Quantum Water Strider Algorithm with Hybrid-Deep-Learning-Based Activity Recognition for Human–Computer Interaction

Areej A. Malibari 1, Jaber S. Alzahrani 2, Ayman Qahmash 3, Mohammed Maray 3, Mohammed Alghamdi 3,4, Reem Alshahrani 5, Abdullah Mohamed 6 and Anwer Mustafa Hilal 7,*

1 Department of Computer Science, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia; aamalibari1@kau.edu.sa
2 Department of Industrial Engineering, College of Engineering at Alqunfudah, Umm Al-Qura University, Mecca 24382, Saudi Arabia; Jzhrani@aqu.edu.sa
3 Department of Information Systems, College of Computer Science, King Khalid University, Abha 62529, Saudi Arabia; a.qahmash@kku.edu.sa (A.Q.); mmarey@kku.edu.sa (M.M.); mhalghamdi@uj.edu.sa (M.A.)
4 Department of Information and technology Systems, College of Computer Science and Engineering, University of Jeddah, Jeddah 21959, Saudi Arabia
5 Department of Computer Science, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia; rashahrani@tu.edu.sa
6 Research Centre, Future University in Egypt, New Cairo 11845, Egypt; Mohamed.a@fue.edu.eg
7 Department of Computer and Self Development, Preparatory Year Deanship, Prince Sattam Bin Abdulaziz University, Al-Kharj 16278, Saudi Arabia
*
Correspondence: a.hilal@psau.edu.sa

Featured Application: This work is based on automatic human activity recognition, which has drawn much attention in video analysis technology due to the growing demands from many applications, such as surveillance environments, entertainment environments, and healthcare systems.

Abstract: Human action and activity recognition are clues that alleviate human behavior analysis. Human action recognition (HAR) becomes a significant challenge in various applications involving human computer interaction (HCI) and intellectual video surveillance for enhancing security in distinct fields. Precise action recognition is highly challenging because of the variations in clutter, backgrounds, and viewpoint. The evaluation method depends on the proper extraction and learning of data. The achievement of deep learning (DL) models results in effective performance in several image-related tasks. In this view, this paper presents a new quantum water strider algorithm with hybrid-deep-learning-based activity recognition (QWSA-HDLAR) model for HCI. The proposed QWSA-HDLAR technique mainly aims to recognize the different types of activities. To recognize activities, the QWSA-HDLAR model employs a deep-transfer-learning-based, neural-architectural-search-network (NASNet)-based feature extractor to generate feature vectors. In addition, the presented QWSA-HDLAR model exploits a QWSA-based hyperparameter tuning process to choose the hyperparameter values of the NASNet model optimally. Finally, the classification of human activities is carried out by the use of a hybrid convolutional neural network with a bidirectional recurrent neural network (HCNN-BiRNN) model. The experimental validation of the QWSA-HDLAR model is tested using two datasets, namely KTH and UCF Sports datasets. The experimental values reported the supremacy of the QWSA-HDLAR model over recent DL approaches.

Keywords: human–computer interaction; activity recognition; deep learning; hyperparameter tuning; NASNet model
1. Introduction

In the digital era, human beings require an advanced level of computer intelligence [1]. Human–computer interaction (HCI) is not restricted to the original hardware-related interaction. Certain smarter interaction techniques have been slowly occurring in the lives of people, namely a sequence of highly intellectual techniques based on voice recognition, face recognition, and gesture recognition [2]. Intellectual mechanisms will help establish interaction between computers and humans. The origin of such highly suitable interaction approaches has become a major advancement trend in the present HCI domain. The main objective of HCI advancement is to make effective computers that adapt to and serve the requirements of humans [3]—people-centered rather than forcing individuals to adapt to the computer. Gaining HCI data allows very efficient learning and creation of smarter systems [4]. Machine learning (ML) is a significant branch of artificial intelligence (AI). It performs well in several domains and illustrates powerful research and development (R&D) potential.

With the advent of ML technology in HCI, machines have become very intelligent. Amongst authors in both the industry and academia whose aim was the development of ubiquitous computing, the most broadly discussed research concept regarding HCI has been human activity recognition (HAR) [5,6]. Recently, the amount of research on HAR has increased rapidly due to the extensive availability of sensors, improvements in power utilization, and reduction in costs, and resulting in technological advances in ML approaches. The Internet of Things (IoT) and AI can now be live streamed [7]. The growth in HAR has alleviated practical implementations in several real-world domains, which include the medical sector, tactical military applications, the detection of crime and violence, and sports science [8]. It was apparent that the extensive range of conditions to which HAR can be applicable presents evidence that the domain consists of powerful capabilities for enhancing living standards [9].

Mathematical methods related to human activity data enable the recognition of a diversity of human activities—for instance, walking, running, sitting, standing, and sleeping. HAR mechanisms are of two major groups: sensor-related systems and video-related systems. Time-series-classifying tasks were major difficulties while utilizing HAR, i.e., whenever the movements of individuals were forecasted with the help of sensory data [10]. These tasks usually include accurately extracting features from raw data using signal processing approaches and deep-field expertise to fit one of the ML models. In recent research, the capability of deep learning (DL) techniques, which include long short-term memory (LSTM) neural networks and convolutional neural networks (CNN), to automatically extract meaningful attributes from the provided raw sensor data and attain the most advanced outcomes has been achieved [11,12].

This paper presents a new quantum water strider algorithm with hybrid-deep-learning-based activity recognition (QWSA-HDLAR) model for HCI. The proposed QWSA-HDLAR technique employs a deep-transfer-learning-based, neural-architectural-search-network (NASNet)-based feature extractor to generate feature vectors. In addition, the presented QWSA-HDLAR model exploits the QWSA-based hyperparameter tuning process of the NASNet model. Finally, the classification of human activities is carried out by the use of a hybrid convolutional neural network with a bidirectional recurrent neural network (HCNN-BiRNN) model. The experimental validation of the QWSA-HDLAR model is tested using two datasets, namely KTH and UCF Sports datasets. In short, the major contributions are listed as follows:

- An automated QWSA-HDLAR technique encompassing NASNet-based feature extraction, QWSA-based hyperparameter tuning, and HCNN-BiRNN-based classification is presented for the identification and classification of human activities on HCI. To the best of our knowledge, the presented QWSA-HDLAR technique does not exist in the literature.
• Employ QWSA-based NASNet model to extract feature vectors, where the QWSA helps in accomplishing enhanced classification results due to the hyperparameter tuning process.
• Validate the performance of the QWSA-HDLAR technique using two datasets, such as KTH and UCF Sports datasets.

2. Related Works

In [13], a novel technique was devised in this study for action recognition. The presented technique was related to the DL features fusion and shape. A two-step-based approach can be performed—human extraction to action recognition. At the initial step, human beings were derived through a simple learning process. During this process, HOG features can be derived from some chosen datasets. After choosing the powerful features with the use of entropy-controlled features, linear support vector machine (LSVM) maximization and detection were executed. Secondly, geometric features were derived from detected areas, and parallel DL features were derived from original video frames. The gained feature vector can be classified through a cubic multiclass SVM. Jaoued et al. [14] recommend a new technique for HAR depending upon a hybrid DL method. The devised method can be assessed on the challenging UCF101, KTH, and UCF Sports datasets.

Zheng et al. [15] examine the segmentation techniques’ impact on DL method performance and compare four data transformation methods. The multichannel technique included three overlapped color channels, generated optimum performance. Additionally, the multichannel method can be implemented for three public datasets and generated satisfying outcomes for multisource acceleration data. Tanberk et al. [16] devise a hybrid deep method for understanding and interpreting videos, aiming at HCR. The devised architecture was built by compiling dense auxiliary movement information and optical flow approach in video datasets with the help of DL approaches. As we know, it was the first research related to a new compilation of LSTM fed by auxiliary data and 3D-CNN fed by optical flow on video frames for HAR.

Abdulazeem et al. [17] devise a structure with three main stages for HCR. The stages are recognition, pretraining, and preprocessing. This structure provides a set of new methods which were three-fold as follows: the first is during the pretraining stage—a standard CNN can be well trained on a generic dataset for adjusting weights; another is performing the recognition procedure—this pretrained method can be then applied to the target dataset; and finally, the recognition stage exploits CNN and LSTM to apply five distinct architectures. Ronald et al. [18] devise iSPLInception, a DL method motivated by the Inception-ResNet architecture from Google, which not only attains high prediction accuracy but utilizes some device sources. The researchers in [19] devise the late fusion of a HAR classifier and visual recognition. Vision can be utilized to recognize the several screws collected in a mock part, while HAR from body-worn inertial measurement units (IMUs) categorizes actions performed while assembling the parts. CNN techniques were utilized in both modes of classifier while or before several late fusion approaches were examined for estimation of a concluding state estimate.

3. The Proposed Model

In this paper, a novel QWSA-HDLAR model was developed for the recognition of human activities in the HCI environment. The proposed QWSA-HDLAR technique initially applies a NASNet model to derive a collection of feature vectors. Additionally, the presented QWSA-HDLAR model utilizes a QWSA-based hyperparameter tuning process to choose the hyperparameter values for the NASNet model optimally. Finally, the classification of human activities is carried out using the CNN-BiRNN model.

3.1. Feature Extraction: NASNet Model

Primarily, the proposed QWSA-HDLAR technique exploits the NASNet model to derive a collection of feature vectors. This is one of the very influential and famous
technologies for handling smaller datasets through a pretrained network. A pretrained network was trained on massive datasets, generally in the tasks of classifying images; later, the architecture and weight were retained. If these primary datasets are relatively large and sufficiently widespread, then the feature subset that the pretrained network has learned could be helpful as a visual model. Thus, the feature might assist various computer vision tasks, although the new task can include completely different classifications from the primary task [20]. The transfer of learning from a pretrained network is exploited in two ways: fine-tuning and feature extraction. Feature extraction includes the convolution base of the pretrained network for extracting the novel data set feature and later training a novel classification on top of the output.

The fine-tuning corresponds to the feature extraction model, which includes unfreezing the final layer of the frozen convolution base employed for extracting features. Then, the unfrozen layer is later retrained alongside the novel classification formerly learned in the feature extraction model. The fine-tuning model aims to alter the pretrained model’s most abstract features to make them more pertinent to the novel task. The following steps are involved in this study:

- A pretrained NASNet is considered, and the classification base is detached.
- The convolution base of pretrained models is frozen.
- A new CNN-BiGRU classification is added and trained on top of the convolution base of pretrained networks.
- C layers of the convolution base of pretrained networks are unfrozen.
- Finally, the unfrozen layer and novel classifiers are trained together.

Equipped with engineering expertise and a large amount of computation power, Google launched NASNet [21] and devised the problem of searching for an optimal CNN model as a reinforcement learning (RL) problem. The RL model is a type of ML approach which allows an agent to discern the best action in virtual environments to attain the goal with feedback from its own experiences and actions. Furthermore, the concept was to search the optimal grouping of parameters of the provided number of layers, searching filter size, strides, output channel, etc. In the RL settings, the reward after every search action was the accuracy of the searched model on the provided datasets. In NASNet, using the general framework is predetermined; the cells or blocks are not predetermined by researchers. Rather, they searched through the RL search technique. The structure of the NASNet model is shown in Figure 1.

![Figure 1. NASNet architecture.](image-url)

Furthermore, the number of early convolution filters and motif repetitions \( N \) are free parameters and utilized for scaling. In particular, they are named reduction and normal cells. A reduction cell is a convolution cell that returns a feature map from which the feature
map’s width and height are minimized by a factor of 2, and a normal cell is a convolution cell that returns a feature map of similar dimensions. NASNet achieved advanced results in the ImageNet competition, but the computational power required for NASNet was bigger than what a small company only capable of using common methodologies could provide.

3.2. Hyperparameter Tuning: QWSA Model

In this study, the QWSA-based hyperparameter tuning process optimally chooses the hyperparameter values of the NASNet model. The WSA has good performance in a majority of the problems; occasionally, it has a chance of becoming trapped in the local optima and early convergence [22]. Here, the concept of quantum computing was taken into account. In quantum space, the location of the male and female WSs is not concurrently defined; therefore, the location of WS must be determined using the wave function $\psi(X, t)$, where $X$ determines the location of WSs. This implies that the square of mode indicates the likelihood density of WS appearing at the $X$-location in space, and it can be expressed as follows [23]:

$$|\psi|^2 dxdydz = Q dxdydz$$ \hspace{1cm} (1)

In Equation (1), $Q$ defines a likelihood density function which fulfills the normalized condition:

$$\int_{-\infty}^{\infty} |\psi|^2 dxdydz = \int_{-\infty}^{\infty} Q dxdydz = 1$$ \hspace{1cm} (2)

The location of WS is accomplished using the Monte Carlo method, and its updated formula is provided as follows.

$$P(t) = \delta \times P_b(t) + (1 - \delta) \times P_g(t)$$ \hspace{1cm} (3)

$$m_b(t) = \frac{1}{M} \sum_{i=1}^{M} P_b_i(t)$$ \hspace{1cm} (4)

$$D(t + 1) = 2\beta \times |m_b(t) - X(t)|$$ \hspace{1cm} (5)

$$\beta = \beta - \left( \beta - \beta_0 \right) \times \left( \frac{t}{G_{\text{max}}} \right)$$ \hspace{1cm} (6)

$$X(t) = P(t) \pm (D/2) \times \ln \left( \frac{1}{\gamma} \right)$$ \hspace{1cm} (7)

From the expression, $M$ defines the population size; $\delta$ and $\gamma$ signify that an arbitrary number lies within $[0, 1]$; $P(t)$ determines the local attraction area of the $t$-th iterations of WSs, which define the location of every WSs is an arbitrary location among the global and the individual locations; $D$ signifies the weighted distance between the candidate and the mean optimum location of population; $m(t)$ defines the mean value of individual optimum location for the WSs; $G_{\text{max}}$ determines the iteration count; $\beta$ denotes the shrinkage–expansion coefficient, i.e., exploited to handle the individual convergence rate lies within $\beta_0$ to $\beta$. In many instances, $[\frac{\beta_0}{\beta}, \beta] = [l, 0.5]$, and $P_b$ and $P_g$ define the candidate and the $g$, respectively. The flowchart of WSA is shown in Figure 2.

The updating is implemented in the mating phase of original WSA:

$$\begin{cases} 
\chi^*_i(t) = P(t) \pm \beta \times |m_b(t) - x_i(t)| \times \ln \left( \frac{1}{\gamma} \right) & \text{if mating happens} \\
\chi^*_i(t) = P(t) \pm \beta \times |m_b(t) - x_i(t)| \times \ln \left( \frac{1}{\gamma} \right) & \text{or else}
\end{cases}$$ \hspace{1cm} (8)
In these conditions, if $X_i^*$ has good outcomes when compared to $X_i(t)$, $X_i(t+1) = X_i^*(t)$; otherwise, $X_i(t+1) = X_i(t)$ as explained in Algorithm 1.
Algorithm 1: Pseudocode of WSA

Inputs: The population size news, number of territories nt, and the maximal number of iterations MaxCycle

Outputs: The richest position of WS and the objective value

Initialize the population randomly
Evaluate the fitness value of WSs

while (ending criteria is not satisfied) do
    Establish nt number of territories and assign the WSs
    For (each territory) do
        The male keystone sends mating ripples, and the designated female decides about the response that is repulsive or attractive signals.
        Upgrade the location of the keystone based on the response of female
        Calculate the novel location for finding food to compensate for the consumed energy at the time of mating
        if (keystone cannot find food) then
            Forage for food resources and approach the food-rich territory
            if (keystone couldn’t find food again) then
                The hungry keystone would be died due to starvation or be killed by the resident keystone of the new territory.
                A larva that is matured replaces the killed keystone as the successor determined
            End if
        End if
    End for

Return WS_optimal

The QWSA method extracts a fitness function for achieving enhanced classifier performance. It fixes a positive numeral for indicating the superior outcome of a candidate solution. In this article, a classifier error rate reduction was regarded as a fitness function, as provided in Equation (9). The optimum resolution contains a minimal error rate, and the poor resolution achieves a higher error rate.

\[
fitness(x_i) = \text{ClassifierErrorRate}(x_i) = \frac{\text{number of misclassified samples}}{\text{Total number of samples}} \times 100 \quad (9)
\]

3.3. Activity Recognition: HCNN-BiRNN Model

At the final stage, the classification of human activities is carried out by the use of CNN-BiRNN model. The CNN-BiRNN hybrid models contain two major mechanisms: a BiRNN with attention model [24] on the top half and a CNN with L convolution layers on the bottom half. These components of the models are jointly trained in an end-to-end manner. One sample \( X \in \mathbb{R}^{1 \times W \times 1} \) is a real value vector (\( W \) refers to the range dimension length); therefore, a 1D-CNN module is applied, from which the convolutional process occurs in the range dimension. For the initial layer, convolutional operations using stride length 1 employ \( K_1 \) filters, \( W(1) \in \mathbb{R}^{1 \times w(1) \times 1} \), to \( X \), resulting in feature maps of layer 1, \( H(1) \in \mathbb{R}^{1 \times W^{(1)} \times K_1} \). For the next \( L - 1 \) convolutional layers, convolution-pooling operations repetitively employ \( K_l \) filters, \( W(l) \in \mathbb{R}^{1 \times w(l) \times K_{l-1}} \), to \( H(l-1) \), and feature map \( H(L) \in \mathbb{R}^{1 \times W^{(L)} \times K_L} \) is obtained after the \( L \) convolution layer.

Every convolution layer is followed by a pooling layer; therefore, the time dimension is shortened, and the temporal dependency increases as the convolution layer grows. After dropping the single dimension, \( H^{(L)} \in \mathbb{R}^{W^{(L)} \times K_L} \) is regarded as a series of length \( W^{(L)} \) with \( K_L \) feature vector at every time step. Next, we replace the \( W^{(L)} \) time step, using \( T \) for notation convenience. The forward recurrent neural network (RNN) reads \( H^{(L)} \) in its novel order and produces hidden state \( f_{ht} \) at all the time steps, and the backward RNN reads \( H^{(L)} \) in its reverse order and generates \( bh_{ht} \) as follows

\[
f_{ht} = f \left( W_{fzh} \cdot H^{(L)}_{ht} + W_{fhh} \cdot f_{ht-1} \right), \quad t = 1 \ldots T \quad (10)
\]
\[ bh_t = f \left( W_{bxh} \cdot H_t^{(L)} + W_{bh} \cdot bh_{t-1} \right), \quad t = T \ldots 1 \]  \hfill (11)

From the expression, \( W_{fxh} \in \mathbb{R}^{m \times KL} \) and \( W_{bxh} \in \mathbb{R}^{m \times KL} \) refer to the input-hidden weights, \( W_{fh} \in \mathbb{R}^{m \times m} \) and \( W_{bh} \in \mathbb{R}^{m \times m} \) denotes the weight that connects the hidden layer, \( m \) denotes the dimensionality of the hidden state, and \( f \) indicates the sigmoid function. Next, the concatenation of \( fh_t \) and \( bh_t \) forward and backward states create \( h_t = [fh_t, bh_t] \).

Consequently, every \( h_t \) hidden state comprises data of the entire target, with stronger emphasis on the part near the region at \( t \) step. In BiRNN, the data at all the time steps might slowly be lost alongside the backward and forward propagation.

To prevent data loss and focus on the discriminatory time step automatically, the attention module is proposed, which, as a byproduct, is capable of relaxing the misalignment problem. In this technique, we adapt a multilayer perceptron for calculating the attention weight depending on the hidden state and indicate \( g \) as an invariant feature vector, i.e.,

\[ g = \sum_{t=1}^{T} \alpha_t \cdot h_t \]  \hfill (12)

The weight, \( \alpha_t \), is calculated using the below equation

\[ \alpha_t = \frac{\exp(e_t)}{\sum_l \exp(e_l)} \]  \hfill (13)
\[
e_t = U_a^T \tanh(W_a \cdot h_t) \]  \hfill (14)

Let \( U_a \in \mathbb{R}^{1 \times n}, W_a \in \mathbb{R}^{n \times 2m} \) be the parameter of the attention module, and weight \( \alpha_t \) stands for the coefficient which scores the matching degree among the recognition task and \( h_t \) hidden state. The invariant feature vector \( g \) incorporates the data at each time step based on the discrimination of the hidden state. Assuming the invariant feature vector \( g \), we adopted the \( \text{softmax} \) function to predict the label vector of \( X \) input sample, as follows

\[ p(c|g; \theta) = \frac{\exp(\theta(c)^T g)}{\sum \exp(\theta(j)^T g)} \]  \hfill (15)

In Equation (15), \( C \) denotes the class count, \( p(c|g; \theta) \) represents the probability of \( g \) belongs to \( c - \text{th} \) class, and \( \theta \) indicates the variable of softmax classification.

4. Performance Validation

The experimental result analysis of the QWSA-HDLAR model is tested using two datasets, namely the KTH dataset \cite{25} and the UCF Sports dataset \cite{26}. The first KTH dataset includes 600 samples with six class labels, as given in Table 1. Next, the UCF Sports dataset contains 1000 samples with ten class labels, as provided in Table 2.

Table 1. Details of KTH dataset.

| KTH Dataset |
|-------------|
| **Class No.** | **Description** | **No. of Samples** |
| 1 | Boxing | 100 |
| 2 | Handclapping | 100 |
| 3 | Handwaving | 100 |
| 4 | Jogging | 100 |
| 5 | Running | 100 |
| 6 | Walking | 100 |
| **Total** | | **600** |
Table 2. Details of UCF Sport dataset.

| Class No. | Description         | No. of Samples |
|-----------|---------------------|----------------|
| 1         | Diving-Side         | 100            |
| 2         | Golf-Swing          | 100            |
| 3         | Kicking-Front       | 100            |
| 4         | Lifting             | 100            |
| 5         | Riding Horse        | 100            |
| 6         | Run-Side            | 100            |
| 7         | StateBoarding-Front | 100            |
| 8         | Swing-Bench         | 100            |
| 9         | Swing-SideAngle     | 100            |
| 10        | Walk-Front          | 100            |

Total Number of Samples 1000

Figure 1 demonstrates the confusion matrices produced by the QWSA-HDLAR model. With the entire dataset, the QWSA-HDLAR model recognized 99 samples under class 1, 97 samples under class 2, 95 samples under class 3, 99 samples under class 4, 98 samples under class 5, and 100 samples under class 6. Similarly, with 70% of the training (TR) dataset, the QWSA-HDLAR model identified 74 samples under class 1, 74 samples under class 2, 63 samples under class 3, 67 samples under class 4, 69 samples under class 5, and 63 samples under class 6.

Table 3 illustrates the overall HAR outcomes of the QWSA-HDLAR model on the test KTH dataset. The experimental output demonstrated that the QWSA-HDLAR model has shown enhanced performance on all datasets. For instance, on the entire dataset, the QWSA-HDLAR model has obtained average accuracy of 99.33%, recall of 98%, specificity of 99.60%, F-score of 97.99%, and an area under the receiver operating characteristic curve (AUROC) score of 98.80%. Eventually, with 70% of the TR dataset, the QWSA-HDLAR model attained average accuracy of 99.21%, recall of 97.63%, specificity of 99.52%, F-score of 97.62%, and AUROC score of 98.58%. Meanwhile, on 30% of the TS dataset, the QWSA-HDLAR model reached average accuracy of 99.63%, recall of 98.80%, specificity of 99.78%, F-score of 98.83%, and an AUROC score of 99.29% as illustrated in Figure 3.

Table 3. HAR results of QWSA-HDLAR model on KTH dataset.

| Class Labels | Accuracy | Recall | Specificity | F-Score | AUROC Score |
|--------------|----------|--------|-------------|---------|-------------|
| Entire Dataset |          |        |             |         |             |
| 1            | 99.67    | 99.00  | 99.80       | 99.00   | 99.40       |
| 2            | 99.00    | 97.00  | 99.40       | 97.00   | 98.20       |
| 3            | 98.83    | 95.00  | 99.60       | 96.45   | 97.30       |
| 4            | 99.33    | 99.00  | 99.40       | 98.02   | 99.20       |
| 5            | 99.33    | 98.00  | 99.60       | 98.00   | 98.80       |
| 6            | 99.83    | 100.00 | 99.80       | 99.50   | 99.90       |
| Average      | 99.33    | 98.00  | 99.60       | 97.99   | 98.80       |
| Training Phase (70%) |          |        |             |         |             |
| 1            | 99.52    | 98.67  | 99.71       | 98.67   | 99.19       |
| 2            | 98.81    | 97.37  | 99.13       | 96.73   | 98.25       |
| 3            | 98.57    | 94.03  | 99.43       | 95.45   | 96.73       |
| 4            | 99.05    | 98.53  | 99.15       | 97.10   | 98.84       |
| 5            | 99.52    | 97.18  | 100.00      | 98.57   | 98.59       |
| 6            | 99.76    | 100.00 | 99.72       | 99.21   | 99.86       |
| Average      | 99.21    | 97.63  | 99.52       | 97.62   | 98.58       |
Table 3. Cont.

| Class Labels | Accuracy | Recall | Specificity | F-Score | AUROC Score |
|--------------|----------|--------|-------------|---------|-------------|
| Entire Dataset |          |        |             |         |             |
| 1            | 100.00   | 100.00 | 100.00      | 100.00  | 100.00      |
| 2            | 99.44    | 95.83  | 100.00      | 97.87   | 97.92       |
| 3            | 99.44    | 96.97  | 100.00      | 98.46   | 98.48       |
| 4            | 100.00   | 100.00 | 100.00      | 100.00  | 100.00      |
| 5            | 98.89    | 100.00 | 98.68       | 96.67   | 99.34       |
| 6            | 100.00   | 100.00 | 100.00      | 100.00  | 100.00      |

| Average      | 99.63    | 98.80  | 99.78       | 98.83   | 99.29       |

Figure 3. Confusion matrix of QWSA-HDLAR model on KTH dataset.
The training and validation accuracies depicted by the QWSA-HDLAR technique with distinct epochs on the KTH dataset are demonstrated in Figure 4. The results assured that the accuracies are found to be higher with increased epochs. Additionally, the training accuracy seems to be superior to testing accuracy.

![Figure 4](image-url)

**Figure 4.** Training and validation accuracies of QWSA-HDLAR model on KTH dataset.

The training and validation losses gained by the QWSA-HDLAR method on the test KTH dataset are reported in Figure 5. The figure identified that the QWSA-HDLAR technique has resulted in lower training and validation loss values.

In order to report the enhanced performance of the QWSA-HDLAR model, a comparison study with existing methods [25–27] is performed on the KTH dataset in Table 4. The results implied that the Gaussian mixture model with Kalman filter (GMM-KF) model has shown poor performance with a lower accuracy of 90.47%, whereas the gated recurrent neural network (GRNN) model has attained a slightly enhanced accuracy of 85.85%. This is followed by the support vector machine with 3DCNN (SVM-3DCNN) and the CNN with convolutional autoencoder (CNN-CAE) models, which obtained improved accuracy values of 90.45% and 92.80%, respectively. Though the GMM-KFGRNN and SDL-HBC models resulted in reasonable accuracy values of 95.52% and 99.38%, the QWSA-HDLAR model gained a higher accuracy of 99.63%.

**Table 4.** Comparative accuracy analysis of QWSA-HDLAR model on KTH dataset.

| Methods            | Accuracy (%) |
|--------------------|--------------|
| QWSA-HDLAR         | 99.63        |
| GMM-KF             | 70.47        |
| GRNN               | 85.85        |
| GMM-KFGGRNN        | 95.52        |
| SDL-HBC model      | 99.38        |
| SVM-3DCNN          | 90.45        |
| CNN-CAE            | 92.80        |
In order to report the enhanced performance of the QWSA-HDLAR model, a comparison study with existing methods [25–27] is performed on the KTH dataset in Table 4. The results implied that the Gaussian mixture model with Kalman filter (GMM-KF) model has shown poor performance with a lower accuracy of 90.47%, whereas the gated recurrent neural network (GRNN) model has attained a slightly enhanced accuracy of 85.85%. This is followed by the support vector machine with 3DCNN (SVM-3DCNN) and the CNN with convolutional autoencoder (CNN-CAE) models, which obtained improved accuracy values of 90.45% and 92.80%, respectively. Though the GMM-KFGRNN and SDL-HBC models resulted in reasonable accuracy values of 95.52% and 99.38%, the QWSA-HDLAR model gained a higher accuracy of 99.63%.

Table 4. Comparative accuracy analysis of QWSA-HDLAR model on KTH dataset.

| Methods          | Accuracy (%) |
|------------------|--------------|
| QWSA-HDLAR       | 99.63        |
| GMM-KF           | 90.47        |
| GRNN             | 85.85        |
| GMM-KFGRNN       | 95.52        |
| SDL-HBC model    | 99.38        |
| SVM-3DCNN        | 90.45        |
| CNN-CAE          | 92.80        |

Figure 6 establishes the confusion matrices produced by the QWSA-HDLAR approach on the UCF Sports dataset. The figure implied that the QWSA-HDLAR technique has proficiently identified all ten class labels effectually on the applied data.

Figure 6. Cont.
The QWSA-HDLAR approach showed enhanced performance on all datasets. For example, on the entire dataset, the QWSA-HDLAR algorithm achieved average accuracy of 99.06%, recall of 95.30%, specificity of 99.48%, F-score of 95.30%, and AUROC score of 97.39%. Finally, with 70% of the TR dataset, the QWSA-HDLAR approach reached average accuracy of 99.06%, recall of 95.40%, specificity of 99.48%, F-score of 95.28%, and AUROC score of 97.44%. At the same time, on 30% of the TS dataset, the QWSA-HDLAR methodology reached average accuracy of 99.07%, recall of 95.26%, specificity of 99.48%, F-score of 95.13%, and AUROC score of 97.37%.

Figure 6. Confusion matrix of QWSA-HDLAR model on UCF Sport dataset.

Table 5 exemplifies the overall HAR outcomes of the QWSA-HDLAR technique on the test UCF Sports dataset. The experimental output illustrated that the QWSA-HDLAR approach showed enhanced performance on all datasets. For example, on the entire dataset, the QWSA-HDLAR algorithm achieved average accuracy of 99.06%, recall of 95.30%, specificity of 99.48%, F-score of 95.30%, and AUROC score of 97.39%. Finally, with 70% of the TR dataset, the QWSA-HDLAR approach reached average accuracy of 99.06%, recall of 95.40%, specificity of 99.48%, F-score of 95.28%, and AUROC score of 97.44%. At the same time, on 30% of the TS dataset, the QWSA-HDLAR methodology reached average accuracy of 99.07%, recall of 95.26%, specificity of 99.48%, F-score of 95.13%, and AUROC score of 97.37%.

The training and validation accuracies depicted by the QWSA-HDLAR methodology with distinct epochs on UCF Sports dataset are demonstrated in Figure 7. The results ensured that the accuracies were higher with increased epochs. In addition to that, training accuracy seems to be better than testing accuracy.

Table 5. HAR outcomes of QWSA-HDLAR model on UCF Sport dataset.

| Class Labels | Accuracy | Recall | Specificity | F-Score | AUROC Score |
|--------------|----------|--------|-------------|---------|--------------|
| Entire Dataset | 99.06  | 95.30  | 99.48       | 95.30   | 97.39        |

(c)
Table 5. Cont.

| Class Labels | Accuracy | Recall | Specificity | F-Score | AUROC Score |
|--------------|----------|--------|-------------|---------|-------------|
|              | Training Phase (70%) |         |             |         |             |
| 1            | 98.57    | 94.37  | 99.05       | 93.06   | 96.71       |
| 2            | 98.86    | 91.03  | 99.84       | 94.67   | 95.43       |
| 3            | 98.57    | 93.65  | 99.06       | 92.19   | 96.35       |
| 4            | 99.71    | 98.57  | 99.84       | 98.57   | 99.21       |
| 5            | 99.29    | 97.18  | 99.52       | 96.50   | 98.35       |
| 6            | 98.71    | 98.44  | 98.74       | 93.33   | 98.35       |
| 7            | 99.00    | 92.31  | 99.84       | 95.36   | 96.07       |
| 8            | 99.14    | 95.59  | 99.53       | 95.59   | 97.56       |
| 9            | 99.14    | 94.44  | 99.68       | 95.77   | 97.06       |
| 10           | 99.57    | 98.46  | 99.69       | 97.71   | 99.07       |
| Average      | 99.06    | 95.40  | 99.48       | 95.28   | 97.44       |
|              | Testing Phase (30%) |         |             |         |             |
| 1            | 98.00    | 86.21  | 99.26       | 89.29   | 92.73       |
| 2            | 99.00    | 95.45  | 99.28       | 93.33   | 97.37       |
| 3            | 99.67    | 100.00 | 99.62       | 98.67   | 99.81       |
| 4            | 99.33    | 100.00 | 99.26       | 96.77   | 99.63       |
| 5            | 99.33    | 96.55  | 99.63       | 96.55   | 98.09       |
| 6            | 99.00    | 94.44  | 99.62       | 95.77   | 97.03       |
| 7            | 99.00    | 95.45  | 99.28       | 93.33   | 97.37       |
| 8            | 98.67    | 93.75  | 99.25       | 93.75   | 96.50       |
| 9            | 99.67    | 96.43  | 100.00      | 98.18   | 98.21       |
| 10           | 99.00    | 94.29  | 99.62       | 95.65   | 96.95       |
| Average      | 99.07    | 95.26  | 99.48       | 95.13   | 97.37       |

Figure 7. Training and validation accuracies of QWSA-HDLAR model on UCF Sport dataset.
The training and validation losses inferred by the QWSA-HDLAR technique on the test UCF Sport dataset are reported in Figure 8. The figure identified that the QWSA-HDLAR algorithm has resulted in lower training and validation loss values.

![Training and Validation Loss - UCF Sport Dataset](image)

**Figure 8.** Training and validation accuracies of QWSA-HDLAR model on KTH dataset.

In order to report the enhanced performance of the QWSA-HDLAR technique, a comparison study with recent methodologies [28,29] is performed on the UCF Sports dataset in Figure 9. The results implied that the AR-DT and LTP-HAR approaches showed poor performance with lower \( accu \) values of 78.21% and 78.84%, respectively.

![UCF Sport Dataset](image)

**Figure 9.** Comparative \( accu \) analysis of QWSA-HDLAR model on UCF Sport dataset.
This is followed by the average two-stream CNN and GMM-KFGRNN algorithms, which reached improved $\text{accu}_y$ values of 88.30% and 88.51%, respectively. Though the DTR-DNN and GS-LOF techniques resulted in reasonable $\text{accu}_y$ values of 95.83% and 95.54%, the QWSA-HDLAR method obtained a higher $\text{accu}_y$ of 99.07%. The detailed results and discussion show that the proposed model has shown effectual performance on HAR over the other models.

5. Conclusions

In this paper, a new QWSA-HDLAR approach has been presented for the recognition of human activities in the HCI environment. The proposed QWSA-HDLAR technique encompasses a series of operations: preprocessing, NASNet feature extraction, QWSA-based hyperparameter tuning, and CNN-BiRNN-based classification. In the QWSA-HDLAR model, the QWSA is applied to optimally choose the hyperparameter values of the NASNet model. The experimental validation of the QWSA-HDLAR model is tested using two datasets, namely the KTH and UCF Sports datasets. The experimental values reported the supremacy of the QWSA-HDLAR model over recent DL approaches with maximum accuracy of 99.63% and 99.07% on the applied KTH and UCF Sport datasets, respectively. In the future, an ensemble of three DL models can be applied to improvise the overall recognition performance of the QWSA-HDLAR method.

Author Contributions: Conceptualization, A.A.M.; Data curation, J.S.A.; Formal analysis, J.S.A. and A.Q.; Investigation, A.Q.; Methodology, A.M.H.; Project administration, M.M.; Resources, M.M.; Software, M.A.; Supervision, M.A.; Validation, R.A. and A.M.; Visualization, R.A. and A.M.; Writing—original draft, A.A.M. and A.M.H. All authors have read and agreed to the published version of the manuscript.

Funding: The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work through the Small Groups Project under grant number (241/43). Taif University Researchers Supporting Project number (TURSP-2020/346), Taif University, Taif, Saudi Arabia. The authors would like to thank the Deanship of Scientific Research at Umm Al-Qura University for supporting this work by Grant Code: 22UQU4340237DSR26.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable to this article, as no datasets were generated during the current study.

Acknowledgments: The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work through the Small Groups Project under grant number (241/43). Taif University Researchers Supporting Project number (TURSP-2020/346), Taif University, Taif, Saudi Arabia. The authors would like to thank the Deanship of Scientific Research at Umm Al-Qura University for supporting this work by Grant Code: 22UQU4340237DSR26.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhang, S.; Li, Y.; Zhang, S.; Shahabi, F.; Xia, S.; Deng, Y.; Alshurafa, N. Deep learning in human activity recognition with wearable sensors: A review on advances. Sensors 2022, 22, 1476. [CrossRef] [PubMed]
2. Boualia, S.N.; Amara, N.E.B. Pose-based human activity recognition: A review. In Proceedings of the 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC), Tangier, Morocco, 24–28 June 2019; pp. 1468–1475.
3. Li, X.; He, Y.; Jing, X. A survey of deep learning-based human activity recognition in radar. Remote Sens. 2019, 11, 1068. [CrossRef]
4. Pareek, P.; Thakkar, A. A survey on video-based human action recognition: Recent updates, datasets, challenges, and applications. Artif. Intell. Rev. 2021, 54, 2259–2322. [CrossRef]
5. Antonik, P.; Marsal, N.; Brunner, D.; Romani, D. Human action recognition with a large-scale brain-inspired photonic computer. Nat. Mach. Intell. 2019, 1, 530–537. [CrossRef]
6. Anwer, M.H.; Hadeel, A.; Fahd, N.A.; Mohamed, K.N.; Abdelwahed, M.; Anil, K.; Ishfaq, Y.; Abu Sarwar, Z. Fuzzy cognitive maps with bird swarm intelligence optimization-based remote sensing image classification. Comput. Intell. Neurosci. 2022, 2022, 4063354.
7. Bijalwan, V.; Semwal, V.B.; Gupta, V. Wearable sensor-based pattern mining for human activity recognition: Deep learning approach. *Ind. Robot. Int. J. Robot. Res. Appl.* **2021**, *49*, 21–33. [CrossRef]

8. Majumder, S.; Kehtarnavaz, N. A review of real-time human action recognition involving vision sensing. In Proceedings of the Real-Time Image Processing and Deep Learning 2021, Online; Volume 11736, pp. 53–64.

9. Pandu, S.B.; Britto, A.; Francis, S.; Sekhar, P.; Vijayarajan, P.; Albraikan, A.A.; Al-Wesabi, F.N.; Al Duhayyim, M. Artificial intelligence based solar radiation predictive model using weather forecasts. *Comput. Mater. Contin.* **2022**, *71*, 109–124.

10. Cao, J.; Lin, M.; Wang, H.; Fang, J.; Xu, Y. Towards activity recognition through multidimensional mobile data fusion with a smartphone and deep learning. *Mob. Inf. Syst.* **2021**, *2021*, 6615695. [CrossRef]

11. Gu, F.; Chung, M.H.; Chignell, M.; Valaee, S.; Zhou, B.; Liu, X. A survey on deep learning for human activity recognition. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–34. [CrossRef]

12. Verma, K.K.; Singh, B.M. Deep Multi-Model Fusion for Human Activity Recognition Using Evolutionary Algorithms. *Int. J. Interact. Multimed. Artif. Intell.* **2021**, *7*, 2. [CrossRef]

13. Khan, M.A.; Zhang, Y.D.; Allison, M.; Kadry, S.; Wang, S.H.; Saba, T.; Iqbal, T. A fused heterogeneous deep neural network and robust feature selection framework for human actions recognition. *Arab. J. Sci. Eng.* **2021**, *1–16*. [CrossRef]

14. Male, J.; Martinez-Hernandez, U. March. Recognition of human activity and the state of an assembly task using vision and inertial sensor fusion methods. In Proceedings of the 2021 22nd IEEE International Conference on Industrial Technology (ICIT), Valencia, Spain, 10–12 March 2021; Volume 1, pp. 919–924.

15. Zheng, X.; Wang, M.; Ordieres-Meré, J. Comparison of data preprocessing approaches for applying deep learning to human activity recognition in the context of industry 4.0. *Sensors* **2018**, *18*, 2146. [CrossRef] [PubMed]

16. Tanberk, S.; Kilimci, Z.H.; Tükel, D.B.; Uysal, M.; Akyokuş, S. A hybrid deep model using deep learning and dense optical flow approaches for human activity recognition. *IEEE Access* **2020**, *8*, 19799–19809. [CrossRef]

17. Abdulazeem, Y.; Balaha, H.M.; Bahgat, W.M.; Badawy, M. Human action recognition based on transfer learning approach. *IEEE Access* **2021**, *9*, 82058–82069. [CrossRef]

18. Ronald, M.; Poulose, A.; Han, D.S. iSPLInception: An inception-ResNet deep learning architecture for human activity recognition. *IEEE Access* **2021**, *9*, 68985–69001. [CrossRef]

19. Male, J.; Martinez-Hernandez, U. March. Recognition of human activity and the state of an assembly task using vision and inertial sensor fusion methods. In Proceedings of the 2021 22nd IEEE International Conference on Industrial Technology (ICIT), Valencia, Spain, 10–12 March 2021; Volume 1, pp. 919–924.

20. Han, T.; Liu, C.; Wu, R.; Jiang, D. Deep transfer learning with limited data for machinery fault diagnosis. *Appl. Soft Comput.* **2021**, *103*, 107150. [CrossRef]

21. Cano, E.; Mendoza-Avilés, J.; Areiza, M.; Guerra, N.; Mendoza-Valdés, J.L.; Rovetto, C.A. Multi skin lesions classification using fine-tuning and data-augmentation applying NASNet. *PeerJ Comput. Sci.* **2021**, *7*, e371. [CrossRef]

22. Karthick, R.; Senthilveli, A.; Meenalochni, P.; Senthil Pandi, S. Design and Analysis of Linear Phase Finite Impulse Response Filter Using Water Strider Optimization Algorithm in FPGA. *Circuits Syst. Signal Proc. 2022*, *1–29*. [CrossRef]

23. Sun, K.; Zhao, T.; Li, Z.; Wang, L.; Wang, R.; Chen, X.; Yang, Q.; Ramezani, E. Methodology for optimal parametrization of the human action recognition using support vector machines and 3D convolutional neural networks. *Int. J. Adv. Intell. Inform.* **2017**, *3*, 47–55. [CrossRef]

24. Geng, C.; Song, J.X. Human action recognition based on support vector machines and 3D convolutional neural networks. *Appl. Sci.* **2022**, *12*, 6848. [CrossRef]

25. Available online: http://www.nada.kth.se/cvap/actions/ (accessed on 15 March 2022).

26. Available online: http://crcv.ucf.edu/data/UCF_Sports_Action.php (accessed on 22 March 2022).

27. Latah, M. Human action recognition using support vector machines and 3D convolutional neural networks. *Sensors* **2021**, *21*, 68985–69001. [CrossRef]

28. Available online: http://www.nada.kth.se/cvap/actions/ (accessed on 15 March 2022).

29. de Oliveira Silva, V.; de Barros Vidal, E.; Romariz, A.R.S. Human action recognition based on a two-stream convolutional network classifier. In Proceedings of the 5th International Conference on Computer Sciences and Automation Engineering (ICCSAE 2015), Sanya, China, 14–15 November 2015. [CrossRef]