence	Classification-Score
Original / (A)	ALSKALSKALSKALSKALSKALSK	0.781
Point Mutation / (B)	ALSKALSKALS**E**ALSKALSKALSK	0.762
Loop Mutation / (C)	ALSKALSKALS**KSQAE**ALSKALSK	0.759
Original / (D)	ACDCRGDCFCGGGGIVRRADRAAVP	0.745
Point Mutation / (E)	ACDCRGD**K**CFCGGGGIVRRADRAAVP	0.705
Point Mutation / (F)	ACDCRGDCFCGGGGIVRRA**K**RAAVP	0.710
Double Point Mutation / (G)	ACDCRGD**K**CFCGGGGIVRRA**K**RAAVP	0.650
Loop Mutation / (H)	ACDCRGDCFC**SSSS**IVRRADRAAVP	0.556

For a single point mutation analysis, one amino acid from the middle of the peptide sequence is substituted with an amino acid having the opposite property (e.g., non-polar to polar, positive to negative charge, etc.). Similar criterion is employed for double point mutations, but the process is repeated for two amino acids. In loop mutation, multiple peptide composition pairs are replaced with their opposite counterparts.

For all the sequences given in Table 5, the classification score is predicted using the proposed ACP-KSRC method. The prediction scores were generated using the OCD of *ACP740* dataset after 10-fold cross-validation and each fold sequence from Table 5 was used as an independent test sample. Finally, the average prediction score was calculated by taking the mean of the individual fold results. It is noteworthy to point out that the mutant sequence dataset was not used in the design of the OCD matrix.

The prediction score, as predicted, is decreasing for both peptides with higher mutation rates. However, the classification score is more sensitive to mutation for sequence (D) as compared to sequence (A). To figure out why this is happening, *Alphafold2* [58] was used to predict the 3D structure of sequences presented in Table 5.

It can be observed in Fig. 9 that the alpha helix structure of sequence (A) is unaffected by point (B) and loop mutation (C), however, the flexible structure of sequence (D) exhibits a notable difference even with a single point mutation (E). This 3D structure-based mutation analysis demonstrates that the prediction score of the proposed ACP-KSRC is sensitive to structural variation and, hence, can be used as a valuable tool for large-scale ACP screening.

IV. CONCLUSION

Cancer, as the most challenging disease due to its complexity and heterogeneity, requires multifaceted therapeutic approaches. Anticancer peptides (ACPs) provide a promising perspective for cancer treatment, but their large-scale identification and synthesis require reliable prediction methods. In this study, we have provided an ACP classification strategy that makes use of sparse representation classification combined with kernel principal component analysis (KSRC). The proposed ACP-KSRC approach relies on the well-understood statistical theory of sparse representation classification, unlike the conventional *black-box* methods.

In particular, we designed the over-complete dictionary matrices using the embedding of the composition of the K-spaced amino-acid pairs (CKSAAP). To deal with non-linearity and dimension reduction, the kernel principal component analysis (KPCA) method is used, while to balance the dictionary, the SMOTE oversampling technique is also utilized. The proposed method is evaluated on two benchmark datasets for well-known statistical parameters and is found to outperform the existing methods. The results indicate the highest sensitivity with the highest balanced accuracy, which can be useful in the understanding of the structural and chemical properties and the development of new ACPs.

REFERENCES

[1] H. Sung, J. Ferlay, R. L. Siegel, M. Laversanne, I. Soerjomataram, A. Jemal, and F. Bray, "Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries," *CA, A Cancer J. Clinicians*, vol. 71, no. 3, pp. 209–249, May 2021.

[2] S. Basith, M. Cui, S. J. Y. Macalino, and S. Choi, "Expediting the design, discovery and development of anticancer drugs using computational approaches," *Current Medicinal Chem.*, vol. 24, no. 42, pp. 4753–4778, Jan. 2018.

[3] I. Kaur, M. N. Doja, and T. Ahmad, "Data mining and machine learning in cancer survival research: An overview and future recommendations," *J. Biomed. Informat.*, vol. 128, Apr. 2022, Art. no. 104026.

[4] D. Basak, S. Arrighi, Y. Darwiche, and S. Deb, "Comparison of anticancer drug toxicities: Paradigm shift in adverse effect profile," *Life*, vol. 12, no. 1, p. 48, Dec. 2021.

[5] H.-C. Yi, Z.-H. You, X. Zhou, L. Cheng, X. Li, T.-H. Jiang, and Z.-H. Chen, "ACP-DL: A deep learning long short-term memory model to predict anticancer peptides using high-efficiency feature representation," *Mol. Therapy -Nucleic Acids*, vol. 17, pp. 1–9, Sep. 2019.

[6] A. Tyagi, P. Kapoor, R. Kumar, K. Chaudhary, A. Gautam, and G. P. S. Raghava, "In silico models for designing and discovering novel anticancer peptides," *Sci. Rep.*, vol. 3, no. 1, pp. 1–8, Oct. 2013.

[7] S. M. Atif, S. Khan, I. Naseem, R. Togneri, and M. Bennamoun, "Multi-kernel fusion for RBF neural networks" *Neural Process. Lett.*, doi: 10.1007/s11063-022-10925-3.

[8] S. Khan, I. Naseem, R. Togneri, and M. Bennamoun, "RAFP-Pred: Robust prediction of antifreeze proteins using localized analysis of *n*-peptide compositions," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 15, no. 1, pp. 244–250, Jan. 2018.

[9] M. Usman, S. Khan, and J.-A. Lee, "AFP-LSE: Antifreeze proteins prediction using latent space encoding of composition of K-spaced amino acid pairs," *Sci. Rep.*, vol. 10, no. 1, pp. 1–13, Apr. 2020.

[10] S. Park, S. Khan, and A. Wahab, "E3-targetPred: Prediction of E3-target proteins using deep latent space encoding," 2020, *arXiv:2007.12073*.

[11] U. M. Al-Saggaf, M. Usman, I. Naseem, M. Moinuddin, A. A. Jiman, M. U. Alsaggaf, H. K. Alshoubaki, and S. Khan, "ECM-LSE: Prediction of extracellular matrix proteins using deep latent space encoding of K-spaced amino acid pairs," *Frontiers Bioeng. Biotechnol.*, vol. 9, p. 878, Oct. 2021.

[12] M. Usman, S. Khan, S. Park, and J.-A. Lee, "AoP-LSE: Antioxidant proteins classification using deep latent space encoding of sequence features," *Current Issues Mol. Biol.*, vol. 43, no. 3, pp. 1489–1501, Oct. 2021.

[13] Z. Chen, P. Zhao, F. Li, A. Leier, T. T. Marquez-Lago, Y. Wang, G. I. Webb, A. I. Smith, R. J. Daly, K.-C. Chou, and J. Song, "iFeature: A Python package and web server for features extraction and selection from protein and peptide sequences," *Bioinformatics*, vol. 34, no. 14, pp. 2499–2502, Jul. 2018.

[14] Z. Hajisharifi, M. Piryaiee, M. M. Beigi, M. Behbahani, and H. Mohabatkar, "Predicting anticancer peptides with Chou's pseudo amino acid composition and investigating their mutagenicity via Ames test," *J. Theor. Biol.*, vol. 341, pp. 34–40, Jan. 2014.

[15] L. Ge, J. Liu, Y. Zhang, and M. Dehmer, "Identifying anticancer peptides by using a generalized chaos game representation," *J. Math. Biol.*, vol. 78, nos. 1–2, pp. 441–463, Jan. 2019.

[16] R. Ge, G. Feng, X. Jing, R. Zhang, P. Wang, and Q. Wu, "EnACP: An ensemble learning model for identification of anticancer peptides," *Frontiers Genet.*, vol. 11, p. 760, Jul. 2020.

[17] X.-G. Chen, W. Zhang, X. Yang, C. Li, and H. Chen, "ACP-DA: Improving the prediction of anticancer peptides using data augmentation," *Frontiers Genet.*, vol. 12, Jun. 2021, Art. no. 698477.

[18] P. Agrawal, D. Bhagat, M. Mahalwal, N. Sharma, and G. P. S. Raghava, "AntiCP 2.0: An updated model for predicting anticancer peptides," *Briefings Bioinf.*, vol. 22, no. 3, May 2021, Art. no. bbaa153.

[19] S. Akbar, M. Hayat, M. Iqbal, and M. A. Jan, "iACP-GAEnsC: Evolutionary genetic algorithm based ensemble classification of anticancer peptides by utilizing hybrid feature space," *Artif. Intell. Med.*, vol. 79, pp. 62–70, Jun. 2017.

[20] S. Akbar, M. Hayat, M. Tahir, S. Khan, and F. K. Alarfaj, "cACP-DeepGram: Classification of anticancer peptides via deep neural network and skip-gram-based word embedding model," *Artif. Intell. Med.*, vol. 131, Sep. 2022, Art. no. 102349.

[21] S. Akbar, A. U. Rahman, M. Hayat, and M. Sohail, "cACP: Classifying anticancer peptides using discriminative intelligent model via Chou's 5-step rules and general pseudo components," *Chemometric Intell. Lab. Syst.*, vol. 196, Jan. 2020, Art. no. 103912.

[22] S. Akbar, M. Hayat, M. Tahir, and K. T. Chong, "cACP-2LFS: Classification of anticancer peptides using sequential discriminative model of KSAAP and two-level feature selection approach," *IEEE Access*, vol. 8, pp. 131939–131948, 2020.

[23] S. Ahmed, R. Muhammod, Z. H. Khan, S. Adilina, A. Sharma, S. Shatabda, and A. Dehzangi, "ACP-MHCNN: An accurate multi-headed deep-convolutional neural network to predict anticancer peptides," *Sci. Rep.*, vol. 11, no. 1, pp. 1–15, Dec. 2021.

[24] A. Ghulam, F. Ali, R. Sikander, A. Ahmad, A. Ahmed, and S. Patil, "ACP-2DCNN: Deep learning-based model for improving prediction of anticancer peptides using two-dimensional convolutional neural network," *Chemometric Intell. Lab. Syst.*, vol. 226, Jul. 2022, Art. no. 104589.

[25] L. T. Phan, H. W. Park, T. Pitti, T. Madhavan, Y.-J. Jeon, and B. Manavalan, "MLACP 2.0: An updated machine learning tool for anticancer peptide prediction," *Comput. Struct. Biotechnol. J.*, vol. 20, pp. 4473–4480, Jan. 2022.

[26] P. B. Timmons and C. M. Hewage, "ENNAACT is a novel tool which employs neural networks for anticancer activity classification for therapeutic peptides," *Biomed. Pharmacotherapy*, vol. 133, Jan. 2021, Art. no. 111051.

[27] J. Chen, H. H. Cheong, and S. W. I. Siu, "XDeep-AcPEP: Deep learning method for anticancer peptide activity prediction based on convolutional neural network and multitask learning," *J. Chem. Inf. Model.*, vol. 61, no. 8, pp. 3789–3803, Aug. 2021.

[28] M. Sun, S. Yang, X. Hu, and Y. Zhou, "ACPNet: A deep learning network to identify anticancer peptides by hybrid sequence information," *Molecules*, vol. 27, no. 5, p. 1544, Feb. 2022.

[29] I. Naseem, S. Khan, R. Togneri, and M. Bennamoun, "ECMSRC: A sparse learning approach for the prediction of extracellular matrix proteins," *Current Bioinf.*, vol. 12, no. 4, pp. 361–368, 2017.

[30] Y. Li, L. Hu, and W. Gao, "Multi-label feature selection via robust flexible sparse regularization," *Pattern Recognit.*, vol. 134, Feb. 2023, Art. no. 109074.

[31] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[32] T. Hofmann, B. Schölkopf, and A. J. Smola, "Kernel methods in machine learning," *Ann. Statist.*, vol. 36, no. 3, pp. 1171–1220, 2008.

[33] M. Usman, S. Khan, S. Park, and A. Wahab, "AFP-SRC: Identification of antifreeze proteins using sparse representation classifier," *Neural Comput. Appl.*, vol. 34, no. 3, pp. 2275–2285, Feb. 2022.

[34] I. Naseem, R. Togneri, and M. Bennamoun, "Sparse representation for speaker identification," in *Proc. 20th Int. Conf. Pattern Recognit.*, Aug. 2010, pp. 4460–4463.

[35] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[36] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. New York, NY, USA: Springer, 2010. [Online]. Available: https://books.google.com.pk/books?id=d5b6lJI9BvAC

[37] S. Li, Q. Li, G. Li, X. He, and L. Chang, "Simultaneous sensing matrix and sparsifying dictionary optimization for block-sparse compressive sensing," in *Proc. IEEE 10th Int. Conf. Mobile Ad-Hoc Sensor Syst.*, Oct. 2013, pp. 597–602.

[38] S. Chen and D. Donoho, "Basis pursuit," in *Proc. 28th Asilomar Conf. Signals, Syst. Comput.*, 1994, pp. 41–44.

[39] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. 27th Asilomar Conf. Signals, Syst. Comput.*, 1993, pp. 40–44.

[40] M. Gharavi-Alkhansari and T. S. Huang, "A fast orthogonal matching pursuit algorithm," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, vol. 3. May 1998, pp. 1389–1392.

[41] L. Zhang and W. Zhou, "On the sparseness of 1-norm support vector machines," *Neural Netw.*, vol. 23, no. 3, pp. 373–385, Apr. 2010.

[42] S. Mandal and A. K. Sao, "Employing structural and statistical information to learn dictionary(s) for single image super-resolution in sparse domain," *Signal Process. Image Commun.*, vol. 48, pp. 63–80, Oct. 2016.

[43] L. Zhang, W. D. Zhou, P. C. Chang, J. Liu, Z. Yan, T. Wang, and F. Z. Li, "Kernel sparse representation-based classifier," *IEEE Trans. Signal Process.*, vol. 60, no. 4, pp. 1684–1695, Apr. 2012.

[44] S. Khan, I. Naseem, R. Togneri, and M. Bennamoun, "A novel adaptive kernel for the RBF neural networks," *Circuits, Syst., Signal Process.*, vol. 36, no. 4, pp. 1639–1653, Apr. 2017.

[45] F. Last, G. Douzas, and F. Bacao, "Oversampling for imbalanced learning based on K-means and SMOTE," 2017, *arXiv:1711.00837*.

[46] W. Chen, H. Ding, P. Feng, H. Lin, and K.-C. Chou, "iACP: A sequence-based tool for identifying anticancer peptides," *Oncotarget*, vol. 7, no. 13, p. 16895, 2016.

[47] L. Wei, C. Zhou, H. Chen, J. Song, and R. Su, "ACPred-FL: A sequence-based predictor using effective feature representation to improve the prediction of anti-cancer peptides," *Bioinformatics*, vol. 34, no. 23, pp. 4007–4016, 2018.

[48] P.-A. Binz, J. Shofstahl, J. A. Vizcaino, H. Barsnes, R. J. Chalkley, G. Menschaert, E. Alpi, K. Clauser, J. K. Eng, L. Lane, and S. L. Seymour, "Proteomics standards initiative extended FASTA format," *J. Proteome Res.*, vol. 18, no. 6, pp. 2686–2692, Jun. 2019.

[49] I. Naseem, R. Togneri, and M. Bennamoun, "Sparse representation for ear biometrics," in *Proc. Int. Symp. Vis. Comput.* Cham, Switzerland: Springer, 2008, pp. 336–345.

[50] A. Gisbrecht, B. Mokbel, and B. Hammer, "Linear basis-function t-SNE for fast nonlinear dimensionality reduction," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jun. 2012, pp. 1–8.

[51] X. D. Zhang, M. Ferrer, A. S. Espeseth, S. D. Marine, E. M. Stec, M. A. Crackower, D. J. Holder, J. F. Heyse, and B. Strulovici, "The use of strictly standardized mean difference for hit selection in primary RNA interference high-throughput screening experiments," *SLAS Discovery*, vol. 12, no. 4, pp. 497–509, Jun. 2007.

[52] S. Park, S. Khan, M. Moinuddin, and U. M. Al-Saggaf, "GSSMD: A new standardized effect size measure to improve robustness and interpretability in biological applications," in *Proc. IEEE Int. Conf. Bioinf. Biomed. (BIBM)*, Dec. 2020, pp. 1096–1099.

[53] B. Scholkopf, A. Smola, and K.-R. Müller, "Kernel principal component analysis," in *Proc. Int. Conf. Artif. Neural Netw.* Cham, Switzerland: Springer, 1997, pp. 583–588.

[54] S. Vijayakumar and L. Ptv, "ACPP: A web server for prediction and design of anti-cancer peptides," *Int. J. Peptide Res. Therapeutics*, vol. 21, no. 1, pp. 99–106, Mar. 2015.

[55] B. Manavalan, S. Basith, T. H. Shin, S. Choi, M. O. Kim, and G. Lee, "MLACP: Machine-learning-based prediction of anticancer peptides," *Oncotarget*, vol. 8, no. 44, pp. 77121–77136, Sep. 2017.

[56] L. Xu, G. Liang, L. Wang, and C. Liao, "A novel hybrid sequence-based model for identifying anticancer peptides," *Genes*, vol. 9, no. 3, p. 158, 2018.

[57] Q. Li, W. Zhou, D. Wang, S. Wang, and Q. Li, "Prediction of anti-cancer peptides using a low-dimensional feature model," *Frontiers Bioeng. Biotechnol.*, vol. 8, p. 892, Aug. 2020.

[58] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Zidek, A. Potapenko, and A. Bridgland, "Highly accurate protein structure prediction with AlphaFold," *Nature*, vol. 596, pp. 583–589, Aug. 2021.

EHTISHAM FAZAL received the B.S. degree in avionics engineering from the College of Engineering, PAF-KIET, Karachi, Pakistan, in 2020, where he is currently pursuing the M.S. degree in electronic engineering. He is also working as a Machine Learning Engineer with Lambda Theta, Karachi. Before joining Lambda Theta, in 2022, he was associated with CLOUDEE.AI, where he worked for more than one year as a Machine Learning Engineer. His research interests include machine learning, computer vision, and bioinformatics.

MUHAMMAD SOHAIL IBRAHIM received the B.E. degree in electronic engineering from Iqra University, Karachi, Pakistan, in 2012, and the M.E. degree in telecommunications from the NED University of Engineering and Technology, Pakistan, in 2016. From 2013 to 2019, he was a Lecturer with the Faculty of Engineering, Science, and Technology, Iqra University. He worked as a Research Assistant with the Embedded Systems Research Group, Karachi Institute of Economics and Technology, Pakistan, from 2013 to 2014. He is currently with the Smart Energy Systems Laboratory, College of Electrical Engineering, Zhejiang University, China. His research interests include deep learning, computer vision, and deep learning applications in energy systems. He was a recipient of the 2020 Highly Cited Review Paper Award from *Applied Energy* (Elsevier), for his review paper titled "Machine Learning Driven Smart Electric Systems: Current Trends and New Perspectives."

SEONGYONG PARK received the Ph.D. degree from the Synergistic Bioinformatics (SynBI) Laboratory, Department of Bio and Brain Engineering, Korea Advanced Institute of Science and Technology (KAIST), South Korea, under the mentorship of Prof. Gwansu-Yi. In 2022, he has completed a Postdoctoral Fellowship at Asan Medical Center (AMC), Seoul, South Korea, where he worked on developing an intelligent patient monitoring system for the operating room. He is currently a Postdoctoral Research Fellow with the Cancer Data Science Laboratory, National Cancer Institute (NCI), National Institutes of Health (NIH), USA. He has been involved in several projects focused on the development of gene expression-based classification models for cancer prognosis and AI-based drug re-purposing models for cancer drug development. Additionally, he has worked on several projects related to the application of machine learning and deep learning in biomedical signal processing, including bioassay, protein sequence, surface-enhanced Raman spectroscopy (SERS), and phonocardiogram (PCG).

IMRAN NASEEM received the B.E. degree in electrical engineering from the NED University of Engineering and Technology, Pakistan, in 2002, the M.S. degree in electrical engineering from the King Fahd University of Petroleum and Minerals (KFUPM), Saudi Arabia, in 2005, and the Ph.D. degree from The University of Western Australia, in 2010. He is currently a Professor with the Department of Electrical Engineering, PAF-KIET, Pakistan. He has an illustrious research background in machine learning with a focus on biometrics and novel adaptive algorithms. He has published over 60 papers in top journals and conferences, including the IEEE Transactions on Pattern Analysis and Machine Intelligence, *Pattern Recognition*, *IJCV*, and the IEEE ICIP. His cornerstone work in face recognition proposed in 2010 is considered to be the basis of the modern deep-learning subspace reconstruction theory with more than 1100 citations to date. He has received several research grants from both national and international funding agencies, including The University of Western Australia, IGNITE (Pakistan), and HEC (Pakistan). His research grant (worth 13.8 million PKR) on developing a face recognition system was highlighted in the national and international media.

ABDUL WAHAB received the M.S. degree from Université Pierre et Marie CURIE (UPMC), Paris, in 2009, and the Ph.D. degree from the École Polytechnique Paris, France, in 2011. He was a Research Professor with the Korea Advanced Institute of Sciences and Technology (KAIST), Daejeon, South Korea, from 2016 to 2018, and a Postdoctoral Researcher with Université Denis Diderot (Paris VII), in 2012. He is currently an Associate Professor with Nazarbayev University, Astana, Kazakhstan. He has published multiple articles in top journals, including the IEEE Transactions on Medical Imaging, the IEEE Transactions on Signal Processing, the IEEE Transactions on Neural Networks and Learning Systems, the *SIAM Journal on Imaging Sciences*, and the *SIAM Journal on Applied Mathematics*. He has coauthored the book *Mathematical Methods in Elasticity Imaging* (Princeton University Press, 2015). His research interests include inverse problems, mathematical imaging, wave propagation, and machine learning. He was a recipient of the Korea Research Fellowship from the National Research Foundation of Korea (NRF), in 2015.

• • •