Journal of Artificial Intelligence and Data Mining (JAIDM), x(x): xxx-xxx, xxxx

Shahrood University of Technology

Journal of Artificial Intelligence and Data Mining (JAIDM)
Journal homepage: http://jad.shahroodut.ac.ir

Research Paper

Selecting Optimal Moments of Chest Images by Partialized-Dual-Hybrid Feature Selection Scheme for Morphological-based COVID-19 Diagnosis

Seyed Alireza Bashiri Mosavi[1*], Mohsen Javaherian[2], Omid Khalaf Beigi[3]

1. Department of Electrical and Computer Engineering, Buein Zahra Technical University, Buein Zahra, Qazvin, Iran.
2. Research Institute for Astronomy and Astrophysics of Maragha, University of Maragheh, 55136-553, Maragheh, Iran.
3. Department of Electrical and Computer Engineering, Kharazmi University, Tehran, Iran.

Article Info

Article History:
Received 24 March 2024
Revised 29 July 2024
Accepted 04 August 2024

DOI:10.22044/jadm.2024.14359.2540

Keywords:
Hybrid feature selection scheme, Hyperplane-based learning methods, Optimal Zernike moments, COVID-19 prediction.

**Corresponding author: abashirimosavi@bzte.ac.ir (S.A Bashiri Mosavi).*

Abstract

One way of analyzing COVID-19 is to exploit X-ray and computed tomography (CT) images of the patients' chests. Employing data mining techniques on chest images can provide significant improvements in the diagnosis of COVID-19. However, in feature space learning of chest images, there exists a large number of features that affect COVID-19 identification performance negatively. In this work, we aim to design the dual hybrid partial-oriented feature selection scheme (DHPFSS) for selecting optimal features to achieve high-performance COVID-19 prediction. First, by applying the Zernike function to the data, moments of healthy chest images and infected ones were extracted. After Zernike moments (ZMs) segmentation, subsets of ZMs ($SZMs^{1:n}$) are entered into the DHPFSS to select $SZMs^{1:n}$-specific optimal ZMs ($OZMs^{1:n}$). The DHPFSS consists of the filter phase and dual incremental wrapper mechanisms (IWMs), namely incremental wrapper subset selection (IWSS) and IWSS with replacement (IWSSr). Each IWM is fed by ZMs sorted by filter mechanism. The dual IWMs of DHPFSS are accompanied with the support vector machine (SVM) and twin SVM (TWSVM) classifiers equipped with radial basis function kernel as $^{SVM}IWSS^{TWSVM}$ and $^{SVM}IWSSr^{TWSVM}$ blocks. After selecting $OZMs^{1:n}$, the efficacy of the union of $OZMs^{1:n}$ is evaluated based on the cross-validation technique. The obtained results manifested that the proposed framework has accuracies of 98.66%, 94.33%, and 94.82% for COVID-19 prediction on COVID-19 image data (CID) including ^{1}CID, ^{2}CID, and ^{3}CID respectively, which can improve the accurate diagnosis of illness in an emergency or the absence of a specialist.

1. Introduction

CORONAVIRUS 2019 (COVID-19) is generally known as a disease with acute respiratory symptoms [1]. Since the coronavirus disease has become one of the dominant infections and the ongoing pandemic during the last two years, lots of research has been carried out to recognize its characteristics. COVID-19, with higher transmission potential, belongs to a large family of viruses whose shape is very similar to monarchical crowns in electron microscope images [2]. The higher death rate of COVID-19 motivated numerous attempts to develop new methods of coronavirus diagnosis [3-5] and analyze both related time series [6, 7] and fatality rate [8, 9]. One of the standard diagnostic methods is using medical X-ray chest images. Pulmonary involvement and lung damages in COVID-19 patients make abnormal patterns in X-ray chest

data differ from healthy chest images [10]. Even for specialists, it is hard to diagnose this disease from images in some cases. Sometimes, health centers cannot correctly diagnose the disease due to the lack of specialists, necessitating automatic methods to detect anomalies and accurately identify patterns. New techniques must also be applicable in a short period with a beneficial treatment procedure.

Because of the applicability of pattern recognition methods in data mining [11-13], there are many papers that exploit various classification approaches in different fields [14-22]. The significance of pattern recognition methods in health care surveys leads to the publishing of many scientific articles for discriminating COVID-19 X-ray images from healthy. In a method developed by [23], more than 96% of X-ray images' classification accuracy was achieved by using transfer learning with convolutional neural networks. Many predictive models have been proposed in the field of automatic COVID-19 disease in X-ray and CT images: Visual Geometry Group (VGG)-16, VGG-19, MobileNet, and InceptionResNetV2 pre-trained models with accuracy more than 90% [24], multi-objective differential evolution method in convolutional neural networks ([25]), ResNet18 convolution neural networks with accuracy more than 96.73% [26], transfer learning with different classifiers leading to more than 96% accuracy for F1-score [27], fine-tuned deep learning algorithms to X-ray images, with the triple accuracy metrics more than 96% [28], VGG19 and U-Net with accuracy of around 97% [29], model based on averaging, normalization, dense, and classification layers with the overall accuracy about 92% [30], coupling the image processing-oriented techniques and deep learning models (DenseNet201, VGG16, and VGG19) with an accuracy of about 95.5% [31]. Some other exploited different classification methods with accuracies of more than 88% are cloud version of Google Auto Machine Learning platforms [32], DenseNet201 [33], explainable deep learning approach [34], Grad-CAM based color visualization approach [35], CNN-based ResNet architecture with using Grad-Cam [36], and residual network-based ReCOV-101 [37]. Other related similar works about the classification of X-ray images can be found in [38], [39], and [40]. Among supervised and unsupervised methods for classifying X-ray images, the feature selection scheme (FSS) is employed as an efficient tool to increase the accuracy of the segregation and reduce the temporal expenses. [41] developed the two-stage procedure including the filter phase and optimization step for FSS. At first, two filtering algorithms (Mutual Information and Relief-F) are used to assign an order to feature attributes. For instance, a meta-heuristic technique (Dragonfly algorithm) is employed to optimize the selected features obtained from the filter phase presented by [41]. In this FS-based COVID-19 predictive models, applying FSS on multivariate features set in the form of one individual window (OIW) leads to neglecting features discarded by survived ones according to FSS modules. Regarding the partitioned-based OIW, we can prevail the defeat of OIW in the multi-window analysis. In the other related work, a hybrid model was proposed to detect lungs infected by COVID-19 in CT scan images [42]. This study involved two phases of recognition based on GoogleNet and ResNet18 methods and a meta-heuristic feature selection technique rooted in the Manta Ray Foraging-based Golden Ratio Optimizer. In these steps, some features are extracted from X-ray chest images using a convolutional neural network in the framework of deep learning, and then the most significant features are selected as final subsets to increase classification accuracies. In this article, the train-test procedures are rooted in conventional approaches that require precise determination of parameters (weight and bias) leading to a hard fine-tuning mechanism. It may not solve the problem of size- and type-independent COVID-19 status prediction vis-à-vis class labeling of new COVID-19 variants.

For solving this concern, the number of learning parameters can be extremely high in the case of using hyperplane-based models. Such a developed mechanism promises high generalization capacity in COVID-19 prediction during different variants. Another feature selection method consisting of both fast and accurate selection stages was proposed by [43]. They obtained the maximum accuracy by applying an enhanced K-nearest neighbor (KNN) classifier to the tested item for segregating the qualified neighbors based on their degree of closeness and strength. In this paper, a hybrid nested multi-level scheme has been introduced in response to vertical design approaches; however, the procedure is exploited by a lazy learner in data classification which can be promoted to the strong learners like support vector machine (SVM) and twin SVM (TWSVM). Having a glance at a few past FSS-based COVID-19 studies shows that designing a comprehensive hybrid scheme for selecting relevant features have remained the most significant concern for timely-accurate COVID-19 prediction.

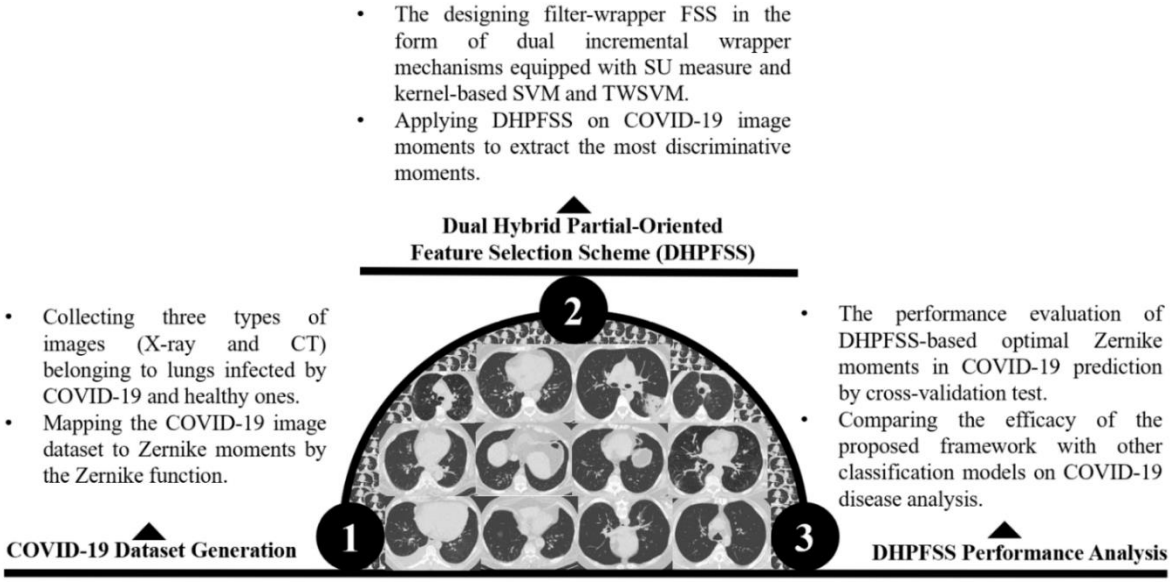


Figure 1. The overall framework of FSS-oriented COVID-19 recognition based on DHPFSS.

Ignoring the polyhedral learning strategy (PLS) leads to labeling optimal-blurred features as redundant. Using vertically one-sided FSS omits the surviving optimal ones involved in recognizing the intrinsic relevance among features. Moreover, proposed FSS in previous studies is applied to COVID-19 multivariate point data in a whole-manner exacerbating sacrificing the optimal-blurred features. Hence, applying PLS to image data based on the partial-oriented strategy can be beneficial in precisely measuring the information rate shared between features per segment.

Summarizing key features and performance metrics of some learning methods (feature selection-classifier-based learning methods (FSCLM) and classifier-based learning methods (CLM)) for COVID-19 prediction is given in Table 1. The key contributions of this paper to address arisen concerns in FSS-based COVID-19 disease analysis are summarized as follows:

- Selecting optimal features based on the novel feature selection algorithm called dual hybrid partial-oriented FSS (DHPFSS) for high-performance COVID-19 recognition is on the agenda of this study. This scheme is designed by information-theoretic criteria and

  hyperplane-based learning models in the form of dual incremental mechanisms.
- The DHPFSS is applied to the high-dimensional COVID-19 dataset to be fragmented into the subsets of Zernike moments ($SZMs^{1:n}$). By exerting the DHPFSS per SZM, the $SZMs^{1:n}$-specific optimal ZMs ($OZMs^{1:n}$) are extracted. The partial view in FSS causes to survive the optimal-blurred moments and promotes the generalization capacity of the learning model in COVID-19 recognition.
- We compared the classification performance of the DHPFSS-based method in COVID-19 prediction with other models.

The rest of the article is organized as follows: the description of the proposed hybrid feature selection algorithm is given in Section 2. The experimental results of applying the proposed framework to the COVID-19 dataset and comparing it with other methods are discussed in Section 3. The concluding remarks are interpreted in Section 4.

2. Proposed method: Dual Hybrid Partial-Oriented Feature Selection Scheme (DHPFSS)

The visual summary of the proposed framework mounted on DHPFSS for COVID-19 disease analysis is depicted in figure 1. As seen in figure 1, in the first step, we gather four types of chest images on COVID-19 grouped based on X-ray and CT scans. Next, we applied the Zernike function to each COVID-19 image data (CID) for extracting the Zernike moments (ZMs) of ones.

Table 1. Summarizing key features and performance metrics of some learning methods (feature selection-classifier-based learning methods (FSCLM) and classifier-based learning methods (CLM)) for COVID-19 prediction

Refs.	key features of FSCM and CM for COVID-19 prediction	Prediction rates	Specific limitations of the method
[41]	FSCLM type: Filter phase (Mutual Information and Relief-F) and Dragonfly algorithm for feature selection and support vector machine (SVM) for data classification	98.39% for SARS-CoV-2 CT images (called ^{2}CID in our paper).	The lack of polyhedral learning strategy (PLS) causes a risk of incorrectly labeling certain features as redundant or unnecessary. This is because without the comprehensive view provided by PLS, valuable features that may appear blurred or less clear in isolation could be mistakenly identified as unimportant. On the other hand, the obtained results are related to the feature selection and data classification process of VCID. The proposed method is not applied to the ^{1}CID and ^{3}CID (x-ray images). Such a learning scenario can cause problems in the classification of X-ray images.
[42]	FSCLM type: Hybrid meta-heuristic FS algorithm (Manta Ray Foraging) based Golden Ratio Optimizer (MRFGRO) and SVM, multilayer perceptron (MLP), and extreme learning machines (ELMs) for data classification	99.42% accuracy for SARS-COV-2 CT images (called ^{2}CID in our paper).	The vertically one-sided FSS is a limited approach that might miss crucial information about the underlying relationships between features. A more sophisticated feature selection method is needed to capture this "intrinsic relevance" among features. On the other hand, the obtained results are related to the feature selection and data classification process of ^{2}CID. The proposed method is not applied to the ^{1}CID and ^{3}CID (x-ray images). Such a learning scenario can cause problems in the classification of X-ray images.
[33]	CLM type: Classification of the COVID-19 infected patients using DenseNet201 based deep transfer learning	۹۶.۲۰% for SARS-CoV-2 CT images (called ^{2}CID in our paper).	In [33], only deep transfer learning has been used for COVID-19 prediction. The train-test processing based on this method in the presence of high-dimensional feature space of COVID data increases the complexity of calculations. Also, the obtained results are related to the feature selection and data classification process of CID^{2}. The proposed method is not applied to the CID^{1} and CID^{3} (x-ray images). Such a learning scenario without discarding the redundant features in features space can cause problems in the classification of X-ray images.
[38]	CLM type: COVID-19 prediction based on convolutional neural network	93.5% for the covid chest x-ray dataset and covid19 x-ray dataset (called ^{1}CID and ^{3}CID respectively in our paper).	Applying convolutional neural networks (CNNs) on COVID-19 prediction without discarding irrelevant data from feature space causes these features to participate in the feature mapping process and consequently negatively affect the accuracy of a CNN-based classification model.

* COVID-19 image data (CID)

After obtaining ZMs per CID ($^{i}CID^{ZMs}; 1 \le i \le 3$), as a preprocessing step, based on the importance of geometric functions (GFs) in the orthogonal polynomials, the normalized magnitude values of ZMs derived from GFs are categorized into the pivotal ZMs (*p*ZMs), secondary ZMs (*s*ZMs), and minor ZMs (*m*ZMs). The *p*ZMs of $^{i}CID^{ZMs}$ ($^{i}CID^{pZMs}$) directly enter into the predictive model without feeding them to DHPFSS. The *s*ZMs of $^{i}CID^{ZMs}$ ($^{i}CID^{sZMs}$) are fed to the proposed feature selection algorithms to select the most relevant moments (MRMs) of $^{i}CID^{sZMs}$. In terms of *m*ZMs of $^{i}CID^{ZMs}$ ($^{i}CID^{mZMs}$), due to the weak role of existing GFs in this bundle for reconstructing images reflected in the low magnitude value of *m*ZMs, the *m*ZMs will not play a role in the learning scenario including the feature selection process and predictive model. In the second step of the proposed framework (See figure 1), we introduce a new hybrid feature selection algorithm called DHPFSS formed by the single filter phase (SFP) and dual incremental wrapper mechanisms. First, $^{i}CID^{sZMs}$ are entered into SFP equipped with symmetric uncertainty (SU) measure to rank the features according to the relevance rate (RR) between feature and class label. It is worth noting that for deep recognition of the intrinsic relationships between features and surviving the features (called optimal-blurred features) that are sacrificed with a slight difference in indices (e.g., RR in the filter phase and prediction accuracy in the wrapper phase) than the other features in the whole-manner strategy ($^{i}CID^{sZMs}$ are entered into the filter phase or wrapper phase in the whole-manner), $^{i}CID^{sZMs}$ are fragmented into *m* subsets (partial-manner strategy). Hence, each $^{i}_{k}CID^{sZMs}$ (Ranked k^{th} subset of ^{i}CID-specific sZMs; $1 \le k \le m$, $1 \le i \le 3$) is entered into the filter phase separately, and then ranked $^{i}_{k}CID^{sZMs}$ ($^{i}_{k}Rank^{sZMs}$) are fed to the integrated dual incremental wrapper phases (IDIWPs) including incremental wrapper-based mechanisms and hyperplane-based classifiers called ^{SVM}IWSS^{TWSVM} and ^{SVM}IWSSr^{TWSVM} blocks. After selecting the MRMs per $^{i}_{k}CID^{sZMs}$ based on dual incremental wrapper mechanisms (IWMs) and dual

hyperplane-based classifiers (${}_k^i CID^{sZMs^{MRMs}}_{IWMs^{Classifiers}}$:[${}_k^i CID^{sZMs^{MRMs}}_{IWSS^{SVM}}$, ${}_k^i CID^{sZMs^{MRMs}}_{IWSS^{TWSVM}}$, ${}_k^i CID^{sZMs^{MRMs}}_{IWSSr^{SVM}}$, ${}_k^i CID^{sZMs^{MRMs}}_{IWSSr^{TWSVM}}$]), the union-intersection operations (UIOs) are applied ${}_k^i CID^{sZMs^{MRMs}}_{IWMs^{Classifiers}}$ for finding the optimal moments (OMs) of ${}^i CID^{sZMs}$ (${}^i CID^{sZMs^{OMs}}$). After selecting ${}^i CID^{sZMs^{OMs}}$, the union of ${}^i CID^{sZMs^{OMs}}$ and ${}^i CID^{pZMs}$ is introduced as the final optimal moments of ${}^i CID^{ZMs}$ (${}^i CID^{ZMs^{FOMs}}$). In the third step, the efficacy of DHPFSS-based ${}^i CID^{ZMs^{FOMs}}$ in COVID-19 analysis is evaluated by the cross-validation test. Furthermore, we compare our proposed framework with other learning models on COVID-19 recognition. According to figure 2, samples of ${}^i CID$ (1st sample of ${}^i CID$ to N^{th} sample of ${}^i CID$) are entered into the Zernike function. The Zernike moments (ZMs) per sample of ${}^i CID$ depend on the importance of GFs in the orthogonal polynomials. Thus, the normalized magnitude values of ZMs derived from GFs is recorded in three formats, namely ${}^i CID^{pZMs}$, ${}^i CID^{sZMs}$, and ${}^i CID^{mZMs}$ (See figure 2; ${}^i CID^{pZMs}$: green face, ${}^i CID^{sZMs}$: blue face, and ${}^i CID^{mZMs}$: red face). From three bundles of the ZMs per sample, only ${}^i CID^{sZMs}$ (medium importance) of samples are fed to the feature selection algorithm to select optimal-blurred moments. In terms of the rest bundles, ${}^i CID^{pZMs}$ goes straight to the next step (predictive model) without feeding to the feature selection algorithm and ${}^i CID^{mZMs}$ are excluded from all learning scenarios (feature selection scheme and predictive model). After extracting ${}^{X\text{-}i} CID^{sZMs}$ per sample (${}^{1\text{-}i} CID^{sZMs}$ to ${}^{N\text{-}i} CID^{sZMs}$), we gathered them in the forms of sZM_1 to sZM_n. Before applying DHPFSS on the ${}^i CID^{sZMs}$-based dataset, the ${}^i CID^{sZMs}$ is fragmented into the m segments (See figure 2; e.g., purple-faced segment, orange-faced segment, and pink-faced segment enclosed by dashed circles). Next, each subset (${}_k^i CID^{sZMs}$; $1 \le k \le m$) is entered into the filter phase as the first step of DHPFSS (See figure 2, filter funnel). Based on information theory concepts in the filter phase, the relevancy rate (RR) of features available in each segment is calculated, and the ranked features (${}_k^i Rank^{sZMs}$) per segment (${}_k^i CID^{sZMs}$) are obtained. In the next step of the proposed FSS, the pair (${}_k^i CID^{sZMs}$, ${}_k^i Rank^{sZMs}$) is fed to the wrapper phase of the DHPFSS (See figure 2; e.g., purple-face square, orange-face square, and pink-face square), which consists of IDWIBs accompanied by ${}^{SVM}IWSS^{TWSVM}$ or ${}^{SVM}IWSSr^{TWSVM}$ blocks. For each pair (${}_k^i CID^{sZMs}$, ${}_k^i Rank^{sZMs}$), for example, in the left spiral of the incremental wrapper phase (LSIWP), the MRMs via ${}^{SVM}IWSS^{TWSVM}$ are obtained (See figure 2; e.g., purple-face star, orange-face star, and pink-face star). Next, the intersection between all pairs (${}_k^i CID^{sZMs^{MRMs}}_{IWSS^{SVM}}$, ${}_k^i CID^{sZMs^{MRMs}}_{IWSS^{TWSVM}}$) is calculated (See figure 2, e.g., the intersection between two purple-face stars, the intersection between two orange-face stars, and the intersection between two pink-face stars). Then, the union operation is exerted on intersected results to achieve the optimal moments derived from LSIWP ($LSIWP^{OMs}$). Such a scenario for each pair (${}_k^i CID^{sZMs}$, ${}_k^i Rank^{sZMs}$) is conducted on the right spiral of the incremental wrapper phase based on ${}^{SVM}IWSS^{TWSVM}$ to achieve $RSIWP^{OMs}$. Finally, the union of $LSIWP^{OMs}$ and $RSIWP^{OMs}$ is obtained as ${}^i CID^{sZMs^{OMs}}$. Regardless of the explanations raised for the overall summary of DHPFSS (See figure 2) mentioned in the previous paragraph, the pseudocode of DHPFSS is shown in table 2. According to table 2, the main body of DHPFSS is mounted on trilateral calculations, namely filter phase (RR analysis), incremental wrapper mechanisms (IWMs), and union-intersection operations (UIOs). After applying the preliminary steps (See Lines 2-5 of Table 2) including specifying the type of ZMs of ${}^i CID^{ZMs}$ (pivotal ZMs, secondary ZMs (*s*ZMs), and minor ZMs) and the feature fragmentation on ${}^i CID^{sZMs}$, the feature ranking is conducted based on RR analysis to obtain ${}_k^i CID^{sZMs}$-specific ${}_k^i Rank^{sZMs}$ (See Line 7 of Table 2). Next, the pair (${}_k^i CID^{sZMs}$, ${}_k^i Rank^{sZMs}$) is entered into IWSS and IWSSr mechanisms (Lines 8 and 9) in twice manner. The two hyperplane-based classifiers (SVM and TWSVM) are situated on branches of IWSS and IWSSr trees for conducting the learning procedures. After selecting MRMs per ${}_k^i CID^{sZMs}$ (${}_k^i CID^{sZMs^{MRMs}}_{IWMs^{Classifiers}}$) based on ${}^{SVM}IWSS^{TWSVM}$ and ${}^{SVM}IWSSr^{TWSVM}$, the ${}_k^i CID^{sZMs^{MRMs}}_{IWMs^{Classifiers}}$ is recorded in the double structure arrays, namely S^{IWSS} and S^{IWSSr} (Lines 10-14).

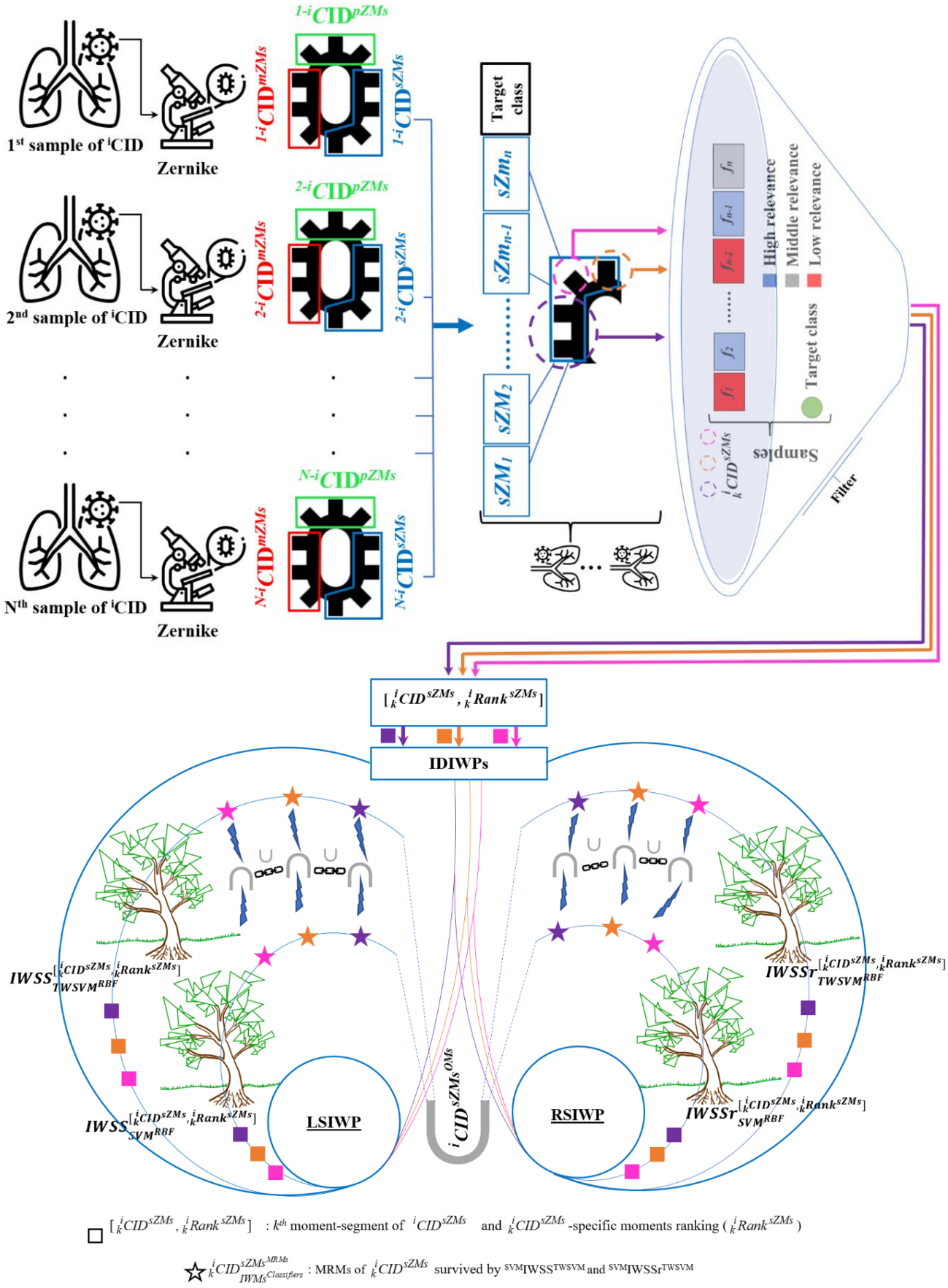


Figure 2. Overall process of DHPFSS.

The S^{IWSS} and S^{IWSSr} contain two columns named R^1 and R^2, which R^1 of S^{IWSS} related to ^{SVM}IWSS results (${}_k^i CID_{IWSS^{SVM}}^{sZMs^{MRMs}}$), R^2 of S^{IWSS} related to ^{TWSVM}IWSS results (${}_k^i CID_{IWSS^{TWSVM}}^{sZMs^{MRMs}}$), R^1 of S^{IWSSr} related to ^{SVM}IWSSr results (${}_k^i CID_{IWSSr^{SVM}}^{sZMs^{MRMs}}$), and R^2 of S^{IWSSr} related to ^{TWSVM}IWSSr results (${}_k^i CID_{IWSSr^{TWSVM}}^{sZMs^{MRMs}}$). Finally, the records of S^{IWSS} and S^{IWSSr} are entered into the UIOs to extract the optimal moments of ${}^i CID^{sZMs}$ called ${}^i CID^{sZMs^{OMs}}$.

For further details on the filter and IWMs phases, refer to Sections 2.1 to 2.2.

2.1. Incremental Wrapper Mechanisms (IWMs)

2.1.1. IWSS

In the proposed FSS, we utilized the IWSS mechanism [44] as a part of the IWMs. This method is an optimal incremental approach based on the combination of features, leading to higher accuracy in the feature selection process. The outputs of the embedded filter and wrapper phase determine the incremental process of IWSS for selecting optimal features. According to the relevancy ratio (RR) in the filter phase, features are arranged in descending order. The empty position in the arrangement is assigned to the selected feature (f_{h1}: feature with the highest rank in RR) in the first increment. The selected feature is then fed to the classification learner (CL) as an input. The prediction accuracy of f_{h1} ($^{Acc}(f_{h1})$) is calculated to identify the candidate features' subset (CFS) during the TTP process. The design of the CFS-based classifier is updated as $^{Acc}(f_{h1}, f_{h2})$ in the second increment by joining the other feature with the next highest rank (RR (f_{h2})) in the arrangement. If the classifier's efficiency with components of $f_{h1}$and f_{h2} shows better performance than with just of f_{h1}, the third increment (joining f_{h3}) will include both previous ones; otherwise, the component f_{h2} is removed. This process is repeated for f_{h3} and all remaining components to achieve the highest efficiency in the CFS-based model. The process of selecting OTFs during the IWSS process is illustrated in figure 3 (left).

2.1.2. IWSSr

The IWSSr [46] algorithm (extended IWSS) is utilized as for the second incremental wrapper mechanism in the DHPFSS. The output of the filter and wrapper method relies on IWSS, where the concept of arranging RRs is applied for features in the first increment. Similar to the previous item, f_{h1} is chosen as CSF, and $^{Acc}(f_{h1})$ is determined by CL trained with f_{h1}. After joining f_{h2} to the CSF in the second increment, there are two scenarios. In the first case, f_{h1} is removed, and the CSF is rebuilt by f_{h2}. In the second case, both f_{h1} and f_{h2} are inputted to the CL model. Here, the code calculates both $^{Acc}(f_{h1})$ and $^{Acc}(f_{h1}, f_{h2})$ in this step. Figure 3 (right) provide the detailed explanations of the IWSSr procedure.

2.2. Integrated Dual Incremental Wrapper Phases (IDIWPs)

The IWSS and IWSSr are considered as the optimal incremental approaches based on the combination of features, leading to higher accuracy in the feature selection process. The outputs of the embedded filter and wrapper phase (SVM and TWSVM) determine the incremental process of IWSS and IWSSr for selecting optimal features. By injecting the results of the single filter phase into dual IWMs, the train-test procedures are triggered on leaves (nodes) of IWSS and IWSSr trees via varied kernelized hyperplane-based classifiers (SVM and TWSVM). Ignoring the incremental spirit of IWSS and IWSSr algorithms causes the features selected by the filter phase to directly enter into the classifiers as wrappers (vertically integrated view/ solid view) and the obtained result is reported as the classification accuracy without giving a chance to share other features in the classification problem. However, IWSS-and IWSSr-specific replacement mechanisms in the constitute candidate feature subset caused the wrapper phase (SVM and TWSVM) to continuously face a changing feature space (based on the IWSS and IWSSr policies in adding features to the leaves of trees) and report the classification accuracy resulting from changes in a candidate feature subset. In such circumstances, IWSS and IWSSr can populate the candidate feature subsets with features that may even have a low SU rate, but these features increase the prediction accuracy. Hence, IWSS and IWSSr are the pivot mechanisms in the proposed learning model. The IDWIBs in DHPFSS are formed based on LSIWP and RSIWP (See figure 2, wrapper phase), wherein LSIWP and RSIWP contain $^{SVM}IWSS^{TWSVM}$ and $^{SVM}IWSSr^{TWSVM}$, respectively. By injecting the results of the single filter phase into dual IWMs, the train-test procedures are triggered on leaves (nodes) of IWSS and IWSSr trees via varied hyperplane-based classifiers (SVM and TWSVM). Generally, in the LSIWP, the FSS-based learning scenario is based on

$IWSS_{SVM^{RBF}}^{[{}_k^iCID^{sZMs},{}_k^iRank^{sZMs}]}$ and $IWSS_{TWSVM^{RBF}}^{[{}_k^iCID^{sZMs},{}_k^iRank^{sZMs}]}$. In respect of RSIWP, $IWSSr_{SVM^{RBF}}^{[{}_k^iCID^{sZMs},{}_k^iRank^{sZMs}]}$, and $IWSSr_{TWSVM^{RBF}}^{[{}_k^iCID^{sZMs},{}_k^iRank^{sZMs}]}$. After completing the FSS-based learning procedures on IDIWPs, UIOs are conducted based on the obtained results. The detailed descriptions of the filter and incremental wrapper models of DHPFSS are elaborated in the following subsections.

2.2.1. Single Filter Phase (SFP) in DHPFSS

Relevancy ratio (RR): In the filter phase of the proposed FSS, the amount of information shared between sZMs of ${}_k^iCID^{sZMs}$ and the target class is calculated via triple information theory-based tools namely, entropy, conditional entropy, and mutual information (MI). These tools are interlaced in the form of symmetrical uncertainty (SU) index. The SU is adopted to measure the relevance between the feature and the class label. The average normalized interaction gains of feature *f*, every other feature, and the class label are calculated to reflect the interaction of feature *f* with other features in the feature set *F*. The SU metric normalizes MI by scaling its range to [0,1]. SU of variables *X* and *Y* is defined based on MI which is a technique to measure the relevance between two random variables. If *X* and *Y* are two features, then MI (*X*; *Y*) measures how much information feature *X* contains about feature *Y*. If *X* is a feature and *Y* is a class label, then MI (*X*; *Y*) measures how much information feature *X* contains about the class label *Y*. The SU index, defined as:

$$SU({}_k^iCID^{sZMs^j},C)=2\frac{MI({}_k^iCID^{sZMs^j};C)}{H({}_k^iCID^{sZMs^j})+H(C)} \quad (1)$$

where *j* in $sZMs^j$ reflects the j^{th} moments of *sZMs* set, and *C* is the target class of ${}^iCID^{sZMs}$. The *H(D)* in (1) is called entropy and expressed as:

$$H(D)=-\sum_{d\in D}p(d)\log p(d) \quad (2)$$

where *D* is a discrete random variable, and $p(d)=\Pr\{D=d\}$ is a probability density function.

Mutual information (*MI*) in (1) is as follow:

$$MI({}_k^iCID^{sZMs^j};C)=H({}_k^iCID^{sZMs^j})-H({}_k^iCID^{sZMs^j}\mid C) \quad (3)$$

where $H({}_k^iCID^{sZMs^j}\mid C)$ in (3) is called conditional entropy and expressed as follows:

$$H({}_k^iCID^{sZMs^j}\mid C)=-\sum_{x\in {}_k^iCID^{sZMs^j}}\sum_{c\in C}p(x,c)\log p(x\mid c) \quad (4)$$

2.2.2. Support Vector-based Wrappers in IDIWPs

2.2.2.1. SVM

The mathematical classification model designed based on the idea of maximizing the margin between binary classes was introduced by [47]. This robust supervised method was named SVM, one of the applicable and efficient machine learning algorithms. According to the structural-risk minimization, linearly separable data in feature space can be grouped in binary or multi-class sets using hard or soft margin in SVM. On the other hand, data sets with a nonlinear distribution in feature space would be separated by using kernel functions. In fact, the aim of SVM is to find a hyperplane (decision boundary) between data sets and then maximize the margin between two parallel hyperplanes as large as possible to achieve the highest accuracy. For the general case that we deal with nonlinearly separable data sets, the kernel (*K*) is embedded in the optimization formulation as follows:

$$a^*=\arg\ \min_\alpha \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j K(x_i,x_j)-\sum_{k=1}^{l}\alpha_k; \quad (5)$$

$$0\le\alpha_i\le C,\quad \sum_{j=1}^{l}\alpha_i y_i=0,\quad i,j=1,...,l$$

The function *K* has the principal role in making data sets linearly separable by mapping them from the original space to the high-dimensional feature space. The radial basis function *(*RBF) [47] is used $K(x,x')$ in (5) and defined as follows:

$$K(x,x')=\exp\left(-\frac{\|x-x'\|^2}{2\sigma^2}\right) \quad (6)$$

The squared Euclidean distance $\|x-x'\|^2$ in (6) calculates the distance between two data points. The separating hyperplane with the maximum margin can be attained by solving the following problem:

$$f(x)=\operatorname{sgn}\left(\sum_{i\in s}\alpha_i y_i K(x_i,x)+b\right);$$

$$b=\frac{1}{s}\sum_{i\in s}\left[y_i-\sum_j\alpha_j y_j K(x_j,x_i)\right] \quad (7)$$

2.2.2.2. TWSVM

According to the idea proposed in the SVM, we are able to construct the structural-risk minimization problem with the condition of using nonparallel hyperplanes for separating data sets. In this case, the distance between data sets becomes larger and causes a reduction in the classification error. Roughly speaking, in binary data classification, each hyperplane takes the closest distance from the samples of a class and the farthest distance from another one. The idea of constructing this type of learning model is primarily introduced in the generalized proximal eigenvalue support vector machine (GEPSVM) [47], and then developed with a new skeleton as TWSVM [48]. To find the equations of nonparallel hyperplanes, one can solve the following optimization problems:

$$\min_{w_1,b_1,q} \frac{1}{2}\| Pw_1 + e_1 b_1 \|^2 + c_1 e_2{}^T q \quad s.t. \quad -(Qw_1 + e_2 b_1) + q \geq e_2, \ q \geq 0 \tag{8}$$

$$\min_{w_2,b_2,q} \frac{1}{2}\| Qw_2 + e_2 b_2 \|^2 + c_2 e_1{}^T q \quad s.t. \quad (Pw_2 + e_1 b_2) + q \geq e_1, \ q \geq 0 \tag{9}$$

where c_1 and c_2 are used as the regularization parameters, and e_1 and e_2 are arrays with element values of one. Using the Karush–Kuhn–Tucker (KKT) conditions and the Lagrangian method in (10) and (11), we can achieve the following dual form of optimization problems:

$$^{dual}TWSVM^1: \ \max\{e_2^T\alpha - \frac{1}{2}\alpha^T G(H^T H)^{-1}G^T\alpha\} \tag{10}$$

$$^{dual}TWSVM^2: \ \max\{e_1^T\psi - \frac{1}{2}\psi^T P(Q^T Q)^{-1}P^T\psi\} \tag{11}$$

A general approach for solving this type of equations in terms of α and ψ is quadratic programming. So, the solution to problems appears as arrays [$w^{(1)}$, $w^{(2)}$] and parameters [$b^{(1)}$, $b^{(2)}$] for the following nonparallel hyperplanes:

$$X^T w^{(1)} + b^{(1)} = 0 \text{ and } X^T w^{(2)} + b^{(2)} = 0 \tag{12}$$

For linearly separable data, the class labels of features are determined by planes held in the following relations:

$$Class \ \ x = \arg_v \min \left| x^T w^{(v)} + b^{(v)} \right|; \ \ v = 1, 2 \tag{13}$$

In the case of dealing with nonlinearly separable data, we need to insert K in equations to obtain the equations of nonparallel hyperplanes [48]:

$$K(x^T, C^T)u^{(1)} + b^{(1)} = 0 \text{ and } K(x^T, C^T)u^{(2)} + b^{(2)} = 0 \tag{14}$$

where $C^T = [A \ B]^T$, and K denotes the RBF kernel (See Equation (6)). Finally, to find $[u^{(1)} \ b^{(1)}]^T$ and $[u^{(2)} \ b^{(2)}]^T$, the following dual optimization problems must be solved:

$$^{K}TWSVM^1: \min_{u^{(1)},b^{(1)},q} \frac{1}{2}\| (K(A,C^T)u^{(1)} + e_1 b^{(1)} \|^2 + c_1 e_2{}^T q \quad s.t. \quad -(K(B,C^T)u^{(1)} + e_2 b^{(1)}) + q \geq e_2, \ q \geq 0 \tag{15}$$

$$^{K}TWSVM^2: \min_{u^{(2)},b^{(2)},q} \frac{1}{2}\| (K(B,C^T)u^{(2)} + e_2 b^{(2)} \|^2 + c_2 e_1{}^T q \quad s.t. \quad (K(A,C^T)u^{(2)} + e_1 b^{(2)}) + q \geq e_1, \ q \geq 0 \tag{16}$$

The proposed method works in the framework of robust information theory based on a novel approach concerning dual incremental mechanisms equipped with hyperplane-based classifiers has high compatibility with high-dimensional datasets and different types of new variants. The hyperplane-based approach like SVM and TWSVM is the best option that maximizes predictive accuracy without overfitting training data. In the case of SVM, it employs a separating hyperplane with low structural risk in the classification of data and is not linearly separable in feature space. Furthermore, by applying the proper kernel, we can increase the generalization capacity of the learning model (optimal matching between transient samples). On the other hand, TWSVM which is decorated by the spirit of the SVM into a new skeleton caused each hyperplane to take the closest distance from the samples of a class and the farthest distance from another one. TWSVM formulation causes that to be more able to construct the structural-risk minimization problem with the condition of using nonparallel hyperplanes for separating data sets. TWSVM helps the distance between data sets become larger and reduces the classification error. Taking into cognizance the above-mentioned points, motivated us to utilize SVM and TWSVM as the basic classifiers in wrapper mechanisms situated on the proposed FSS.

According to what was discussed above, the computational cost of the DHPFSS accompanied by IWMs and hyperplane-based predictive models can be approximated by analyzing main functions. The most expensive cases of operating IWSS and IWSSr have the complexities of $O(n)$ and $O(n^2)$, respectively. In the presence of the SVM and TWSVM within the IWMs tree, the complexities are $O(n^3)$ and $O(2\times(n/2)^3)$, respectively. Hence, $IWSS^{SVM/TWSVM}$ and $IWSSr^{SVM/TWSVM}$ have the complexities of $O(\max\{(n\times n^3), (n\times 2\times(n/2)^3)\})$ and $O(\max\{(n^2\times n^3), (n^2\times 2\times(n/2)^3)\})$, respectively.

Table 2. The pseudocode of the DHPFSS

Main body of DHPFSS

Input: ${}^{i}CID^{sZMs}$; $\{1 \le i \le 3, 1 \le k \le m\}$.

Output: Optimal moments of ${}^{i}CID^{sZMs}$ (${}^{i}CID^{sZMs^{OMs}}$).

(1) **for** *i*=1 to 3 // three Covid-19 image data
(2) ${}^{i}\text{CID}^{pZMs} = [{}^{1\text{-}i}\text{CID}^{pZMs}, {}^{2\text{-}i}\text{CID}^{pZMs}, \ldots, {}^{N\text{-}i}\text{CID}^{pZMs}]$; // No participate in feature selection process.
(3) ${}^{i}\text{CID}^{sZMs} = [{}^{1\text{-}i}\text{CID}^{sZMs}, {}^{2\text{-}i}\text{CID}^{sZMs}, \ldots, {}^{N\text{-}i}\text{CID}^{sZMs}]$; // participate in feature selection process.
(4) ${}^{i}\text{CID}^{mZMs} = [{}^{1\text{-}i}\text{CID}^{mZMs}, {}^{2\text{-}i}\text{CID}^{mZMs}, \ldots, {}^{N\text{-}i}\text{CID}^{mZMs}]$; // No participate in the feature selection process.
(5) $[{}^{i}_{1}CID^{sZMs}, {}^{i}_{2}CID^{sZMs}, \ldots, {}^{i}_{m}CID^{sZMs}]$ =feature fragmentation (${}^{i}\text{CID}^{sZMs}$);
(6) **for** *k*=1 to m // m=number of subset of ${}^{i}CID^{sZMs}$ (${}^{i}_{k}CID^{sZMs}; 1 \le k \le m$)
(7) ${}^{i}_{k}Rank^{sZMs}$ = sort [calculate relevancy rate (RR) of ${}^{i}_{k}CID^{sZMs}$];
(8) ${}^{i}_{k}CID^{sZMs^{MRMs}}_{IWSS^{SVM}}$ = **IWSS** (${}^{i}_{k}Rank^{sZMs}$, ${}^{i}_{k}CID^{sZMs}$, SVM^{RBF}); ${}^{i}_{k}CID^{sZMs^{MRMs}}_{IWSS^{TWSVM}}$ = **IWSS** (${}^{i}_{k}Rank^{sZMs}$, ${}^{i}_{k}CID^{sZMs}$, $\text{TWSVM}^{\text{RBF}}$);
(9) ${}^{i}_{k}CID^{sZMs^{MRMs}}_{IWSSr^{SVM}}$ = **IWSSr** (${}^{i}_{k}Rank^{sZMs}$, ${}^{i}_{k}CID^{sZMs}$,SVM^{RBF}); ${}^{i}_{k}CID^{sZMs^{MRMs}}_{IWSSr^{TWSVM}}$ = **IWSSr** (${}^{i}_{k}Rank^{sZMs}$, ${}^{i}_{k}CID^{sZMs}$, $\text{TWSVM}^{\text{RBF}}$);
(10) **if** k=1
(11) S^{IWSS}=struct (k, 'R¹', ${}^{i}_{k}CID^{sZMs^{MRMs}}_{IWSS^{SVM}}$, 'R²', ${}^{i}_{k}CID^{sZMs^{MRMs}}_{IWSS^{TWSVM}}$); S^{IWSSr}=struct (k, 'R¹', ${}^{i}_{k}CID^{sZMs^{MRMs}}_{IWSSr^{SVM}}$, 'R²', ${}^{i}_{k}CID^{sZMs^{MRMs}}_{IWSSr^{TWSVM}}$);
(12) **else**
(13) S^{IWSS}(end+1)=struct (k, 'R¹', ${}^{i}_{k}CID^{sZMs^{MRMs}}_{IWSS^{SVM}}$, 'R²', ${}^{i}_{k}CID^{sZMs^{MRMs}}_{IWSS^{TWSVM}}$); S^{IWSSr}(end+1)=struct (k, 'R¹', ${}^{i}_{k}CID^{sZMs^{MRMs}}_{IWSSr^{SVM}}$, 'R²', ${}^{i}_{k}CID^{sZMs^{MRMs}}_{IWSSr^{TWSVM}}$);
(14) **end**
(15) **end**
(16) // union-intersection operations (UIOs)
(17) $\text{UIOs}^{\text{IWSS}} = [(S^{\text{IWSS}}(1).R^1) \cap (S^{\text{IWSS}}(1).R^2)] \cup [(S^{\text{IWSS}}(2).R^1) \cap (S^{\text{IWSS}}(2).R^2)] \cup \ldots \cup [(S^{\text{IWSS}}(m).R^1) \cap (S^{\text{IWSS}}(m).R^2)]$;
(18) $\text{UIOs}^{\text{IWSSr}} = [(S^{\text{IWSSr}}(1).R^1) \cap (S^{\text{IWSSr}}(1).R^2)] \cup [(S^{\text{IWSSr}}(2).R^1) \cap (S^{\text{IWSSr}}(2).R^2)] \cup \ldots \cup [(S^{\text{IWSSr}}(m).R^1) \cap (S^{\text{IWSSr}}(m).R^2)]$;
(19) ${}^{i}CID^{sZMs^{OMs}}$ = ($\text{UIOs}^{\text{IWSS}}$) ∪ ($\text{UIOs}^{\text{IWSSr}}$);
(20) **end**

Function: IWSSr (RZMs, Data, Learning Model)

(1) Sel= ${}^{i}_{k}Rank^{sZMs}$ {1}; // first moments ${}^{i}_{k}Rank^{sZMs}$ array (feature with highest SU insert in Sel).
(2) AccSel=PerEval ($\lceil Data \rceil_{N \times Sel\ with\ C}$,Learning model); // C: [target class] $_{N\times1}$; N: number of sample of ${}^{i}CID$, PerEval: Performance Evaluation.
(3) **for** v=2 to length (${}^{i}_{k}Rank^{sZMs}$)
(4) OptFea= $\varnothing$;
(5) **for** r=1 to length (Sel)
(6) Sel^{temp}=$\text{update}^{\text{SelSub}}$(copy(Sel), swap(Sel{r}, ${}^{i}_{k}Rank^{sZMs}$ {v}));
(7) $\text{AccSel}^{\text{temp}}$= PerEval ($\lceil Data \rceil_{N \times Sel^{temp}\ with\ C}$, Learning Model);
(8) **if** ($\text{AccSel}^{\text{temp}}$>AccSel)
(9) OptFea=swap (Sel{r}, ${}^{i}_{k}Rank^{sZMs}$ {v});
(10) Acc= $\text{AccSel}^{\text{temp}}$;
(11) **end**
(12) **end**
(13) Sel^{temp}=$\text{update}^{\text{SelSub}}$(copy(Sel), add (${}^{i}_{k}Rank^{sZMs}$ {v}));
(14) $\text{AccSel}^{\text{temp}}$=PerEval ($\lceil Data \rceil_{N \times Sel^{temp}\ with\ C}$, Learning model);
(15) **if** ($\text{AccSel}^{\text{temp}}$>AccSel)
(16) OptFea=Sel^{temp};
(17) AccSel=$\text{AccSel}^{\text{temp}}$;
(18) **end**
(19) **if** (OptFea != null)
(20) update (Sel, OptFea);
(21) **end**
(22) **end**
(23) **return** Sel

Function: IWSS (RZMs, Data, Learning Model)

(1) Sel= ${}^{i}_{k}Rank^{sZMs}$ {1}; AccSel=PerEval ($\lceil Data \rceil_{N \times Sel\ with\ C}$, Learning Model);
(2) **for** v=2 to length (${}^{i}_{k}Rank^{sZMs}$); Sel^{temp}=add (copy(Sel), ${}^{i}_{k}Rank^{sZMs}$ {v});
(3) $\text{AccSel}^{\text{temp}}$=PerEval ($\lceil Data \rceil_{m \times Sel^{temp}\ with\ C}$, Learning Model);
(4) **if** ($\text{AccSel}^{\text{temp}}$>AccSel)
(5) add (Sel, ${}^{i}_{k}Rank^{sZMs}$ {v});
(6) AccSel=$\text{AccSel}^{\text{temp}}$;
(7) **end**
(8) **end**
(9) **return** Sel;

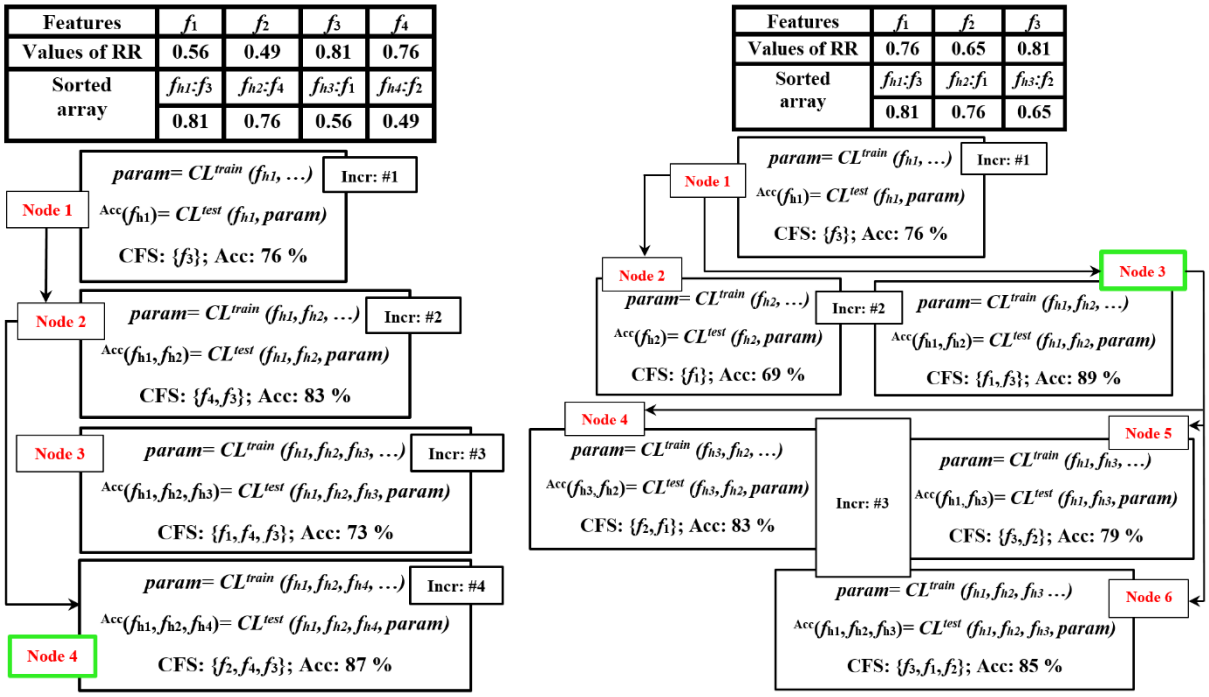


Figure 3. The IWSS (Left) & IWSSr (Right) algorithms [45]

Due to the complexity of the TWSVM being four times smaller than the SVM, we found that the complexities of $IWSS^{SVM/TWSVM}$ and $IWSSr^{SVM/TWSVM}$ are $O(n^4)$ and $O(n^5)$, respectively. So, in the worst case, the DHPFSS has the complexity of order $O(n^5)$.

3. Experimental Design

3.1. Description of Datasets

For our purposes, we used three dataset of chest images including lungs infected by COVID-19 and healthy ones. The Kaggle team has provided two links for two different data sets that are distinguished by the terms COVID-19 X-ray dataset([1]CID:https://www.kaggle.com/khoongweihao/covid19-xray-dataset-train-test-sets) and sarscov2 CT scan dataset ([2]CID: https://www.kaggle.com/datasets/plameneduardo/sarscov2-ctscan-dataset), respectively. The reader can find a repository provided by [49] that gathered normal chest X-ray images and COVID-19 cases [49, 50]. A vast amount of data images containing CT scans of chests infected and non-infected by SARS-CoV-2 have been arranged in the second link. These images belonged to patients hospitalized in Sao Paulo, Brazil [35]. The third set was collected from Github. The third dataset ([3]CID: https://github.com/ieee8023/covid-chestxray-dataset) provides the chest X-ray and CT scan images of normal lungs and abnormal ones infected by viral or bacterial pneumonia such as COVID-19, MERS, SARS, and ARDS. More details of datasets can be found in [38] and [51]. For dataset[2,3] ([2]CID, [3]CID), the number of 300 images belonged to infected lungs; and the remaining 300 images related to healthy ones are used for the learning scenario. In dataset[1] ([1]CID), the number of 74 images belonged to infected lungs, and the number of 74 images belonged to healthy ones are fed to the learning models.

3.2. Zernike Moments (ZMs)

Among various functions (Hu, Legendre, etc.), the ZMs are a more common way of describing an image. The Zernike polynomials, firstly introduced by Frits Zernike [52], are widely used in optics and image processing [53]. They are described in two-dimensional polar coordinates (ρ,θ) forming a complete set of polynomials. The orthogonality of Zernike moments over the interior of a unit circle, $\rho \le 1$(See figure 4), dictates that each moment is unique [52]. The Zernike polynomials are defined as [54]:

$$V_s^t(\rho,\theta) = R_s^t(\rho)\exp(it\theta) \quad (17)$$

$$R_s^t(\rho) = \sum_{l=0}^{(s-|t|)/2} (-1)^l \frac{(s-l)!}{l!\left(\frac{s+|t|}{2}-l\right)!\left(\frac{s-|t|}{2}-l\right)!}\rho^{s-2l} \quad (18)$$

where s and t are integers indicating the order and repetition numbers of functions, respectively.

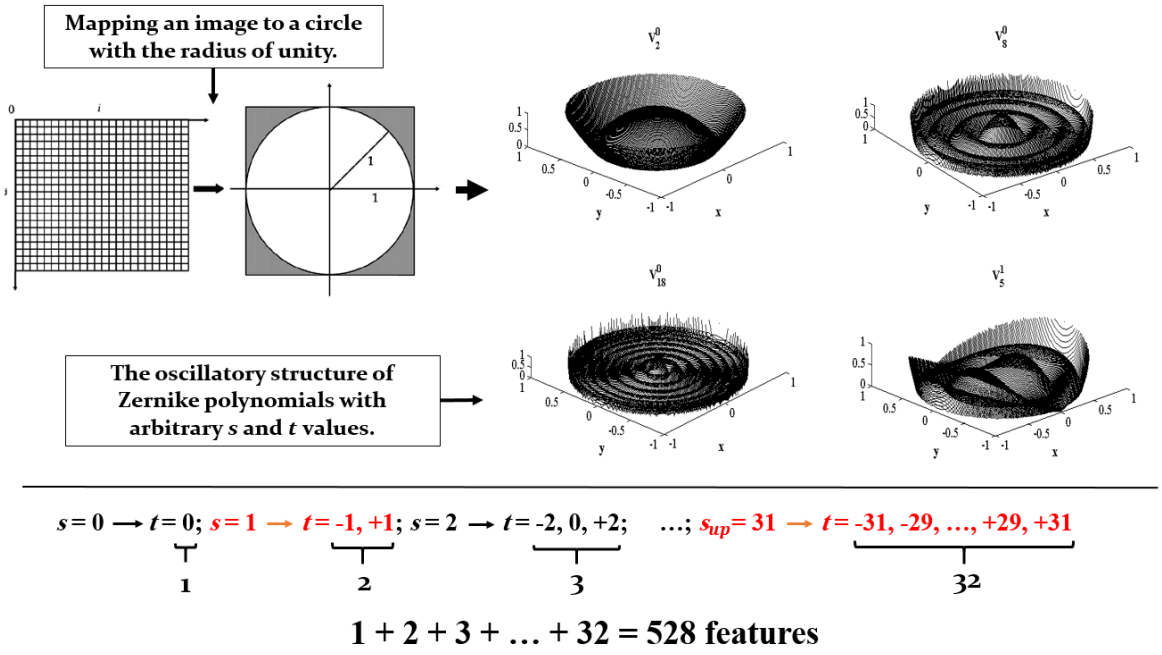


Figure 4. This schematic shows how an image is mapped to a circle and illustrates the oscillatory behavior of Zernike functions (with arbitrary *s* and *t* values based on Equations 17 and 18). The bottom portion highlights the extraction of 528 features from the image using Zernike moments based on Equation 22.

These parameters must satisfy the constraints $0 \le |t| \le s$, and $s-|t|$ is even. The image intensity function in the polar coordinates $F(\rho,\theta)$ can be expressed in terms of the Zernike polynomials as follows [55]:

$$F(\rho,\theta) = \sum_{s=0}^{s_{up}} \sum_{t=0}^{s} Z_{s,t} V_s^t(\rho,\theta) \tag{19}$$

where $Z_{s,t}$ is the ZMs that can be obtained as:

$$Z_{s,t} = \frac{s+1}{2} \iint_{\rho \le 1} V_s^{t*}(\rho,\theta) F(\rho,\theta) \rho d\rho d\theta \tag{20}$$

Moreover, the orthogonality property of the Zernike polynomials is expressed as:

$$\int_0^{2\pi} \int_0^1 V_s^{t^*} V_{s'}^{t'} \rho d\rho d\theta = \frac{\pi}{s+1} \delta_{s,s'} \delta_{t,t'} \tag{21}$$

where V^* indicates the complex conjugate of V, and the symbol δ denotes the Kronecker delta function. One may ask why we use the ZMs for describing an image. As seen in Eq. (1), the Zernike function is intrinsically rotation invariant because of the nature of the exponential Fourier term [54, 56]. Moreover, we can make each image scale and translation invariant by putting the target (here, the chest) within a square (Preprocessing step). Then, this surrounded square is mapped to the unit circle [54, 55].

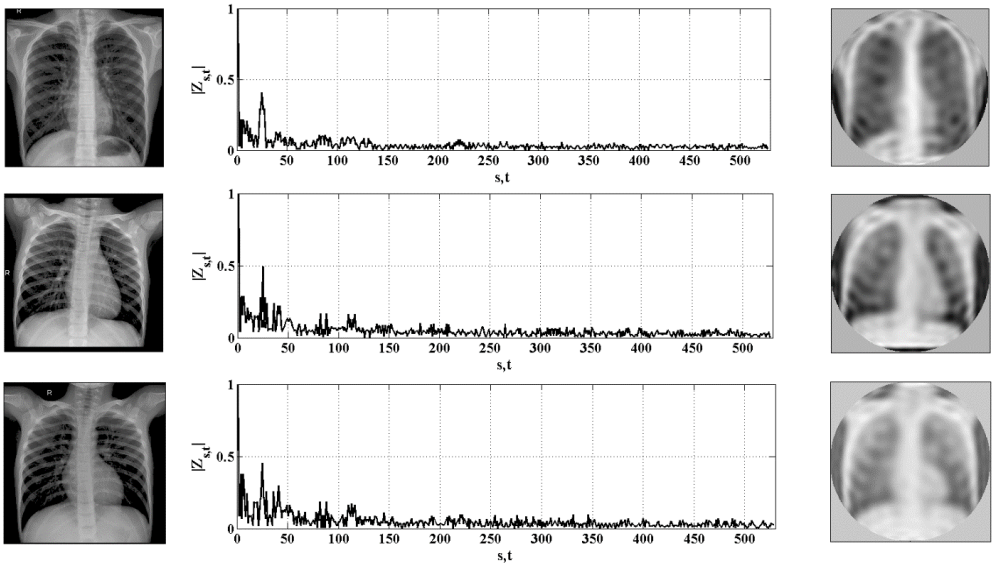
The relation for finding the number of Zernike moments (NZMs) is as follows:

$$NZMs = \sum_{s=0}^{s=s_{up}} (s+1) = \frac{(s_{up}+1)(s_{up}+2)}{2} \tag{22}$$

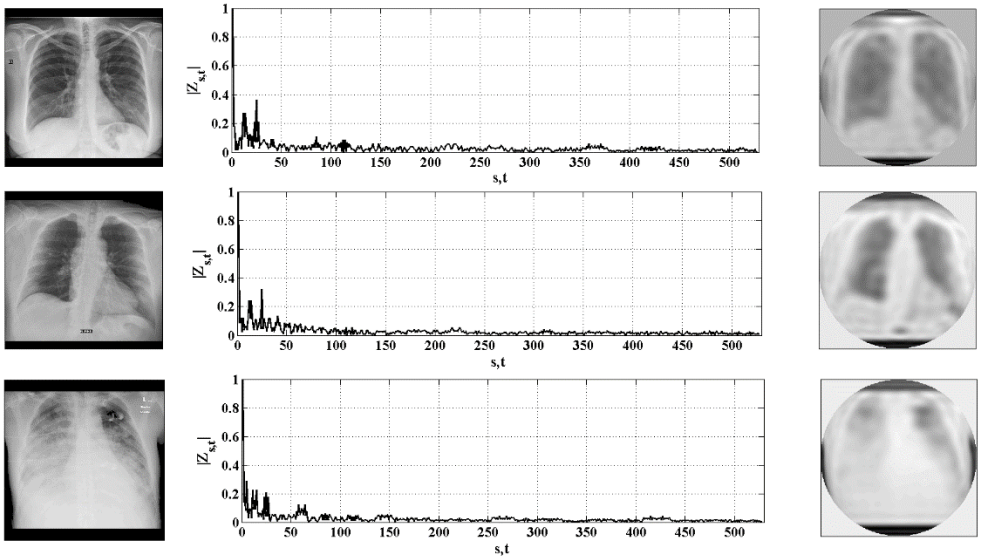
Using least reconstruction error has revealed that the optimum order number of Zernike is S_{up}=31 [54, 56]. So, the NZMs is obtained to be 528 (See Figure 4), and the original image can be reconstructed from pair of (*s*, *t*). For each image, the number of features equals to NZMs carrying the information of the image. For example, the first dataset of the *Github* repository (^{3}CID), figure 5 shows the magnitude values of ZMs for healthy (figure 5 (a)) and infected (figure 5 (b)) chest data with their reconstructed images, respectively.

3.3. Selecting FOMs per ^{i}CID^{sZMs}

By categorizing ZMs per ^{i}CID^{ZMs} $(1 \le i \le 3)$ in the form of triple set moments namely (^{i}CID^{pZMs}, ^{i}CID^{sZMs}, and ^{i}CID^{mZMs}), and following the partial-manner strategy through applying fragmentation function on ^{i}CID^{sZMs} in *k* bundles ($^1_k CID^{sZMs}; 1 \le k \le m$) (For more details refer to Section 2), the preprocessing steps in our proposed framework is completed. In terms of categorizing the sZMs based on the importance of GFs in the orthogonal polynomials, the first 20 ZMs were selected from 528 ZMs as ^{i}CID^{pZMs}, ZMs 21 to 220 are labeled as ^{i}CID^{sZMs}, and the rest of the ZMs are considered as ^{i}CID^{mZMs}.



(a) Original X-ray images of healthy chest (left column), their normalized magnitude values of ZMs with 528 features (middle column), and their reconstructed images from ZMs mapped on unit circle (last column).



(b) Original X-ray chest images infected by COVID-19 (left column), their normalized magnitude values of ZMs with 528 features (middle column), and their reconstructed images from ZMs mapped on unit circle (last column).

Figure 5. The magnitude values of ZMs for healthy (a) and infected (b) chest data (^{3}CID) with their reconstructed images.

An important point to note is that only ^{i}CID^{sZMs} enter into the feature selection process (^{i}CID^{pZMs} enter directly into the predictive model without FSS-based learning, ^{i}CID^{sZMs} enter into the feature selection process, and survived features are fed to the predictive model, and ^{i}CID^{mZMs} is not involved in any of the learning scenarios). Also, based on partial-manner feature selection, we fragmented the ^{i}CID^{sZMs} into 20 segments (m=20) wherein there are 10 sZMs are in each segment (Segment^{1}: sZMs 21:30, Segment^{2}: sZMs 31:40, …, Segment^{19}: sZMs 201:210, Segment^{20}: sZMs 211:220). After conducting the preliminary steps, selecting the ^{i}CID^{sZMs}-specific FOMs set is called ^{i}CID^{sZMs}. DHPFSS is applied to each ^{i}CID^{sZMs} discussed in this section. According to figure 2, in the first step of DHPFSS, each ${}_{k}^{i}CID^{sZMs}$ of ^{i}CID^{sZMs} enters into the filter phase. Hence, the SU amounts of moments per ${}_{k}^{i}CID^{sZMs}$ of ^{i}CID^{sZMs} are calculated based on RR analysis. Taking into consideration the formulations (See (1) to (4)) about the filter phase elaborated in Section 2.2.1, for example, the SU amounts of moments related to ${}_{11}^{3}CID^{sZMs}$ (3: the first dataset from *Github* repository; 11: segment 11 of $^{3}CID^{sZMs}$) based on (1), is analyzed. Based on SU values of ${}_{11}^{3}CID^{sZMs}$, all moments (10 moments) in ${}_{11}^{3}CID^{sZMs}$ have the

same SU value (0.2007). After calculating the SU amount of ${}_k^i CID^{sZMs}$ (for i=1:3, calculating the SU value of moments per segment (totally 20 segments)) and sorting the SU value of ${}_k^i CID^{sZMs}$ in descending order called ${}_k^i Rank^{sZMs}$ (for i=1:3, sorting the SU value of moments per segment (totally 20 segments)), the pair $({}_k^i CID^{sZMs}, {}_k^i Rank^{sZMs})$ are fed to the IDIWPs including ${}^{SVM}\text{IWSS}^{TWSVM}$ and ${}^{SVM}\text{IWSSr}^{TWSVM}$. The obtained MRMs of ${}_k^i CID^{sZMs}$ based on the IWMs-based learning model embedded in the LSIWP (${}^{SVM}\text{IWSS}^{TWSVM}$) and RSIWP (${}^{SVM}\text{IWSSr}^{TWSVM}$) are saved in quad sets namely, ${}_k^i CID_{IWSS^{SVM}}^{sZMs^{MRMs}}$, ${}_k^i CID_{IWSS^{TWSVM}}^{sZMs^{MRMs}}$, ${}_k^i CID_{IWSSr^{SVM}}^{sZMs^{MRMs}}$, and ${}_k^i CID_{IWSSr^{TWSVM}}^{sZMs^{MRMs}}$ (See Lines 10-14 of table 2). By applying the IDIWPs (LSIWP and RSIWP) of DHPFSS on ${}_k^i CID^{sZMs}$, the survived MRMs in the form of dual structure arrays namely IWSS-based struct (S^{IWSS}: ${}_k^i CID_{IWSS^{SVM}}^{sZMs^{MRMs}}$ and ${}_k^i CID_{IWSS^{TWSVM}}^{sZMs^{MRMs}}$) and IWSSr-based struct (S^{IWSSr}: ${}_k^i CID_{IWSSr^{SVM}}^{sZMs^{MRMs}}$ and ${}_k^i CID_{IWSSr^{TWSVM}}^{sZMs^{MRMs}}$) are given in tables 3 to 6. For example, table 3 shows the obtained MRMs of ${}_k^1 CID^{sZMs}$ (the first dataset from *Kaggle* repository; all segment k=1:20) based on ${}^{SVM}\text{IWSS}^{TWSVM}$ (column 1 and column 2) and ${}^{SVM}\text{IWSSr}^{TWSVM}$ (column 3 and column 4). In terms of selected MRMs of ${}_2^1 CID^{sZMs}$ (sZMs-based second segment of first dataset), by applying IWSS^{SVM} and IWSS^{TWSVM} (LSIWP) on ${}_2^1 CID^{sZMs}$, the ${}_2^1 CID_{IWSS^{SVM}}^{sZMs^{MRMs}}$ (sZMs:{31, 32, 33}; 100%) and ${}_2^1 CID_{IWSS^{TWSVM}}^{sZMs^{MRMs}}$ (sZMs:{31, 38}: 90%) are obtained, respectively (See second row; column 1 and column 2 of Table 3). Also, the extracted MRMs of ${}_2^1 CID^{sZMs}$ based on RSIWP are ${}_2^1 CID_{IWSSr^{SVM}}^{sZMs^{MRMs}}$ (sZMs:{31, 32, 33}; 100%) and ${}_2^1 CID_{IWSSr^{TWSVM}}^{sZMs^{MRMs}}$ (sZMs:{32}: 90%) (See second row; columns 3 and 4 in table 3). For example, by applying the IWSSr^{TWSVM} learning model on ${}_{16}^3 CID^{sZMs}$, figure 6 shows the tree growing of IWSSr^{TWSVM} for selecting the MRMs of ${}_{16}^3 CID^{sZMs}$ (node 30: sZMs={171, 174, 175, 180}; Accuracy: 90.24%). In the train-test procedure, the Accuracy (Acc) metric (23) is used for performance evaluation of the learning scenario. Also, the optimal pair of learning parameters, namely C (parameter of classifiers) and σ (parameter of RBF kernel) (See Equation (24)) are selected for reporting the high Acc value in each iteration of IWMs. For example, Acc variations in the optimal node (node 30) via fine-tuning on learning parameters are shown in figure 6.

$$\begin{aligned} &Accuracy\,(Acc) = (TP+TN)/(TP+TN+FP+FN) \\ &\begin{cases} P: Covid\ \ sample; \quad T: predicted\ \ correctly \\ N: NonCovid\ \ sample; \quad F: predicted\ \ incorrectly \end{cases} \end{aligned} \tag{23}$$

$$SVM^{RBF} \mid TWSVM^{RBF} \begin{Bmatrix} C = 2^i \mid i = 0,1,...,15 \\ \sigma = 2^j \mid j = -5,-4,...,15 \end{Bmatrix} \tag{24}$$

After selecting the quad-MRMs sets (two arrays related to the IWSS-based results (S^{IWSS} struct) and two arrays related to the IWSSr-based results (S^{IWSSr} struct)) per segment of ${}^i CID^{sZMs}$ (${}_k^i CID^{sZMs}$) relying on IDIWPs of DHPFSS, the UIOs (See Table 3; Line 17-19) is conducted on S^{IWSS} and S^{IWSSr} for selecting ${}^i CID^{sZMs^{OMs}}$. Table 6 shows the ${}^i CID^{sZMs}$-specific OMs obtained by UIOs situated in IDIWPs (See figure 2). After extracting the ${}^{1:3} CID^{sZMs^{OMs}}$, the *pZMs* per ${}^i CID^{ZMs}$ (The first 20 ZMs) were not included in the feature selection process due to retrieved procedure of the high importance GFs in the orthogonal polynomials, i.e., ${}^i CID^{ZMs}$-specific pZMs combined with ${}^i CID^{sZMs}$-specific OMs (union of ${}^i CID^{pZMs}$ and ${}^i CID^{sZMs^{OMs}}$). The obtained results are considered as final optimal moments of ${}^i CID^{ZMs}$ called ${}^i CID^{ZMs^{FOMs}}$ which are shown in table 6. ${}^i CID^{ZMs}$-specific FOMs is used for COVID-19 prediction in the next section.

3.4. COVID-19 Prediction via ${}^{1:3}CID^{ZMs^{FOMs}}$

After extracting ${}^i CID^{ZMs}$-specific FOMs based on DHPFSS, the efficacy of obtained FOMs per COVID-19 dataset (See table 6) in achieving high performance COVID-19 prediction is addressed in this section. The performance evaluation of survived FOMs per dataset is done based on the conducting 10-fold cross-validation technique.

Table 3. The obtained MRMs of ^{1}CID^{sZMs} based on IDIWPs (LSIWP (^{SVM}IWSS^{TWSVM}), RSIWP (^{SVM}IWSSr^{TWSVM}))

Segment	$^{1}_{k}CID^{sZMs^{MRMs}}_{IWSS^{SVM}}$; Pre. Acc.	$^{1}_{k}CID^{sZMs^{MRMs}}_{IWSS^{TWSVM}}$; Pre. Acc.	$^{1}_{k}CID^{sZMs^{MRMs}}_{IWSSr^{SVM}}$; Pre. Acc.	$^{1}_{k}CID^{sZMs^{MRMs}}_{IWSSr^{TWSVM}}$; Pre. Acc.
k=1	sZMs:{21, 22}; 100%	sZMs:{21, 22, 23}; 90%	sZMs:{21, 22}; 100%	sZMs:{24}; 92.50%
k=2	sZMs:{31, 32, 33}; 100%	sZMs:{31, 38}: 90%	sZMs:{31, 32, 33}; 100%	sZMs:{32}; 90%
k=3	sZMs:{41, 42}; 100%	sZMs:{41, 42, 44, 50}; 92.50%	sZMs:{41, 42}; 100%	sZMs:{42, 47}; 92.50%
k=4	sZMs:{51, 53}; 100%	sZMs:{51, 60}: 87.50%	sZMs:{51, 53}; 100%	sZMs:{55}; 92.50%
k=5	sZMs:{61, 62}; 100%	sZMs:{61, 62, 65}; 95%	sZMs:{61, 62}; 100%	sZMs:{61, 62}; 92.50%
k=6	sZMs:{71, 76, 77}; 100%	sZMs:{71, 80}: 90%	sZMs:{76, 77}; 100%	sZMs:{71, 80}; 90%
k=7	sZMs:{81, 83}; 100%	sZMs:{81}; 82.50%	sZMs:{81, 83}; 100%	sZMs:{81}; 82.50%
k=8	sZMs:{91, 92, 93}; 100%	sZMs:{91, 92, 94}: 90%	sZMs:{91, 92, 93}; 100%	sZMs:{91, 92, 94}; 92.50%
k=9	sZMs:{101, 103, 104}; 100%	sZMs:{101, 102, 103, 104, 105}; 92.50%	sZMs:{103, 104, 105}; 100%	sZMs:{102, 105}; 87.50%
k=10	sZMs:{111, 112}; 100%	sZMs:{111}: 90%	sZMs:{111, 112}; 100%	sZMs:{111}; 90%
k=11	sZMs:{121, 124, 125}; 100%	sZMs:{121, 122, 124}; 90%	sZMs:{121, 125}; 100%	sZMs:{122, 124}; 90%
k=12	sZMs:{131, 132, 133}; 100%	sZMs:{131}: 87.50%	sZMs:{132, 133}; 100%	sZMs:{133}; 90%
k=13	sZMs:{141, 145}; 100%	sZMs:{141, 142, 143}; 95%	sZMs:{141, 145}; 100%	sZMs:{141, 143, 144}; 97.50%
k=14	sZMs:{151, 152}; 100%	sZMs:{151, 153}: 92.50%	sZMs:{151, 152}; 100%	sZMs:{154, 155}; 90%
k=15	sZMs:{161, 162}; 100%	sZMs:{161, 162, 164, 170}; 90%	sZMs:{161, 162}; 100%	sZMs:{161, 162, 166}; 90%
k=16	sZMs:{171, 173, 174}; 100%	sZMs:{171, 172}: 85%	sZMs:{171, 173, 174}; 100%	sZMs:{172, 174}; 87.50%
k=17	sZMs:{181, 182, 183}; 100%	sZMs:{181, 182, 189}; 87.50%	sZMs:{182, 183}; 100%	sZMs:{182}; 85%
k=18	sZMs:{191, 192, 193}; 100%	sZMs:{191, 194, 195, 196, 199}: 90%	sZMs:{191, 192, 193}; 100%	sZMs:{195, 197, 199}; 90%
k=19	sZMs:{201, 203, 204}; 100%	sZMs:{201, 203}; 90%	sZMs:{202, 205, 206, }; 100%	sZMs:{201, 203}; 90%
k=20	sZMs:{211, 212, 213}; 100%	sZMs:{211, 212, 214}: 87.50%	sZMs:{211, 212, 213}; 100%	sZMs:{212}; 85%

$^{1}_{k}CID^{sZMs^{MRMs}}_{IWMs^{Classifiers}}$; 1: the first dataset from *Kaggle* repository (^{1}CID)

Table 4. The obtained MRMs of ^{2}CID^{sZMs} based on IDIWPs (LSIWP (^{SVM}IWSS^{TWSVM}), RSIWP (^{SVM}IWSSr^{TWSVM}))

Segment	$^{2}_{k}CID^{sZMs^{MRMs}}_{IWSS^{SVM}}$; Pre. Acc.	$^{2}_{k}CID^{sZMs^{MRMs}}_{IWSS^{TWSVM}}$; Pre. Acc.	$^{2}_{k}CID^{sZMs^{MRMs}}_{IWSSr^{SVM}}$; Pre. Acc.	$^{2}_{k}CID^{sZMs^{MRMs}}_{IWSSr^{TWSVM}}$; Pre. Acc.
k=1	sZMs:{21, 22}; 100%	sZMs:{21, 22}; 95.23%	sZMs:{21, 22}; 100%	sZMs:{21, 22}; 95.23%
k=2	sZMs:{31}; 71.42%	sZMs:{31, 34, 35, 38}; 85.71%	sZMs:{37}; 80.95%	sZMs:{31, 37}; 95.23%
k=3	sZMs:{41, 42, 46}; 71.42%	sZMs:{41, 42, 47}; 85.71%	sZMs:{45}; 80.95%	sZMs:{42, 48}; 85.71%
k=4	sZMs:{51}; 71.42%	sZMs:{51, 54, 59}; 80.95%	sZMs:{59 }; 90.47%	sZMs:{54, 59, 60}; 90.47%
k=5	sZMs:{61, 62}; 57.14%	sZMs:{61, 62, 64, 65, 67, 68}; 80.95%	sZMs:{63}; 90.47%	sZMs:{63}; 95.23%
k=6	sZMs:{71, 74, 78}; 66.66%	sZMs:{71, 77, 78}: 85.71%	sZMs:{71, 74, 78}; 66.66%	sZMs:{77}; 76.19%
k=7	sZMs:{81, 83}; 66.66%	sZMs:{81, 82, 83, 84, 88, 89}; 90.47 %	sZMs:{83, 90 }; 85.71%	sZMs:{83, 85}; 85.71%
k=8	sZMs:{91, 92}; 71.42%	sZMs:{91, 92}: 76.19%	sZMs:{92, 94}; 76.19%	sZMs:{91, 96}; 80.95%
k=9	sZMs:{101}; 66.66%	sZMs:{101, 104}; 57.14%	sZMs:{104}; 71.42%	sZMs:{103, 105}; 76.19%
k=10	sZMs:{111, 117}; 61.90%	sZMs:{111, 112}: 76.19%	sZMs:{119}; 71.42%	sZMs:{111, 112}; 76.19%
k=11	sZMs:{121}; 85.71%	sZMs:{121, 128}; 80.95%	sZMs:{121}; 85.71%	sZMs:{121, 128}; 80.95%
k=12	sZMs:{131, 137}; 76.19%	sZMs:{131, 132}: 52.38%	sZMs:{136, 137}; 90.47%	sZMs:{135, 139}; 85.71%
k=13	sZMs:{141, 142}; 57.14%	sZMs:{141, 142, 143}; 76.19%	sZMs:{142}; 80.95%	sZMs:{141, 142, 143}; 76.19%
k=14	sZMs:{151, 153, 157, 158}; 80.95%	sZMs:{151, 152, 156}: 66.66%	sZMs:{153, 156}; 80.95%	sZMs:{151, 156}; 85.71%
k=15	sZMs:{161, 164}; 71.42%	sZMs:{161, 162}; 61.90%	sZMs:{169, 170}; 85.71%	sZMs:{162}; 76.19%
k=16	sZMs:{171, 172, 175}; 61.90%	sZMs:{171, 172, 174, 179}: 71.42%	sZMs:{175, 180}; 80.95%	sZMs:{180}; 85.71%
k=17	sZMs:{181, 182, 187}; 71.42%	sZMs:{181, 182}; 80.95%	sZMs:{182, 189}; 90.47%	sZMs:{182, 185}; 90.47%
k=18	sZMs:{191, 192}; 80.95%	sZMs:{191, 192, 193}: 80.95%	sZMs:{191, 193}; 85.71%	sZMs:{191, 193}; 80.95%
k=19	sZMs:{201, 205}; 66.66%	sZMs:{201, 202}; 66.66%	sZMs:{205, 207}; 80.95%	sZMs:{208, 209}; 85.71%
k=20	sZMs:{211, 212}; 71.42%	sZMs:{211, 213}: 71.42%	sZMs:{212, 213}; 76.19%	sZMs:{213}; 76.19%

$^{2}_{k}CID^{sZMs^{MRMs}}_{IWMs^{Classifiers}}$; 2: the second dataset from *Kaggle* repository (^{2}CID)

Table 5. The obtained MRMs of ^{3}CID^{sZMs} based on IDIWPs (LSIWP (^{SVM}IWSS^{TWSVM}), RSIWP (^{SVM}IWSSr^{TWSVM}))

Segment	$^{3}_{k}CID^{sZMs^{MRMs}}_{IWSS^{SVM}}$; Pre. Acc.	$^{3}_{k}CID^{sZMs^{MRMs}}_{IWSS^{TWSVM}}$; Pre. Acc.	$^{3}_{k}CID^{sZMs^{MRMs}}_{IWSSr^{SVM}}$; Pre. Acc.	$^{3}_{k}CID^{sZMs^{MRMs}}_{IWSSr^{TWSVM}}$; Pre. Acc.
k=1	sZMs:{21, 29}; 85.36%	sZMs:{21, 22, 30}; 80.48%	sZMs:{21, 29}; 85.36%	sZMs:{21, 24, 27}; 85.36%
k=2	sZMs:{31, 32, 33}; 85.36%	sZMs:{31, 35, 39}: 82.92%	sZMs:{31, 32, 33}; 85.36%	sZMs:{31, 35, 39}; 82.92%
k=3	sZMs:{41}; 90.24%	sZMs:{41, 49}; 92.68%	sZMs:{41}; 90.24%	sZMs:{41, 49}; 92.68%
k=4	sZMs:{51, 52, 53}; 78.04%	sZMs:{51, 53, 54, 55, 60}: 85.36%	sZMs:{52, 53, 55}; 82.92%	sZMs:{54, 60}; 85.36%
k=5	sZMs:{61, 65, 66}; 73.17%	sZMs:{61, 63, 65, 66, 67, 68}; 85.36%	sZMs:{65, 66}; 78.04%	sZMs:{62, 68, 70}; 82.92%
k=6	sZMs:{71}; 78.04%	sZMs:{71, 72}: 82.92%	sZMs:{71}; 78.04%	sZMs:{71, 77}; 85.36%
k=7	sZMs:{81, 82, 89}; 70.73%	sZMs:{81, 82, 83}; 78.04 %	sZMs:{83, 90 }; 75.60%	sZMs:{82, 83, 85}; 82.92%
k=8	sZMs:{91, 93}; 82.92%	sZMs:{91}: 78.04%	sZMs:{93, 97}; 85.36%	sZMs:{91}; 78.04%
k=9	sZMs:{101, 103, 104, 105}; 85.36%	sZMs:{101, 103, 104, 105, , 106}; 85.36%	sZMs:{101, 104}; 82.92%	sZMs:{103, 104, 105}; 82.92%
k=10	sZMs:{111, 112, 117}; 78.04%	sZMs:{111, 112, 113, 115, , 119}; 85.36%	sZMs:{111, 113}; 80.48%	sZMs:{111, 115, 118}; 87.80%
k=11	sZMs:{121, 123, 124, 126}; 85.36%	sZMs:{121, 124}; 85.36%	sZMs:{121, 124, 128}; 85.36%	sZMs:{121, 124}; 85.36%
k=12	sZMs:{131, 132, 136}; 82.92%	sZMs:{131, 132, 133, 136}: 85.36%	sZMs:{132, 139}; 82.92%	sZMs:{133, 136, 139}; 90.24%
k=13	sZMs:{141, 142, 150}; 85.36%	sZMs:{141, 142, 143, 144}; 82.92%	sZMs:{142, 143}; 87.80%	sZMs:{142, 143, 148}; 87.80%
k=14	sZMs:{151, 160}; 78.04%	sZMs:{151, 152, 155}: 78.04%	sZMs:{158}; 82.92%	sZMs:{151, 155, 160}; 87.80%
k=15	sZMs:{161, 165, 167}; 80.48%	sZMs:{161, 167}; 78.04%	sZMs:{167}; 82.92%	sZMs:{161, 167}; 78.04%
k=16	sZMs:{171, 173}; 78.04%	sZMs:{171, 174, 175, 180}: 87.80%	sZMs:{173, 175, 178}; 85.36%	sZMs:{171, 174, 175, 180}; 90.24%
k=17	sZMs:{181, 182}; 80.48%	sZMs:{181, 182, 187}; 85.36%	sZMs:{188, 90}; 90.24%	sZMs:{182, 183}; 85.36%
k=18	sZMs:{191, 192, 193}; 80.48%	sZMs:{191, 192, 194}: 85.36%	sZMs:{193, 195}; 87.80%	sZMs:{191, 193, 196}; 87.80%
k=19	sZMs:{201, 206, 208}; 82.92%	sZMs:{201, 203, 207}; 80.48%	sZMs:{206, 208}; 87.80%	sZMs:{208, 210}; 82.92%
k=20	sZMs:{211, 212, 215, 219}; 85.36%	sZMs:{211, 212, 215, 218}: 85.36%	sZMs:{211, 215, 217}; 85.36%	sZMs:{212, 219}; 82.92%

$^{3}_{k}CID^{sZMs^{MRMs}}_{IWMs^{Classifiers}}$; 3: the third dataset from *Github* repository (^{3}CID)

Table 6. The obtained OMs per ^{i}CID^{sZMs} via applying UIOs on $^{i}_{k}$CID$^{sZMs^{MRMs}}_{IWMs^{Classifiers}}$ and FOMs per ^{i}CID^{ZMs} (^{i}CID$^{ZMs^{FOMs}}$)

The Obtained OMs Per ^{i}CID^{sZMs} via Appling UIOs on $^{i}_{k}$CID$^{sZMs^{MRMs}}_{IWMs^{Classifiers}}$	
Input (S^{IWSS}; S^{IWSSr})	$^{1:4}CID^{sZMs^{OMs}}$
$[{}_{1:20}{}^{1}CID^{sZMs^{MRMs}}_{IWSS^{SVM}}, {}_{1:20}{}^{1}CID^{sZMs^{MRMs}}_{IWSS^{TWSVM}}; {}_{1:20}{}^{1}CID^{sZMs^{MRMs}}_{IWSSr^{SVM}}, {}_{1:20}{}^{1}CID^{sZMs^{MRMs}}_{IWSSr^{TWSVM}}]$	sZMs: {21, 22, 31, 32, 41, 42, 51, 61, 62, 71, 81, 91, 92, 101, 103, 104, 105, 111, 121, 124, 131, 133, 141, 151, 161, 162, 171, 174, 181, 182, 191, 201, 203, 211, 212}
$[{}_{1:20}{}^{2}CID^{sZMs^{MRMs}}_{IWSS^{SVM}}, {}_{1:20}{}^{2}CID^{sZMs^{MRMs}}_{IWSS^{TWSVM}}; {}_{1:20}{}^{2}CID^{sZMs^{MRMs}}_{IWSSr^{SVM}}, {}_{1:20}{}^{2}CID^{sZMs^{MRMs}}_{IWSSr^{TWSVM}}]$	sZMs: {21, 22, 31, 37, 41, 42, 51, 59, 61:68, 71, 77, 78, 81:84, 88, 89, 91, 92, 111, 112, 121, 131, 137, 141, 142, 143, 151, 153, 156:158, 161, 164, 171, 172, 174, 180:182, 191:193, 211, 213}
$[{}_{1:20}{}^{3}CID^{sZMs^{MRMs}}_{IWSS^{SVM}}, {}_{1:20}{}^{3}CID^{sZMs^{MRMs}}_{IWSS^{TWSVM}}; {}_{1:20}{}^{3}CID^{sZMs^{MRMs}}_{IWSSr^{SVM}}, {}_{1:20}{}^{3}CID^{sZMs^{MRMs}}_{IWSSr^{TWSVM}}]$	sZMs: {21, 31, 41, 51, 53, 61, 65, 66, 71, 81: 83, 91, 101, 103:105, 111, 112, 121, 124, 131, 132, 136, 139, 141, 142, 143, 151, 161, 167, 171, 175, 181, 182, 191:193, 201, 208, 211, 215}
The Final Optimal Moments (FOMs) per ^{i}CID^{ZMs} (^{i}CID$^{ZMs^{FOMs}}$)	
Input	$^{1:4}CID^{ZMs^{FOMs}}$
$[{}^{1}CID^{sZMs^{OMs}}, {}^{1}CID^{pZMs}]$	${}^{1}CID^{ZMs^{FOMs}}$: {1:20, 21, 22, 31, 32, 41, 42, 51, 61, 62, 71, 81, 91, 92, 101, 103, 104, 105, 111, 121, 124, 131, 133, 141, 151, 161, 162, 171, 174, 181, 182, 191, 201, 203, 211, 212}
$[{}^{2}CID^{sZMs^{OMs}}, {}^{2}CID^{pZMs}]$	${}^{2}CID^{ZMs^{FOMs}}$: {1:20, 21, 22, 31, 37, 41, 42, 51, 59, 61:68, 71, 77, 78, 81:84, 88. 89, 91, 92, 111, 112, 121, 131, 137, 141, 142, 143, 151, 153, 156:158, 161, 164, 171, 172, 174, 180:182, 191:193, 211, 213}
$[{}^{3}CID^{sZMs^{OMs}}, {}^{3}CID^{pZMs}]$	${}^{3}CID^{ZMs^{FOMs}}$: {1:20, 21, 31, 41, 51, 53, 61, 65, 66, 71, 81: 83, 91, 101, 103:105, 111, 112, 121, 124, 131, 132, 136, 139, 141, 142, 143, 151, 161, 167, 171, 175, 181, 182, 191:193, 201, 208, 211, 215}

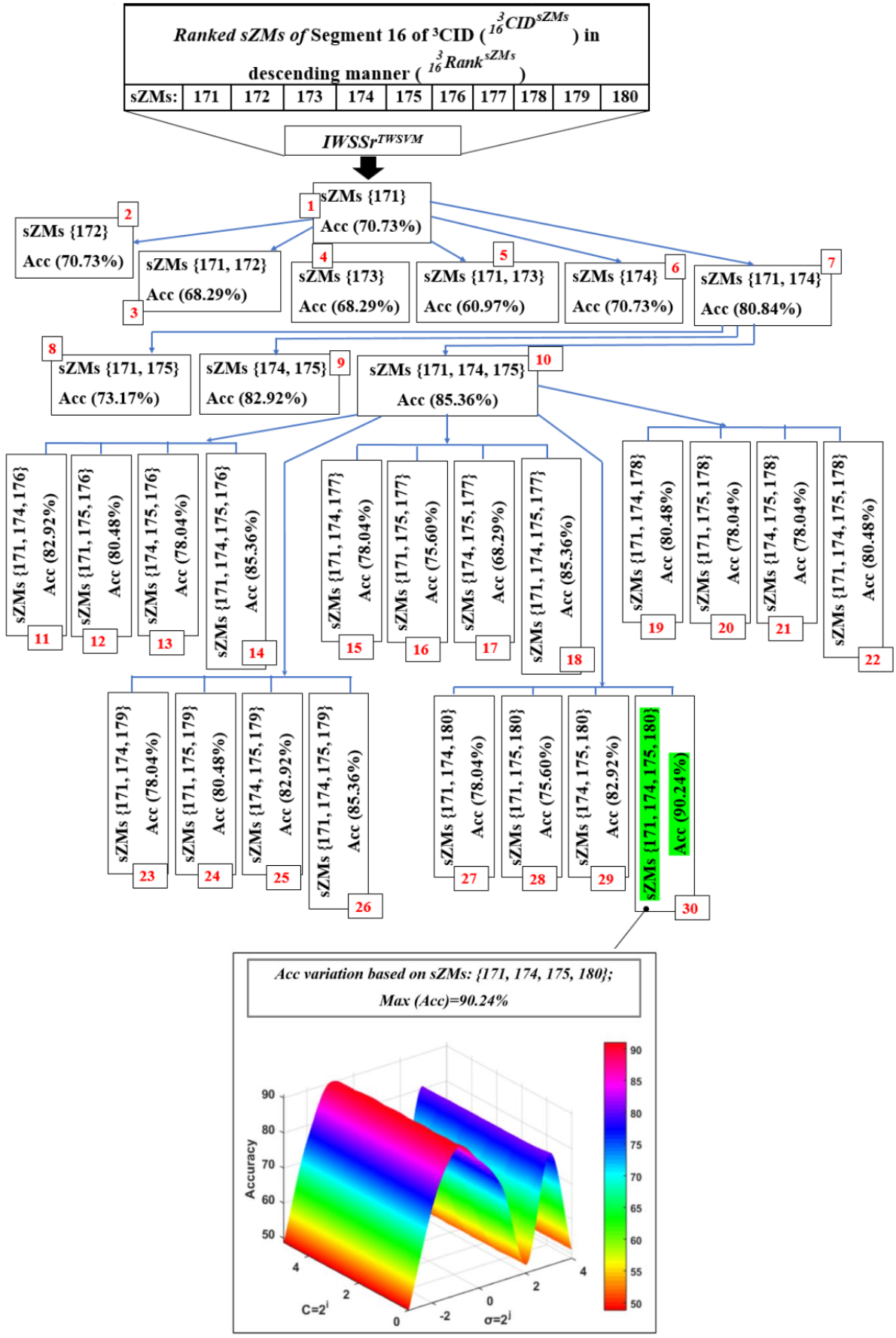


Figure 6. Structure of IWSSr tree in selecting MRMs of $^{3}_{16}CID^{sZMs}$ based on IWSSr^{TWSVM} with regarding Acc variations in optimal node (node 30).

The SVM classifier is used for the train-test procedure running for each fold. Also, for finding the optimal values among learning parameters, namely (C, σ), we conducted train-test procedures per fold based on setting C as $\{C = 2^i \mid i = 0,...,15\}$ and σ as $\{\sigma = 2^j \mid j = -5,...,15\}$. The classification metrics for evaluating the performance of the proposed framework in predicting COVID-19 status per dataset are accuracy (Acc=(TP+TN)/(TP+TN+FP+FN)), sensitivity (True-Positive Rate (TPR)=TP/(TP+FN)), and Specificity (True-Negative Rate (TNR)= TN/(TN+FP)). Taking into cognizance points regarding requirements of train-test procedures for COVID-19 prediction, the values of triple indices in COVID-19 prediction (Acc, TPR, and TNR) per dataset in each fold are shown in table 7. By setting the different values for learning parameters, the maximum value of Acc among Acc variations is captured per fold. These values are recorded in table 7. Furthermore, according to the maximum Acc per fold, TPR and TNR values corresponding to the maximum Acc value are also shown in each fold. For more clarity, the Acc variations of fold[2], fold[3], fold[5], fold[6], fold[8], and fold[9] related to using FOMs of ^2CID in COVID-19 prediction are depicted in figure 7.

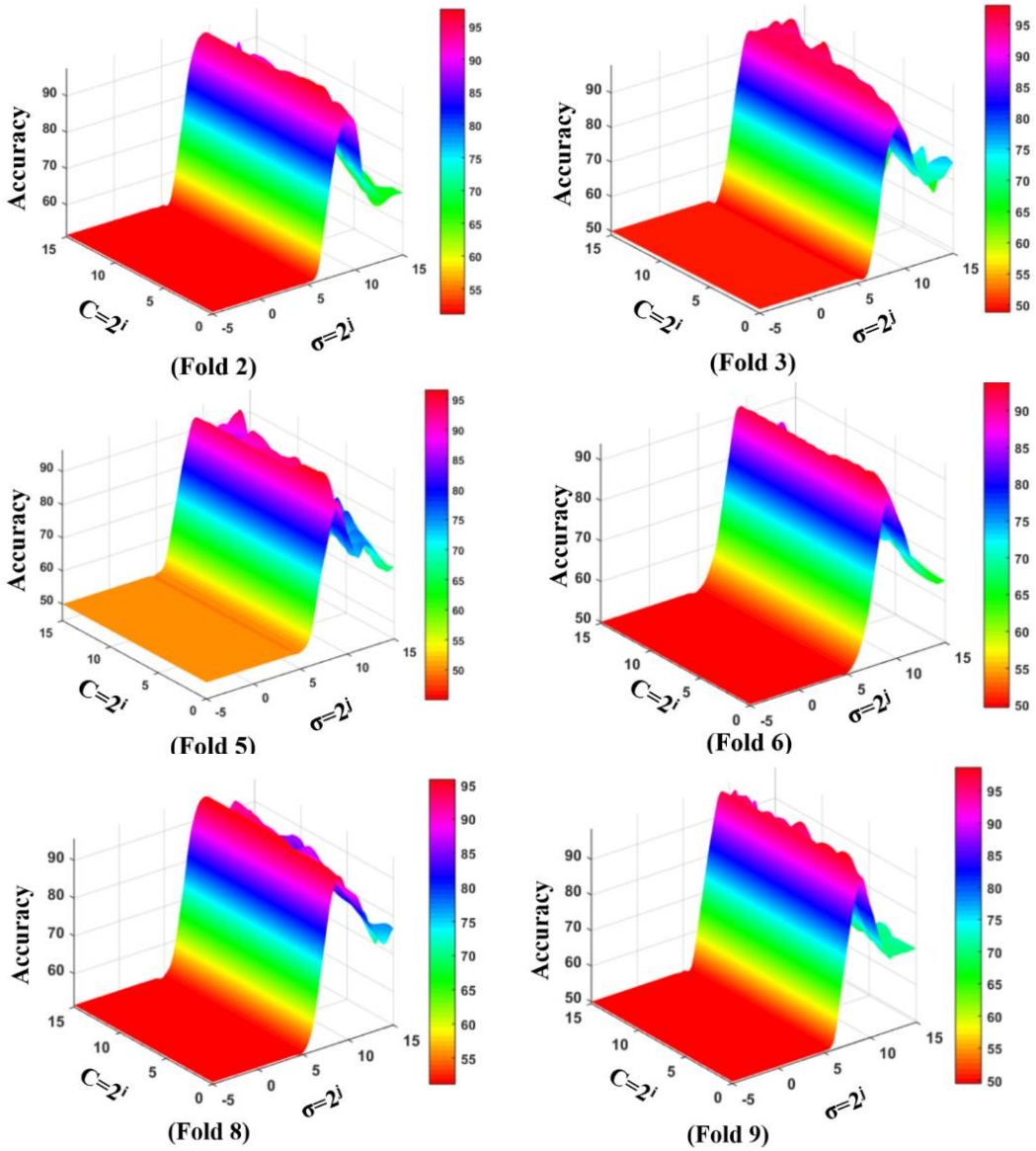


Figure 7. Acc variations based on learning parameters in some folds (fold [2, 3, 5, 6, 8, 9]) for COVID-19 prediction based on FOMs of [2]CID

Table 7. Results of COVID-19 prediction based on selected FOMs per datasets

Classifier	Dataset	10-fold cross validation Max(Acc.) per fold based on fine-tuning on C and σ Accuracy [TPR / TNR]			
		fold 1	**fold 2**	**fold 3**	**fold 4**
		100 [100 / 100]	93.33 [100 / 87.5]	93.33 [100 / 87.5]	100 [100 / 100]
		fold 5	**fold 6**	**fold 7**	**fold 8**
SVM^{RBF}	[1]CID	100 [100 / 100]	100 [100 / 100]	100 [100 / 100]	100 [100 / 100]
		fold 9		**fold 10**	
		100 [100 / 100]		100 [100 / 100]	
		Mean(measure) of folds: Accuracy [TPR / TNR]			
		98.66 [100 / 97.5]			
		fold 1	**fold 2**	**fold 3**	**fold 4**
		90 [96.67 / 83.33]	96.66 [93.33 / 100]	96.66 [96.67 / 96.67]	95 [93.33 / 96.67]
		fold 5	**fold 6**	**fold 7**	**fold 8**
SVM^{RBF}	[2]CID	95 [96.67 / 93.33]	93.33 [90 / 96.67]	91.66 [93.33 / 90]	95 [93.33 / 96.67]
		fold 9		**fold 10**	
		96.66 [96.67 / 96.67]		93.33 [90 / 96.67]	
		Mean(measure) of folds: Accuracy [TPR / TNR]			
		94.33 [94 / 94.66]			
		fold 1	**fold 2**	**fold 3**	**fold 4**
		91.66 [93.33 / 90]	91.66 [93.33 / 90]	95 [93.33 / 96.67]	96.66 [93.33 / 100]
		fold 5	**fold 6**	**fold 7**	**fold 8**
SVM^{RBF}	[3]CID	95 [96.67 / 93.33]	96.66 [100 / 93.33]	96.66 [96.67 / 96.67]	98.33 [100 / 96.67]
		fold 9		**fold 10**	
		95 [100 / 90]		91.66 [86.67 / 96.67]	
		Mean(measure) of folds: Accuracy [TPR / TNR]			
		94.82 [95.33 / 94.33]			

Table 8. Comparison of methods

Methods	Dataset	Accuracy (%)
Our proposed method	[1]CID	98.66
Our proposed method	[2]CID	94.33
Our proposed method	[3]CID	94.82
Jaiswal et al. [33]	[2]CID	96.25
Soares et al. [34]	[2]CID	97.38
Sarki et al. [38]	[1]CID	93.5
Sarki et al. [38]	[3]CID	93.5
Sen et al. [41]	[2]CID	98.39
Dey et al. [42]	[2]CID	99.5
Ibrahim et al. [57]	[2]CID	97.59
Gupta et al. [58]	[2]CID	98.91

3.5. Comparison of Experimental Methods: DPHFSS Vs. Other Methods in COVID-19 Prediction

It seems that the difference between a code and the other ones is its ability to compete with the most accuracies obtained in the optimal classifications. Of course, the flexibility of an algorithm to cope with different types of data has not to be ignored. In this section, we recount the results of some previous works focused on the same data sets that we used in this study. Reference [38] have used the first and third datasets ([1]CID and [3]CID) and achieved a maximum accuracy of 93.5%. Among performances of classifications published in valid journals about accuracies of the developed algorithms in the second type of our dataset SARS-CoV-2-CT ([2]CID) [33, 34, 57, 58], the highest accuracy belonged to [58] with the value of 98.91%; on the other hand, the lowest accuracy value 96.25% belongs to [33]. As we mentioned in the section Introduction, two surveys employed the feature selection technique in their methods [41, 42]. The maximum accuracy of a method introduced in [41] reaches 98.39% in the [2]CID. Reference [42] obtained accuracies of their proposed algorithm about 99.5% for [2]CID. In our feature selection-based approach, for each one of the three datasets, the classifier can reach the accuracies of 98.66%, 94.33%, and 94.82%, respectively. As seen in table 8, the proposed approach (DHPFSS) has better performance than

other techniques in ^{1}CID-and ^{3}CID-based COVID-19 prediction. For more information about comparison results refer to table 8.

4. Conclusions and Future Work

We developed an automated multi-level method based on image moments and a feature selection approach for classifying normal chest images and infected ones. Our supervised algorithm was applied to three different image sets comprising infected (by bacterial and viral pneumonia such as COVID-19) and non-infected (healthy) chest X-ray and CT scan images. In this method, a robust approach of Zernike polynomials was employed to extract image moments. These long series of moments can be reversed to the original image because they convey image information. Then, a novel feature selection approach was added to the algorithm to extract optimal features from a series of image moments. In this procedure named DHPFSS, we employed the filter phase and dual incremental wrapper mechanisms. Thus, we exploited incremental wrapper subset selection (IWSS) and IWSS with replacement (IWSSr) to optimize the algorithm. The SVM and TWSVM classifiers with RBF kernel are involved in the classification process. According to the obtained accuracy metric, we achieved a higher classification accuracy of 98.66% in ^{1}CID data classification. Also, the results show more than 94% accuracy in ^{2}CID and ^{3}CID data classification. Furthermore, comparing the output of our selection algorithm with previous ones demonstrates reliability and flexibility of our method in the face of different types of data. Another advantage of this method is its mathematical robustness in solving classification problems. Generally, the potential impact of the proposed method on real-world COVID-19 diagnosis workflows is related to compacting the high-dimensional space of the COVID-19 dataset and its ability to run in the presence of computers with low processing power. The high-dimensional space increases the prediction time of the patient's coronavirus disease and delays the treatment process. Hence, compacting the feature space for selecting the most relevant features in COVID-19 diagnosis based on the proposed method, decreases computational complexity and brings the timely and accurate diagnosis of illness in an emergency or the absence of a specialist. On the other hand, considering the substantial costs of equipping computers with high processing power (e.g., high graphics processing units (GPUs)) and the potential lack of accessibility to such systems in medical and treatment centers, our proposed approach is more practical. Unlike methods reliant on feature extraction and deep learning, which demand the most powerful computers, our proposed feature selection scheme enables high-performance COVID-19 prediction even with limited processing systems (in terms of hardware and software). In future work, the authors will work on the proposed technique to extend its ability to classify different brain tumor and breast cancers.

Acknowledgements

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

[1] D. Petrakis, D. Margină, K. Tsarouhas, F. Tekos, M. Stan, D. Nikitovic, D. Kouretas, D. A. Spandidos, and A. Tsatsakis, "Obesity - a risk factor for increased COVID-19 prevalence, severity and lethality (Review)," *Molecular medicine reports*, vol. 22, no. 1, pp. 9-19, July 2020.

[2] Y. Fang, Y. Nie, and M. Penny, "Transmission dynamics of the COVID-19 outbreak and effectiveness of government interventions: A data-driven analysis," *Journal of medical virology*, vol. 92, no. 6, pp. 645-659, June 2020.

[3] M. Rafiee, F. Parsaei, S. Rahimi Pordanjani, V. Amiri, and S. Sabour, "A Review on Applicable and Available Paraclinical Methods for Diagnosis of Coronavirus Disease-19," *Arch Iran Med*, vol. 23, no. 11, pp. 794-800, 2020.

[4] M. Yüce, E. Filiztekin, and K. G. Özkaya, "COVID-19 diagnosis - A review of current methods," *Biosensors & bioelectronics*, vol. 172, pp. 1-15, 2021.

[5] M. Yan, K. Herman, K. Muhammad, N. Aaron, and L. Chuen-Yen, "Review of Current COVID-19 Diagnostics and Opportunities for Further Development," *Frontiers in Medicine*, vol. 8, pp. 1-24, 2021.

[6] S. Maurya, and S. Singh, "Time Series Analysis of the Covid-19 Datasets," in *IEEE International Conference for Innovation in Technology (INOCON)*, 2020. pp. 1-6.

[7] J. Luo, Z. Zhang, Y. Fu, and F. Rao, "Time series prediction of COVID-19 transmission in America using LSTM and XGBoost algorithms," *Results in Physics*, vol. 27, pp. 1-9, 2021.

[8] G. Meyerowitz-Katz, and L. Merone, "A systematic review and meta-analysis of published research data on COVID-19 infection fatality rates", *International Journal of Infectious Diseases*, vol. 101, pp. 138-148, 2020.

[9] S. I. Jabbar, "Automated analysis of fatality rates for COVID 19 across different countries," *Alexandria Engineering Journal*, vol. 60, no. 1, pp. 521-526, 2021.

[10] W. Kong and P. P. Agarwal, "Chest Imaging Appearance of COVID-19 Infection," *Radiology: Cardiothoracic Imaging*, vol. 2, no. 1, 2020.

[11] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York: Springer, 2006.

[12] A. R. Webb and K. D. Copsey, *Statistical Pattern Recognition*. 3rd ed., West Sussex, United Kingdom: John Wiley & Sons Ltd, 2011.

[13] M. Kantardzic, *Data Mining: Concepts, Models, Methods, and Algorithms*. 3rd ed. New Jersey: Wiley-IEEE Press, John Wiley & Sons, Inc., 2011.

[14] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, "Supervised learning with quantum-enhanced feature spaces," *Nature*, vol. 567, pp. 209-212, 2019.

[15] D. Dell'Aquila and M. Russo, "Automatic classification of nuclear physics data via a Constrained Evolutionary Clustering approach," *Computer Physics Communications*, vol. 259, 2021.

[16] M. Bevilacqua, R. Nescatelli, R. Bucci, A. D. Magrì, A. L. Magrì, and F. Marini, "Chemometric classification techniques as a tool for solving problems in analytical chemistry," *J AOAC Int*., vol 97, no. 1, pp. 19-28. 2014.

[17] S. Arish, M. Javaherian, H. Safari, and A. Amiri, "Extraction of active regions and coronal holes from EUV images using the unsupervised segmentation method in the Bayesian framework," *Solar Physics*, vol. 291, pp. 1209-1224. 2016.

[18] M. Noori, M. Javaherian, H. Safari, and H. Nadjari, "Statistics of photospheric supergranular cells observed by SDO/HMI," *Advances in Space Research*, vol. 64, no. 2, pp. 504-513, 2019.

[19] T. Merembayev, R. Yunussov, and A. Yedilkhan, "Machine Learning Algorithms for Classification Geology Data from Well Logging," in *14th International Conference on Electronics Computer and Computation (ICECCO)*, 2018, pp. 206-212.

[20] V. Santucci, F. Santarelli, L. Forti, and S. Spina, "Automatic Classification of Text Complexity," *Appl. Sci.*, vol. 10, no. 20, 2020.

[21] N. Lotfi, M. Javaherian, B. Kaki, A. H. Darooneh, and H. Safari, "Ultraviolet solar flare signatures in the framework of complex network," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 30, no. 4, 2020.

[22] C. Li, K. Shirahama, M. Grzegorzek, F. Ma, and B. Zhou, "Classification of environmental microorganisms in microscopic images using shape features and support vector machines," in *Proc. IEEE International Conference on Image Processing*, 2013, pp. 2435-2439.

[23] I. D. Apostolopoulos, and T. A. Mpesiana, "Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks," *Phys. Eng. Sci. Med*., vol. 43, pp. 635-640, 2020.

[24] R. Mohammadi, M. Salehi, H. Ghaffari, A. A. Rohani, and R. Reiazi, "Transfer Learning-Based Automatic Detection of Coronavirus Disease 2019 (COVID-19) from Chest X-ray Images," *J. Biomed. Phys. Eng.*, vol. 10, no. 5, pp. 559-568, 2020.

[25] D. Singh, V. Kumar, Vaishali, and M. Kaur, "Classification of COVID-19 patients from chest CT images using multi-objective differential evolution-based convolutional neural networks," *European journal of clinical microbiology & infectious diseases: official publication of the European Society of Clinical Microbiology*, vol. 39, no. 7, pp. 1379-1389, 2020.

[26] R. A. Al-Falluji, Z. D. Katheeth, and B. Alathari, "Automatic Detection of COVID-19 Using Chest X-Ray Images and Modified ResNet18-Based Convolution Neural Networks," *Computers, Materials & Continua*, vol. 66, no. 2, pp. 1301-1313, 2021.

[27] E. F. Ohata, G. M. Bezerra, J. V. S. d. Chagas, A. V. L. Neto, A. B. Albuquerque, V. H. C. d. Albuquerque, and P. P. R. Filho, "Automatic Detection of COVID-19 Infection Using Chest X-Ray Images Through Transfer Learning," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 1, pp. 239-248, 2021.

[28] M. M. Taresh, N. Zhu, T. A. A. Ali, A. S. Hameed, and M. L. Mutar, "Transfer Learning to Detect COVID-19 Automatically from X-Ray Images Using Convolutional Neural Networks," *International Journal of Biomedical Imaging*, May 2021.

[29] D. Arias-Garzón, J. A. Alzate-Grisales, S. Orozco-Arias, H. B. Arteaga-Arteaga, M. A. Bravo-Ortiz, A. Mora-Rubio, J. M. Saborit-Torres, J. Á. M. Serrano, M. de la Iglesia Vayá, O. Cardona-Morales, and R. Tabares-Soto, "COVID-19 detection in X-ray images using convolutional neural networks," *Machine Learning with Applications*, vol. 6, 2021.

[30] J. Manokaran, F. Zabihollahy, A. Hamilton-Wright, and E. Ukwatta, "Detection of COVID-19 from chest x-ray images using transfer learning," *J Med Imaging (Bellingham)*, vol. 8, 2021.

[31] A. Badawi and K. Elgazzar, "Detecting Coronavirus from Chest X-rays Using Transfer Learning," *COVID*, vol. 1, pp. 403-415, 2021.

[32] A. Sakagianni, G. Feretzakis, D. Kalles, C. Koufopoulou, and V. Kaldis, "Setting up an easy-to-use machine learning pipeline for medical decision support: A case study for COVID-19 diagnosis based on deep learning with CT scans," *Stud Health Technol Inform*, vol. 272, pp. 13-16, June 2020.

[33] A. Jaiswal, N. Gianchandani, D. Singh, V. Kumar, and M. Kaur, "Classification of the COVID-19 infected patients using DenseNet201 based deep transfer

learning," *Journal of biomolecular structure & dynamics*, vol. 39, no. 15, pp. 5682-5689, 2021.

[34] E. Soares, P. Angelov, S. Biaso, M. H. Froes, and D. K. Abe, "(2020) SARS-CoV-2 CT-scan dataset: A large dataset of real patients CT scans for SARS-CoV-2 identification," *MedRxiv*, pp. 1-8, 2020.

[35] H. Panwar, P. K. Gupta, M. K. Siddiqui, R. Morales-Menendez, P. Bhardwaj, and V. Singh, "A deep learning and grad-CAM based color visualization approach for fast detection of COVID-19 cases using chest X-ray and CT-scan images," *Chaos Solitons Fractals*, vol. 140, 2020.

[36] S. Sharma, "Drawing insights from COVID-19-infected patients using CT scan images and machine learning techniques: a study on 200 patients," *Environmental science and pollution research international*, vol. 27, no. 29, pp. 37155-37163, 2020.

[37] V. S. Rohila, N. Gupta, A. Kaul, and D. K. Sharma, "Deep learning assisted COVID-19 detection using full CT-scans," *Internet of Things*, vol. 14, 2021.

[38] R. Sarki, K. Ahmed, H. Wang, Y. Zhang, and K. Wang, "Automated detection of COVID-19 through convolutional neural network using chest x-ray images," *PLOS ONE*, vol. 17, no. 1, 2022.

[39] X. Li, Y. Zhou, P. Du, G. Lang, M. Xu, and W. Wu, "A deep learning system that generates quantitative CT reports for diagnosing pulmonary tuberculosis. *Applied Intelligence*, vol. 51, no. 6, pp. 4082-4093, 2021.

[40] Shan, Y. Gao, J. Wang, W. Shi, N. Shi, M. Han, Z. Xue, D. Shen, and Y. Shi, "Abnormal lung quantification in chest CT images of COVID-19 patients with deep learning and its application to severity prediction," *Medical physics*, vol. 48, no. 4, pp. 1633-1645, 2021.

[41] S. Sen, S. Saha, S. Chatterjee, S. Mirjalali, and R. Sarkar, "A bi-stage feature selection approach for COVID-19 prediction using chest CT images," *Applied Intelligence*, vol. 51, pp. 8985-9000, 2021.

[42] A. Dey, S. Chattopadhyay, P. K. Singh, A. Ahmadian, M. Ferrara, N. Senu, and R. Sarkar, "MRFGRO: a hybrid meta-heuristic feature selection method for screening COVID-19 using deep features," *Scientific Report*, vol. 11, pp. 1-15, 2021.

[43] W. M. Shaban, A. H. Rabie, A. I. Saleh, and M. A. Abo-Elsoud, "A new COVID-19 Patients Detection Strategy (CPDS) based on hybrid feature selection and enhanced KNN classifier. *Knowledge-based systems*, vol. 205, 2020.

[44] R. Ruiz, J. C. Riquelme, and J. S. Aguilar-Ruiz, "Incremental wrapper-based gene selection from microarray data for cancer classification. *Pattern Recognition*, vol. 39, no. 12, pp. 2383-2392, 2006.

[45] S. A. Bashiri Mosavi, "Applying Cross-Permutation-Based Quad-Hybrid Feature Selection Algorithm on Transient Univariates to Select Optimal Features for Transient Analysis. *IEEE Access*, vol. 10, pp. 41131-41151, 2022.

[46] P. Bermejo, J. A. Gámez, and J. M. Puerta, "(2009) Incremental wrapper-based subset selection with replacement: An advantageous alternative to sequential forward selection," in *Proc. IEEE Symposium Series on Computational Intelligence and Data Mining (CIDM)*, Nashville, USA, 2009, pp. 367-374.

[47] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, pp. 273-297, 1995.

[48] O. L. Mangasarian and E. W. Wild, "Multisurface Proximal Support Vector Classification via Generalized Eigenvalues,". *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 1, pp. 69-74, 2006.

[49] J. P. Cohen, P. Morrison, and L. Dao, "COVID-19 Image Data Collection," *ArXiv*, 2020.

[50] J. P. Cohen, P. Morrison, L. Dao, K. Roth, T. Q. Duong, and M. Ghassemi, "COVID-19 Image Data Collection: Prospective Predictions Are the Future," *ArXiv*, 2020.

[51] D. S. Kermany, M. Goldbaum, W. Cai, C. C. S. Valentim, H. Liang, S. L. Baxter, A. McKeown, G. Yang, X. Wu, F. Yan, J. Dong, M. K. Prasadha, J. Pei, M. Y. L. Ting, J. Zhu, C. Li, S. Hewett, J. Dong, I. Ziyar, A. Shi, and K. Zhang, "Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning," *Cell*, vol. 172, no. 5, pp. 1122-1131, 2018.

[52] V. F. Zernike, "Beugungstheorie des schneidenver-fahrens und seiner verbesserten form, der phasenkontrastmethode," *Physica*, vol. 1, pp. 689-704, 1934.

[53] J. Schwiegerling, "Review of Zernike polynomials and their use in describing the impact of misalignment in optical systems," in *Proc. SPIE 10377, Optical System Alignment, Tolerancing, and Verification XI*, 2017.

[54] M. Javaherian, H. Safari, A. Amiri, and S. Ziaei, "Automatic Method for Identifying Photospheric Bright Points and Granules Observed by Sunrise," *Solar Physics*, vol. 289, pp. 3969-3983, June 2014.

[55] M. Sadeghi, M., Javaherian, and H. Miraghaei, "Morphological-based Classifications of Radio Galaxies Using Supervised Machine-learning Methods Associated with Image Moments," *The Astronomical Journal*, vol. 161, no. 2, pp. 94-102, 2021.

[56] S. A. Bashiri Mosavi, M. Javaherian, M. Sadeghi, and H. Miraghaei, "Modularized Filter-Wrapper Feature Selection Scheme for Finding Optimal Image Moments in Maps of Radio Galaxies," *International Journal of Intelligent Engineering and System*, vol. 14, no. 6, pp. 220-233, 2021.

[57] M. R. Ibrahim, S. M. Youssef, and K. M. Fathalla, "(2023) Abnormality detection and intelligent severity assessment of human chest computed tomography scans using deep learning: a case study on SARS-COV-2 assessment," *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, 2023.

[58] K. Gupta and V. Bajaj, "Deep learning models-based CT-scan image classification for automated screening of COVID-19," *Biomedical Signal Processing and Control*, vol. 80, 2023.

انتخاب ویژگی‌های بهینه تصاویر قفسه سینه توسط طرح انتخاب ویژگی جزئی-دوگانه-ترکیبی برای تشخیص کووید-۱۹ مبتنی بر ریخت‌شناسی

سید علیرضا بشیری موسوی[1*]، محسن جواهریان[2] و امید خلف بیگی[3]

[1]گروه مهندسی برق و کامپیوتر، مرکز آموزش عالی فنی و مهندسی بوئین زهرا، بوئین زهرا، قزوین، ایران

[2]مرکز تحقیقات نجوم و اخترفیزیک مراغه (ریام)، دانشگاه مراغه، ۵۵۳-۵۵۱۳۶، مراغه، ایران

[3]گروه مهندسی برق و کامپیوتر، دانشگاه خوارزمی، تهران، ایران

ارسال ۲۰۲۴/۰۳/۲۴؛ بازنگری ۲۰۲۴/۰۷/۲۹؛ پذیرش ۲۰۲۴/۰۸/۰۴

چکیده:

یکی از راه‌های تجزیه و تحلیل بیماری کووید ۱۹، بهره‌برداری از تصاویر اشعه ایکس و توموگرافی کامپیوتری از قفسه سینه بیماران است. بکارگیری تکنیک‌های داده‌کاوی بر روی تصاویر قفسه سینه می‌تواند سبب پیشرفت‌های قابل توجهی در تشخیص بیماری مذکور گردد. با این حال، در فضای ویژگی تصاویر قفسه سینه، تعداد زیادی ویژگی وجود دارد که بر عملکرد شناسایی کووید ۱۹ تأثیر منفی می‌گذارد. در پژوهش حاضر، طراحی یک روش انتخاب ویژگی جزئی‌گرا دوگانه ترکیبی برای انتخاب ویژگی‌های بهینه برای دستیابی به پیش‌بینی کووید ۱۹ با کارایی بالا در دستور کار قرار گرفته است. ابتدا با اعمال تابع زرنیک بر روی داده‌ها، ویژگی‌های زرنیک از تصاویر قفسه سینه سالم و افراد آلوده استخراج می‌شود. پس از تقسیم‌بندی ویژگی‌های زرنیک، زیرمجموعه ویژگی‌های بدست آمده وارد الگوریتم انتخاب ویژگی پیشنهادی می‌شوند تا ویژگی‌های زرنیک بهینه انتخاب شوند. الگوریتم پیشنهادی از فاز فیلتر و مکانیزم‌های دوگانه بسته‌بندی افزایشی، تشکیل شده است. هر بسته‌بندی افزایشی، توسط ویژگی‌هایی که براساس مکانیزم فیلتر مرتب شده‌اند، تغذیه می‌شود. بسته‌بندی‌های افزایشی دوگانه با استفاده از ماشین بردار پشتیبان و نسخه دوقلو آن که مجهز به هسته تابع پایه شعاعی هستند، تجهیز شده‌اند. پس از انتخاب ویژگی‌های بهینه، کارایی الگوریتم پیشنهادی براساس تکنیک اعتبارسنجی متقابل ارزیابی می‌شود. نتایج بدست آمده نشان می‌دهد که چارچوب پیشنهادی دارای دقت ۹۸,۶۶، ۹۴,۳۳ و ۹۴,۸۲ درصد بر روی داده‌های سه گانه برای پیش‌بینی کووید ۱۹ است که می‌تواند در تشخیص دقیق بیماری در شرایط اضطرار یا عدم حضور متخصص، مفید واقع شود.

کلمات کلیدی: طرح انتخاب ویژگی ترکیبی، روش‌های یادگیری مبتنی بر ابر صفحه، ویژگی‌های زرنیک بهینه، پیش‌بینی کووید ۱۹.