Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 10, October 2024

Sign Language Recognition using Feature Selection Based on Binary Black-winged Kite Optimization Algorithm

Jixi Gu
Nanjing No.1 Middle School
International Department
Nanjing, China

Abstract—**Sign language recognition aims to help understand sign language and bridge the communication gap between speech or hearing-impaired and other non-impaired individuals. Sign language recognition belongs to the area of pattern recognition. Feature selection is an important basic approach for solving the pattern recognition problem, and it reduces and refines the feature set of data so that the generated feature subset can further improve the pattern recognition accuracy. In this paper, a new binary black-winged kit optimization algorithm-based feature selection method, BBKAFS, is proposed. When applying to sign language recognition, BBKAFS can not only reduce the feature dimensions, but also obtain higher recognition accuracies on three public sign language data sets.**

Keywords—**sign language recognition; feature selection; swarm intelligence; pattern recognition; black-winged kit algorithm**

I. INTRODUCTION

Sign language is different from oral language, and it is a sophisticated manual language employing hand gestures, body movements, and facial expressions to express individual feelings, thoughts, intentions, needs, and so on[1]. Therefore, sign language becomes a dominant tool of connection between speech or hearing-impaired communities and the general people. The understanding of sign language routinely depends on visual inputs, and automatic sign language recognition can use pattern recognition techniques to realize the translation of this visual-gestural language into the normal spoken or written language[2]. The higher the sign language recognition accuracy is, the smoother the communication between deaf and dumb people and the social world will be.

In the era of big data and artificial intelligence, pattern recognition algorithms or models require to have the faster learning speed, the stronger generalization capability, the better recognition accuracy, and so on. Feature Selection, which is often regarded as one of basic techniques related to the area of pattern recognition, can take advantage of some evaluation criteria to create the new feature subset based on original data. When recognition algorithms or models are applied to the new generated feature subset of data, the better performance will be achieved. Feature selection methods are broadly categorized into three kinds: embedded methods, wrapper methods, and filter methods[3]. Wrapper methods refine the feature subset by the operational results of one classification algorithm[4], and in this paper, the proposed approach based on binary swarm intelligence optimization is classified as this type.

Swarm intelligence optimization is a class of optimization methods, and it can search for the global solutions to problems by mimicking and modelling the behavioral characteristics of one group, and its algorithms not only are employed flexibly, but also are easy to be understood. Popular swarm intelligence optimization algorithms include particle swarm optimization[5], grey wolf optimization[6], whale optimization algorithm[7], Harris Hawk Optimization(HHO)[8], Black Kite optimization Algorithm (BKA)[9], and so on. It is worth noting that BKA is a latest meta-heuristic optimization algorithm inspired by the migration and attack behaviors of black-winged kites, and it exhibits the satisfactory performance in complex functions and engineering cases. Since feature selection is a binary discrete problem in fact, at present, many binary swarm intelligence optimization algorithms are commonly used in the wrapper based feature selection method, such as binary particle swarm optimization[10], binary grey wolf optimization[11] and Binary HHO(BHHO)[12]. Like BBHO based Feature Selection (BBHOFS) in [12], these binary versions of methods consider the classification error rate of a learning model as the optimization objective, and can help to search for and obtain better feature subsets in a discrete space.

In this paper, first we design a Binary BKA(BBKA), which employs the V-type function to transform continuous space into discrete space, and next present a new BBKA based Feature Selection method(BBKAFS), and finally apply this proposed BBKAFS to sign language recognition. Experimental results on three public sign language data sets demonstrate that when compared to other approaches, BBKAFS can not only find a feature subset with smaller size, but also acquire higher recognition accuracies.

II. BINARY BLACK-WINGED KITE ALGORITHM (BBKA)

BBKA based on the currently presented BKA[9] that mimics and models the predatory and migration behavior of black-winged kites mainly consists of three stages as follows:

A. Initialization Stage

Assume that X is a pack of n black-winged kites, and let $X=\{x_1,x_2,\cdots,x_i,\cdots,x_n\}$ where x_i is the position of the ith black-winged kite and each $x_i=\{x_{i1},x_{i2}\cdots,x_{ij},\cdots,x_{id}\}$ where x_{ij} represents the jth dimension of the position of the ith

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 10, October 2024

black-winged kite and d is the number of dimensions of the position space. Each x_{ij} can be initialized as follows:

$$x_{ij} = rand\,\mathrm{int}([0,1]) \quad (1)$$

where $rand\,\mathrm{int}([0,1])$ generates a random number of 0 or 1. Assume that $f(\cdot)$ is the optimization object of BBKA, that is, the fitness function. $f(x_i)$ is used to evaluate the position of the *i*th black-winged kite, and a kite located at the best position x_{best} is considered as the leader of kites. The position x_{best} is defined as:

$$x_{best} = \mathop{\arg\min}_{\forall x_i, x_i \in X} (f(x_i)) \quad (2)$$

, and $f(x_{best})$ is the optimal value of the fitness function.

B. Attacking Stage

When hunting small prey in the air, the black-winged kite needs to frequently adjust its position. First, to mimic this behavior, a mathematical model of updating the position x_i of the *i*th black-winged kite is defined as:

$$\hat{x}_i^c = \begin{cases} x_i + \beta \cdot (\sin(\theta)+1) \cdot x_i & \alpha < \theta \\ x_i + \beta \cdot (2\theta - 1) \cdot x_i & \alpha \ge \theta \end{cases} \quad (3)$$

where $\hat{x}_i^c$ is the new generated position of the *i*th black-winged kite in the attacking stage, like the original BKA, θ is a random number between 0 and 1, $\alpha = 0.9$, and β is gotten as follows:

$$\beta = 0.05 \cdot e^{-2\cdot\left(\frac{q}{Q}\right)^2} \quad (4)$$

where q is the current iteration number, and Q is the maximum number of iterations. Moreover, note that θ takes the different value for different i.

Second, BBKA asks to search the optimal position in the 0-1 binary space, but all new positions generated by Eq.(3) are in the continuous space. Therefore, a continuous to binary map is needed to fulfil the space conversion, and it is defined as:

$$\hat{x}_{ij}^b = \begin{cases} \neg x_{ij} & r1 < V(\hat{x}_{ij}^c) \\ x_{ij} & r1 \ge V(\hat{x}_{ij}^c) \end{cases} \quad (5)$$

where x_{ij} the *j*th dimension of the position x_i that is not updated by Eq.(3), $\hat{x}_{ij}^c$ is the *j*th dimension of the positions $\hat{x}_i^c$ in the continuous space, $\hat{x}_{ij}^b$ is the *j*th dimension of the positions $\hat{x}_i^b$ in the binary space, $\neg x_{ij}$ is the complement of x_{ij}, $r1$ is a random number between 0 and 1, $V(\cdot)$ is a V-type transfer function defined as:

$$V(t) = \left| \frac{\pi}{2} \cdot \arctan(\frac{\pi}{2} \cdot t) \right| \quad (6)$$

, and the curve image of this V-shaped function is shown in Fig.1. Furthermore, note that $r1$ takes the different value for different i and different j.

Last, after the completion of the continuous to binary conversion, x_i and $\hat{x}_i^b$ should be evaluated by the fitness function. If $f(\hat{x}_i^b) < f(x_i)$, then let $x_i = \hat{x}_i^b$; otherwise x_i remains unchanged.

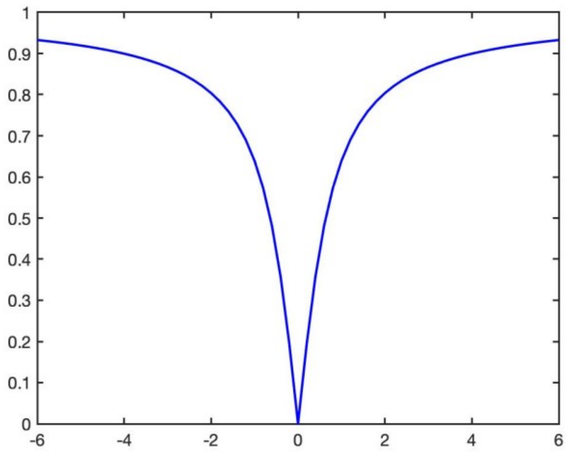

Fig.1. The curve image of the V-type function

C. Migration Stage

First, like BKA, a hypothetical migration behavior of the black-winged kites is mimicked and a related mathematical model is defined as:

$$\breve{x}_i^c = \begin{cases} x_i + w \cdot (x_i - x_{best}) & f(x_i) < f(x_g) \\ x_i + w \cdot (x_{best} - h \cdot x_i) & f(x_i) \ge f(x_g) \end{cases} \quad (7)$$

where $\breve{x}_i^c$ is the new generated position of the *i*th black-winged kite in the migration stage, x_{best} is the position of the black-winged kites' leader and is currently optimal, g is a random integer with the range of $[1,n]$, and w and h can be respectively computed as:

$$h = 2 \cdot \sin(\theta + \frac{\pi}{2}) \quad (8)$$

and

$$w = \tan((r2 - 0.5) \cdot \pi) \quad (9)$$

where θ takes the same value as θ of Eq.(3), w is a Cauchy random number related to Cauchy distribution, and $r2$ is a random number between 0 and 1. Besides, note that g takes the different values for different i, and $r2$ also takes the distinct values for distinct i.

Second, a continuous to binary space conversion is still employed in the migration stage, and a map similar to Eq.(5) is defined as:

$$\breve{x}_{ij}^b = \begin{cases} \neg x_{ij} & r3 < V(\breve{x}_{ij}^c) \\ x_{ij} & r3 \ge V(\breve{x}_{ij}^c) \end{cases} \quad (10)$$

where x_{ij} the jth dimension of the position x_i that is not updated by Eq.(7) after the attacking stage, $\breve{x}_{ij}^{c}$ is the jth dimension of the positions $\breve{x}_i^c$ in the continuous space, $\breve{x}_{ij}^{b}$ is the jth dimension of the positions $\breve{x}_i^b$ in the binary space, $\neg x_{ij}$ is the complement of x_{ij}, $r3$ is a random number between 0 and 1, and $V(\cdot)$ is Eq.(6). Additionally, note that $r3$ takes the different value for different i and different j.

Last, after the completion of the continuous to binary conversion, x_i and $\hat{x}_i^b$ should be also evaluated by the fitness function. If $f\left(\breve{x}_i^b\right) < f\left(x_i\right)$, then let $x_i = \breve{x}_i^b$; otherwise x_i remains unchanged.

III. BBKA-BASED FEATURE SELECTION (BBKAFS)

BBKA-based feature selection, that is, BBKAFS, belongs to the wrapper method, and for it, this paper designs a fitness function $f(\cdot)$ related to the classification error rate obtained from a simple classification algorithm, that is, K-Nearest Neighbors (KNN). Assume that each x_i ($i \in \{1, 2, \cdots, n\}$) is viewed as a candidate feature subset, and this fitness function $f(\cdot)$ [13] is defined as:

$$f(x_i) = (1-\delta)\cdot\Phi(x_i) + \delta\cdot\frac{\Psi(x_i)}{\Psi_{\max}} \tag{11}$$

where $\Phi(x_i)$ is the classification error rate on the data with the feature subset x_i, $\Psi_{\max}$ is the size of the original whole feature set, $\Psi(x_i)$ is the size of the feature subset x_i, and $\delta = 0.01$.

The detailed procedure for BBKAFS is given as follows:

Step 1: Let n be the number of candidate feature subsets, and Q be the maximum number of iterations. Set $q = 1$, $\alpha = 0.9$, and $\delta = 0.01$.

Step 2: Let $TrnFS$ and $ValFS$ respectively be the training and validation data specially used for feature selection, X be a set of all feature subsets, and each x_i ($i \in \{1, 2, \cdots, n\}$) in X be a candidate feature subset.

Step 3: Initialize each x_i by Eq.(1). For each x_i, carry out classification of KNN and obtain the classification error rate $\Phi(x_i)$ on $TrnFS$ and $ValFS$, and calculate the fitness function value $f(x_i)$ by Eq.(11). Sort all $f(x_i)$ ($i = 1, 2, \cdots, n$), and acquire x_{best} by Eq.(2).

Step 4: Set $i = 1$.

Step 5: Set the random numbers θ and $r1$, update x_i and get $\hat{x}_i^c$ by Eq.(3), and convert $\hat{x}_i^c$ to $\hat{x}_i^b$ by Eq.(5).

Step 6: For $\hat{x}_i^b$, perform KNN and get the classification error rate on $TrnFS$ and $ValFS$, and calculate the fitness function value $f\left(\hat{x}_i^b\right)$. If $f\left(\hat{x}_i^b\right) < f\left(x_i\right)$, then let $x_i = \hat{x}_i^b$; otherwise x_i remains unchanged.

Step 7: Set the random numbers g, $r2$ and $r3$, update x_i and get $\breve{x}_i^c$ by Eq.(7), and convert $\breve{x}_i^c$ to $\breve{x}_i^b$ by Eq.(10).

Step 8: For $\breve{x}_i^b$, perform KNN and get the classification error rate on $TrnFS$ and $ValFS$, and calculate the fitness value $f\left(\breve{x}_i^b\right)$. If $f\left(\breve{x}_i^b\right) < f\left(x_i\right)$, then let $x_i = \breve{x}_i^b$; otherwise x_i remains unchanged.

Step 9: Set $i = i+1$. If $i \le n$, then goto Step 5.

Step 10: Sort all $f(x_i)$ ($i = 1, 2, \cdots, n$), and acquire x_{best} by Eq.(2) again.

Step 11: Set $q = q+1$. If $q \le Q$, then goto Step 4; otherwise BBKAFS ends, and x_{best} is the finally attained optimal feature subset.

IV. SIGN LANGUAGE RECOGNITION EXPERIMENTS

A. Experimental Settings and Data Sets

First, since the essence of sign language recognition is regarded as the classification problem of sign language, and in order to measure the performance of the above proposed BBKAFS, we employed KNN for the data classification and the combination with the wrapper-based feature selection, and we also compared BBKAFS-based KNN with BBHOFS-based KNN and KNN without feature selection.

Second, the parameter K of KNN classifier was set to 5, For BBHOFS and BBKAFS, the number n of candidate feature subsets was set to 10, the maximum number Q of iterations was set to 100, Eq.(11) was used as the fitness function, and other parameters of BBHOFS was set according to [12].

Next, three sign language data sets from Kaggle[14] were used in experiments. There three data sets were as follows: British Sign Numbers(BSN), German Sign Language Alphabet(GSLA) and Leap Motion American Sign Language(LMASL). Each data set was divided into the training and testing sets with the approximate ratio of 8:2. KNN, BBHOFS-based KNN and BBKAFS-based KNN were performed and evaluated on these training and testing data. Moreover, for each data set, this training data that accounts for 80% of the total data was again split into $TrnFS$ and $ValFS$ in the approximate ratio of 5:5, and $TrnFS$ and $ValFS$ were specially used for the feature selection of BBHOFS and BBKAFS. Table I lists main characteristics of three sign language data sets, and Table II gives the number of samples in the testing data, the training data, $TrnFS$ and $ValFS$ for each data set.

Finally, experiments were done by Matlab on MacOS running on the personal computer with Inter Core i7 and 16GB

Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 10, October 2024

TABLE I. SIGN LANGUAGE DATA SETS USED IN EXPERIMENTS

Data set	The number of samples	The size of the feature set	The number of classes
BSN	1194	63	11
GSLA	7306	63	24
LMASL	5145	428	18

TABLE II. THE NUMBER OF SAMPLES IN THE TESTING, TRAINING, *TRNFS* AND VALFS DATA

Data set	Testing data	Training data	*TrnFS*	*ValFS*
BSN	239	955	477	478
GSLA	1461	5845	2922	2923
LMASL	1029	4116	2058	2058

TABLE III. COMPARISON OF THE SIZES OF OPTIMAL FEATURE SUBSETS

Data set	No feature selection	BBHOFS	BBKAFS
BSN	63	30	7
GSLA	63	32	10
LMASL	428	207	29

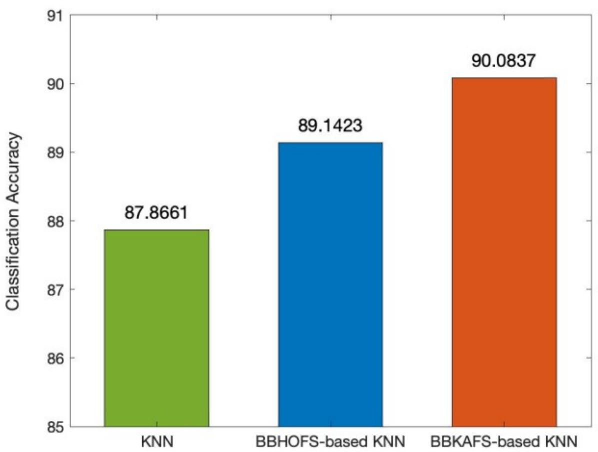

Fig.2. Comparison of classification accuracies on BSN

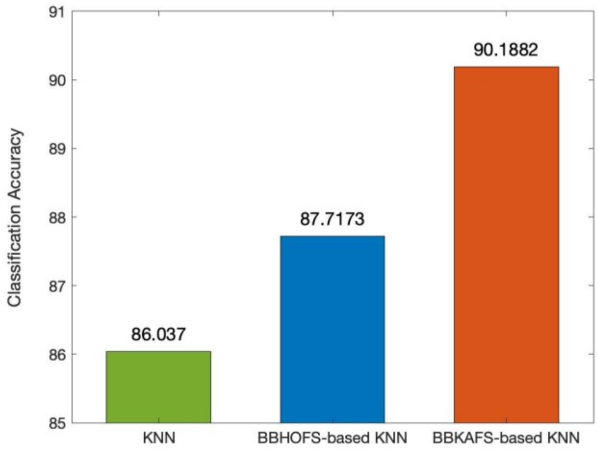

Fig.3. Comparison of classification accuracies on GSLA

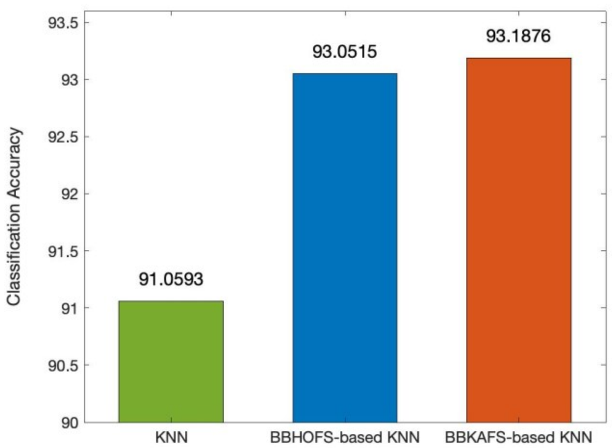

Fig.4. Comparison of classification accuracies on LMASL

RAM. All experimental results were averaged on 20 runs for each data set.

B. Experimenal Results

First, the comparison of the sizes of optimal feature subsets generated by BBHOFS and BBKAFS is reported in Table III. As shown in Table III, on the one hand, when no feature selection is employed, the optimal feature subset is the original entire feature set, and on the other hand, BBKAFS can acquire the lower size of the optimal feature subset than BBHOFS on each sign language data set.

Second, the classification accuracies on three sign language data sets are respectively shown in Fig.2, Fig.3 and Fig.4. In Fig.2 and Fig.3, over 90% classification accuracies are attained only by BBKAFS-based KNN, and in Fig.4, although there is little different classification performance between BBHOFS-based KNN and BBKAFS-based KNN, the latter, that is, the proposed BBKAFS-based KNN, still can achieve the highest accuracy. Furthermore, Fig.2, Fig.3 and Fig.4 describe that when compared with feature selection based KNN, such as BBHOFS-based and BBKAFS-based KNN, the simple KNN without feature selection underperforms on three sign language data sets.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 10, October 2024

V. CONCLUSIONS

To improve the accuracy of sign language recognition, this paper proposes an effective black-winged kite optimization algorithm based feature selection method, that is, BBKAFS. Compared to other approaches for sign language recognition, our method (i) enables to refine and get the optimal feature subset with the smaller size, and (ii) has the better sign language recognition performance.

REFERENCES

[1] S. Alyami, H. Luqman, and M. Hammoudeh, "Reviewing 25 years of continuous sign language recognition research: advances, challenges, and prospects," Information Processing & Management, vol.61, No.5, pp.103774, September 2024.
[2] Y.Q. Zhang, X.W. Jiang, "Recent advances on deep learning for sign language recognition," Computer Modeling in Engineering & Sciences, vol.139, no.3, pp.2399-2450, March 2024.
[3] A. Moslemi, "A tutorial-based survey on feature selection: recent advancements on feature selection," Engineering Applications of Artificial Intelligence, vol.126, part D, pp.107136, November 2023.
[4] B.H. Nguyen, B. Xue, and M.J. Zhang, "A survey on swarm intelligence approaches to feature selection in data mining,"Swarm and Evolutionary Computation, vol.54, pp.100663, May 2020.
[5] A.G. Gad, "Particle swarm optimization algorithm and its applications: a systematic review," Archives of Computational Methods in Engineering, vol.29, pp.2531-2561, April 2022.
[6] S. Mirjalili, S.M. Mirjalili, and A. Lewis, "Grey wolf optimizer," Advances in Engineering Software, vol.69, pp.46-61, March 2014.
[7] S. Mirjalili, A. Lewis, "The whale optimization algorithm," Advances in Engineering Software, vol.95, pp.51-67, May 2016.
[8] A.A. Heidari, S.Mirjalili, H. Faris, and et al., "Harris hawks optimization: algorithm and applications," Future Generation Computer Systems, vol.97, pp.849-872, August 2019.
[9] J. Wang, W.C. Wang, X.X. Hu, and et al., "Black-winged kite algorithm: a nature-inspired meta-heuristic for solving benchmark functions and engineering problems," Artifical Intelligence Rewiew, vol.57, March 2024.
[10] J.W. Too, A.R. Abdullah, and N.M. Saad, "A new co-evolution binary particle swarm optimization with multiple inertia weight strategy for feature selection," Informatics, vol.6, no.2, May 2019.
[11] J.W. Too, A.R. Abdullah, N.M. Saad, and et al., "A new competitive binary grey wolf optimizer to solve the feature selection problem in EMG signals classification," Computers, vol.7, no.4, November 2018.
[12] J.W. Too, A.R. Abdullah, and N.M. Saad, "A new quadratic binary harris hawk optimization for feature selection," Electronics, vol.8, no.10, October 2019.
[13] T. Dokeroglu, A. Deniz, H.E. Kiziloz, "A comprehensive survey on recent metaheuristics for feature selection," Neurocomputing, vol.494, no.14, pp.269-296, July 2022.
[14] Kaggle datasets website, https://www.kaggle.com/datasets.