Research Article

Hybrid Metric K-Nearest Neighbor Algorithm and Applications

Chao Zhang [ID],[1] **Peisi Zhong** [ID],[1] **Mei Liu** [ID],[1] **Qingjun Song** [ID],[2] **Zhongyuan Liang** [ID],[1] **and Xiao Wang** [ID] [1]

[1]*College of Mechanical and Electronic Engineering, Shandong University of Science and Technology, Shandong, Qingdao 266590, China*
[2]*College of Intelligent Equipment, Shandong University of Science and Technology, Shandong, Taian 271019, China*

Correspondence should be addressed to Peisi Zhong; pszhong_sdust@163.com and Mei Liu; lm_sdust@163.com

The K-Nearest Neighbor (KNN) algorithm is a classical machine learning algorithm. Most KNN algorithms are based on a single metric and do not further distinguish between repeated values in the range of K values, which can lead to a reduced classification effect and thus affect the accuracy of fault diagnosis. In this paper, a hybrid metric-based KNN algorithm is proposed to calculate a composite metric containing distance and direction information between test samples, which improves the discriminability of the samples. In the experiments, the hybrid metric KNN (HM-KNN) algorithm proposed in this paper is compared and validated with a variety of KNN algorithms based on a single distance metric on six data sets, and an HM-KNN application method is given for the forward gait stability control of a bipedal robot, where the abnormal motion is considered as a fault, and the distribution of zero moment points when the abnormal motion is generated is compared. The experimental results show that the algorithm has good data differentiation and generalization ability for different data sets, and it is feasible to apply it to the walking stability control of bipedal robots based on deep neural network control.

1. Introduction

KNN algorithm is a classical nonparametric machine learning classification algorithm. The trained KNN model is usually used as a fault classifier in engineering. Reliable KNN classification models can provide accurate fault detection and diagnosis (FDD) information. The obtained FDD information is used for system recovery or system decision-making, such as using an undamaged redundant system or reprogramming, to achieve the purpose of reliable operation of the robot and the safety of the robot and its surrounding environment [1, 2]. FDD approaches are usually divided into three categories: model-based, knowledge-based, and data-driven [3–5]. Knowledge-based [6] approaches typically relate identified behaviors to predefined known faults and diagnoses. Model-based approaches must construct a system model to describe real processes and perform fault detection and diagnosis by analyzing redundancies. This approach usually requires accurate modeling, and therefore, it is only effective in less than 10% of real-world applications [7, 8].

Unlike model-based approaches, data-driven approaches are gaining importance because they rely only on measurement data samples [9]. More importantly, FDD for robots in performing tasks requires fast, online, and small computational effort, and machine learning algorithms are well suited to meet the needs of robotic FDD tasks. Both traditional machine learning-based classification and deep neural network classification are typical data-driven FDD approaches. Although many scholars have applied deep learning techniques to the field of FDD [10, 11], deep neural networks are sensitive to hyperparameters, and for different hyperparameters, the model classification effect can vary dramatically, so the KNN algorithm, which does not depend on the initial hyperparameters, is a more desirable alternative algorithm. Therefore, the KNN algorithm is widely used in fault prediction [12–14] and fault diagnosis [15–20].

K-value selection, *k*-nearest neighbor voting rules, sample space partitioning, and intersample distance metric are the main influencing factors of KNN algorithm performance. There are adaptive *k*-values [21–25] and

k-nearest neighbor weighted voting methods [26–32] for the optimization of K-value research. Optimizing the K value selection method and the voting rules within the K value range can improve the performance of the KNN algorithm, but too large a sample space will have a negative impact on the performance of the KNN algorithm. When the sample space is too large, it also affects the classification accuracy of the KNN model to some extent, which is due to the limitation of the k-value taking, leading to the unbalanced samples in the k-value range. For this situation, the mainstream optimization method is to divide the whole sample space before calculating the distance, and the divided sample clusters are more targeted, and at the same time, since the number of samples contained in the sub-clusters is much smaller than the number of samples in the original set, dividing the sample space can also reduce the computation of KNN classifier regarding the distance between samples, such as KD tree [33], ball tree [34], clustering [35], and other methods. Although the sample space division can improve the performance of the algorithm to some extent, the distance metric method between samples has a greater impact on the performance of KNN algorithm because a reliable distance metric method is a guarantee to distinguish the basis of different kinds of samples. The optimization for the distance metric at this stage is the weighted calculation of each dimensional distance [36, 37] or the improvement of a single metric [38].

Despite the fact that many methods have been proposed to improve the performance of the KNN algorithm, the methods to increase the distinguishability between data still need further study. In this paper, we proposed a hybrid metric-based KNN algorithm, which can further distinguish data that are difficult to be distinguished by a single distance within the K-neighborhood, thus improving the classification performance of the algorithm. Specifically, firstly, a hybrid metric is designed, which contains the distance information and phase information between two samples. To avoid unnecessary consumption of computational resources, a matching mechanism is designed to detect equidistant samples in the k-value range, and according to the type of detection results, different cases of equidistant samples in the k-value range are distinguished and the mixed metric of equidistant samples is obtained to further distinguish the class of equidistant samples.

Although scholars have done a lot of research work on the KNN algorithm, there is less research on equidistant sample differentiation when sorting in the K range. Aiming at the problem that equidistant samples between samples within the range of K value affect the classification accuracy, a hybrid metric KNN (HM-KNN) algorithm is proposed, and a simple application example is given. In the example, the abnormal motion input is regarded as a fault affecting the normal gait, the abnormal motion position is determined according to the classification results, and the action is reoutput according to the HM-KNN model, to improve the walking stability of biped robot.

The main contributions of our work can be viewed from the following aspects:

(1) The existence of equidistant samples in the K-value range is less studied, so a hybrid metric KNN algorithm is proposed to further distinguish equidistant samples in the K-value range and achieve the effect of optimizing the KNN algorithm

(2) A mechanism is designed to detect duplicate values in the range of k values, which avoids the need to find the hybrid metric for all samples and reduces the computational effort of the algorithm

(3) The classification effect of the HM-KNN algorithm is verified on multiple datasets, and the experimental results show that HM-KNN performs better compared with other KNN models

(4) A biped robot gait controller framework based on HM-KNN, a data-driven bipedal robot gait controller, is designed, and the experimental results show that the control framework has a positive effect on improving the gait stability of the bipedal robot

The rest of this paper is organized as follows. Section 2 introduces the details of the proposed HM-KNN method. Experimental results are illustrated in Section 3. Section 4 concludes this paper and offers future work.

2. Methods

2.1. Background. KNN is a very efficient nonparametric classification method, which is essentially a predictive supervised classification algorithm. It is also a typically inert learning method, constructing a model at the last moment before classifying a given test tuple. Only when a test tuple is seen it is generalized to classify the tuple based on the similarity of the stored training tuples. The most common way to measure the similarity of two samples is based on the Euclidean distance, Marxian distance, etc., between the data. The classical KNN algorithm flow is as follows.

(1) Calculate the distances between the samples to be measured and the known class samples

(2) Sort the samples in increasing order of distance

(3) Select the k points with the smallest distance from the current point

(4) Count the frequency of the category in which the first k points are located

(5) Return the category with the highest frequency among the first k points as the predicted classification of the current point

Figure 1 is a schematic diagram of KNN classification. It is known that there are three classes, which are blue dots, orange squares, and green triangles. The yellow area has 6 sampling points, which means the k value is 6. The red point is the sample to be tested. The category with the highest frequency among the first k points in the statistics is the green triangle, so it is determined that the red points belong to the category represented by the green triangle.

The KNN algorithm has the advantages of simplicity and efficiency, low retraining cost, low complexity of the
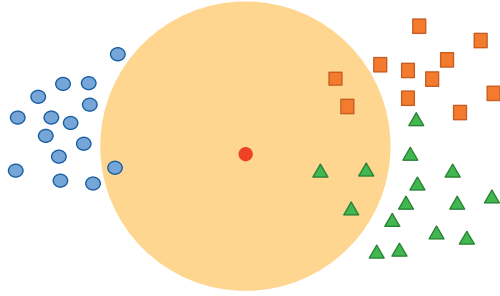
FIGURE 1: A KNN-based triple classification problem.

FIGURE 2: Two causes of equidistant samples.

FIGURE 3: Two cases of equidistant samples.

algorithm, and being suitable for automatic classification of large samples. However, KNN for classification depends on the similarity metric between data, and a single similarity metric is sometimes difficult to achieve the classification requirements, so this paper proposes a hybrid metric-based KNN algorithm.

2.2. Hybrid Metric. KNN classification is reliable when the distances between the samples to be tested and the known samples of different categories can be clearly distinguished. Otherwise, the reliability of the KNN classification results in decreases. Such points with indistinguishable distances become equidistant sample points, which are defined as follows.

Known sample points that are not sufficiently differentiated by a single distance measure or similarity measure in the range of K values of the sample points to be measured are called equidistant sample points.

Two causes of equidistant samples are given in Figure 2. The red pentagon indicates the sample to be tested, and the green triangle and blue dots indicate two different known categories of sample points. d_1 is the distance between the sample to be tested and the green triangle sample. d_2 is the distance between the sample to be tested and the blue dot sample.

Figure 2(a) represents the case where a single metric based on the space distance between samples cannot distinguish equidistant samples. In this case, the distances between different types of known samples and the samples to be measured are the same. A single distance metric, in this case, cannot distinguish the data. For example, when a bipedal robot walks with asymmetric gait, the joint angle data of the left leg support phase and the right leg support phase is the case represented in Figure 2(a). Figure 2(b) represents the case where a single vector pinch angle based metric is unable to distinguish between equidistant samples. In this case, the directional vector angle between the known samples of different kinds and the sample to be measured is 0. The case of Figure 2(b) may arise if there is a need to distinguish the sizes of objects with the same shape.

The hybrid metric consists of distance information L and a phase information P. Figure 3 illustrates the hybrid metric based on the distance metric, and when the single distance information L cannot be differentiated, the phase information P is used for the equidistant samples for further differentiation.
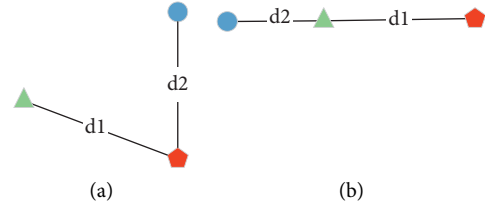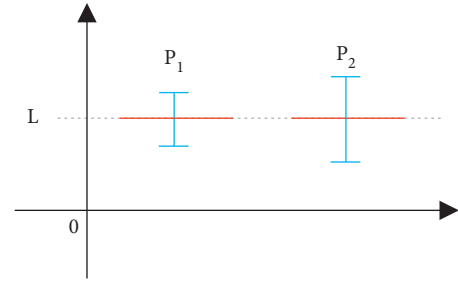
2.2.1. Distance Information L. To eliminate the dimensional differences between the dimensions of the samples, the Mahalanobis distance between the sample to be measured and the known sample is calculated according to equations (1) and (2). The covariance of the training set samples is calculated according to equation (1), and the distance is calculated according to equation (2).

$$\Sigma = \mathrm{cov}(X_0, \ldots, X_n = E[(X_0 - \mu_0) \ldots (X_n - \mu_n)]), \quad (1)$$

where X_i is the training set sample and μ_i denotes the mean value corresponding to the sample.

$$d(X_i, Y_i) = \sqrt{(X_i - Y_i)^T \Sigma^{-1} (X_i - Y_i)}, \quad (2)$$

where $d(X_i, Y_i)$ indicates the distance between samples X_i and Y_i.

2.2.2. Phase Information P. The calculated Mahalanobis distance is called distance information. The direction information is calculated by the cosine similarity. Calculate the cosine similarity according to the following equation:

$$\cos(X_i, Y_i) = \sum_{i=1}^{n} \frac{(X_i \cdot Y_i)}{\|X_i\|\|Y_i\|}. \quad (3)$$

Based on the distance information L and phase information P between samples, the hybrid metric d_{hm} is calculated according to the following equation:

$$d_{hm} = L^s + w(1 - P),$$
$$w\varepsilon[0, 1], \quad (4)$$

where d_{hm} is the hybrid metric and w is the weighting factor. S is an identifier with a value of 0 or 1, which is set to 0 if there are equidistant samples.

When equidistant data exist in the range of K values, it is known from equation (4) that d_{hm} floats up and down at the

central value 1. To increase the data differentiation, the mapping function is defined as equation (5). The function curve is shown in Figure 4.

$$F(x) = \frac{1}{1 + e^{-10(x-1)}}. \tag{5}$$

The final hybrid metric is calculated as shown in the following equation:

$$D_{hm} = F(d_{hm}), \tag{6}$$

where D_{hm} is the value of the hybrid metric after mapping by the mapping function $F(x)$. According to the function image shown in Figure 4, ω is taken as 0.3 in order to ensure that the value of the hybrid metric falls within the linear interval of the mapping function.

2.3. K-Nearest Neighbor Set Update.

Three cases of known occurrence of equidistant sample points when K is a fixed value are shown in Figure 5. These three cases are defined in Figure 5, where C indicates the classification labels of the known samples returned according to the distance between the sample points to be tested and the sample points known to the classification, the subscript represents the classification, the dashed box A is the size of K values, and the dashed box B is the number of points indicating equal distances between the sample points to be tested and the sample points known to the classification.

Figure 5(a) indicates that the repetition area is within the range of K values, which will not change the frequency of the sample categories falling within the range of K. In this case, the sample points of equal distance will not affect the classification result, and no secondary differentiation of the samples is needed, Figure 5(b) indicates that the repetition area covers the range of K values, which will change the frequency of the sample categories falling within the range of K, and a mixed metric differentiation of the repetition values within the range of dashed box B is needed, Figure 5(c) indicates that the repetition area is at the end of both sides of K values, which will also change the frequency of the sample categories falling within the range of K. This case also changes the frequency of the sample categories falling in the K-range and requires a mixture of metrics to distinguish the Neighbors at the end of the K-value region.

The cases in Figures 5(b) and 5(c) affect the final classification results, and the hybrid measure between the samples in region B and the samples to be tested needs to be calculated one by one. The samples contained in region B are sorted according to the order of the hybrid measure from smallest to largest. After obtaining the ascending series of the hybrid measure, the classification with the highest frequency is counted according to equations (7) and (8).

The set of samples in region A is named $A(k)$. k is the value of K taken in KNN. The number of samples in region B is m. The set of samples in region B is named $B(m)$.
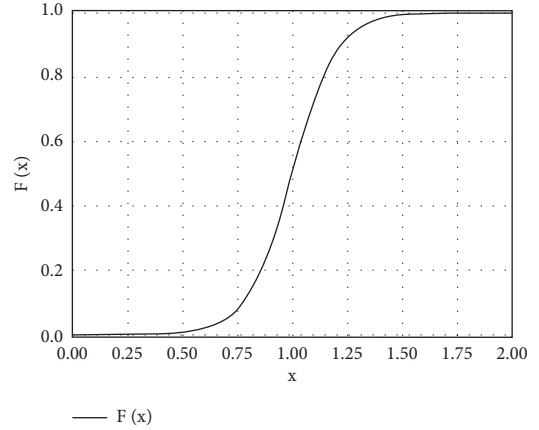

FIGURE 4: Function curve of $F(x)$.

$$S(y_i, c_j) = \begin{cases} 0, & y_i \neq c_j, \\ 1, & y_i = c_j, \end{cases} \tag{7}$$

where y_i is the classification label of the samples in $B(m)$. c_j is the classification label contained in the dataset. When y_i is the same as c_j, $S(y_i, c_j)$ returns 1.

$$y = \arg\max_{c_j} \sum_{x_i \in A(k)} S(y_i, c_j), \quad i = 1, 2, \ldots, n; \; j = 1, 2, \ldots, k, \tag{8}$$

where y_i is the final determination of the sample to be tested.

Equation (8) can determine the class of the sample to be tested based on the samples within the set A(k). The set $A(k)$ in Figures 5(b) and 5(c) needs further clarification.

The sample hybrid metric values within $B(m)$ are calculated one by one, and the samples in $B(m)$ are arranged in ascending order according to the hybrid metric values. The obtained ascending sequence is defined as $D(m)$ in the form of the following equation:

$$D(m) = \{(x_1, y_1), \ldots, (x_i, y_i) \mid x_i \in B(m), \quad i \in (0, m]\}, \tag{9}$$

where y_i is the final determination of the sample to be tested.

In Figure 5(b) $A(k)$ is a subset of $B(m)$. The updated $A(k)$ used for the final classification is given in the following equation:

$$A(k)_{new}^{5b} = \{(x_1, y_1), \ldots, (x_j, y_j) \mid x_j \in D(m), \quad j \in (0, k]\}. \tag{10}$$

In Figure 5(c), there exists an intersection of $A(k)$ and $B(m)$, which is defined as $C(p)$. n samples are included in $C(n)$. The updated $A(k)$ used for the final classification is given in the following equation:

$$A(k)_1 = \{(x_1, y_1), \ldots, (x_i, y_i) \mid x_i \in A(k), \quad i \in (0, k - n]\}, \tag{11}$$

$$A(k)_2 = \{(x_1, y_1), \ldots, (x_j, y_i) \mid x_j \in D(m), \quad j \in (0, p]\}, \tag{12}$$
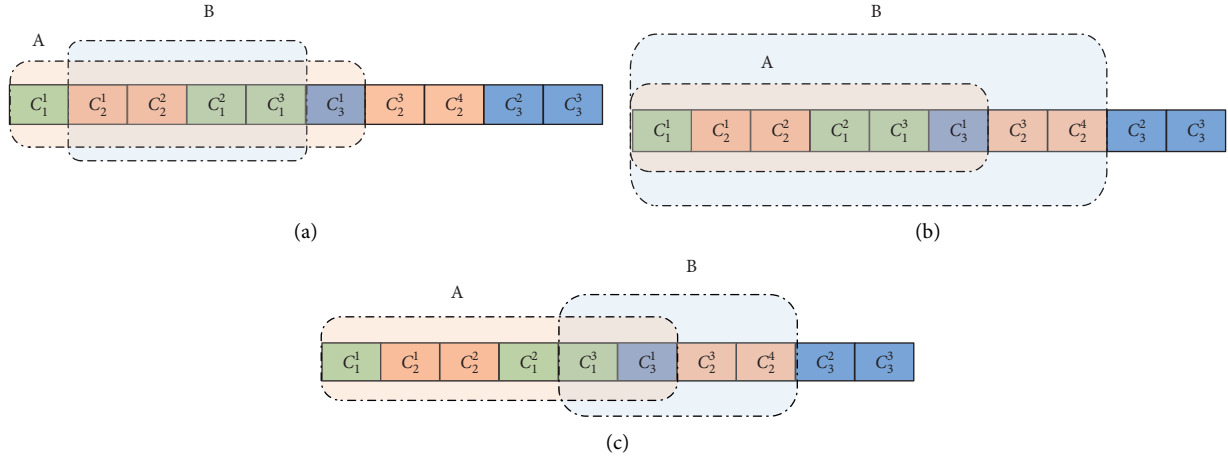
FIGURE 5: Classification of cases with duplicate values within the first K minimum values.

$$A(k)_{\text{new}}^{5c} = \{A(k)_1, A(k)_2\}. \tag{13}$$

Combining the three cases in Figure 5, the update rule of $A(k)$ is given in the following equation:

$$A(k)_{\text{new}} = \begin{cases} A(k), & \text{case in Figure 5 (a)}, \\ A(k)_{\text{new}}^{5b}, & \text{case in Figure 5 (b)}, \\ A(k)_{\text{new}}^{5c}, & \text{case in Figure 5 (c)}. \end{cases} \tag{14}$$

The three cases shown in Figure 5 are more explicitly expressed as described below. In Figure 5(a), $B(m) \subset A(k)$. In Figure 5(b), $B(m) \supseteq A(k)$. In Figure 5(c), $B(m) \cap A(k) \neq \varnothing$ and $B(m) \cup A(k) \neq B(m)$ and $B(m) \cup A(k) \neq A(k)$.

2.4. Algorithm Time Complexity Analysis.

The KNN based on the Euclidean Distance metric short for E-KNN, the KNN based on the Manhattan Distance metric short for MH-KNN, and the KNN based on the cosine similarity metric short for C-KNN, the KNN algorithm based on Chebyshev distance short for CH-KNN, the KNN algorithm based on Mahalanobis Distance metric short for M-KNN, and the KNN algorithm based on the hybrid metric short for HM-KNN are proposed in this paper.

Algorithm time complexity is an important indicator of algorithm performance, and the HM-KNN algorithm proposed in this paper differs from other comparative KNN algorithms in the main difference in the mixing metric and $A(k)$ update. So, the following discussion addresses these two main factors that affect the time complexity.

An important factor affecting the time complexity of the KNN algorithm is the method of measuring the variance between samples. Assume that the time frequency of calculating the distance between a single dimensional known sample and a single dimensional sample to be measured using a single metric is 1. Among the above KNN models, only the HM-KNN model uses the hybrid metric, and the rest of the KNN models are single metric. The time frequencies of the 5 KNN models using a single metric are shown in the following equation:

$$T(n)_d^{\text{other KNN}} = dn, \tag{15}$$

where d is the sample dimension. n is the number of known data samples.

The calculation of the discrepancy between the samples to be measured and the known samples in the HM-KNN algorithm consists of two processes. First, a single metric, such as the Marxian distance, Euclidean distance, etc., is calculated between the sample to be measured and the known sample, and then the hybrid metric between the elements in the $B(m)$ set and the sample to be measured is calculated based on the detected $B(m)$ set. The hybrid metric is calculated according to equations (2) to (6), and the time frequency of the hybrid metric can be estimated as shown in the following equation:

$$T(n)_d^{\text{HM-KNN}} = d(n + 3m), \tag{16}$$

where m is the number of samples in $B(m)$.

Another important factor that affects the time complexity of the KNN algorithm is the sample ordering. In the KNN models described in this paper, the sorting methods all use the insertion sorting method. Except for HM-KNN, the highest ranking time frequency of the other five KNN models is shown in the following equation:

$$\text{MAX}\left(T(n)_Q^{\text{other KNN}}\right) = n^2. \tag{17}$$

The HM-KNN has an additional updated mechanism of $A(k)$ based on a single metric ranking. According to equations (7) to (14), the time frequency of $A(k)$ update can be estimated as in the following equation:

$$\text{MAX}\left(T(n)_Q^{\text{HM-KNN}}\right) = n^2 + m^2. \tag{18}$$

Comparing equations (15) and (16), the difference in the time frequency between HM-KNN and other KNN algorithms is $3md$ when calculating the difference between samples. Comparing equations (17) and (18), the difference in the time frequency between HM-KNN and other KNN algorithms is m^2 when sorting to obtain $A(k)$. Generally, $m \ll n$, when n increases, the difference in time frequency generated by m can be neglected. The highest subterm of the final time frequency of HM-KNN is n^2. Based on the calculated time frequency, the time complexity of HM-KNN and other HNN algorithms can be derived, as shown in Table 1.

According to Table 1, it can be concluded that although HM-KNN increases the hybrid measure of samples in the local range and the update mechanism of $A(k)$, it does not affect the increase of the highest subterm of time frequency, which means that HM-KNN does not increase the algorithm time complexity significantly while increasing the classification accuracy.

2.5. HM-KNN Algorithm.

The HM-KNN pseudocode is based on the hybrid metric method proposed in Section 2.2 and the k-neighborhood set update method proposed in Section 2.3, as Figure 6.

In Figure 6, d denotes the value of any single metric between the input samples to be tested and the samples of known class, such as Euclidean distance, Marxian distance, Manhattan distance, etc.

3. Results and Discussion

3.1. Results.

The experiment is divided into two parts. The first part compares the performance of HM-KNN with that of the single metric-based KNN on five UCI (University of California Irvine) public datasets. The HM-KNN algorithm flow in the first part is shown in the classical KNN algorithm flow in Section 2.1, but the way to measure the difference between the samples to be measured and the known category samples is the mixed metric. This is to compare the difference in performance between the mixed metric and other single metric approaches on a generic dataset. The second part compares the performance of the HM-KNN algorithm with other KNN algorithms on a collected bipedal robot forward gait dataset containing equidistant samples. The flow of the HM-KNN algorithm used in this section is shown in Figure 6. And the optimized HM-KNN algorithm is used in the bipedal robot forward gait walking task.

3.1.1. Performance Comparison of Different Classification Models on UCI Dataset.

In this section, 5 UCI datasets were selected to validate the algorithm. The four datasets are the Iris dataset, the heart disease dataset, the wine dataset, and the breast cancer dataset. The basic information of the datasets is given in Table 2.

Figure 7 gives a comparison of the classification results of the hybrid metric-based KNN proposed in this paper and the other five commonly used metric-based KNN algorithms. The other five metrics are Euclidean distance, Manhattan

TABLE 1: Maximum time complexity of different KNN algorithms.

Algorithm name	Time complexity
E-KNN	$O(k \times n^2)$
MH-KNN	$O(k \times n^2)$
C-KNN	$O(k \times n^2)$
CH-KNN	$O(k \times n^2)$
M-KNN	$O(k \times n^2)$
HM-KNN	$O(k \times n^2)$

distance, cosine similarity, Chebyshev distance, and Mahalanobis distance.

In Figure 7(a), this paper compares the classification accuracy rates of 6 KNN classification models on the Iris dataset. When $K < 5$, the HM-KNN model performs the best among the 6 KNN models with the highest classification accuracy of 0.9873 and the lowest of 0.9774. When $4 < K < 16$, the mean value of the classification accuracy of the HM-KNN model is 0.9678, but there is some fluctuation, which is caused by the change of the sample number in the range of K. However, as the value of K increases, after $K > 14$, the HM- KNN model classification accuracy rises to the highest value of 1. The results indicate that HM-KNN has better generalization ability on this dataset.

Comparing the 6 KNN models in terms of model stability, the HM-KNN model performs the best, with a mean classification accuracy of 0.9773 and a standard deviation of 0.0143, and has the highest mean classification accuracy and the lowest variance compared with the other five KNN classification models. In summary, the HM-KNN model has a higher classification accuracy, better generalization ability, and better stability compared with the other five KNN models on this dataset.

In Figure 7(b), this paper compares the classification accuracy rate of the 6 KNN classification models on the heart disease dataset. When $K < 7$, except for the CH-KNN model, which fluctuates less, all the KNN models fluctuate more because as the value of K increases, the frequency of sample categories falling within the range of K values changes, causing accurate classification rate to fluctuate, and the classification models are not stable at this time. When $K > 6$, the models are relatively stable. At $K = 8$ and $K = 9$, the HM-KNN model achieves a maximum of 0.9673. At $6 < K < 18$, the HM-KNN model was significantly better than the other five classification models in terms of accuracy. When $K = 18$, the classification accuracy is slightly lower than that of the M-KNN model and C-KNN model, and the classification accuracy is 0.9442. When $K = 19$, the classification accuracy increases and is slightly higher than that of M-KNN. When K is not greater than 19, HM-KNN has relatively higher classification accuracy and better generalization ability compared with other models except for individual values of K. When K is not greater than 19, HM-KNN has relatively higher classification accuracy and better generalization ability compared with other models.

Comparing the 6 KNN models in terms of model stability, the HM-KNN model has a mean classification accuracy of 0.9417 and a standard deviation of 0.0326. The HM-KNN model has the highest mean classification

Algorithm Hybrid Metric KNN
Input: x_i
Initial: k; g is Equidistant sample marker;
while do
Calculate the distance d_i between x_i and each known data;
Store the known samples in Z according to the increasing order of d_i;
Select the first k samples from Z and store them in A (k);
if equidistant samples exist do
Obtain the set of equidistant samples B (m);
g = True;
else do
g = False;
if g = False do
Select the first k minimum values according to the ordering in A (k);
else do
Determine the distribution of equidistant samples and update A (k) according to Eqs. (7) to (14);
end if
Return the classification label with the highest frequency in A (k) as the predicted classification of the test data.
end while

FIGURE 6: Pseudocode of hybrid metric KNN algorithm.

TABLE 2: Data set parameters.

Dataset	Instances	Attributes
Iris	150	4
Heart disease	304	14
Wine	178	13
Breast cancer	569	30
Hydraulic system condition monitoring	2205	43680

accuracy but the largest variance compared with the remaining five KNN classification models, which is caused by the initial dataset cut, and it is obvious from the figure that when $K = 1$, each classification model classification. It is obvious from the figure that the classification accuracy of each model is lower than 0.88 when $K = 1$ and the classification accuracy of HM-KNN classification increase the fastest, causing the variance to be too large. In summary, the HM-KNN model has a higher classification accuracy compared with the other five KNN models in this dataset and has better stability when $K > 6$. After $K > 14$, the generalization ability is slightly lower than that of M-KNN as the value of K increases but better than the other four classification models.

In Figure 7(c), this paper compares the classification accuracy of the 6 KNN classification models on the wine dataset. The classification accuracy of the 6 KNN models is relatively stable without significant fluctuations. When $K = 4$ is, the HM-KNN classification accuracy reaches the highest value of 1, and the classification accuracy of the rest K values is stable at 0.9777. The results indicate that HM-KNN has better generalization ability on this dataset. Comparing the 6

KNN models in terms of model stability, the HM-KNN model has a mean classification accuracy of 0.9789 and a standard deviation of 0.0051. It has the highest mean classification accuracy and the smallest variance compared with the other five KNN classification models. In summary, the HM-KNN model has higher classification accuracy, better generalization ability, and better stability than the other five KNN models in this dataset.

In Figure 7(d), this paper compares the classification accuracy of 6 KNN classification models on the breast cancer dataset. The results in the figure show that the classification accuracy of the 6 types of KNN classification models on this dataset is relatively satisfactory, with the classification accuracy rate above 0.95. The HM-KNN classification accuracy reached the maximum value of 0.9930 when $K = 3$ and $K = 4$. The classification accuracy of all 6 KNN models decreased as the value of K increased. When $K > 7$, HM-KNN classification stability is better than other classification models. However, the classification accuracy decreases and fluctuates with increasing K values, which is due to the presence of interference data in the sample.

Comparing the 6 KNN models in terms of model stability, the HM-KNN model has a mean value of 0.9806 and a standard deviation of 0.0072. It has the highest mean value of classification accuracy and the lowest variance compared to the other five KNN classification models. In summary, the HM-KNN model has higher classification accuracy, better generalization ability, and better stability than the other five KNN models in this dataset.

HM-KNN is a secondary classification of repetitive data based on M-KNN. Combining subfigures Figures 7(a)–7(d),
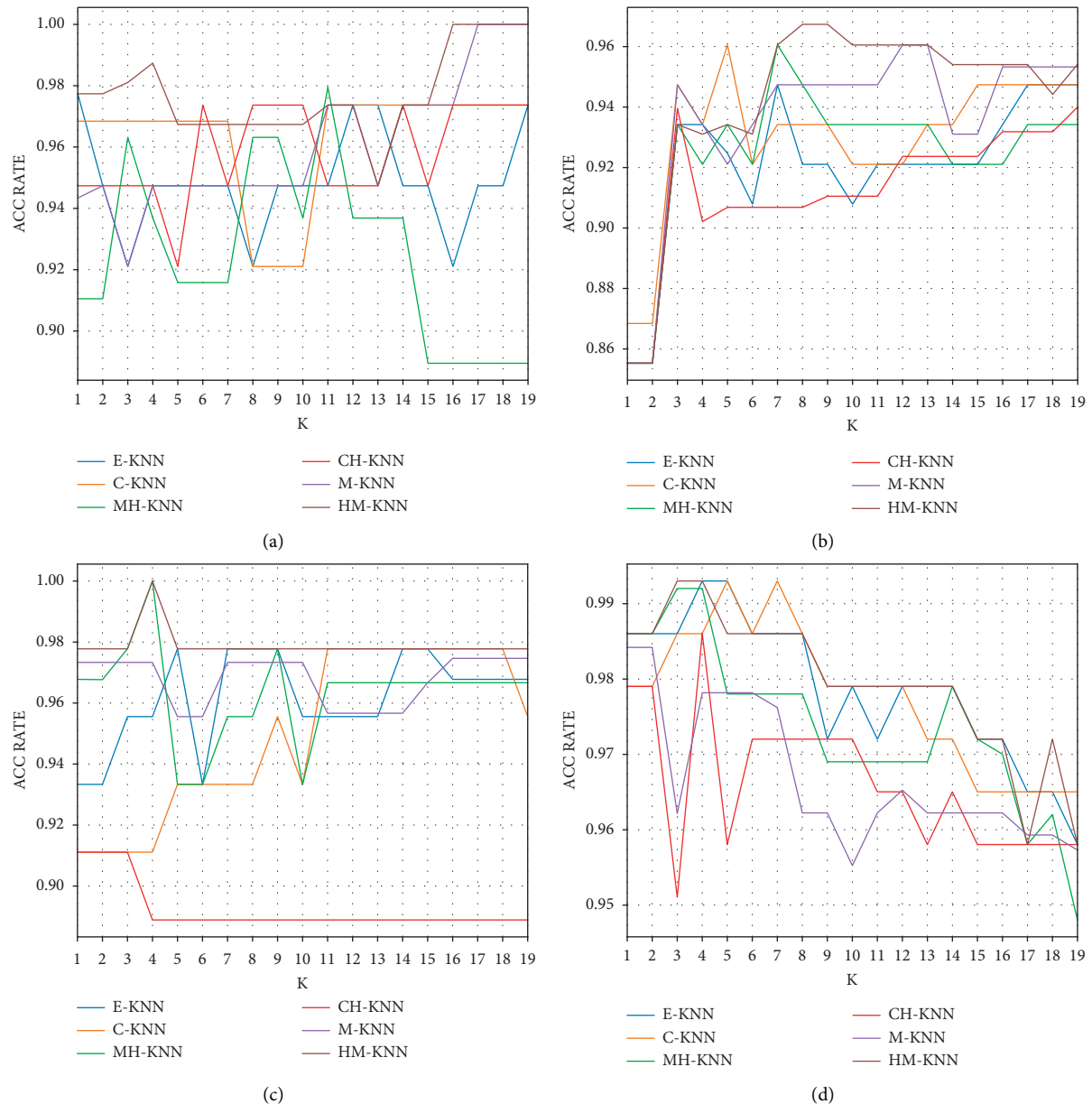
(a)

(b)

(c)

(d)

Figure 7: Comparison of classification accuracy of different KNN models on UCI classical dataset (cross validation 25%).

it can be concluded that the HM-KNN proposed in the paper is more general than the other five KNN models on the four classical UCI datasets, and has better generalization ability, and significantly outperforms M-KNN. It can also be seen that the hybrid metric has a boosting effect on the correct classification rate at suitable K values when a single metric cannot distinguish the samples.

Using the validation of the hydraulic system condition monitoring dataset [39], the dataset was obtained experimentally through a hydraulic test stand. The test stand consists of a main working unit and an auxiliary cooling and filtering circuit, which are connected through the oil tank.

The system periodically repeats constant load cycles and measures process values while the state of the four hydraulic components changes quantitatively. The data set

characteristics include pressure data, temperature data, and flow data, among others.

The Hydraulic System Conditions Monitoring dataset has five target categories corresponding to the five parameters describing the system, and the specific classification is given in Table 3. The distribution of the data in the table shows that the samples are distributed relatively evenly and there are no cases where there are very few or very many samples in a particular category. Therefore, it is feasible to use the classification accuracy to describe the classification model on this dataset.

In Figure 8, we compare the accuracy of six KNN classification models on the hydraulic system condition monitoring dataset. When $K < 10$ and $12 < K < 17$, the HM-KNN classification rate is lower than that of M-KNN. When

TABLE 3: Parameters for hydraulic system condition monitoring dataset.

| Target item name	Conditions	Instances
Cooler condition	Close to total failure	732
	Reduced efficiency	732
	Full efficiency	741
Valve condition	Optimal switching behavior	1125
	Small lag	360
	Severe lag	360
	Close to total failure	360
Internal pump leakage	No leakage	1221
	Weak leakage	492
	Severe leakage	492
Hydraulic accumulator	Optimal pressure	599
	Slightly reduced pressure	399
	Severely reduced pressure	399
	Close to total failure	808
Stable flag	Conditions were stable	1449
	Static conditions might not have been reached yet	756
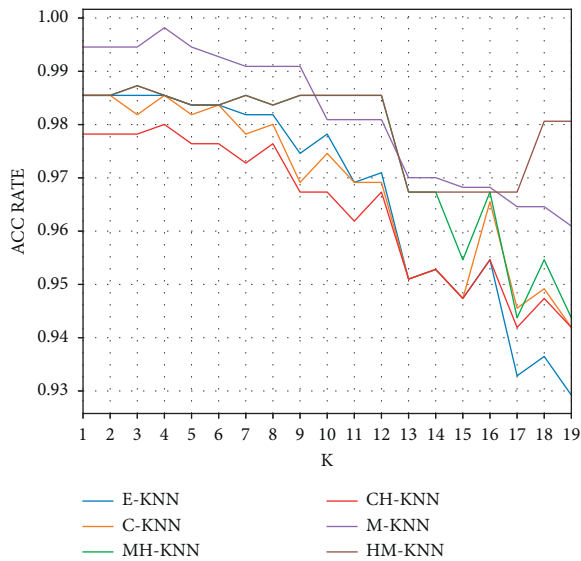
FIGURE 8: Comparison of classification accuracy of different KNN models on hydraulic system condition monitoring (HSCM) dataset (cross validation 25%).

$8 < K < 13$ and $K > 16$, the HM-KNN model outperforms the other five KNN models. Compared with the other five KNN models, the generalization ability of HM-KNN is relatively better with the increase of K. When the value of K increases from 18 to 19, the classification accuracy of HM-KNN improves to 0.9806, which indicates that HM-KNN can make more reliable judgments in a larger range of K. This indicates that HM-KNNNN can make more reliable classifications.

Comparing the 6 KNN models in terms of model stability, the mean value of the classification accuracy of the HM-KNN model is 0.9827 and the standard deviation is 0.0078. The mean value of the classification accuracy of M-KNN is equal to that of HM-KNN, but the standard deviation of the classification accuracy of the M-KNN model is larger than that of the HM-KNN model. The standard deviation of the M-KNN model is 0.0093. In summary, the

HM-KNN model has a relatively higher classification accuracy, better generalization ability, and better stability than the other KNN models on this dataset.

3.1.2. Performance Comparison of Different Classification Models on Biped Robot Forward Walking Dataset. The biped robot forward walking dataset used for the experiments is based on the NAO robot collection shown in Figure 9. Nao robot with robot body version H25. The robot has a total of 6 degrees of freedom for each leg, which are 3 degrees of freedom for the hip joint, 1 degree of freedom for the knee joint, and 2 degrees of freedom for the ankle joint.

The experimental dataset consisted of one normal gait cycle bipedal joint angle data and 19 abnormal gait cycle bipedal joint angle data of NAO robot forward walking based on deep network control. The data set angle data to include hip pitch angle, hip roll angle, knee pitch angle, ankle pitch angle, and ankle roll angle.

The dataset includes the biped motor angles for one normal forward walking gait cycle of the biped robot and multiple forward walking abnormal gait cycles of the biped motor angles, of which there are 28 items of data in each class, and the information of the dataset is given in Table 4.

The dataset is relatively balanced in each class, and there are no classes with large differences in sample size, and it is feasible to use the classification model as a measure of the classification accuracy. The classification results are shown in Figure 10.

The results in Figure 10 show that M-KNN, CH-KNN, C-KNN, and E-KNN had better performance than HM-KNN when $K \leq 8$. In general, when the value of K is small, it is easy to lead to overfitting and insufficient generalization ability, so it is difficult to evaluate the classification performance of the KNN models. When $K \in [9, 19]$, the classification accuracy of the HM-KNN model is higher than that of other KNN algorithms, and the classification accuracy reaches 0.987 and is more stable. It indicates that for this dataset, HM-KNN has good generalization ability. When $K > 16$, the classification accuracy decreases, but its classification effect is better than other KNN models.

FIGURE 9: Nao robot.

TABLE 4: Data set parameters.

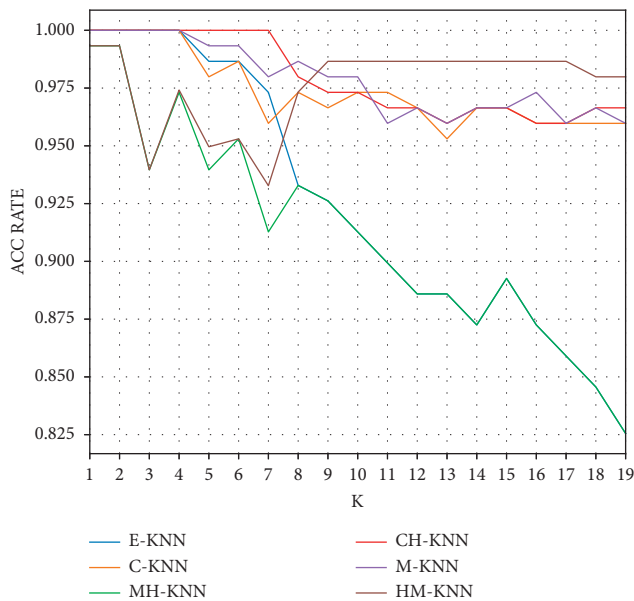| Dataset	Instances	Attributes
Biped robot forward walking	594	10

FIGURE 10: Comparison of classification accuracy of different KNN models on biped robot forward walking (BRFW) dataset (cross-validation 25%).

3.1.3. Summary. Comparing the 6 KNN models in terms of model stability, the M-KNN model has the best performance with the mean value of 0.9784 and the standard deviation of 0.0155. The mean value of the HM-KNN classification accuracy is slightly lower than that of M-KNN with a value of 0.9762. The standard deviation of the HM-KNN classification accuracy is slightly higher than that of M-KNN, with a value of 0.0183. However, the median classification accuracy

of HM-KNN is higher than the median accurate rate of M-KNN with a value of 0.9866, indicating that the accuracy of HM-KNN classification is higher than that of M-KNN when $K > 10$.

Compared with the remaining 5 KNN classification models, the median accuracy of classification is the largest and the standard deviation is the smallest when $K > 10$. In summary, when $K > 10$, the HM-KNN model has higher classification accuracy, better generalization ability, and better stability compared with the other five KNN models on this dataset.

Figure 11 shows the distribution of the mean and standard deviation relationship of the classification accuracy of the 6 KNN models on the 6 datasets. From the results in the figure, it is observed that except for the heart disease dataset and the biped robot forward walking dataset, HM-KNN classification is better than the other KNN models on the other four datasets. HM-KNN has the largest mean accurate classification rate but also the largest standard deviation on the heart disease dataset, which, combined with the Figure 7(b), is due to the HM-KNN model classification model has lower classification accuracy when $k < 3$, while there is a significantly higher classification accuracy and better stability when $k \geq 3$.

HM-KNN has lower average classification accuracy than M-KNN and CH-KNN on biped robot forward walking dataset and also has the largest standard deviation. Combined with the analysis in Figure 11, it is known that HM-KNN has the largest and most stable classification accuracy when $k \geq 9$. The accuracy of HM-KNN classification is lower than that of M-KNN and CH-KNN when $1 < k < 9$, which causes the problem of small mean value and large standard deviation of HM-KNN. Combining Figures 7, 8, 10, and 11, it can be seen that HM-KNN has the largest average accurate classification rate and the best stability when $k \geq 10$, while HM-KNN has relatively better generalization ability compared with other KNN models.

3.1.4. Application of HM-KNN for Deep Neural Network Control of Biped Robot Walking Tasks. In this section, the proposed HM-KNN is applied to the walking task of a deep neural network-controlled bipedal robot to improve the robot walking stability.

In a deep neural network-controlled biped robot walking task, when the neural network model controlling the robot is trained, it receives inputs from the robot sensors and outputs actions based on the inputs, but the deep network is sensitive to the sensor inputs, and perturbation changes in the sensor values can lead to abnormal output actions.

To solve the above problem, this paper proposes a simple HM-KNN based control framework, where HM-KNN comes with a small amount of data for discriminating the output actions of the deep neural network. When the robot action is generated by the deep neural network, it is fed into the HM-KNN model. If the action is normal, the action will act directly on the robot to make the robot move. When the action value is abnormal, there are two cases, one is the abnormal action deviates from the normal actions within the
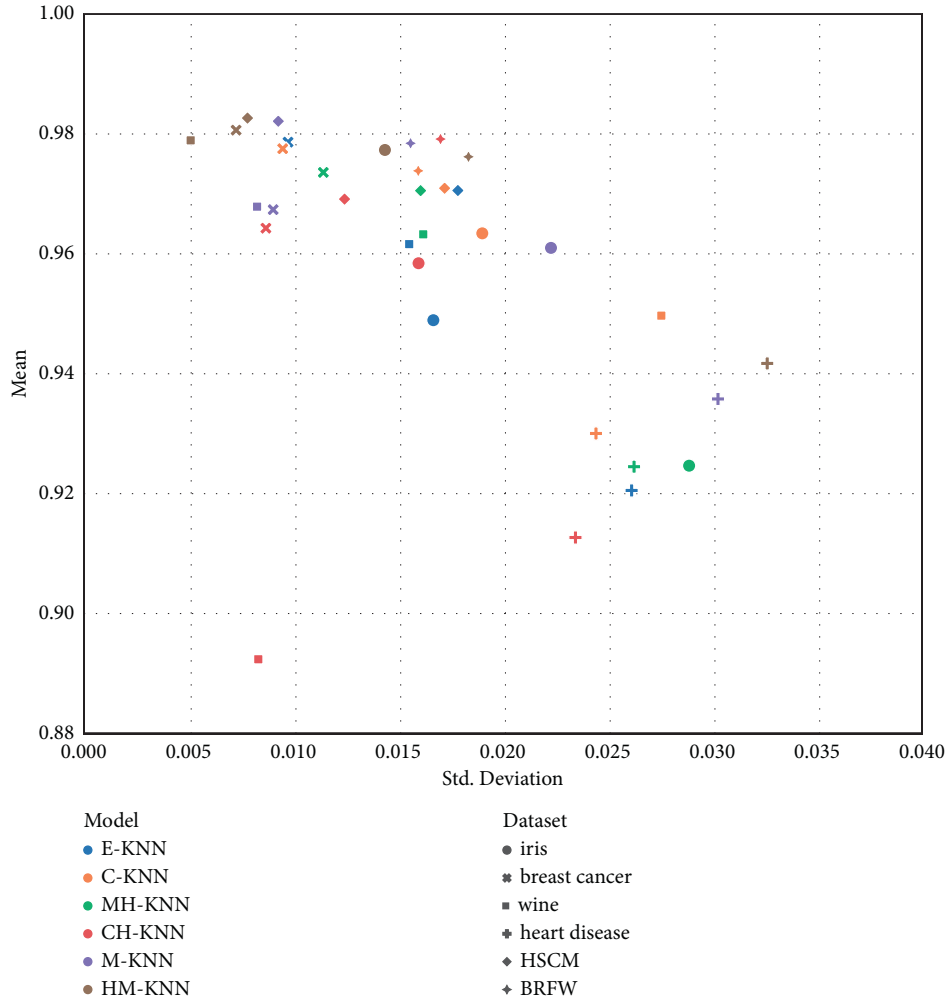
Figure 11: Effectiveness of multi-KNN on multiple datasets.

safety limit, HM-KNN will output the nearest normal action value to the abnormal action, and the second is the abnormal value deviates from the normal value beyond the safety limit, HM-KNN model will output the fault location to the operator, who will carry out the next. The safety limit is the absolute value of the distance of the abnormal value, which is statistically derived according to the different tasks required and is taken here as 0.3. The KM-KNN and the safety limit are used to ensure robot safety and improve the stability of robot walking. The control process is given in Figure 12.

To address the above problems, this paper proposes a simple control framework for robots based on HM-KNN, in which the action values output from the neural network are fed into a model of HM-KNN with a small amount of data, and the gait reconstruction of the bipedal robot is performed based on the output of HM-KNN, to improve the stability of the bipedal robot in the walking process.

Figure 13 shows the classification results of HM-KNN on the biped robot forward walking dataset for different w parameters. When $w = 0$ indicates that the directional information in the hybrid metric does not play a role, and the samples are simply classified by the Mahalanobis distance. In Figure 13, the classification accuracy of the HM-KNN model with $w = 0$ is marked with blue "+," and the classification accuracy of the HM-KNN model with $w = 0.3$ is marked with "o."

The results in Figure 13 show that at $K = 13$, the classification results for the remaining w parameters are significantly better than $w = 0$. When $K = 16$, the HM-KNN model taking $w = 0.4$ achieves the highest classification accuracy and significantly outperforms the HM-KNN model based on the other w values. When $K = 17$, the HM-KNN model taking $w = 0.3$ achieves the highest classification accuracy and significantly outperforms the HM-KNN model based on other w values.

Figure 14 shows the distribution of the mean and standard deviation of the classification accuracy for the HM-KNN model with different parameters at $K \in [1, 19]$. Where Max_f denotes the number of occurrences of the maximum classification accuracy value, which represents the size of the points in the plot. Max_K denotes the maximum K value corresponding to the maximum classification accuracy. Max_f and Max_K represent the generalization ability of the model. Combining the mean value of classification accuracy, the standard deviation of classification accuracy, Max_f and Max_K, $w = 0.3$ is relatively better.
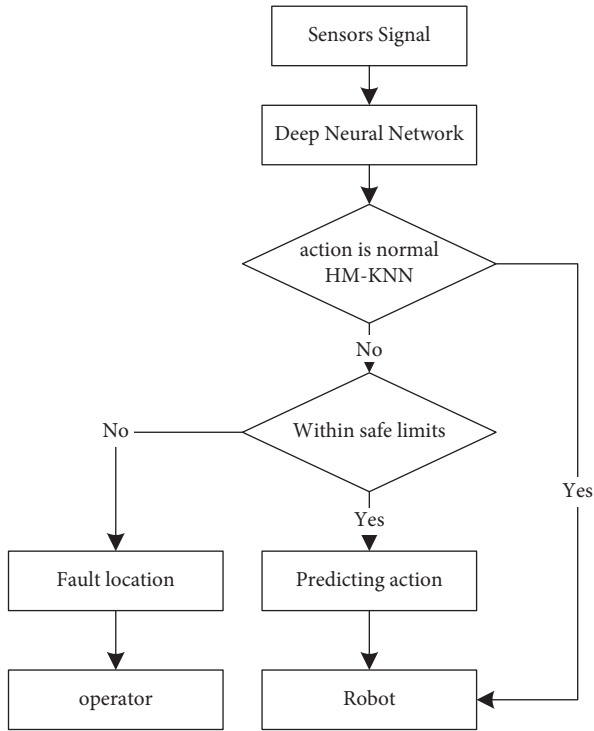
Figure 12: Deep neural network control framework of a biped robot based on HM-KNN.
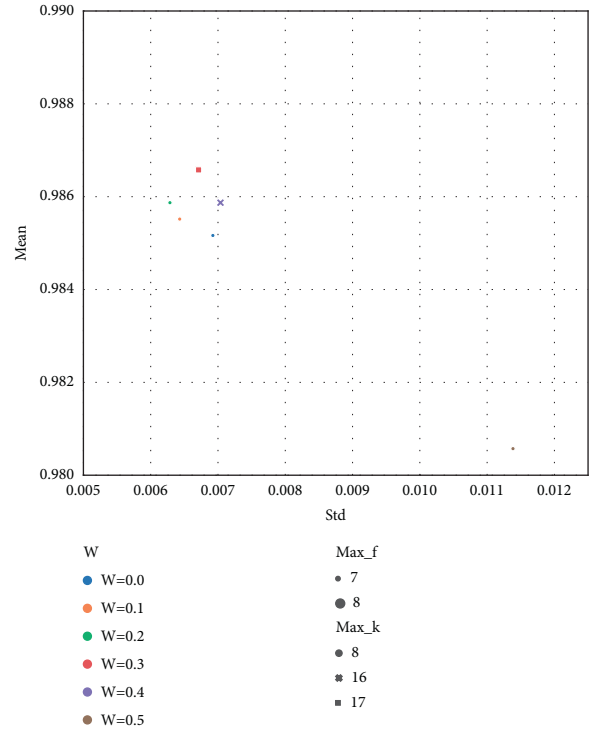

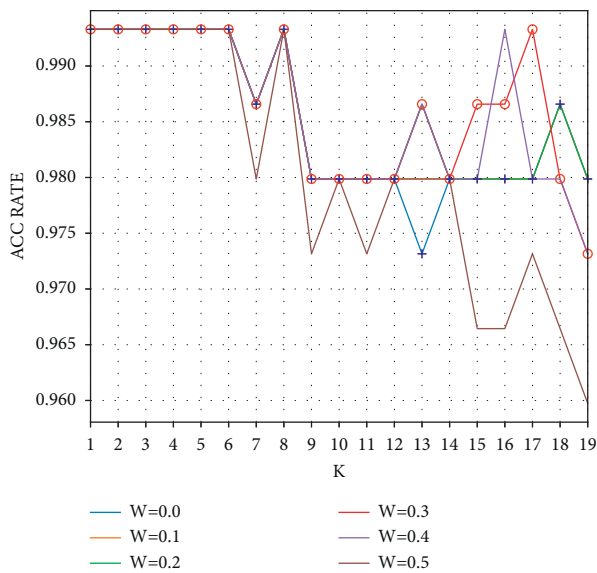Figure 14: Data analysis of HM-KNN classification results based on different parameter w.

Figure 13: Comparison of classification results of HM-KNN with different w parameters.

Combining Figures 13 and 14, it can be seen that the HM-KNN model performs best in the biped robot forward walking dataset when $w = 0.3$.

When abnormal data are generated and input to the value HM-KNN model, the model classifies the abnormal values and outputs the normal values that are closest to the abnormal values. The common discriminant of biped robot

walking stability is the Zero Moment Point (ZMP) [40]. Figure 15 shows the stability of the biped robot with a single leg support phase, the shaded area indicates the stable support area, the coordinate origin indicates the robot's right foot plantar coordinate system, the green point indicates the desired ZMP, i.e., the optimal ZMP for the current step, and the red star indicates the actual ZMP. Figures 15(a)–15(d) show the bipedal robot stability adjustment process after anomalous zmp generation, which is done based on the framework shown in Figure 12. Figure 15(a) shows the zmp distribution when the abnormal motion occurs. Figures 15(b) and 15(c) show the variation process of the actual zmp position. Figure 15(d) shows the zmp position after the adjustment is completed.

The position of the green dot in the support foot co-ordinate system in Figure 15 is [0.0207, 0.0003], and the position of the red star in the support foot coordinate system in Figure 15(a) is [0.0378, −0.0211], which is obviously beyond the stable support area and can lead to unstable robot walking and even damage the robot. After the zmp position adjustment process is shown in Figures 15(b) and 15(c), the position of the red star in the support foot co-ordinate system in Figure 15(d) is [0.0212, −0.0061], and it can be seen that the zmp is close to the ideal zmp by combining Figures 15(a) and 15(d). Although it does not reach the ideal zmp position, the zmp falls within the stable support area, which can ensure that the robot is this moment is stable, and it can be seen from Figure 15 that the robot control framework proposed in Figure 12 is effective in improving the stability of the robot.
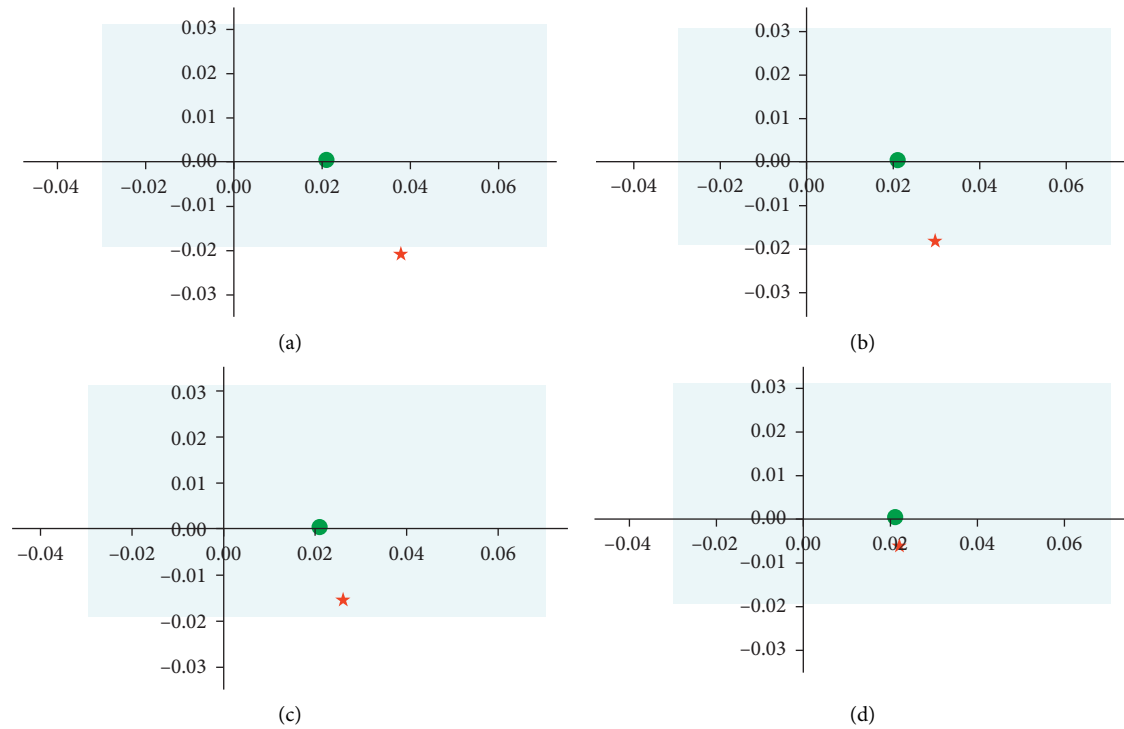
FIGURE 15: Comparison of the distribution of ZMP in the stability support area.

4. Conclusions

In this paper, a hybrid metric-based KNN algorithm is proposed to further improve the performance of the algorithm by further distinguishing equidistant samples in k-nearest neighbors. More specifically, firstly, the hybrid metric method is proposed for the problem that a single metric cannot distinguish equidistant samples. Secondly, a distribution discrimination method for equidistant samples is designed to calculate the hybrid metric between samples and update the k-nearest neighbors more specifically.

In the experimental part, HM-KNN is compared with several KNN models on several data sets and an application example of HM-KNN is given. More specifically, the experiments are divided into two parts. In the first part of the experiment, HM-KNN is compared with the other 5 single metric-based KNN models on 5 UCI public datasets. The results show that HM-KNN has good classification and generalization ability compared to other single-metric-based KNN algorithms. In the second part of the experiment, HM-KNN is compared with the other 5 single-metric-based KNN models on a self-collected bipedal robot forward walking dataset, which includes equidistant samples, such as joint data of the left and right single-legged support phases. The results demonstrate that HM-KNN has good classification and generalization ability on this data and compared to other single metric-based KNN algorithms. A deep neural network based on HM-KNN is designed to control the forward travel controller of the bipedal robot. The effectiveness of the HM-KNN-based deep neural network controller is demonstrated by comparing the ZMP distribution.

Although the HM-KNN algorithm proposed in this paper can further increase the distinguishability of the data compared with the traditional single metric KNN algorithm, there are still shortcomings, which is the focus of the next step. First, the proposed hybrid metric method can further increase the distinguishability of data, but it cannot overcome the distance failure caused by the dimensional catastrophe. Second, the parameter w in the hybrid metric is designed empirically, and this fixed parameter often does not adapt well to different data sets, and how to design an adaptive method for the w parameter in the hybrid metric is a key issue that needs to be studied urgently. Improving the above deficiencies is the focus of future work.

Data Availability

All data included in this study are available upon request by contact with the corresponding author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] B. S. Dhillon, *Robot Reliability and Safety*, Springer, New York, NY, USA, 1991.

[2] J.-H. Shin and J.-J. Lee, "Fault detection and robust fault recovery control for robot manipulators with actuator failures," in *Proceedings of the 1999 IEEE International Conference on Robotics & Automation Detroit*, pp. 861–866, I E E E, Detroit, MI, USA, May 1999.

[3] E. Khalastchi and M. Kalech, "On fault detection and diagnosis in robotic systems," *ACM Computing Surveys*, vol. 51, no. 1, pp. 1–24, 2018.

[4] S. Yin, G. Wang, and H. R. Karimi, "Data-driven design of robust fault detection system for wind turbines," *Mechatronics*, vol. 24, no. 4, pp. 298–306, 2014.

[5] G. Heredia and A. Ollero, "Virtual sensor for failure detection, identification and recovery in the transition phase of a morphing aircraft," *Sensors*, vol. 10, no. 3, pp. 2188–2201, 2010.

[6] R. Davis, "Knowledge-based systems," *Science*, vol. 4741, no. 231, pp. 957–963, 1986.

[7] R. Isermann, "Model-based fault detection and diagnosis - status and applications," *IFAC Proceedings Volumes*, vol. 37, no. 6, pp. 49–60, 2004.

[8] J. Zarei, M. A. Tajeddini, and H. R. Karimi, "Vibration analysis for bearing fault detection and classification using an intelligent filter," *Mechatronics*, vol. 24, no. 2, pp. 151–157, 2014.

[9] S. Yin, S. X. Ding, A. Haghani, H. Hao, and P. Zhang, "A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee eastman process," *Journal of Process Control*, vol. 22, no. 9, pp. 1567–1581, 2012.

[10] F. Jia, Y. Lei, N. Lu, and S. Xing, "Deep normalized convolutional neural network for imbalanced fault classification of machinery and its understanding via visualization," *Mechanical Systems and Signal Processing*, vol. 110, pp. 349–367, 2018.

[11] R. Huang, Y. Liao, S. Zhang, and W. Li, "Deep decoupling convolutional neural network for intelligent compound fault diagnosis," *IEEE Access*, vol. 7, pp. 1848–1858, 2019.

[12] M. F. Yaqub, I. Gondal, and J. Kamruzzaman, "Inchoate fault detection framework: adaptive selection of wavelet nodes and cumulant orders," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 3, pp. 685–695, 2012.

[13] L. Wang, J. Pan, Y. Gao, B. Wang, K. Hong, and X. Chen, "Incipient fault diagnosis of limit switch based on a ARMA model," *Measurement*, vol. 135, pp. 473–480, 2019.

[14] A. Glowacz, "DC motor fault analysis with the use of acoustic signals, coiflet wavelet transform, and K-nearest neighbor classifier," *Archives of Acoustics*, vol. 40, no. 3, pp. 321–327, 2015.

[15] Y. Li and X. Zhang, "Diffusion maps based k-nearest-neighbor rule technique for semiconductor manufacturing process fault detection," *Chemometrics and Intelligent Laboratory Systems*, vol. 136, pp. 47–57, 2014.

[16] Z. Zhe Zhou, C. Chenglin Wen, and C. Chunjie Yang, "Fault detection using random projections and k-nearest neighbor rule for semiconductor manufacturing processes," *IEEE Transactions on Semiconductor Manufacturing*, vol. 28, no. 1, pp. 70–79, 2015.

[17] J. Yang, Z. Sun, and Y. Chen, "Fault detection using the clustering-kNN rule for gas sensor arrays," *Sensors*, vol. 16, no. 12, p. 2069, 2016.

[18] J. Xiong, Q. Zhang, G. Sun, X. Zhu, M. Liu, and Z. Li, "An information fusion fault diagnosis method based on dimensionless indicators with static discounting factor and KNN," *IEEE Sensors Journal*, vol. 16, no. 7, pp. 2060–2069, 2016.

[19] J. Tian, C. Morillo, M. H. Azarian, and M. Pecht, "Motor bearing fault detection using spectral kurtosis-based feature extraction coupled with K-nearest neighbor distance analysis," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 3, pp. 1793–1803, 2016.

[20] A. Glowacz, "Fault diagnosis of single-phase induction motor based on acoustic signals," *Mechanical Systems and Signal Processing*, vol. 117, pp. 65–80, 2019.

[21] F. Bulut and M. F. Amasyali, "Locally adaptive k parameter selection for nearest neighbor classifier: one nearest cluster," *Pattern Analysis & Applications*, vol. 20, no. 2, pp. 415–425, 2017.

[22] N. Garcia-Pedrajas, J. A. Romero del Castillo, and G. Cerruela-Garcia, "A plvnnr," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 2, pp. 470–475, 2017.

[23] S. Zhang, X. Li, M. Zong, X. Zhu, and D. Cheng, "Learning k for kNN classification," *ACM Transactions on Intelligent Systems and Technology*, vol. 8, no. 3, pp. 1–19, 2017.

[24] S. Zhang, X. Li, M. Zong, X. Zhu, and R. Wang, "Efficient kNN classification with different numbers of nearest neighbors," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 5, pp. 1774–1785, 2018.

[25] X.-F. Zhong, S.-Z. Guo, L. Gao, H. Shan, and J.-H. Zheng, "An improved k-NN classification with dynamic k," in *Proceedings of the 9th International Conference on Machine Learning and Computing*, pp. 211–216, ACM, New York, NY, USA, February 2017.

[26] M. Bicego and M. Loog, "Weighted k-nearest neighbor revisited," in *Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 1642–1647, IEEE, Cancun, Mexico, December 2016.

[27] C. Bo, H. Lu, and D. Wang, "Weighted generalized nearest neighbor for hyperspectral image classification," *IEEE Access*, vol. 5, pp. 1496–1509, 2017.

[28] S. A. Dudani, "The distance-weighted k-nearest-neighbor rule," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-6, no. 4, pp. 325–327, 1976.

[29] J. Gou, "A new distance-weighted k -nearest neighbor classifier," *Journal of Information and Computational Science*, vol. 9, no. 6, pp. 1429–1436, 2012.

[30] J. Gou, T. Xiong, and Y. Kuang, "A novel weighted voting for K-nearest neighbor rule," *Journal of Computers*, vol. 6, no. 5, pp. 833–840, 2011.

[31] P. Li, J. Gou, and H. Yang, "The distance-weighted K-nearest centroid neighbor classification," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 8, no. 3, pp. 611–622, 2017.

[32] A. Vidyarthi and N. Mittal, "AVNM: avnmric," *Computer Methods and Programs in Biomedicine*, vol. 137, pp. 195–201, 2016.

[33] B. C. Ooi, *Spatial Kd-Tree: A Data Structure for Geographic Database*, pp. 247–258, Springer, Berlin, Germany, 1987.

[34] B. Leibe, K. Mikolajczyk, and B. Schiele, "Efficient clustering and matching for object class recognition," in *Proceedings of theBMVC 2006 - Proceedings of the British Machine Vision Conference 2006*, pp. 789–798, Edinburgh, Scotland, September 2006.

[35] S. Zhao and C. Jian, "Fast KNN classification algorithm based on three- way clustering," *Journal of Chinese Mini-Micro Computer Systems*, vol. 42, no. 9, pp. 1845–1851, 2021.

[36] R. Ao, K. Lingjun, L. Zhen, and W. Qian, "Spectral reconstruction training sample selection based on weighted euclidean distance," *Packing Engineering*, vol. 41, no. 15, pp. 253–259, 2020.

[37] G. Verdier and A. Ferreira, "Adaptive mdnnrfdsm," *IEEE Transactions on Semiconductor Manufacturing*, vol. 24, no. 1, pp. 59–68, 2011.

[38] J. Gou, H. Ma, W. Ou, S. Zeng, Y. Rao, and H. Yang, "A generalized mean distance-based k-nearest neighbor classifier," *Expert Systems with Applications*, vol. 115, pp. 356–372, 2019.

[39] N. Helwig, E. Pignanelli, and A. Schütze, "Condition monitoring of a complex hydraulic system using multivariate statistics," in *Proceedings of the 2015 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings*, pp. 210–215, Pisa, Italy, May 2015.

[40] M. Vukobratović and J. Stepanenko, "On the stability of anthropomorphic systems," *Mathematical Biosciences*, vol. 15, no. 1–2, pp. 1–37, 1972.