

Parameter Pliable Variational Mode Decomposition and Clustering Based Gravitational Search Algorithm Techniques for Bird Sound Classification

Sunil Kumar Prabhakar¹, Dogeun Park¹, Young-Gi Ju¹, Dong-Ok Won^{1*}

¹Department of Artificial Intelligence Convergence, Hallym University, Chuncheon, Korea

Corresponding author email ID^{1*}: dongok.won@hallym.ac.kr

Abstract—In the field of ecological monitoring, a significant aspect is bird sound classification. As different movements are observed by birds, classification of its sounds is quite important. The target information can be fully explained with the help of suitable feature extraction and selection schemes. In this paper, the first approach utilizes a parameter pliable Variational Mode Decomposition (PP-VMD) technique for extracting the features and then for feature selection Colliding Bodies Optimization (CBO) algorithm is utilized. The chosen features are then classified with the conventional machine learning classifiers. The second approach uses the concept of spectral clustering (SC) and K-means algorithm with clustering the initial centers (KIC) for feature extraction and Gravitational Search Algorithm (GSA) is used for feature selection before classifying it with machine learning classifiers. The proposed two strategies are implemented on publicly available bird sound classification dataset and the best results are obtained when PP-VMD with CBO and SVM classifier is used reporting an accuracy of 94.45%.

Keywords— *Feature Extraction, Feature Selection, Bird Sounds, Classification.*

I. INTRODUCTION

In the field of environmental monitoring, the classification of bird sounds plays a vital role [1]. For the acquisition of bird sounds, acoustic sensors are widely used. The acoustic sensor generates many gigabytes of data and so it becomes extremely difficult to analyze the data manually and so automated classification is necessary. A famous category of environmental sounds are bird sounds and to extract it some of the commonly used techniques are wavelet transform, Short Time Fourier Transform (STFT), Mel-Frequency Cepstral Coefficient (MFCC), Code Excited Linear Prediction (CELP) etc. [2]. Due to the exponential growth of samples and features, the curse of dimensionality problem occurs. To prevent the curse of dimensionality issue, efficient selection of features is necessary before leading to classification [3]. Some of the works done in bird sound classification are as follows. The code book features were used for bird sound classification by Labao et al [4]. An efficient bird sound separation network based on end-to-end single channel was proposed by Zhang et al [5]. A continuous frame sequence along with spectrogram frame linear network was used for bird sound classification by Zhang et al [6]. A novel deep transfer learning model was proposed by Kumar et al for bird sound classification [7] and various Convolutional Neural Networks (CNN) based models were investigated for bird sound classification by Xie et al [8]. A

low-cost microcontroller is used for a real time bird sound recognition process in [9]. Classifying woodpecker sounds was done efficiently using a two-stage approach by Vidana-Vila et al [10]. Two dimensional cepstral coefficients were used for classification of bird species by Lee et al [11]. Acoustic patterns were detected using multi-label classification by Zhang et al [12]. The bird and amphibian calls were automatically classified by using machine learning by Acevedo et al [13]. The ensemble extraction for detecting bird species was done by Kasten et al [14]. Penalized maximum likelihood was used for acoustic recognition of multiple bird species by Jancovic et al [15]. The concept of multilevel ternary pattern (TP) idea with repeated and iterated ReliefF (IRF) technique and machine learning technique was carried out by Tuncer et al [16]. In this work, once the basic pre-processing is done using Independent Component Analysis (ICA), VMD based CBO concept is used for feature extraction and selection followed by machine learning classification as the first proposed work, and clustering based GSA technique is proposed as the second technique followed by machine learning classification. The organization of the paper is as follows. Section 2 discusses the proposed technique 1, section 3 discusses the proposed technique 2, section 4 discusses the results and discussion followed by conclusion in section 5.

II. PROPOSED TECHNIQUE 1: PARAMETER DEPENDENT VMD TECHNIQUE:

For non-stationary signal analysis, one of the freshly used signal processing techniques is VMD technique [17]. In this advanced type of signal processing technique, the audio signals are decomposed into a fixed number of band limited nodes. The total number of nodes helps to reproduce the input audio signal by utilizing their sparsity properties. If the audio signal is considered as $S(t)$ and the number of modes is considered as m , then it is written as follows:

$$S(t) = \sum_{m=1}^M S_m(t) = \sum_{m=1}^M A_m(t) \cos \phi_m(t) \quad (1)$$

In a non-recursive VMD technique, every node has a center frequency. The major building blocks of VMD technique comprise of Wiener filtering, Harmonic shifting enabled frequency shifting and Hilbert transform. The VMD algorithm helps to decompose an audio signal into various modes which involves three parameters. The three parameters utilized are penalty parameter (α), Lagrangian multiplier (λ) and the mode number (m). If 'c' is indicated as an iteration count which is 1, the center

frequency of the m^{th} mode represented by $[f_m]$ and the spectrum is represented by $[\hat{d}_m]$. For the first iteration, the Lagrangian Multiplier (LM) is indicated as $[\lambda_1]$. For the $(c+1)^{th}$ iteration and modes from $m=1, \dots, M$, the mathematical specification of the m^{th} mode spectrum is expressed as follows:

$$\hat{d}_m^{c+1}(f) = \frac{\hat{d}(f) - \sum_{i < m} \hat{d}_i^{c+1}(f) - \sum_{i > m} \hat{d}_i^{c+1}(f) + \hat{\lambda}^c(f)/2}{1 + 2\alpha(f - f_m)^2} \quad (2)$$

The mathematical representation of every center mode frequency is given as:

$$f_m^{c+1} = \frac{\int_0^\infty f |\hat{d}_m^{c+1}(f)|^2 df}{\int_0^\infty |\hat{d}_m^{c+1}(f)|^2 df} \quad (3)$$

The updated formulation of LM is specified as:

$$\hat{\lambda}^{c+1}(f) = \hat{\lambda}^c(f) + T(\hat{d}(f) - \sum_m \hat{d}_m^{c+1}(f)) \quad (4)$$

where the updated parameters of LM are represented as τ . Until the convergence is achieved, the iteration count is increased. The mathematical specification of the convergence criterion is specified as follows:

$$\sum_m \frac{\|\hat{d}_m^{c+1} - \hat{d}_m^c\|_2^2}{\|\hat{d}_m^c\|_2^2} < \varepsilon \quad (5)$$

2.1 Parameter Pliable VMD (PP-VMD) technique:

In real life applications, VMD cannot be suitable for all the models. It depends on two essential parameters such as the mode number m and the penalty parameter α . Selecting the optimum value for α and m is somewhat difficult and hence it is optimized by an algorithm here. For any kind of optimization problem, the objective function is quite important. For feature identification purposes, correlation coefficient and kurtosis are quite important. The values of the kurtosis can change significantly with various types of disturbances but still it is used as an impact measurement index so that the disturbances are monitored. The similarity of 2 signals can be characterized well by the correlation coefficient and again it can be susceptible to different noises. For both the indexes, the merits and demerits are considered and their multiplication term acts as the objective function of the VMD parameter optimization. Autocorrelation and cross correlation are the 2 types of computation. The degree of similarity between 2 signals is measured by cross correlation and as its features are quite distinct and can be easily distinguishable, it has been chosen here and it is represented as:

$$CC_{\text{coeff}} = \frac{E[(k - \bar{k})(l - \bar{l})]}{E[(k - \bar{k})^2] E[(l - \bar{l})^2]} \quad (6)$$

The two signals are indicated as k and l respectively. The original signal is represented as k and the disturbed signal is represented as l . Based on Schwartz inequality, the

mathematical expectation is indicated by $E[\cdot]$. When Kurtosis index is multiplied with CC_{coeff} , it is represented as: $KI * CC_{\text{coeff}}$.

2.2 Colliding Bodies Optimization Algorithm:

A famous meta-heuristic technique influenced by the one-dimensional collision theory is colliding bodies optimization [18]. Figure 1 illustrates the implementation of the CBO algorithm to the PP-VMD process classified with the machine learning classifiers.

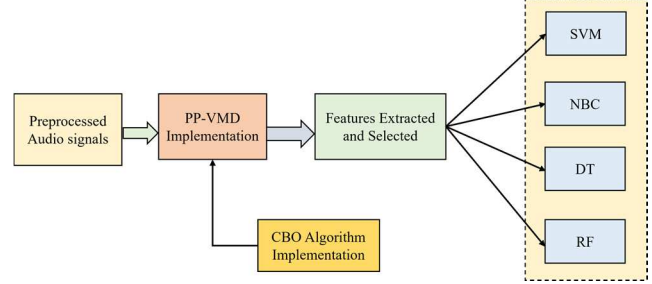


Figure 1 CBO based PP-VMD Process

The basic formulation of CBO is written as follows.

2.2.1 Basic formulation of CBO:

The modelling of each search agent is done as a body with mass and velocity. The start position of the i^{th} body is randomly given in a j -dimensional search space assigned by the user and is represented as:

$$p_{ij} = p_{j,\min} + \text{rand} \cdot (p_{j,\max} - p_{j,\min}) \quad (7)$$

where a random number between 0 and 1 is expressed as rand . Between 2 bodies and their positions, a collision occurs and depending on the one-dimensional collision laws, the update is done. For the body p_k , the mass is defined as follows:

$$m_k = \frac{1/C_k}{1/\sum_{i=1}^n (1/C_i)}, k = 1, \dots, n \quad (8)$$

where the cost function value of the k^{th} particle is expressed as C_k and n represents the total number of bodies utilized in the optimization process.

Based on the objective function values, the sorting of n colliding bodies are done in an ascending order and then it is divided into 2 equal groups such as the upper half comprising of the moving objects and the lower half comprising of the stationary groups. To drive the stationary objects position towards a better position, the moving group objects collide with the stationary objects group objects. The establishment of the colliding pairs are done in an ascending order with respect to the object function. The collision happens between the best moving particles and the best stationary ones. Before the collision, the velocity of the bodies is assigned as follows:

$$\text{Stationary bodies: } v_i = 0, i = 1, \dots, \frac{n}{2} \quad (9)$$

$$\text{Moving bodies: } v_i = p_{i-(n/2)} - p_i, i = \frac{n}{2} + 1, \dots, n \quad (10)$$

In the search space, the velocities are expressed as displacements. The velocity after the collision is computed as follows based on the colliding bodies theory and is expressed as:

$$\text{Stationary bodies } v_i' = \frac{(m_{i+(n/2)} + \varepsilon m_{i-(n/2)}) v_{i+(n/2)}}{m_i + m_{i+(n/2)}}, i = 1, \dots, \frac{n}{2} \quad (11)$$

$$\text{Moving bodies: } v_i' = \frac{(m_i - \varepsilon m_i - (n/2)) v_i}{m_i + m_{i-(n/2)}}, i = \frac{n}{2} + 1, \dots, n \quad (12)$$

where the coefficient of restitution is expressed as ε . It is nothing but the ratio of the relative velocities between the two bodies before and after the collision.

$$\varepsilon = \frac{|v_{i+1}' - v_i'|}{|v_{i+1} - v_i|} \quad (13)$$

To ensure a correct balance between the exploration and exploitation during the optimization process, the coefficient varies linearly between 1 and 0. The novel positions of the stationary bodies and moving bodies are determined once the displacement is computed as follows:

$$\text{Stationary bodies: } p_i^{new} = p_i + rand \cdot v_i', i = 1, \dots, \frac{n}{2} \quad (14)$$

$$\text{Moving bodies: } p_i^{new} = p_{i-(n/2)} + rand \cdot v_i', i = \frac{n}{2} + 1, \dots, n \quad (15)$$

where a uniformly distributed random vector is expressed as *rand* and it is in the range of [-1,1]. Unless a stopping criterion is achieved, the process is repeated for all the particles at every iteration. The pseudocode of CBO is shown in point wise as follows:

- (1) The CBO population is initialized in the search space as per equation (7).
- (2) The objective functions are evaluated and the masses are defined as per equation (8).
- (3) The population is sorted so that the stationary groups and moving groups are identified.
- (4) The velocities are computed as shown in equation (9) and equation (10).
- (5) The velocity after the collisions is computed using equation (11) and equation (12).
- (6) The determination of the new positions is done using equation (14) and equation (15).
- (7) Proceed to step 8 if the termination criteria are fulfilled, otherwise go to step 2.
- (8) The best solution traced by the algorithm is repeated.
- (9) End

III. PROPOSED TECHNIQUE 2: CLUSTERING TECHNIQUES BASED GRAVITATIONAL SEARCH ALGORITHM:

3.1 Gravitational Search Algorithm:

Inspired and influenced by Newton's law of gravity and mass interactions, a famous meta-heuristic method was developed and termed the gravitational search algorithm [19]. The gravitational force produced by the alteration in between

masses is the main idea of GSA. Under gravitation, an agent progresses towards the agent with heavier masses dependent on the Newtonian gravity and the laws of motion. The performance of agents is generally measured by their masses in GSA. Heavy masses are obtained by good solutions and lighter masses are obtained by poor solutions. For a solution set with N_a agents, the i^{th} agent is specified as:

$$P_i = (p_i^1, p_i^2, \dots, p_i^d), i = 1, 2, \dots, N_a$$

The computation of the mass value of the agent P_i is done as follows:

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \quad (16)$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^{N_a} m_j(t)} \quad (17)$$

where the mass values of the agent p_i at time t is represented as $M_i(t)$ and the fitness value of the agent p_i at time t is represented as $fit_i(t)$.

For a minimization issue, $best(t)$ and $worst(t)$ are defined as follows:

$$best(t) = \min_{j \in \{1, N_a\}} fit_j(t) \quad (18)$$

$$worst(t) = \max_{j \in \{1, N_a\}} fit_j(t) \quad (19)$$

The computation of force between the agent p_i and p_j is done as follows:

$$F_{ij}^d(t) = G(t) \frac{M_i(t) \times M_j(t)}{D_{ij}(t) + \varepsilon} (p_j^d(t) - p_i^d(t)) \quad (20)$$

where the passive mass of agent i is represented as M_{pi} and the active mass of agent j is represented as M_{aj} respectively. At a time t , $D_{ij}(t)$ represents the Euclidean distance in between two agents i and j hence it is computed by:

$$D_{ij}^{(t)} = \sqrt{\sum_{d=1}^d (p_i^d(t) - p_j^d(t))^2} \quad (21)$$

A heavier mass is possessed by more efficient agents implying that a higher attraction is possessed by better agents and it progresses more slowly. The definition of the gravitational force can be given as:

$$F_i^d(t) = \sum_{j \in Kbest, j \neq i} rand_j F_{ij}^d(t) \quad (22)$$

where a randomly distributed random number is represented as $rand_j$ in the interval [0,1]. The set of first K agents with a high mass value and best fitness value is denoted by $Kbest$. Based on the law of motion, the acceleration is expressed as:

$$a_i^d(t) = \frac{F_i^d(t)}{Z_i(t)} \quad (23)$$

where the inertial mass of the i^{th} agent is expressed as Z_i .

Supposing that

$$Z_{ai}(t) = Z_{pi}(t) = Z_{ui}(t) = Z_i(t), i = 1, 2, \dots, N_a \quad (24)$$

The recalculation of the acceleration can be done as follows:

$$a_i^d(t) = G(t) \sum_{j \in Kbest, j \neq i}^N \text{rand}_j \frac{Z_j(t)}{D_{ij}(t) + \epsilon} (p_j^d(t) - p_i^d(t)) \quad (25)$$

where a gravitational constant at time t is expressed by $G(t)$. To avoid division by zero, ϵ is used and considered here as a small positive number. The acceleration added to fraction of its current velocity is the agent's next velocity. Its velocity $v_i^d(t+1)$ and position $p_i^d(t+1)$ is computed as follows:

$$v_i^d(t+1) = \text{rand}_i \times v_i^d(t) + a_i^d(t) \quad (26)$$

$$p_i^d(t+1) = p_i^d(t) + v_i^d(t+1) \quad (27)$$

3.2 Spectral Clustering (SC):

For a particular dataset $P = \{p_i\}_{i=1}^n$, the main intention of clustering is to split or partition P into ' c ' clusters [20]. For the training data, a cluster assignment matrix is found by the SC by means of implementing a weighted graph G whose vertices are over P . There are many variations of the SC algorithms and in this paper, SC algorithm with K-way normalized cuts is discussed. An undirected weight graph is specified by $G = \{P, W\}$, where $W \in R^{n \times n}$ indicates an affinity matrix and P denotes a vertex set. Between a pair of vertices of G , the similarity relationship is characterized and for the symmetric matrix W and every entry W_{ij} is used to record the edge weights.

$$W_{ij} = \begin{cases} \exp\left(-\frac{\|p_i - p_j\|^2}{\sigma^2}\right), & \text{if } p_i, p_j \text{ are neighbours} \\ 0, & \text{otherwise} \end{cases} \quad (28)$$

The Laplacian graph is expressed as $L = D - W$, where the diagonal matrix is represented as D comprising of the diagonal elements as $D_{ii} = \sum_j W_{ij}, \forall_i$. The weights of its edges are measured by the size of a subset of a graph depending on the normalized cut criterion. Then a normalized Laplacian matrix is utilized and then the transformation of the optimization problem is done into a trace maximization problem

$$\max_{F^T F = I_q} \text{tr}\left(F^T D^{-1/2} W D^{-1/2} F\right) \quad (29)$$

where the identity matrix of size q by q is indicated by I_q .

The cluster assignment matrix is indicated as $F \in \mathbb{R}^{n \times q}$ by means of continuous values by relaxation. By means of Eigenvalue decomposition of the matrix $D^{-1/2} W D^{-1/2}$, the optimal solution F for (29) is obtained.

3.3 K-means algorithm with clustering the Initial Centers (KIC):

One of the celebrated and most pragmatic algorithms for clustering analysis is the K-means algorithm [21]. K samples are initiated into the algorithm besides the input datasets and

it is assigned as the inceptive centers of K clusters. The first 'K' data samples are usually indicated by K's. The execution of the K-means algorithm is influenced by the K representations. Then four important stages of this algorithm are expressed as follows:

Stage 1: In a random manner, the K data items are chosen from $P = \{p_1, p_2, \dots, p_n\}$ as cluster centers of $(m_1, m_2, m_3, \dots, m_k)$

Step 2: To the relevant cluster, every data item is added based on equation (30). If the following equation holds then the object p_i from the set of $P = \{p_1, p_2, \dots, p_n\}$ is appended to the cluster c_j

$$\|p_i - m_j\| < \|p_i - m_s\| \quad 1 \leq s \leq k; j \neq s \quad (30)$$

Step 3: The novel cluster centers $(m_1^*, m_2^*, \dots, m_k^*)$ are computed using equation (31) depending on the clustering of step 2 where the number of objects in the cluster i is indicated as n_i .

$$m_i^* = \frac{1}{n_i} \sum_{p_j \in C_i} p_j \quad 1 \leq i \leq k \quad (31)$$

Step 4: If there is a change in cluster centers, then the algorithm is repeated from Step 2 otherwise depending on the resulting centers the clustering is done.

On initial centers, the accomplishment of the K-means clustering algorithm is dependent mostly and it still progresses towards achieving local optimality. There are very sturdy ties among the data points in this algorithm and therefore the cluster centers do not withdraw from their local dense span limits.

3.3.1 Algorithm for tracing the cluster initial centers:

Using the K-means algorithm, all data objects are initially congregated and then the initial cluster centers are found out. The design for an object is produced depending on the generated clusters from each characteristic at every stage. The objects with similar patterns are situated in one cluster and then the obtained clusters will be much higher than the original clusters. In two different stages, the completion of the clusters is done. The initial stage is accomplished above and in the next stage, the integration of homogeneous clusters with each other is done unless a given number of clusters are achieved. The achieved cluster centers are called seed points and the technique for initial clustering of data points is shown in Algorithm 1.

Algorithm 1: Initial clustering of data points

Input: Dataset $(P = \{p_1, p_2, \dots, p_n\})$

Attribute set $(A = \{A_1, A_2, \dots, A_q\})$

Cluster numbers (K)

Output: Cluster seed-set (CSS = $\{Cs_1, Cs_2, \dots, Cs_n\}$)

Begin

While $(\forall A_j \in A)$

Determine Standard Deviation (σ_j)

Compute Mean (μ_j)

Determine cluster centers ($c = 1, 2, \dots, k$)

$$P_e = Z_e * \sigma_j + \mu_j; Z_e = \frac{2 * e - 1}{2 * k}$$

Compute K-means on this characteristic.

To every data pattern, the cluster labels are allocated.

Trace the distinctive patterns ($H \geq k$)

Cluster each data with the obtained patterns.

Return CS

End

A cluster label is generated for every attribute of data objects. This label is finally appended to the data object design. This label is finally affixed to the data object design. The objects with similar patterns are accommodated in one cluster. For all the data objects, the mean and Standard Deviation of that characteristic feature is calculated so that data object labels are produced on each attribute. The attribute value ranges are smashed into K indistinguishable intervals depending on the mean and standard deviation, so that an initial cluster center is formed. Thus, using the K-means technique, the clustering of the data points is done utilizing the initial centers.

3.4 GSA-SC and GSA-KIC:

An amalgamated data clustering technique dependent on GSA and SC/KIC algorithm is utilized which utilizes the properties of both these algorithms. The best clustering distribution is obtained under the frequency of GSA depending on the optimized cluster set selected by the SC and KIC.

3.4.1 Clustering problem:

Given a dataset D , the main intention of the clustering technique is to get a partition $B = \{B_1, B_2, \dots, B_k\}$ (i.e., $B_b \neq \phi$, for $\forall_b$; $\bigcup_{b=1}^k B_b = D$, $B_b \cap B_c = \phi$, for $\forall_b \neq c$, for which the objects in the similar cluster are quite similar to each other. The entire mean-square quantization error is one of the widely used clustering finding quality measure. In the datapoint set D , the fitness parameter of the clustering with cluster set S is expressed as

$$fit(D, S) = \sum_{c=1}^K \sum_{d_i \in S_c} \|d_i - S_c\|^2 \quad (32)$$

where $\|d_b - S_j\|$ expresses the similarity between the cluster center S_c and the point d_b .

3.4.2 Solution encoding:

Depending on the implementation of SC and KIC, the initialization of the candidate solutions for data congregation are done. Assigning $P = \{P_1, P_2, \dots, P_t\}$ considered as a collection of t agents which corresponds to the cluster center set. Every agent t_i comprises of K cluster centers. The b^{th} agent is considered as a $K * l$ -dimensional vector indicated as:

$$P_b = \{p_{11}^b, p_{12}^b, \dots, p_{1l}^b, \dots, p_{c1}^b, p_{c2}^b, \dots, p_{cl}^b, \dots, p_{k1}^b, p_{k2}^b, \dots, p_{kl}^b\} \quad (33)$$

where the center of the b^{th} cluster is corresponded by S_b . Figure 2 shows the implementation of SC and KIC with GSA and machine learning classifiers.

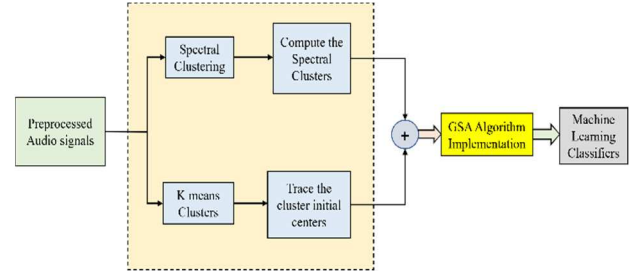


Figure 2 SC and KIC with GSA and Classifiers

3.4.3 Main implementation of GSA-SC and GSA-KIC:

The optimized initial clusters are selected by the proposed techniques. Depending on the collection of candidate cluster centers, the initial population generation is executed. Using GSA technique, the candidate solutions are implemented for data clustering. Once the group of candidate cluster centers is selected, then the data clustering process in the GSA framework is implemented. To assemble the initial population of GSA, the initial cluster centers which are generated and optimized are utilized. The quality of the initial population can be highly improved when compared to the standard GSA thereby there is a huge improvement in the convergence speed of the algorithm. The dataset samples are considered as feature space objects by the GSA mechanism, which progresses from one to another through Newton's law of gravity and motion.

Thus, the transformation of the clustering problem is done into an optimization problem by GSA. The primary steps of the GSA-SC and GSA-KIC is shown in Algorithm 2 as follows.

Algorithm 2: Implementation of GSA-SC and GSA-KIC

Input: collection of candidate cluster centers $S \in_{l,k+l}$

Output: Cluster assignment B

Initialize an initial population $P = \{P_1, P_2, \dots, P_t\}$

$P_1 \leftarrow S$

For $b = 2$ to t do

$$P_b = [p_{11}^b, p_{12}^b, \dots, p_{1l}^b, \dots, p_{cl}^b, \dots] \times rand$$

(where rand is denoted as a uniformly distributed random number in the range of [0,1])

End For

Do

Implement SC and KIC

Data point allocation to the closest cluster center

$$BDP_t(t = 1, 2, \dots, t)$$

Evaluate the fitness according to equation (32)

Compute the mass value of the agent (16) – (19)

Compute the acceleration of the agent using (25)

Compute the velocity of the agent using (26)

Search space updation P by mixing the agents in (27)

Unless the stop criterion is reached

Trace and chose the best cluster assignment B

IV. RESULTS AND DISCUSSION

To assess the classification performance, the dataset used here comprises of sound signals of 18 various species of birds. The dataset is publicly available online at <http://web.firat.edu.tr/turkertuncer/BIRDS.rar> [16]. For every

sound, the average length is about 5 seconds. From 10 to 20 various sound recordings, every bird species was recorded from YouTube and Xeno-Canto website were highly useful to get the mixed form of data. For the bird sounds, the sampling frequency was 48 KHz. A 10-fold cross validation method was used in this work for classifying with machine learning classifiers. Table 1 shows the performance analysis of PP-VMD technique with CBO and machine learning classifiers along with the performance analysis of SC and KIC with GSA technique and machine learning classifiers.

Table 1. Consolidated Results -Accuracy (%)

	SVM	NBC	DT	RF
PP-VMD+CBO	94.45	93.45	92.42	88.65
SC-GSA	93.24	92.01	89.43	90.12
KIC-GSA	91.03	89.87	91.23	92.87

4.1 Comparison with previous works

The present results have been compared with the previous works and shown in Table 2. On analyzing Table 2, the proposed works produce good results when comparing most of the works and in future it will be improved more.

Table 2 Comparison with previous works

Reference	Technique	Number of classes	Accuracy
[9]	Minimum distance classifier + MFCC	21 classes	72%
[10]	MFCC and Zero Crossing Rate	2 classes	94.02%
[11]	MFCC and Gaussian Mixture Model	28 classes	84.06%
[12]	KNN, MLP, DT	5 classes	85.3%
[13]	LDA, DT, SVM	12 classes	94.95%
[14]	Ensemble extraction	10 classes	92%
[15]	Frequency track estimation	30 classes	70%
[16]	Multilevel TP + IRF concept	18 classes	96.67%
Proposed	PP-VMD + CBO + SVM	18 classes	94.45%
	SC - GSA+SVM	18 classes	93.24%
	KIC - GSA + RF	18 classes	92.87%

V. CONCLUSION

To maintain a good ecosystem in the world, a significant indicator of good diversity is birds. Each bird has its own acoustic signal and so the species and number could be determined easily by the sound frequencies. At the same time, as the sounds of the birds are collected from noise environments, the automated classification of these sounds is quite difficult. Deep learning, transfer learning has been implemented to classify the bird sound signals but due to its high computational complexity, hand-crafted features with machine learning classifiers are considered best. Therefore, in this work two different strategies were proposed. The highest classification accuracy of 94.45% is obtained when the PP-VMD with CBO and SVM classifier is used. The second highest classification accuracy of 93.24% is obtained when SC-GSA with SVM classifier is used. The third highest classification accuracy of 92.87% is obtained when KIC-GSA with SVM classifier is used. Future works plan to

incorporate modifications in the VMD technique and the clustering technique to check its effectiveness and versatility.

ACKNOWLEDGEMENT

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2022R1A5A8019303).

REFERENCES

1. E. Tsau, S.-H. Kim, and C.-C. J. Kuo, "Environmental sound recognition with CELP-based features," in Proceedings of the ISSCS 2011—International Symposium on Signals, Circuits and Systems, pp. 1–4, Iasi, Romania, July 2011.
2. S. Zhao, Y. Zhang, H. Xu, and T. Han, "Ensemble classification based on feature selection for environmental sound recognition," Mathematical Problems in Engineering, vol. 2019, Article ID 4318463, 7 pages, 2019.
3. A. V. Bang and P. P. Rege, "Recognition of bird species from their sounds using data reduction techniques," in Proceedings of the 7th International Conference on Computer and Communication Technology, pp. 111–116, Allahabad, India, November 2017.
4. A. B. Labao, M. A. Clutario, and P. C. Naval, "Classification of bird sounds using codebook features," in Proceedings of the Asian Conference on Intelligent Information and Database Systems, pp. 223–233, Dong Hoi City, Vietnam, March 2018.
5. C. Zhang, Y. Chen, Z. Hao, X. Gao, An Efficient Time-Domain End-to-End Single-Channel Bird Sound Separation Network. Animals 2022, 12, 3117.
6. X. Zhang, A. Chen, G. Zhou, Z. Zhang, X. Huang, X. Qiang, Spectrogram-frame linear network and continuous frame sequence for bird sound classification. Ecol. Inform. 2019, 54, 101009.
7. Y. Kumar, S. Gupta, W. Singh, A novel deep transfer learning models for recognition of birds sounds in different environments. Soft Comput. 2022, 26, 1003–1023.
8. J. Xie, K. Hu, M. Zhu, J. Yu, Q. Zhu, Investigation of Different CNN-Based Models for Improved Bird Sound Classification. IEEE Access 2019, 7, 175353–175361.
9. O. Küçüktopcu, E. Masazade, C. Ünsalan, P.K. Varshney, A real-time bird sound recognition system using a low-cost microcontroller. Applied Acoustics. 2019; 148:194-201.
10. E. Vidaña-Vila, J. Navarro, R.M. Alsina-Pagès, A. Ramírez, A two-stage approach to automatically detect and classify woodpecker (Fam. Picidae) sounds. Applied Acoustics. 2020; 166:107312.
11. C.H. Lee, C.C. Han, C.C. Chuang Automatic classification of bird species from their sounds using two-dimensional cepstral coefficients IEEE Trans Audio Speech Lang Process, 16 (2008), pp. 1541-1550.
12. L. Zhang, M. Towsey, J. Xie, J. Zhang, P. Roe Using multi-label classification for acoustic pattern detection and assisting bird species survey, Appl Acoust, 110 (2016), pp. 91-98.
13. M.A. Acevedo, C.J. Corrada-Bravo, H. Corrada Bravo, L.J. Villanueva-Rivera, T.M. Aide Automated classification of bird and amphibian calls using machine learning: A comparison of methods Eco Inf, 4 (2009), pp. 206-214.
14. E.P. Kasten, P.K. McKinley, S.H. Gage Ensemble extraction for classification and detection of bird species Ecol Inf, 5 (2010), pp. 153-166.
15. P. Jančović, M. Kökür Acoustic recognition of multiple bird species based on penalized maximum likelihood IEEE Signal Process Lett, 22 (2015), pp. 1585-1589.
16. T. Tuncer, E. Akbal, S. Dogan, Multileveled ternary pattern and iterative ReliefF based bird sound classification, Applied Acoustics, Vol 176, May 2021, 107866
17. Y. Liu, G. Yang, M. Li, and H. Yin, "Variational mode decomposition denoising combined the detrended fluctuation analysis," Signal Processing, vol. 125, pp. 349–364, 2016.
18. A. Kaveh and V. R. Mahdavi, "Colliding bodies optimization: a novel meta-heuristic method," Computers & Structures, vol. 139, pp. 18–27, 2014.
19. B. Shaw, V. Mukherjee, and S. P. Ghoshal, "A novel opposition-based gravitational search algorithm for combined economic and emission dispatch problems of power systems," International Journal of Electrical Power and Energy Systems, vol. 35, no. 1, pp. 21–33, 2012.
20. A. Mohamed and G. B. Jasmon, "A new clustering technique for power system voltage stability analysis," Electric Machines and Power Systems, vol. 23, no. 4, pp. 389–403, 1995.
21. D. Kiran, A. R. Abhyankar, and B. K. Panigrahi, "Hierarchical clustering-based zone formation in power networks," in Proceedings of the 2016 National Power Systems Conference (NPSC), Bhubaneswar, India, December