applied sciences

MDPI

Article

Deep Learning Based Human Activity Recognition Using Spatio-Temporal Image Formation of Skeleton Joints

Nusrat Tasnim [1], Mohammad Khairul Islam [2] and Joong-Hwan Baek [1,*]

[1] School of Electronics and Information Engineering, Korea Aerospace University, Goyang 10540, Korea; tasnim.nishu70@kau.kr

[2] Department of Computer Science and Engineering, University of Chittagong, Chittagong 4331, Bangladesh; mkislam@cu.ac.bd

* Correspondence: jhbaek@kau.ac.kr; Tel.: +82-2-300-0125

Abstract: Human activity recognition has become a significant research trend in the fields of computer vision, image processing, and human–machine or human–object interaction due to cost-effectiveness, time management, rehabilitation, and the pandemic of diseases. Over the past years, several methods published for human action recognition using RGB (red, green, and blue), depth, and skeleton datasets. Most of the methods introduced for action classification using skeleton datasets are constrained in some perspectives including features representation, complexity, and performance. However, there is still a challenging problem of providing an effective and efficient method for human action discrimination using a 3D skeleton dataset. There is a lot of room to map the 3D skeleton joint coordinates into spatio-temporal formats to reduce the complexity of the system, to provide a more accurate system to recognize human behaviors, and to improve the overall performance. In this paper, we suggest a spatio-temporal image formation (STIF) technique of 3D skeleton joints by capturing spatial information and temporal changes for action discrimination. We conduct transfer learning (pretrained models- MobileNetV2, DenseNet121, and ResNet18 trained with ImageNet dataset) to extract discriminative features and evaluate the proposed method with several fusion techniques. We mainly investigate the effect of three fusion methods such as element-wise average, multiplication, and maximization on the performance variation to human action recognition. Our deep learning-based method outperforms prior works using UTD-MHAD (University of Texas at Dallas multi-modal human action dataset) and MSR-Action3D (Microsoft action 3D), publicly available benchmark 3D skeleton datasets with STIF representation. We attain accuracies of approximately 98.93%, 99.65%, and 98.80% for UTD-MHAD and 96.00%, 98.75%, and 97.08% for MSR-Action3D skeleton datasets using MobileNetV2, DenseNet121, and ResNet18, respectively.

Keywords: spatio-temporal image formation; human activity recognition; deep learning; fusion strategies; transfer learning

Citation: Tasnim, N.; Islam, M.K.; Baek, J.-H. Deep Learning Based Human Activity Recognition Using Spatio-Temporal Image Formation of Skeleton Joints. *Appl. Sci.* **2021**, *11*, 2675. https://doi.org/10.3390/app11062675

Academic Editor: Hyo Jong Lee

Received: 22 February 2021
Accepted: 15 March 2021
Published: 17 March 2021

1. Introduction

Human action recognition has been grabbing more attention among researchers due to its multitude of real-world applications, for example, human–machine or human–object interaction [1–3], smart and intelligent surveillance system [4–6], content-based data retrieval [7], virtual reality/augmented reality [8,9], health care system [10,11], autonomous driving [12], and games [13]. The demand for human action recognition as well as pose estimation [14] is also expanding significantly, as a result, to manage the time, avoid intimate contact during the pandemic of diseases, and provide comfortable interaction for the rehabilitees. The human action recognition system focuses on identifying the activity accurately about what type of behavior is undertaken in a sequence of frames of so-called video.

With the progress of modern technology, various sensor devices such as Microsoft Kinect, Z-Depth, Leap Motion Controller, and Intel RealSense [15] have been invented to

capture human activity data. Different modalities, for instance, RGB, depth, and skeleton data [16] are mainly used by the researchers for video-based human action recognition. Even though notable success has been appeared in the human action recognition system in the last few years, it is still challenging to accurately predict human activity for various restrictions such as camera orientation, occlusions, background, the variation of length, and speed of action [17]. Several methods offered a variety of ideas about human action recognition using hand-crafted features as well as using deep-learning features.

Hand-crafted features are usually referred to as the extraction of meaningful information (such as edges and corners) present in the images or videos using various descriptors e.g., local binary pattern (LBP) [18], space-time interest points (STIP) [19], scale-invariant feature transform (SIFT) [20], speeded up robust features (SURF) [21], histograms of oriented gradient (HOG) [22], and histograms of optic flow (HOF) [23]. These descriptors generate discriminative features by extracting information from locally important patches to represent action sequences. Abdul et al. [19] proposed a method for action recognition using trajectory-based feature representation where meaningful spatial information was preserved by detecting STIPs with SIFT and temporal change between the consecutive frames in an RGB video was computed by matching the interest points. Another hand-crafted feature-based method was introduced by Mahjoub et al. [24] with spatio-temporal interest point to detect the interest points in the video. They took the help of HOG and HOF descriptors to extract appearance and motion features to perform classification using support vector machine (SVM). In [25], Akam et al. combined local and global features extracted from RGB sequences and performed classification with SVM. They designed the shape descriptor as a local descriptor by integrating 3D trajectory and motion boundary histogram and extracted global features with gist descriptor for classification.

The engagement of modern researchers in the field of deep learning helps to design several learning models such as convolutional neural network (CNN) [26], recurrent neural network (RNN) [27], and long short-term memory (LSTM) [27]. These well-known models are widely being used to extract deep features for human action recognition. A two-stream CNN model was designed by Simonyan et al. [28] for human action recognition in RGB videos. The proposed CNN models consisting of spatial and temporal networks covered both local and global changes in the sequences for action discrimination. Zhang et al. [29] considered the time complexity to calculate the optical flow and suggested a deeply transferred motion vector CNN model to take the scope of optical flow. In [30], an effective deep 3D CNN model was introduced by Tran et al. to dig up the spatio-temporal features and recognized the action classes. Donahue et al. [31] proposed a long-term recurrent convolutional network with doubly deep compared to a fixed simple spatio-temporal receptive field for sequential processing. By using a long-term recurrent network, they captured complex dynamic to classify action groups.

Dataset captured in RGB format suffers from view dependency, background and illumination sensitivity, and computational complexity. While acquired an image or video of action in RGB format, it generates a lot of pixel values that makes it more complicated to differentiate from the background. The afore-mentioned obstacles hinder the performance of RGB video-based human action recognition and persuade to adopt the capturing devices to generate the depth and other formats of images or videos. Cheng et al. [32] proposed an efficient method for action recognition by extracting LBP features from depth motion map representation of three different views in depth sequences. They performed the classification by combining two fusion methods (feature-level fusion and decision-level). The integration of local and global features generated from depth sequences was introduced by Wu et al. [33]. They took the advantages of the correlation among the poses in neighboring frames of action to get the modified bag-of-words model called the bag of correlated poses. The dimensionality of the feature map was reduced with the help of principal components analysis and classified using SVM. Trelinski et al. [34] presented a CNN features-based method for human action recognition using depth sequences. The features were extracted by training a CNN model with multi-channel inputs such as two consecutive frames and

projected view on the Cartesian plane. Finally, a LSTM was trained to determine the classification results. In [35], Wang et al. encoded the depth maps into weighted hierarchical depth motion maps (WHDMM) in three orthogonal planes. Then a three-branch CNN was conducted to pick out the discriminative features from WHDMM for the classification of human action.

Even though the depth images or videos require very-little storage compared to RGB, it also sustains from color, texture, and shape information. Meanwhile, both RGB and depth videos are captured using traditional cameras in 2D space. Thus, human action recognition based on RGB and depth sequences lack more deep information due to its 2D orientation and cannot capture the 3D structure of the action. By considering the following limitation, many sensor devices such as Microsoft Kinect provides more optimized information about the human body in terms of twenty skeleton joints. The skeleton information is represented in a deep 3D structure that is view-independent. Thus, the acquisition of an image or video in 3D skeleton format is much faster, lightweight in storage, and easy to use in the fields of human action recognition. Human action recognition based on skeleton data requires the exploitation of spatial and temporal changes of 3D skeleton joints in the sequences of action. There are several methods that suggest skeleton-based human action discrimination ranging from hand-crafted features based on human action classification using traditional machine learning algorithms to deep features based on human action recognition using deep learning. To provide more discriminative local and temporal features to improve recognition performance, skeleton joints information is represented in spatial format by capturing motion. Thus, it is very important to map the 3D skeleton joints in such a way that can cumulate both spatial and temporal information.

In this paper, we propose a new 3D skeleton-based human action recognition method by mapping the skeleton joints into spatio-temporal image by joining line between the same joints in two adjacent frames. Initially, we draw the position of joints by putting pixels with different colors in the jet color map (a color generated from the jet color map is the array of red, green, and blue intensities ranging from 0 to 1) in each frame. Then we draw lines between the same joints in two consecutive frames with different colors in the jet color map and combine them to get the final spatio-temporal image that helps to maintain both intra-frame and inter-frames information. To overcome the view dependency, we map the 3D skeleton joint coordinates along XY, YZ, and ZX planes (front, side, and top views). As the popularity of deep learning is increasing along with the recognition performance, we conduct the pretrained deep learning models to perform the classification of human action. First, we use the pretrained model to extract the discriminative features from the spatio-temporal images obtained from the front, side, and top views and then fuse the feature maps to get the final outputs. We also fine-tune the pretrained model to reduce the complexity and improve the recognition performance.

The remaining sections of this paper are illustrated as follows: The primitive knowledge including transfer learning and dataset are described in Section 2. In Section 3, we try to discuss the major ideas and limitations of the state-of-the-art studies about skeleton-based human action recognition. Section 4 elaborately explains the methodology of the proposed system in a step-by-step manner. The recognition results and performance comparisons are included in Section 5. We provide an analysis and discussion in Section 6. Finally, we add the conclusive words about the proposed method in Section 7.

2. Background Study

In this section, we present the elementary knowledge required for the proposed system. First, we interpret deep learning, more specifically transfer learning. Then we clarify the dataset used in overall illustrations and experiments.

2.1. Transfer Learning

With the tremendous improvement of modern technology over the past few years, we have seen much success of deep neural networks in different fields, particularly in recogni-

tion, classification, and machine translation tasks. These significant changes are also accelerated to the network architectural design such as AlexNet, ResNet, SuffleNet, GoogleNet, DenseNet, and MobileNet [36]. Even though the achievement of the deep neural network design has significant contributions in this domain, it still requires advanced knowledge and huge time to design an effective and efficient model. Due to the design and time complexity of the deep learning, we use the pretrained model trained with the ImageNet dataset for classification tasks known as transfer learning for human action recognition. We fine-tune the pretrained model to get better results for our proposed method. We consider three well-known pretrained deep learning models: MobileNetV2 [37], DenseNet121 [38], and ResNet18 [39] as the backbone of CNN models which are commonly applied in detection, recognition, and classification problems. To emphasize on the strength of STIF representation from the 3D skeleton data, we consider both less parameterized model (MobileNetV2) and heavy parameterized models (DenseNet121 and ResNet18).

MobileNetV2 integrates inverted residual blocks [37] in which the shortcut connection is established between the thin bottleneck layers. In MobileNetV2, an inverted residual block with a linear bottleneck first increases the dimensionality of the feature maps from low-dimensional inputs to high-dimensional outputs. Then a light-weight depth-wise separable convolution is introduced to filter the outputs. The feature maps are processed from low-dimension to high-dimension and again back into low-dimension. The main scenario is similar to narrow-wide-narrow concepts that reduce the number of parameters significantly. Figure 1 shows an inverted residual block of MobileNetV2 in which there are three blocks. The first block performs 1×1 convolution along with batch normalization and rectified linear unit (ReLU) operations to generate wide-dimensional feature maps. This wide-dimensional feature maps are then passed through the second block that accomplishes the depth-wise 3×3 separable convolution to reduce the computation. Finally, the third block conducts 1×1 convolution and batch normalization and reduces the dimensionality of the feature maps.

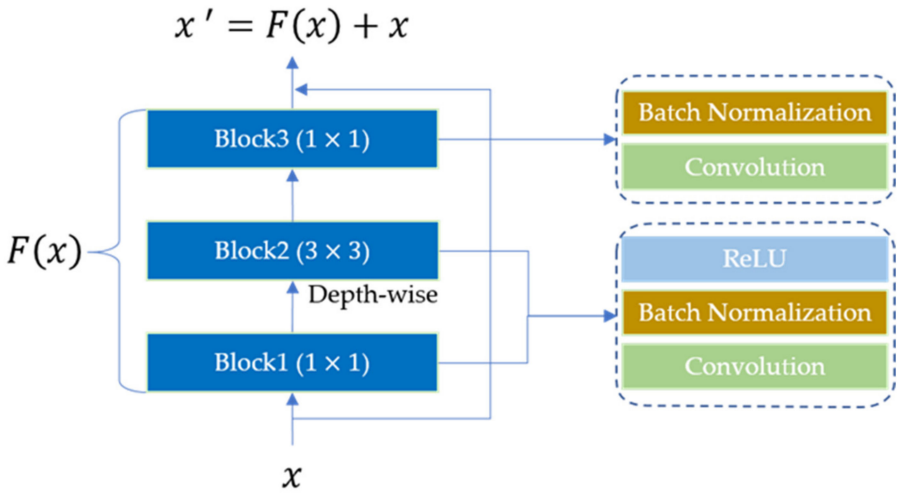


Figure 1. Inverted residual block in MobileNetV2.

DenseNet (dense convolutional network) focuses on the extraction of deeper features and tries to make it more efficient for training. DenseNet consists of two basic blocks: (i) dense block and (ii) transition block [38]. Each layer in DenseNet is connected to all other deeper layers in the network. The first layer is connected to the second, third, fourth, and so on. The second layer is joined with the third, fourth, fifth, and so on. Figure 2a shows a dense block in DenseNet in which each layer is connected with all other deeper layers. The transition layer is inserted to reduce the dimensionality of the output features in which a batch normalization, a convolution with 1×1 kernel, and an average pooling with 2×2 kernel operations are performed. Figure 2b depicts the graphical representation

of the transition block in the DenseNet. In our experiments, we consider DenseNet121 for human behaviors classification.

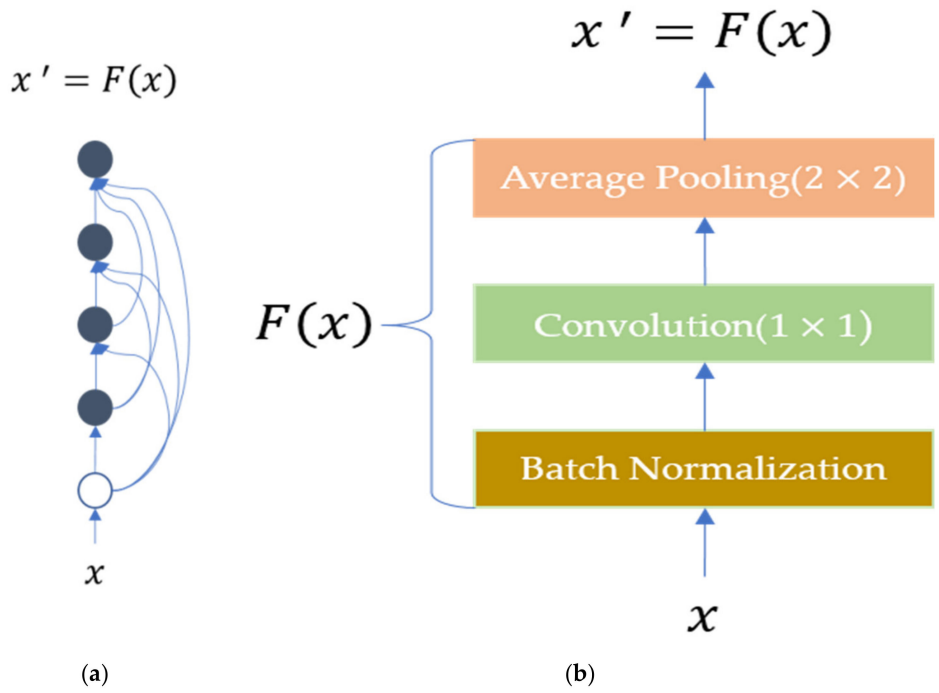


Figure 2. DenseNet (**a**) dense and (**b**) transition blocks.

ResNet (residual network) is designed by adding skip or shortcut connections from the previous layer to the current layer known as residual information. Compared with other well-known deep learning models, ResNet helps us to train up to hundreds or thousands of layers to get deeper features and improve the performance, particularly in recognition and classification tasks. The main component of ResNet is the residual blocks [39]. Figure 3 visualizes two kinds of residual blocks. The first residual block integrates skip connection directly to the output features, as shown in Figure 3a. However, the second residual block in Figure 3b adjusts the channels and resolution by conducting a 1×1 convolution before joining the skip connection. We adopt ResNet18 for our experiments.

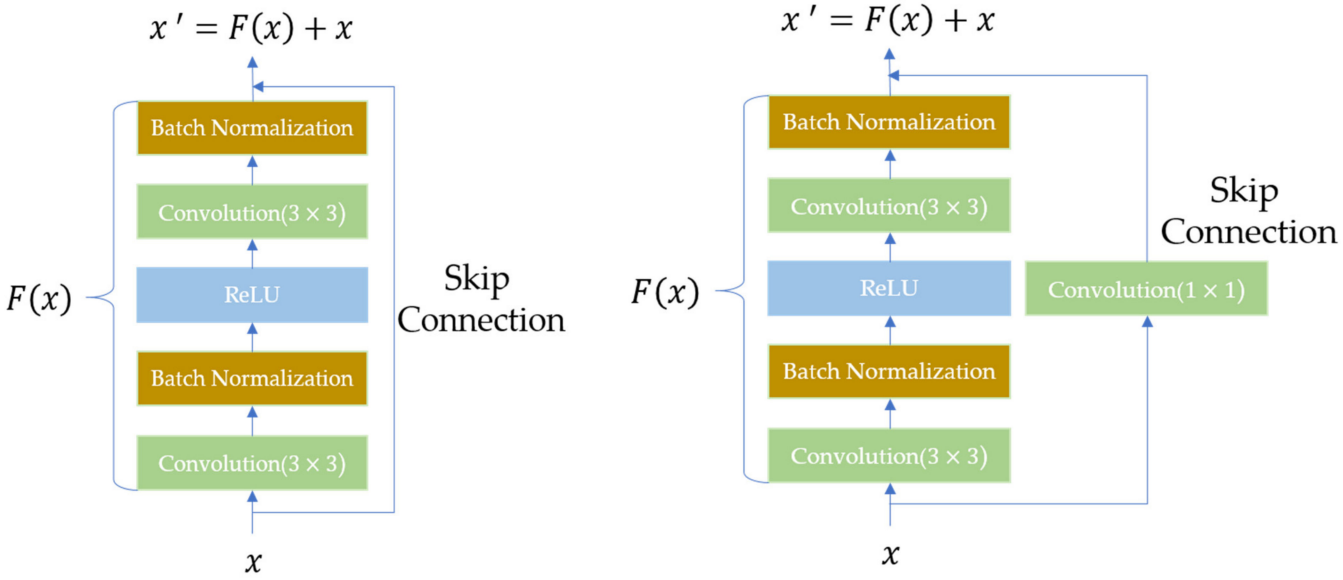
2.2. Dataset

In this section, we provide a detail description of the datasets used in the proposed system. We use two publicly available datasets: UTD multi-modal human action dataset (UTD-MHAD) [40] and MSR-Action3D dataset [41].

2.2.1. UTD-MHAD

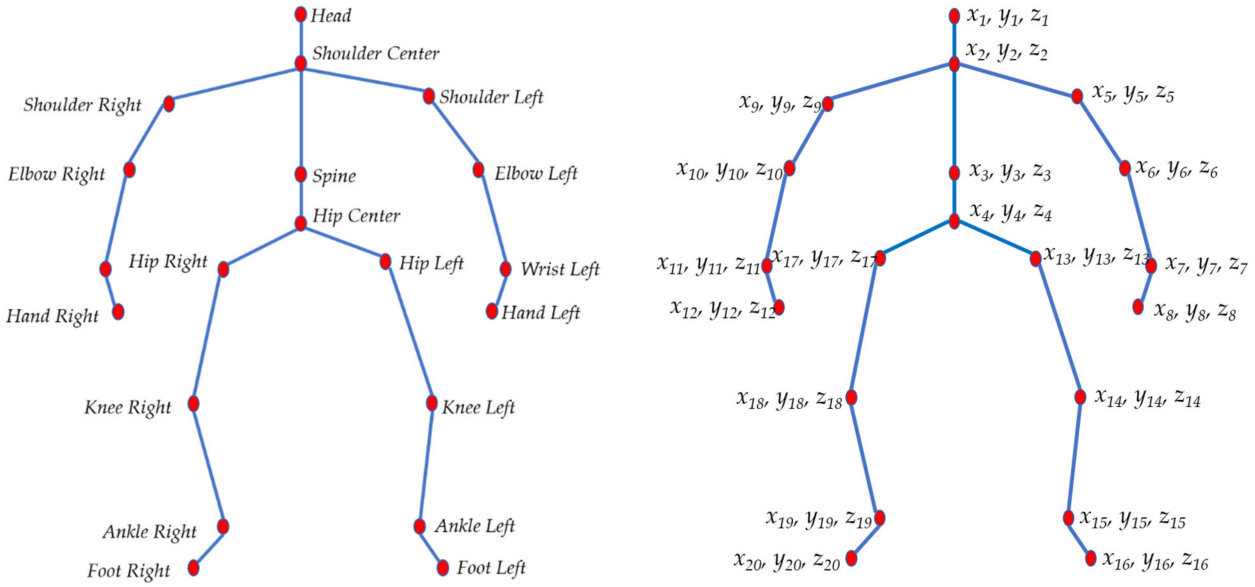
The members at embedded systems and signal processing (ESSP) Laboratory of the University of Texas at Dallas captured the UTD-MHAD skeleton dataset by using Microsoft Kinect camera. They extracted twenty skeleton joint coordinates of human body along *X*, *Y*, and *Z* axes as shown in Figure 4. Figure 4a shows the representation of the twenty skeleton joints with the corresponding names: *'Head'*, *'Shoulder Center'*, *'Spine'*, *'Hip Center'*, *'Shoulder Left'*, *'Elbow Left'*, *'Wrist Left'*, *'Hand Left'*, *'Shoulder Right'*, *'Elbow Right'*, *'Wrist Right'*, *'Hand Right'*, *'Hip Left'*, *'Knee Left'*, *'Ankle Left'*, *'Foot Left'*, *'Hip Right'*, *'Knee Right'*, *'Ankle Right'*, *'Foot Right'*. For better understanding and analysis, we order the joints from $(x_1,\ y_1, z_1)$ to $(x_{20},\ y_{20},\ z_{20})$ as shown in Figure 4b. UTD-MHAD contains a total of 27 actions dataset which is accomplished by 8 persons (4 females and 4 males). Each person performs each action 4 times. There are three corrupted data removed and a total of 861 sequences are kept for the experiments. The names of the action classes are as follows: *'SwipeLeft'*, *'SwipeRight'*, *'Wave'*, *'Clap'*, *'Throw'*, *'ArmCross'*, *'BasketballShoot'*, *'DrawX'*, *'DrawCircle(CLW)'*,

'DrawCircle(counter CLW)', *'DrawTriangle'*, *'Bowling'*, *'Boxing'*, *'BaseballSwing'*, *'TennisSwing'*, *'ArmCurl'*, *'TennisServe'*, *'Push'*, *'Knock'*, *'Catch'*, *'PickUpandThrow'*, *'Jog'*, *'Walk'*, *'SitToStand'*, *'StandToSit'*, *'Lunge'*, *'Squat'*. Most of actions in this dataset are captured by hands. Three actions such as *'Jog'*, *'Walk'*, *'Lunge'* is performed by the leg, and only two actions *'SitToStand'* and *'StandToSit'* are done by the full body. We consider the dataset obtained by the first 5 subjects for training and 3 subjects for testing.



(**a**) (**b**)

Figure 3. ResNet residual blocks (**a**) without and (**b**) with 1 × 1 convolution block.



(**a**) (**b**)

Figure 4. Human skeleton joints views; (**a**) twenty joints names and (**b**) corresponding joints numbering.

2.2.2. MSR-Action3D

Wanqing Li captured the MSR-Action3D dataset with the help of the Communication and Collaboration Systems Group at Microsoft Research Red-mond. This dataset contains 20 classes of actions which are done by 10 persons. Each person repeated each action three times. There are several corrupted sequences that are discarded and a total of 567 action sequences are maintained. As described in the previous Section 2.2.1 for UTD-MHAD dataset, the MSR-Action3D dataset also has twenty skeleton joint coordinates of human body along X, Y, and Z axes. The names and order are the same as the UTD-MHAD dataset as depicted in Figure 4. The action classes are as follows: *'HighArmWave'*, *'HoizontalArmWave'*, *'Hammer'*, *'HandCatch'*, *'ForwardPunch'*, *'HighThrow'*, *'DrawCross'*, *'DrawTick'*, *'DrawCircle'*, *'HandClap'*, *'TwoHandWave'*, *'SideBoxing'*, *'Band'*, *'ForwardKick'*, *'SideKick'*, *'Jogging'*, *'TennisSwing'*, *'TennisServe'*, *'GolfSwing'*, *'PickUpandThrow'*. The dataset from the first 6 subjects is used for training and the rest of the 4 subjects for testing.

3. Related Works

The shortcoming [17] of human behaviors capturing devices leads to the optimization of information about the human body. The 3D human skeleton dataset provides optimal and meaningful joint coordinates to recognize the genre of human attitudes. Several state-of-the-art review letters have been published on human action recognition particularly, 3D skeleton-based human action recognition [42–68]. The interpretations about the categories of 3D skeleton-based action recognition are separated into broad groups: trajectory and pose-based classification [42], joint and part-based classification [16,43], hand-crafted and deep learning-based classification [16,17,42,44], joints, mined joints, and dynamic based action classification [45], and spatio-temporal representation-based classification [42]. The spatio-temporal representation with CNN, RNN, LSTM, and the integration of CNN with RNN are also conducted for human action recognition using skeleton data [42]. All the referred categories fall into two major groups: 3D skeleton-based human action recognition with traditional classifiers using hand-crafted features and with deep learning features. The principal ideas of the prior works are briefly composed in the later sections.

3.1. Traditional Features-Based Classifiers with Skeleton Dataset for Human Action Recognition

The commonly used classifiers including SVM, k-nearest neighbors, hidden markov model (HMM), dynamic time warping (DTW), extreme learning machine (ELM), and Bayesian classifier are considered for skeleton-based human activity recognition. The above-mentioned discrimination methods require meaningful hand-crafted features for the classification of human behaviors. Video-based human action recognition depends on the spatial information of each frame and temporal changes of all frames. To perform hand-crafted features-based human action discrimination, the spatial information of each frame and temporal information between neighboring frames must have to be captured and combined.

Xia et al. [46] introduced new features called the histogram-based representation of 3D human posture from the projected views of depth sequences using linear discriminant analysis. The temporal information was kept by using HMM. In [47], Yang et al. presented a novel approach where they extracted features based on EigenJoints of joint positions differences and conducted Naïve Bayes Nearest Neighbor classifier for action recognition. Zhu et al. [48] demonstrated a new method for human action recognition by fusing spatio-temporal motion information and frame difference with the pairwise distance of skeleton joint coordinates. The spatio-temporal information determined by 3D interest point detection and local feature descriptor using Harris3D detector. The geometric relationship between various human body parts was exposed by Vemulapalli et al. [43] using rotations and transformation in 3D space. The performance was evaluated with DTW, Fourier temporal pyramid, and linear SVM by the representation of the lie group. Evangelidis et al. [49] illustrated a local descriptor from skeleton joints that maintained view-invariant features. They conducted Fisher kernel to explain the skeleton quads in an

action and performed the classification using traditional SVM. A fast and powerful method for human action recognition using the 3D skeleton dataset was published by Zanfir et al. [50] in which they considered a moving pose descriptor containing pose information, speed, and acceleration of joints. Agahian et al. [51] assumed the skeleton sequences of action as a set of spatio-temporal poses. First, they normalized the joint coordinate and computed temporal variations. Then SVM was used to differentiate among the bag of poses (BOP) and finally histogram features were mined for the purposes of action classification using ELM. An effective multi-instance learning idea was brought out by Yang et al. [52] to find the discriminative multi-instance multi-task learning (MIMTL) features to expose the relationship among the skeleton joints. They also conducted multi-task learning model to recognize human action with MIMTL.

The previous hand-crafted features-based methods require an active engagement along with a lot of efforts to extract the feature of spatial and temporal information from the skeleton sequences. Sometimes, it becomes more complicated to design discriminative features from the 3D skeleton videos that degrades the performance of the system.

3.2. Deep Convolutional Neural Network for Skeleton-Based Human Action Recognition

Deep learning such as CNN, RNN, and LSTM naturalizes the process of human action classification by assembling the automated feature extraction and discrimination stages. Diverse methods revealed different techniques to recognize human behaviors from skeleton sequences using deep learning [53–68]. The most prominent issues in the prior methods based on deep learning can be divided into two categories: the use of raw 3D skeleton joint coordinates and the spatial representation of 3D skeleton joints.

Various skeleton-based human action recognition methods focused on the design of more powerful deep learning models to dig up the significant features from the skeleton joints. Du et al. [53] proposed an end-to-end hierarchical RNN with five branches to perform human action classification using raw skeleton sequence (RSS). They partitioned the skeleton joints into five segments and separately passed through the five branches of RNN. Finally, they combined the generated features and integrated a fully connected layer along with a softmax layer to make the decision. A two-stream RNN was introduced by Wang et al. [54] for human action recognition based on skeleton dataset. They considered both spatial and temporal information that was captured by two branches of RNN called temporal RNN and spatial RNN. Zhu et al. [55] designed a deep learning model by arranging LSTM and feed-forward fusion layer subsequently to learn co-occurrence of the human skeleton joints. They also provided a new dropout algorithm to train the model. In [56], Liu et al. explored the drawback of RNN based contextual learning to spatio-temporal learning by representing the skeleton joints into tree-structure. They again explained a novel gating technique to handle the noise and occlusion and accomplished the classification by using a spatio-temporal LSTM network. Song et al. [57] suggested an end-to-end spatio-temporal deep learning model using RNN with LSTM and got the help of spatial and temporal attention blocks to maintain the intra-frame and inter-frame relationship. A CNN-based approach for human action detection and classification was proposed by Li. et al. [58] in which they extracted discriminative features from raw skeleton joints and from the difference of skeleton joints in neighboring frames. They concatenated the extracted feature maps and applied two fully connected layers with a softmax layer to make the final prediction. Si et al. [59] separated skeleton joints of the human body into five parts and computed spatial features using a hierarchical spatial reasoning network. Then they built a temporal stack learning network consisting of velocity and position networks to capture temporal information. The dependency of RNN on the relationships between different parts of the human body inspired Zhang et al. [60] to use a simple universal geometric feature for skeleton-based action recognition. They modeled a 3-layer LSTM network to classify the geometric feature.

Some methods concentrated to map the 3D skeleton joints into a spatial format such as image format and fine-tuned the well-known deep learning models for features extraction

and recognition. Du et al. [61] reorganized the skeleton sequence as a matrix by chronological order of joint coordinates in each frame along the temporal direction. Then the matrix was encoded into an image format with red, blue, and green for the coordinate values along X, Y, and Z axes. The recognition of human activity was obtained by a CNN model with spatio-temporal image representation of the skeleton joints. In [62], Liu et al. represented the skeleton sequences into image format by distinctive joint locations arrangement. They additionally mapped the relative joint velocities and used the CNN model to perform the action classification. The skeleton joint sequences were transformed into static images called skeleton optical flow guided feature (SOFGF) with determining the displacement of joints, angles, and joint distances by Ren et al. [63] and trained multi-stream CNN for discrimination. Li et al. [64] transformed the 3D skeleton sequences to color images using red, green, and blue colors which were translation and scale invariant. They developed a multi-scale CNN of image classification for human action recognition. A method for spatio-temporal information to color images of skeleton sequences called temporal pyramid skeleton motion map (TPSMM) was encoded by Chen et al. [65] in the frame to segment-wise manners. Then they conducted six CNN branches to extract features and performed classification. Li et al. [66] calculated the pair-wise distance from skeleton sequences and represented them into joint distance map (JDM) for three different views along XY, YZ, and ZX planes. Four branches of the CNN model were implemented to find out the distinctive features for action recognition using JDM. Hou et al. [67] formatted the skeleton sequences into skeleton optical spectra (SOS) using hue, saturation, and brightness colors to capture the spatial and motion information. They mapped three different views (front, side, and top) of SOS and deployed three branches of CNN to measure the classification results. Another similar method was suggested by Wang et al. [68], in which they first rotated the skeleton sequences to make view-invariant and increase the dataset. They represented the rotated skeleton sequences into joint trajectory maps (JTMs) and integrated fusion methods to get the recognition results.

3.3. Limitations of the State-of-the-Works and Our Contributions

As the hand-crafted features-based methods [43,46–52] require explicit contact by the researchers, it is always a tedious task to find out meaningful features from the skeleton sequences. It is also very hard to design a powerful deep learning model to process the raw skeleton joints to capture spatial and temporal information [53–60]. Above all, the deep learning-based human action recognition methods using raw skeleton dataset show comparatively worse performance than the deep learning-based methods using spatio-temporal representation of skeleton sequence. Sometimes, the spatio-temporal encoding cannot maintain local information as well as the global information significantly from the skeleton joints. The methods described [61–68] suffer from spatio-temporal design complexities such as view dependency, lack of motion, and deficiency of spatial and temporal information. These disadvantages lead to the low performance in skeleton-based human action recognition.

Thus, we figure out a simple yet robust, effective, and efficient way to represent the 3D skeleton joint coordinates into image format called STIF that defends both spatial and temporal variation of human behaviors specific movements. The STIF mapping facilitates to train the fine-tuned deep learning models that can automatically generate the meaningful features and performs the classification of human activity.

The major contributions of this paper are summarized as follows:

1. We idealize a novel technique to map 3D skeleton joints into spatio-temporal image. Our spatio-temporal image provides more discriminable features.
2. We adopt several fusion strategies (element-wise average, multiplication, and maximization) to expose the performance variations. Element-wise maximization shows better performance than average and multiplication.

3. We justify the robustness of our approach using both light weight and heavy weight deep learning models. We consider MobileNetV2 as the light weight and DenseNet121 and ResNet18 as the heavy weight models.
4. We compare the experimental results with prior works to show the effectiveness of our proposed method.

4. Proposed Methods

The description of the proposed methodology alongside the analysis is integrated with this section. First, the key ideas are analyzed, and then provided a detailed visual representation of the proposed system. Then we provide the spatial-temporal representation of the skeleton joint coordinates, knowledge transfer using well-known pretrained model, and finally conduct several fusion techniques to achieve better classification accuracy.

4.1. Research Motivation

With the massive advancement of modern technology, the application areas of human action recognition are spreading at a high speed in the fields of computer vision, image processing, and human–machine or human–object interaction. Various methods proposed for human action recognition using RGB [20,24,25,29–31], depth [32–35], and skeleton [42–68] dataset as described in Sections 1 and 3. However, it is still a challenging research topic to provide an effective and efficient method for human action classification. Many methods provided excellent and efficient ways of discriminating human activity using skeleton dataset. Even though the suggested methods facilitated great performance for human activity recognition using skeleton dataset, they suffered from several scarcities as illustrated in Section 3.3. The major concerns considered that most of the methods discriminated the human actions with accuracy below 90%. The weakness of the previous methods, for example, the inability to capture adequate intra-frame and inter-frame variation while representing the skeleton sequences into spatial format lessened the overall performance.

While we perform any meaningful action, for instances, drawing a circle, it has three different views along XY, YZ, and ZX planes that provide the spatial as well as temporal information. To capture the actual spatial views such as the circular shape of circle drawing, we connect lines between joints in adjacent frames. The temporal information is preserved by using color information that varies in every frame with the temporal changes. The line between joints in two neighboring frames with different colors defines the velocity of joints movement. By combining both spatial and temporal information, we generate spatio-temporal image which contains sufficient discriminative properties to perform the recognition.

4.2. System Architecture

Figure 5 depicts the overall architecture of the proposed system. There are four major modules: (i) spatio-temporal image formation, (ii) knowledge transfer, (iii) fusion, and (iv) classification. First, we convert the skeleton joints into a spatial format called STIF by covering both spatial information and temporal changes for the three different views XY, YZ, and ZX planes (front, side, and top views). Then the images are passed through the pareto frontier pretrained model referred to as the backbone network (MobileNetV2, DenseNet121, and ResNet18) trained with the ImageNet dataset. A fully connected layer is attached to each branch of the backbone network for extracting discriminative features. The generated features obtained from three different views (f_{xy}, f_{yz}, and f_{zx}) are fused in three different manners. Again, we conduct a fully connected layer to reduce the dimensionality of the features map (f_{xyz}). Finally, a fully connected (fc) and a softmax (sf) layers are added to perform the classification.

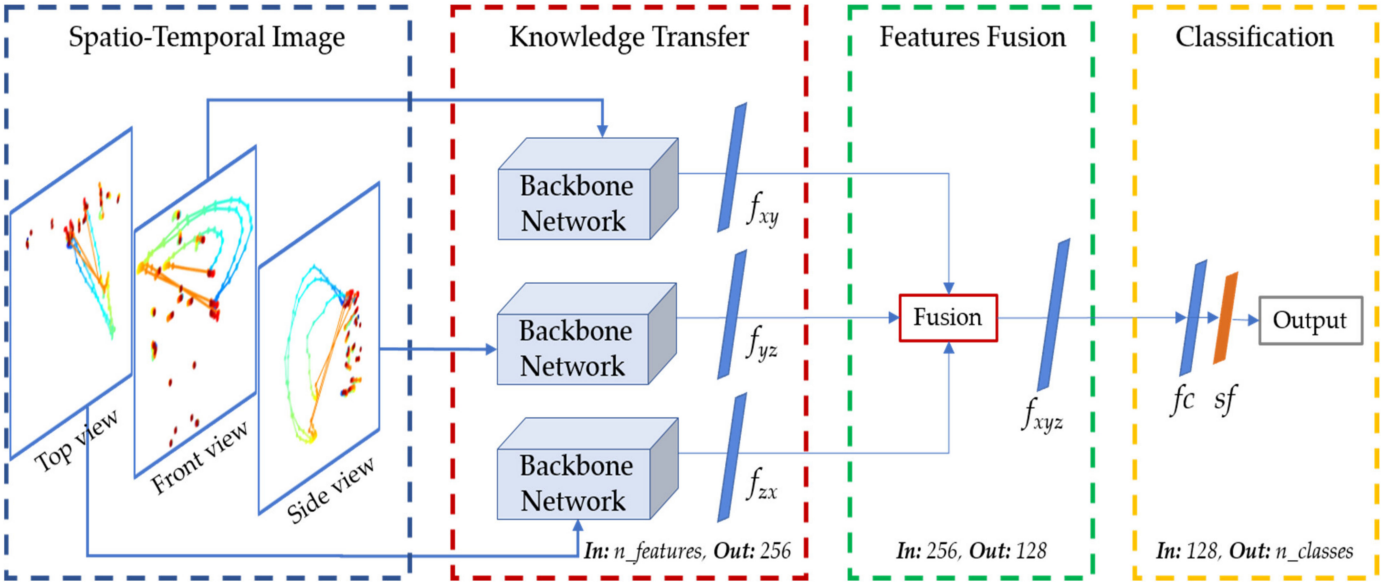


Figure 5. Architecture of the proposed system.

The input to the knowledge transfer module is the STIF representation of skeleton sequence which is indicated by n_features. Each branch including the front, side, and top of the backbone network produced 256-dimensional richer features that are fused in the fusion module. The input to the fusion module is the 256-dimensional features which are again passed through two fully connected layers and a softmax layer. The first fully connected layer reduces the 256-dimensional features to the 128-dimensional feature map. Finally, the second fully connected layer generates 27 output probabilities by following a softmax layer. The details of each module are described in the next sections.

4.2.1. Spatio-Temporal Image Formation (STIF)

Human action consists of a sequence of frames in which the spatial and temporal information changes occur over time while performing an action. The sequence of frames can be RGB, depth, skeleton, or any other format. In the proposed method, we consider only the 3D skeleton joints information for human action recognition. Skeleton joints information is usually encoded in a 3-dimensional coordinate system. Most of the devices invented for capturing skeleton joints information of the human body consider only twenty joints, specifically Microsoft Kinect captures skeleton data with twenty joint coordinates values along X, Y, and Z axes in each frame. The positions of the joints change from frame to frame as the action is rendered. We observe the changes of both the spatial and temporal information of the joint positions and represent them into spatio-temporal image. To generate the spatio-temporal image first, we map all the twenty joints in a frame with the same color in the jet color map [17] and then change the colors as the time step passed. Finally, the STIF is created by connecting lines between joints in adjacent frames subsequently.

Let us consider two joints, for example, $A(X_{i,j}, Y_{i,j})$ and $B\left(X_{(i+1),j}, Y_{(i+1),j}\right)$ along XY plane at two adjacent frames in an action where i indicates the index of frame and j represents index of joints (in our case j = 1, 2, . . . , 20). The spatio-temporal image representation of an action along XY plane (front view) can be obtained by joining line between A and B using Equation (1).

$$Y - Y_{i,j} = m\left(X - X_{i,j}\right) \tag{1}$$

where m is the slope of the line passing through the points A and B given by Equation (2).

$$m = \frac{Y_{(i+1),j} - Y_{i,j}}{X_{(i+1),j} - X_{i,j}} \tag{2}$$

Similarly, we get the spatio-temporal image representation of an action along YZ plane (side view) and ZX plane (top view).

To map the spatial and temporal changes of joints information in spatial format such as image format for an action, we use the jet color map. First, we generate the jet colors information with length equal to the number of frames in an action as given in Equation (3).

$$Colors = JET(length(action)) \tag{3}$$

Figure 6 shows the jet colors in 3D space with bar chart that represents the variation of colors starting from blue to red.

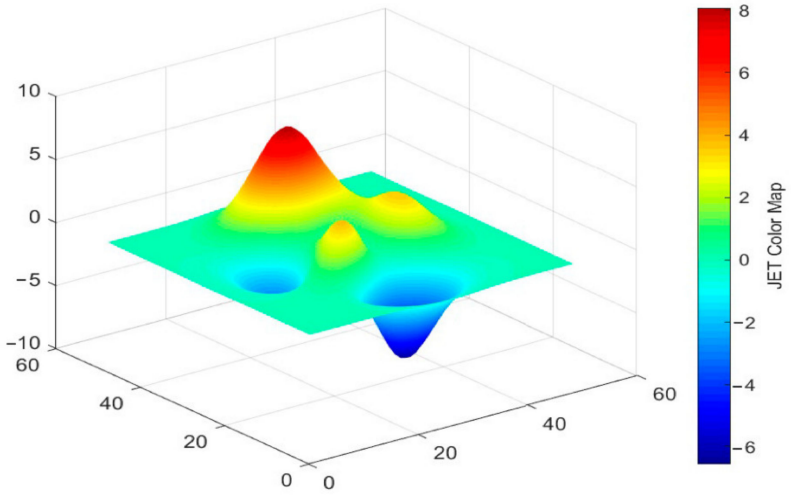


Figure 6. Jet colors generation process.

The generated colors are mapped by putting pixels for each joint as well as for the line passing through current and next frames. Equations (4) and (5) indicate the joints and line mapping with different colors to maintain spatial and temporal changes.

$$jointsMapping = putPixel(A, B, Color) \tag{4}$$

$$lineMapping = drawLine(A, B, Color) \tag{5}$$

The final spatio-temporal image is obtained by the mapping combination of joints and line as given in Equation (6).

$$spatioTemporalImage = concateMapping(jointsMapping, lineMapping) \tag{6}$$

The detailed of the STIF representation of human skeleton joints is summarized in Algorithm 1. The inputs to the algorithm are a sequence of frames containing twenty joint values along X, Y, and Z axes and different colors with the same length as the number of frames minus 1. The proposed method first computes the spatio-temporal image between two adjacent frames subsequently and finally combined all of them to generate the ultimate image.

Figure 7a–c shows the spatio-temporal image representation of *'Clap'* action in UTD-MHAD dataset along XY, YZ, and ZX planes (front, side, and top views) in which 1st frame indicates the mapping between frames at positions 1 and 2 in the action. The mapping of frames from positions 1 to $(i + 1)$ be shown as ith frame and similarly $(i + k)$th frame

is the combined mapping from frames at positions 1 to ($i + k + 1$). The nth frame is the final spatio-temporal representation that concatenates all the previous mapping in an action. Figure 8a–c depicts the spatio-temporal visualization of the '*HighArmWave*' action in MSR-Action3D dataset along XY, YZ, and ZX planes (front, side, and top views).

Algorithm 1. Steps in spatio-temporal image representation from 3D human skeleton joints

```
1.  spatioTemporalImage = spatioTemporalFormation(V, C)
2.  //Input: Sequence of frames (V), Different Colors(C).
3.  //Output: Spatio-temporal representation of an action.
4.  f = readSequence (V)
5.  n = f.lengthOfSequence
6.  for i = 1:n−1 do      //Loop over all frames in a sequence.
7.          currentFrame = f(i)
8.          nextFrame = f(i+1)
9.          color = C(i)
10.         for j = 1:20 do      //Loop over 20 skeleton joints.
11.                 jointsInCurentFrame = currentFrame(j)
12.                 jointsInNextFrame = nextFrame (j)
13.                 jointsMapping = putPixel(jointsInCurrentFrame, jointsInNextFrame, color)
14.                 lineMapping = drawLine(jointsInCurrentFrame, jointsInNextFrame, color)
15.                 spatioTemporalImage = concateMapping(jointsMapping, lineMapping)
16.         end for
17. end for
```

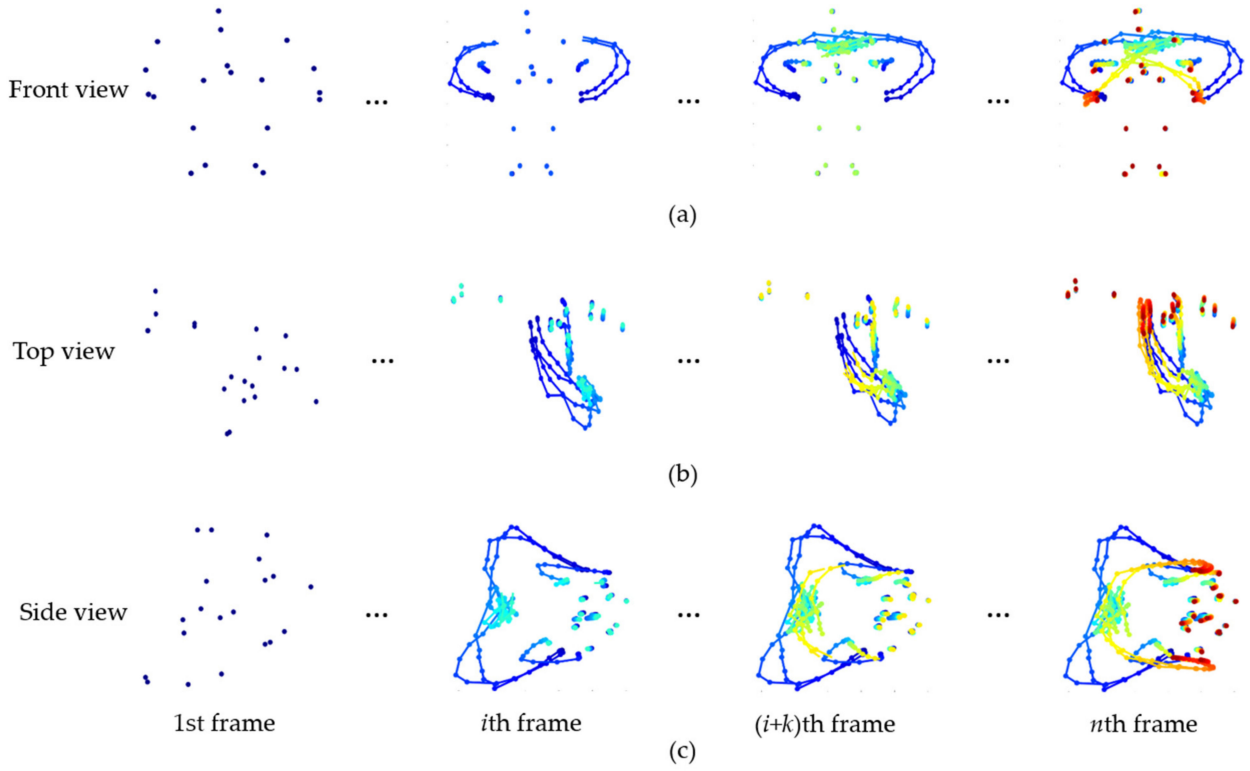


Figure 7. Spatio-temporal image formation (STIF) representation of *'Clap'* action in UTD-MHAD dataset; (**a**) front view (along XY plane), (**b**) side view (along YZ plane), and (**c**) top view (along ZX plane).

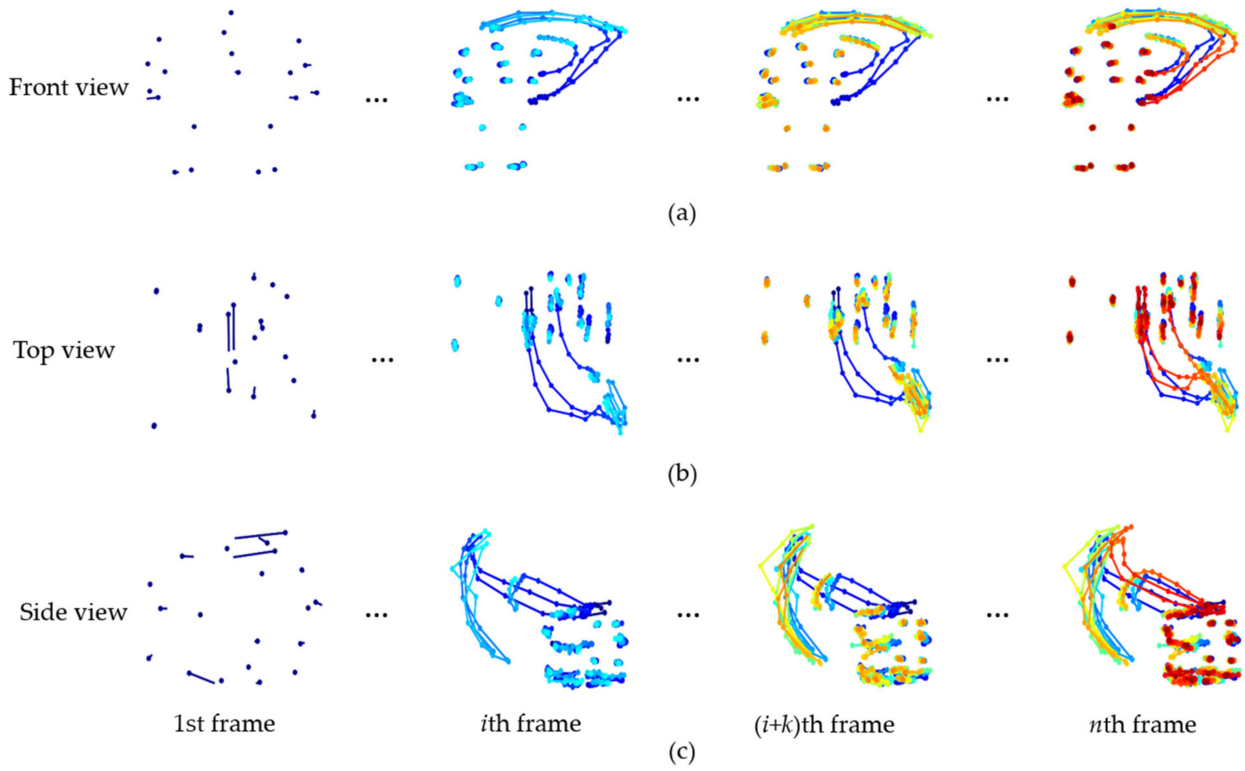


Figure 8. STIF representation of *'HighArmWave'* action in MSR-Action3D dataset; (**a**) front view (along *XY* plane), (**b**) side view (along *YZ* plane), and (**c**) top view (along *ZX* plane).

Due to the insufficiency of the dataset, we increase the dataset by using data augmentation such as rotation and scaling. Figure 9a–c visualizes the original, rotated, and scaled data along the front view of the *'SwipeLeft'* action class.

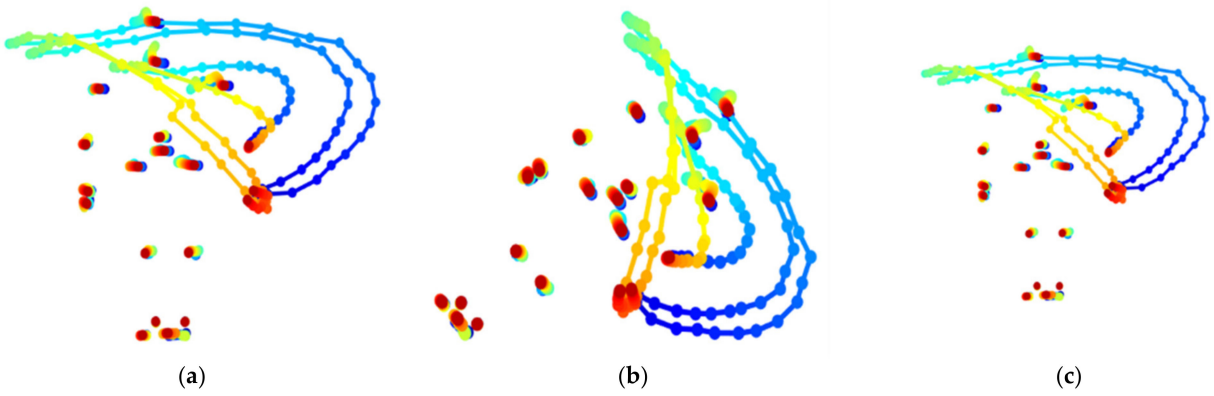


Figure 9. (**a**) Original, (**b**) rotated, and (**c**) scaled front views of '*SwipeLeft*' action data in UTD-MHAD dataset.

4.2.2. Knowledge Transfer

The benchmark human skeleton dataset is limited in volume, for instances, UTD-MHAD and MSR-Action3D datasets having fewer sequences. Even though we adopt data augmentation techniques such as scaling and rotation, it still very few in number for training deep learning models. Thus, we use the pretrained models trained with the ImageNet dataset and then the pretrained knowledge is transferred along with fully connected layers to extract the meaningful features from the UTD-MHAD and MSR-Action3D datasets for human activity recognition. We introduce three pretrained models; MobileNetV2, DenseNet121, and ResNet18 for performing the classification among the

action classes. A brief description of the pretrained models used in our experiment is provided in Section 2.1.

4.2.3. Features Fusion and Classification

As described in the previous section, we consider pre-trained models to extract the discriminative features from the spatiotemporal images. We integrate a fully connected layer after the backbone model to reduce the dimensionality of the feature map. Then we apply several fusion mechanisms to compute the better classification accuracy. The fusion techniques allow us to integrate feature maps obtained from different branches of the backbone networks as shown in Figure 5.

Three different fusion techniques are introduced for comparing the performance of the proposed system. The first method is the straightforward average of the feature maps defined as f_{avg} and given in Equation (7).

$$f_{avg}(i) = \frac{1}{3}\sum[f_{xy}(i),\ f_{yz}(i),\ f_{zx}(i)] \tag{7}$$

where $i = 1, 2, \ldots, 256$ is the dimensionality of the feature map.

Each branch of the backbone network along with the fully connected layer generates 256-dimensional features namely as f_{xy}, f_{yz}, and f_{zx}. These feature maps are averaged to pass a 256-dimensional feature map through a fully connected layer to produce the feature map called f_{xyz} of size 256. This 256-dimensional feature map is then passed through a fully connected layer and a softmax layer to get the final output.

One of the most popular fusion methods is the element-wise multiplication of the feature maps defined as f_{mul} in Equation (8). Since three features with the same dimensionality (256-dimensional) are generated from the front, side, and top views images, the dimension of the resultant feature map is 256.

$$f_{mul}(i) = \prod[f_{xy}(i),\ f_{yz}(i),\ f_{zx}(i)] \tag{8}$$

where $i = 1, 2, \ldots, 256$ is the dimensionality of the feature map.

The last fusion technique is the element-wise maximization of the feature maps that yields a 256-dimensional feature map defined as f_{max} in Equation (9).

$$f_{max}(i) = \max[f_{xy}(i),\ f_{yz}(i),\ f_{zx}(i)] \tag{9}$$

where $i = 1, 2, \ldots, 256$ is the dimensionality of the feature map.

Among the above mentioned three fusion techniques, element-wise maximization is more robust compare to straight-forward average and multiplication.

5. Experimental Results

The experimental environment, training, and testing configurations, performance evaluation, and comparison are elaborately described in this section. First, we present the hardware and software used to implement the proposed system. Then, the parameters setting of training the deep learning models and performance are described in detail.

5.1. Training and Testing Configurations

We use Intel (R) Core (TM) i7 CPU, GeForce GTX 1080 GPU, Windows 10, and Linux 16.04 to accomplish the overall experiment. We conduct MATLAB 2019b and Python 3.5 as the programming language. We perform the preprocessing (spatio-temporal image representation) using MATLAB 2019b and implement deep learning using PyTorch toolbox in Python 3.5. To use the pre-trained model, we resize the STIF images generated from UTD-MHAD and MSR-Action3D datasets into $224 \times 224 \times 3$.

We evaluate the proposed system by applying it on the UTD-MHAD and MSR-Action3D datasets as described in Section 2.2. The dataset is partitioned into training and testing data as described in Section 2.2. At the initial step, we set the learning rate as 0.001,

and the learning rate decreases at the interval of 10 epoch with a factor of 0.9. We set the batch size to 16 that shuffles during the reading. The Adam optimizer is conducted for optimization purposes with a momentum value of 0.999. The training process continues until 100 epochs. Table 1 lists the hardware, software, and parameters used for training and testing the proposed system.

Table 1. Hardware, software, and parameters configurations.

Parameters	Values
Hardware	Intel (R) Core(TM) i7 CPU, GeForce GTX 1080 GPU
Software	Windows 10, Linux 16.04, MATLAB 2019b, Python 3.5
Initial learning rate	0.001
Learning rate dropping factor	0.9
Learning rate dropping period	10
Optimizer	Adam

5.2. Performance Evaluations

We calculate classification accuracies for emphasizing the effectiveness and efficiency of the proposed system using Equation (10) for each action.

$$\text{Accuracy}(\%) = \frac{\text{Total Correctly Predicted Observations}}{\text{Total Number Observations}} \times 100 \quad (10)$$

We carry out the evaluation process in six different ways to provide more clarification about the discriminability of human activity using the proposed method. We initially train the deep learning models with three different views along XY, YZ, and ZX planes separately and compute the test results. Then the features extracted from three different views are fused in three different ways and enumerate the classification results. We conduct three well-known deep learning models including MobileNetV2, DenseNet121, and ResNet18 to represent the robustness of the proposed system.

Table 2 enlists human activity recognition results with different configurations of data and models using the UTD-MHAD dataset. The proposed method achieves classification accuracies about 97.29% and 98.21% for DenseNet121 using front and top views of the UTD-MHAD dataset respectively. However, while we apply side view data, MobileNetV2 (96.06%) shows better performance than DenseNet121 (95.98%) and ResNet18 (93.17%).

Table 2. Human activity recognition with different modality and models using UTD-MHAD dataset.

Methods	MobileNetV2	DenseNet121	ResNet18
Front view (XY plane)	96.52%	97.29%	95.29%
Side view (YZ plane)	96.06%	95.98%	93.17%
Top view (ZX plane)	97.08%	98.21%	94.67%
Fusion (f_{avg})	97.98%	98.51%	95.93%
Fusion (f_{mul})	98.93%	98.89%	97.18%
Fusion (f_{max})	98.89%	99.65%	98.80%

As we mentioned in the previous discussion, we fuse the features obtained from the front, side, and top views dataset using three techniques to improve the classification accuracies in MobileNetV2, DenseNet121, and ResNet18. While fusing the features by applying the conventional average and maximization fusion techniques, the proposed method gets 98.51% and 99.65% highest classification accuracies using DeNseNet121. The element-wise multiplication of the features provides 98.93% discrimination accuracies using MobileNetV2. By considering the recognition results in Table 2, it is clear that DenseNet121 outperforms both MobileNetV2 and ResNet18 because it can generate deeper features from the STIF representation of the skeleton data.

We again investigate the performance of human activity recognition on the MSR-Action3D dataset using MobileNetV2, DenseNet121, and ResNet18 for showing the robustness of the proposed method. Table 3 shows the human action recognition results on the MSR-Action3D dataset. Like the UTD-MHAD dataset, we obtain better accuracies of about 94.83% for DenseNet121 compared with MobileNetV2 (93.83%) and ResNet18 (93.04%) on the front view dataset. A similar trend appears in the case of the top view dataset in which DenseNet121 classifies human action with an accuracy of about 93.00% while MobileNetV2 and ResNet18 secure accuracies of about 91.67% and 92.08% respectively.

Table 3. Human activity recognition with different modality and models using MSR-Action3D dataset.

Methods	MobileNetV2	DenseNet121	ResNet18
Front view (XY plane)	93.83%	94.83%	93.08%
Side view (YZ plane)	91.25%	91.25%	92.50%
Top view (ZX plane)	91.67%	93.00%	92.08%
Fusion (f_{avg})	95.42%	96.67%	97.50%
Fusion (f_{mul})	95.50%	96.67%	96.25%
Fusion (f_{max})	96.00%	98.75%	97.08%

ResNet18 obtains the highest accuracy about 97.50% using average fusion on the MSR-Action3D dataset. For both the multiplication and maximization fusions, DenseNet121 secures about 96.67% and 98.75% discrimination results.

To render more clarification about the performance of the proposed method for each backbone model along with each experiment, we visualize the graphical results shown in Figure 10a,b. From the Figure 10, we can argue that DenseNet121 works much better in comparison with MobileNetV2 and ResNet18 for both UTD-MHAD and MSR-Action3D datasets. At the same time, it can be stated that the three fusion techniques contribute a lot to improve the classification performance of the proposed system in case of either UTD-MHAD or MSR-Action3D dataset.

To examine the complexities such as memory consumption, operational requirements, and time complexity, we express the parameters in millions, floating-point operation per seconds (FLOPS) in giga, and time in second as listed in Table 4. The inverted-residual blocks in MobileNetV2 reduce the parameters as well as the number of operations than DenseNet121 and ResNet18. The operations increase as the features are extracted from deeper in DenseNet121. The pre-trained MobileNetV2, DenseNet121, and ResNet18 require 0.00129, 0.00235, and 0.00108 s respectively for running an action.

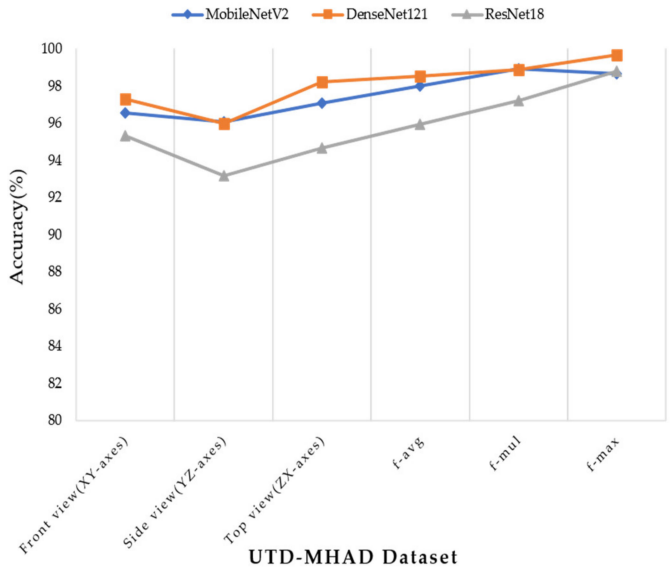
Table 4. Complexity analysis of different models.

Models	Parameters (M)	FLOPs (G)	Time (s)
MobileNetV2	2.56	0.33	0.00129
DenseNet121	7.22	2.90	0.00235
ResNet18	11.31	1.82	0.00108

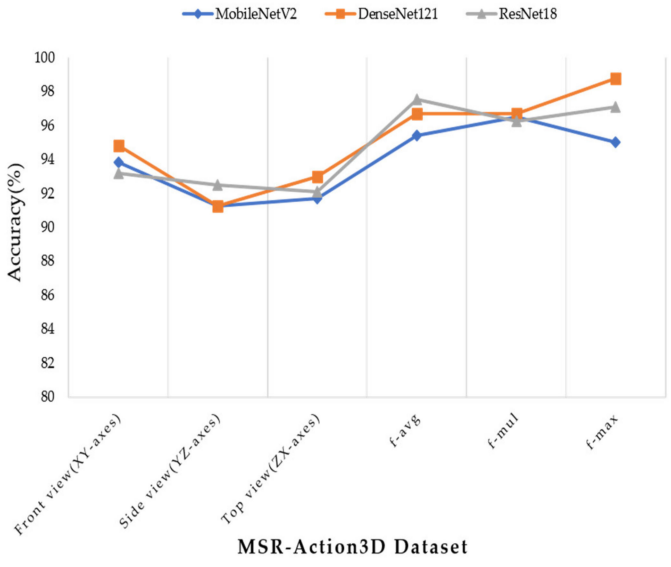
The further inquisition of performance on different actions individually explains that the proposed method assures the recognition accuracy above 90% for the UTD-MHAD dataset. From Figure 11a, it can be said that the action classes *'SwipeLeft'*, *'SwipeRight'*, *'Clap'*, *'Throw'*, *'ArmCross'*, *'Boxing'*, *'BaseballSwing'*, *'TennisSwing'*, *'TennisServe'*, *'Push'*, *'Knock'*, *'Catch'*, *'PickUpandThrow'*, *'Jog'*, *'Walk'*, *'SitToStand'*, *'StandToSit'*, *'Lunge'*, and *'Squat'* secure highest classification accuracies with any one of the three pre-trained models.

The recognition results decrease to 58.33%, 91.67%, and 83.33% using MobileNetV2, DenseNet121, and ResNet18 in MSR-Action3D dataset for *'HandCatch'* action due to the irregularities in the dataset. On top of that, the overall discrimination performance is satisfactory for any other action classes in the MSR-Action3D dataset with three pre-trained models. Figure 11b illustrates the classification results for individual classes in the

MSR-Action3D dataset with MobileNetV2, DenseNet121, and ResNet18. The reduction of accuracies in some classes of action such as *'HandCatch'* in the MSR-Action3D dataset happens due to the irregular movement in few points of the sequences.



(**a**)

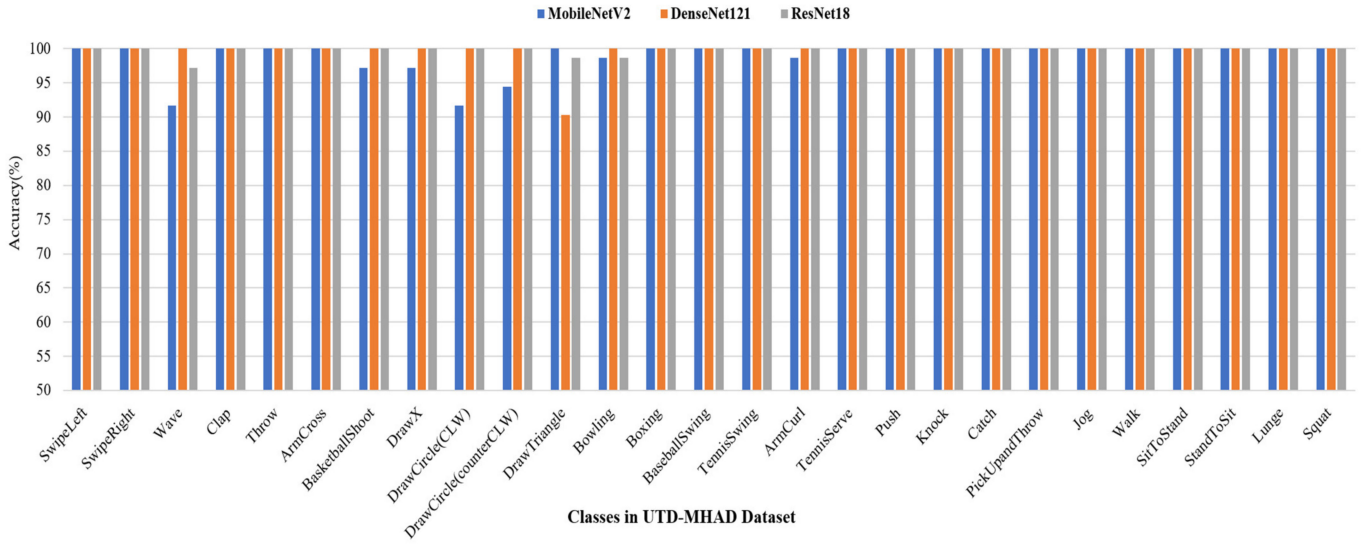


(**b**)

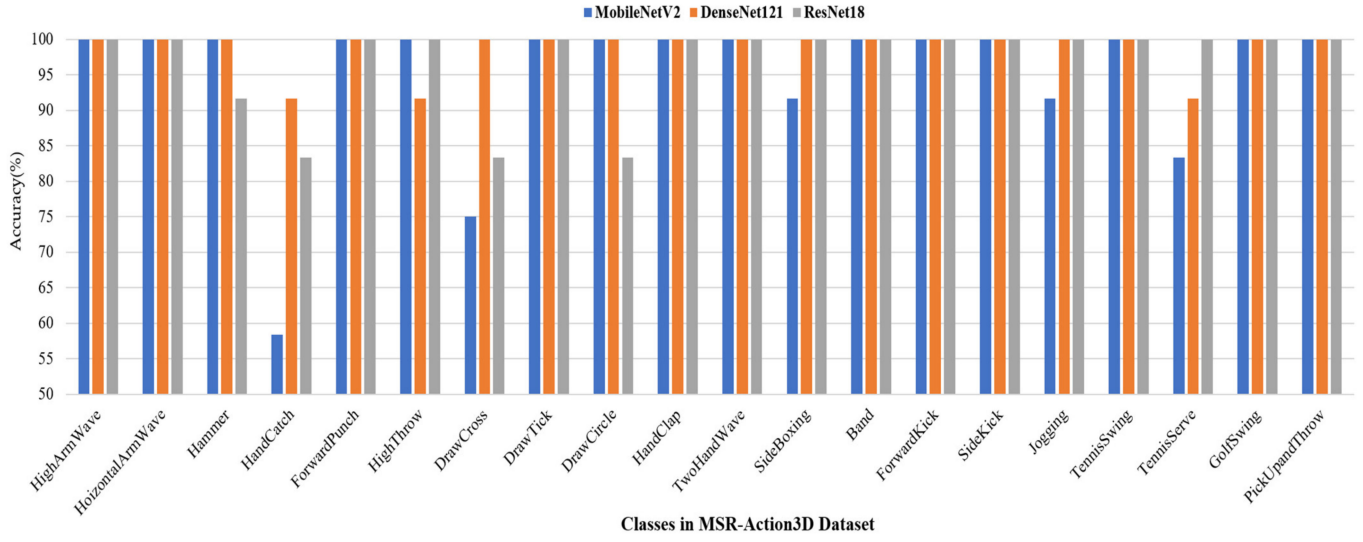
Figure 10. Visual comparisons of different views and fusions of dataset and models; (**a**) UTD-MHAD and (**b**) MSR-Action3D datasets.

5.3. State-of-the-Art Comparisons

We have already described that the proposed method fulfills the desired objective with better performance compared to the state-of-the-art methods for human action recognition using 3D skeleton dataset. For the fairness of the experimental results, we have separately mentioned the strategies with classifiers to compare the recognition accuracies between the proposed system and the prior works for UTD-MHAD and MSR-Action3D datasets as depicted in Tables 5 and 6. Among the six different experimental results (front, side, top, average, multiplication, and maximization), we only list the best one for each model.



(**a**)



(**b**)

Figure 11. Classification performance of the proposed system based on individual classes in (**a**) UTD-MHAD and (**b**) MSR-Action3D datasets.

Table 5. Performance comparisons of human action recognition using UTD-MHAD dataset.

Methods	Accuracy
STIF with MobileNetV2	98.93%
STIF with DenseNet121	99.65%
STIF with ResNet18	98.80%
BOP with ELM [51]	95.30%
TPSMM with CNN [65]	88.10%
JDM with CNN [66]	93.26%
SOS with CNN [67]	86.97%
JTM with CNN [68]	85.81%

Table 6. Performance comparisons of human action recognition using MSR-Action3D dataset.

Methods	Accuracy
STIF with MobileNetV2	96.00%
STIF with DenseNet121	98.75%
STIF with ResNet18	97.08%
BOP with ELM [51]	91.90%
MIMTL with SVM [52]	93.63%
RSS with RNN [53]	94.48%
SOFGF with CNN [62]	97.25%

The previous method in [51] attains about 95.30% outcomes for human activity recognition using the UTD-MHAD dataset which is the highest among the references [51,64,66–68]. The proposed method achieves accuracy of 99.65%, the highest experimental results, with STIF representation of UTD-MHAD skeleton dataset using DenseNet121 which is around 4% greater than the methods using TPSMM [64], JDM [66], SOS [67], and JTM [68] with CNN. Even though we apply MobileNetV2 and ResNet18, our method secures about 3% better accuracy than the state-of-the-art methods.

We further provide the comparative results for MSR-Action3D skeleton dataset to boost up on the suggested system. The proposed method ensures 98.75% accuracy with DenseNet121 which is the better than the defending best accuracy of 97.25% accuracy achieved by SOFGF with CNN [62]. The other methods [51–53] classified human action with accuracies of about 4% lower than the proposed method. On the other hand, our method can obtain about 96.00% and 97.08% recognition accuracies with MobileNetV2 and ResNet18 correspondingly.

6. Analysis and Discussion

As the prior works partially lack capturing the spatial and temporal variations explicitly, we frankly provide a spatio-temporal representation of 3D skeleton joint values along the front, side, and top views. We adopt data augmentation (rotation and scaling) to increase the experimental data due to the data deficiency in UTD-MHAD and MSR-Action3D datasets. By considering the design and time complexities of the deep learning model, we fine-tune the well-known pretrained models such as MobileNetV2, DenseNet121, and ResNet18 for the human action recognition in the proposed method. We also investigate the classification results with three different fusion techniques to improve the performance.

The spatio-temporal image obtained by assembling joints mapping and line mapping between the same joints in two consecutive frames can maintain the spatial information and temporal changes with different colors in the jet color map. The variations of spatial information as well as temporal information of each action are easily distinguishable for both UTD-MHAD and MSR-Action3D datasets. The effect of the network architecture doesn't affect more on the performance of the proposed method since the lightweight MobileNetV2 works well comparable with heavy-weight DenseNet121 and ResNet18.

The fusion techniques also accelerate a bit on the performance of the proposed method. Most of the cases, element-wise maximization ensures the highest classification results compared to average and multiplication strategies.

The classification accuracies of individual classes as shown in Figure 11a,b indicate that the action performed by any parts of the human body can be classified effortlessly using deep learning with the proposed spatio-temporal image formation method.

However, the spatio-temporal representation of the 3D skeleton joints is fully confined to the regularities of the frames in an action. The irregular frames generate indiscipline spatial and temporal information that makes it more complicated to predict the correct classes of action.

7. Conclusions

In this paper, a new approach for human action recognition is suggested using deep learning with spatio-temporal image formation from 3D skeleton joints. We analyze the 3D skeleton joints and propose to encode the spatio-temporal image from 3D skeleton joints by mapping the line between the same joints in two neighboring frames. The spatial and temporal information is extracted by preserving the shape of the action and joining the line with different colors along with the temporal changes. We deploy pretrained deep learning models to evaluate the usefulness of spatio-temporal representation of the proposed method. We accomplish the experiments in two separate ways: (i) with individual views (front, side, and top views) and (ii) with fusion mechanisms (average, multiplication, and maximization). The experimental results with individual views show that the front view dataset works well. While applying the features fusion, maximization improves the recognition rate significantly.

We also compare the recognition accuracies with three deep learning models to reveal the sturdiness of our work. The features mined from the spatio-temporal image are invariant to views and speed of the action. Thus, both the pretrained lightweight and heavy weight deep learning models can individualize the actions without any difficulties.

Even though we perform the experiments with individual views along *XY*, *YZ*, and *ZX* planes, the discrimination accuracies of the proposed method outperform the state-of-the-art works with UTD-MHAD and MSR-Action3D benchmark skeleton datasets. The investigation of the experimental results with three different fusion methods are also conducted to bring out the most perfect approach. The overall experimental results of the proposed system using pretrained deep learning with UTD-MHAD and MSR-Action3D skeleton datasets shows better performance.

Author Contributions: Conceptualization, analysis, methodology, manuscript preparation, and experiments, N.T.; data curation, writing—review and editing, N.T., M.K.I. and J.-H.B.; supervision, J.-H.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the GRRC program of Gyeonggi province [GRRC Aviation 2017-B04, Development of Intelligent Interactive Media and Space Convergence Application System].

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We would like to acknowledge Korea Aerospace University with much appreciation for its ongoing support to our research.

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

References

1. Eum, H.; Yoon, C.; Lee, H.; Park, M. Continuous human action recognition using depth-MHI-HOG and a spotter model. *Sensors* **2015**, *15*, 5197–5227. [CrossRef]
2. Dawar, N.; Kehtarnavaz, N. Continuous detection and recognition of actions of interest among actions of non-interest using a depth camera. In Proceedings of the IEEE International Conference on Image Processing, Beijing, China, 17–20 September 2017. [CrossRef]
3. Chu, X.; Ouyang, W.; Li, H.; Wang, X. Structured feature learning for pose estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016. [CrossRef]
4. Ziaeefard, M.; Bergevin, R. Semantic human activity recognition: A literature review. *Pattern Recognit.* **2015**, *48*, 2329–2345. [CrossRef]
5. Chaaraoui, A.A.; Padilla-Lopez, J.R.; Ferrandez-Pastor, F.J.; Nieto-Hidalgo, M.; Florez-Revuelta, F. A vision-based system for intelligent monitoring: Human behaviour analysis and privacy by context. *Sensors* **2014**, *14*, 8895–8925. [CrossRef] [PubMed]
6. Wei, H.; Laszewski, M.; Kehtarnavaz, N. Deep Learning-Based Person Detection and Classification for Far Field Video Surveillance. In Proceedings of the 13th IEEE Dallas Circuits and Systems Conference, Dallas, TX, USA, 2–12 November 2018. [CrossRef]
7. Zhu, H.; Vial, R.; Lu, S. Tornado: A spatio-temporal convolutional regression network for video action proposal. In Proceedings of the CVPR, Venice, Italy, 22–29 October 2017. [CrossRef]

8. Wen, R.; Nguyen, B.P.; Chng, C.B.; Chui, C.K. In Situ Spatial AR Surgical Planning Using projector-Kinect System. In Proceedings of the Fourth Symposium on Information and Communication Technology, Da Nang, Vietnam, 5–6 December 2013. [CrossRef]
9. Azuma, R.T. A survey of augmented reality. *Presence: Teleoperators Virtual Environ.* **1997**, *6*, 355–385. [CrossRef]
10. Jalal, A.; Kamal, S.; Kim, D. A Depth Video Sensor-Based Life-Logging Human Activity Recognition System for Elderly Care in Smart Indoor Environments. *Sensors* **2014**, *14*, 11735–11759. [CrossRef]
11. Zheng, Y.; Ding, X.; Poon, C.C.Y.; Lo, B.P.L.; Zhang, H.; Zhou, X.; Yang, G.; Zhao, N.; Zhang, Y. Unobtrusive Sensing and Wearable Devices for Health Informatics. *IEEE Trans. Biomed. Eng.* **2014**, *61*, 1538–1554. [CrossRef] [PubMed]
12. Chen, L.; Ma, N.; Wang, P.; Li, J.; Wang, P.; Pang, G.; Shi, X. Survey of pedestrian action recognition techniques for autonomous driving. *Tsinghua Sci. Technol.* **2020**, *25*, 458–470. [CrossRef]
13. Bloom, V.; Makris, D.; Argyriou, V. G3D: A gaming action dataset and real time action recognition evaluation framework. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Providence, RI, USA, 16–21 June 2012. [CrossRef]
14. Kim, S.T.; Lee, H.J. Lightweight Stacked Hourglass Network for Human Pose Estimation. *Appl. Sci.* **2020**, *10*, 6497. [CrossRef]
15. Tasnim, N.; Islam, M.; Baek, J.H. Deep Learning-Based Action Recognition Using 3D Skeleton Joints Information. *Inventions* **2020**, *5*, 49. [CrossRef]
16. Sun, Z.; Liu, J.; Ke, Q.; Rahmani, H. Human Action Recognition from Various Data Modalities: A Review. *arXiv* **2020**, arXiv:2012.11866.
17. Pham, H.H.; Salmane, H.; Khoudour, L.; Crouzil, A.; Zegers, P.; Velastin, S.A. Spatio–temporal image representation of 3D skeletal movements for view-invariant action recognition with deep convolutional neural networks. *Sensors* **2019**, *19*, 1932. [CrossRef]
18. Arivazhagan, S.; Shebiah, R.N.; Harini, R.; Swetha, S. Human action recognition from RGB-D data using complete local binary pattern. *Cogn. Syst. Res.* **2019**, *58*, 94–104. [CrossRef]
19. Abdul-Azim, H.A.; Hemayed, E.E. Human action recognition using trajectory-based representation. *Egypt. Inform. J.* **2015**, *16*, 187–198. [CrossRef]
20. Lowe, D.G. Object recognition from local scale-invariant features Computer Vision. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Corfu, Greece, 20–25 September 1999. [CrossRef]
21. Bay, H.; Ess, A.; Tuytelaars, T.; Van Gool, L. Speeded-up robust features (SURF). *Comput. Vis. Image Underst.* **2008**, *110*, 346–359. [CrossRef]
22. Yang, X.; Zhang, C.; Tian, Y. Recognizing actions using depth motion maps-based histograms of oriented gradients. In Proceedings of the 20th ACM International Conference on Multimedia, Nara, Japan, 27–31 October 2012. [CrossRef]
23. Oreifej, O.; Liu, Z. HON4D: Histogram of oriented 4D normals for activity recognition from depth sequences. In Proceedings of the Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013. [CrossRef]
24. Mahjoub, A.B.; Atri, M. Human action recognition using RGB data. In Proceedings of the 11th International Design & Test Symposium (IDT), Tunisia, Hammamet, 18–20 December 2016. [CrossRef]
25. Al-Akam, R.; Paulus, D. Local and Global Feature Descriptors Combination from RGB-Depth Videos for Human Action Recognition. In Proceedings of the ICPRAM, Funchal, Madeira, Portugal, 16–18 January 2018. [CrossRef]
26. Li, Y.D.; Hao, Z.B.; Lei, H. Survey of convolutional neural network. *J. Comput. App.* **2016**, *36*, 2508–2515.
27. Sherstinsky, A. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Phys. D Nonlinear Phenom.* **2020**, *404*, 132306. [CrossRef]
28. Simonyan, K.; Zisserman, A. Two-stream convolutional networks for action recognition in videos. *arXiv* **2014**, arXiv:1406.2199. [CrossRef]
29. Zhang, B.; Wang, L.; Wang, Z.; Qiao, Y.; Wang, H. Real-time action recognition with deeply transferred motion vector cnns. *IEEE Trans. Image Proc.* **2018**, *27*, 2326–2339. [CrossRef] [PubMed]
30. Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning spatiotemporal features with 3d convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015. [CrossRef]
31. Donahue, J.; Hendricks, L.A.; Guadarrama, S.; Rohrbach, M.; Venugopalan, S.; Saenko, K.; Darrell, T. Long-term recurrent convolutional networks for visual recognition and description. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015.
32. Chen, C.; Jafari, R.; Kehtarnavaz, N. Action recognition from depth sequences using depth motion maps-based local binary patterns. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 5–9 January 2015. [CrossRef]
33. Wu, D.; Shao, L. Silhouette analysis-based action recognition via exploiting human poses. *IEEE Trans. Circuits Syst. Video Technol.* **2012**, *23*, 236–243. [CrossRef]
34. Trelinski, J.; Kwolek, B. Convolutional Neural Network-Based Action Recognition on Depth Maps. In Proceedings of the International Conference on Computer Vision and Graphics, Warsaw, Poland, 17–19 September 2018.
35. Wang, P.; Li, W.; Gao, Z.; Zhang, J.; Tang, C.; Ogunbona, P.O. Action recognition from depth maps using deep convolutional neural networks. *IEEE Trans. Hum. Mach. Syst.* **2015**, *46*, 498–509. [CrossRef]
36. Torchvision Master. Available online: https://pytorch.org/docs/stable/torchvision/models.html (accessed on 1 December 2020).

37. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018. [CrossRef]
38. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, Honolulu, HI, USA, 21–26 July 2017. [CrossRef]
39. Feichtenhofer, C.; Pinz, A.; Wildes, R.P. Spatiotemporal residual networks for video action recognition. *arXiv* **2016**, arXiv:1611.02155.
40. Chen, C.; Jafari, R.; Kehtarnavaz, N. UTD-MHAD: A Multimodal Dataset for Human Action Recognition Utilizing a Depth Camera and a Wearable Inertial Sensor. In Proceedings of the IEEE International Conference on Image Processing, Quebec City, QC, Canada, 27–30 September 2015. [CrossRef]
41. Li, W.; Zhang, Z.; Liu, Z. Action recognition based on a bag of 3D points. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010. [CrossRef]
42. Warchoł, D.; Kapuściński, T. Human Action Recognition Using Bone Pair Descriptor and Distance Descriptor. *Symmetry* **2020**, *12*, 1580. [CrossRef]
43. Vemulapalli, R.; Arrate, F.; Chellappa, R. Human action recognition by representing 3d skeletons as points in a lie group. In Proceedings of the IEEE conference on computer vision and pattern recognition, Columbus, OH, USA, 23–28 June 2014. [CrossRef]
44. Ke, Q.; Bennamoun, M.; An, S.; Sohel, F.; Boussaid, F. A new representation of skeleton sequences for 3d action recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, Honolulu, HI, USA, 21–26 July 2017. [CrossRef]
45. Presti, L.L.; Cascia, L.M. 3D skeleton-based human action classification: A survey. *Pattern Recognit.* **2016**, *53*, 130–147. [CrossRef]
46. Xia, L.; Chen, C.C.; Aggarwal, J.K. View invariant human action recognition using histograms of 3d joints. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Providence, RI, USA, 16–21 June 2012. [CrossRef]
47. Yang, X.; Tian, Y.L. Eigenjoints-based action recognition using naive-bayes-nearest-neighbor. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Providence, RI, USA, 16–21 June 2012. [CrossRef]
48. Zhu, Y.; Chen, W.; Guo, G. Fusing spatiotemporal features and joints for 3d action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Portland, OR, USA, 23–28 June 2013. [CrossRef]
49. Evangelidis, G.; Singh, G.; Horaud, R. Skeletal quads: Human action recognition using joint quadruples. In Proceedings of the 22nd International Conference on Pattern Recognition, Stockholm, Sweden, 24–28 August 2014. [CrossRef]
50. Zanfir, M.; Leordeanu, M.; Sminchisescu, C. The moving pose: An efficient 3d kinematics descriptor for low-latency action recognition and detection. In Proceedings of the IEEE international conference on computer vision, Sydney, Australia, 1–8 December 2013. [CrossRef]
51. Agahian, S.; Negin, F.; Köse, C. Improving bag-of-poses with semi-temporal pose descriptors for skeleton-based action recognition. *Vis. Comput.* **2019**, *35*, 591–607. [CrossRef]
52. Yang, Y.; Deng, C.; Gao, S.; Liu, W.; Tao, D.; Gao, X. Discriminative multi-instance multitask learning for 3D action recognition. *IEEE Trans. Multimed.* **2017**, *19*, 519–529. [CrossRef]
53. Du, Y.; Wang, W.; Wang, L. Hierarchical recurrent neural network for skeleton based action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 8–10 June 2015. [CrossRef]
54. Wang, H.; Wang, L. Modeling temporal dynamics and spatial configurations of actions using two-stream recurrent neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017. [CrossRef]
55. Zhu, W.; Lan, C.; Xing, J.; Zeng, W.; Li, Y.; Shen, L.; Xie, X. Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks. In Proceedings of the AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016.
56. Liu, J.; Shahroudy, A.; Xu, D.; Wang, G. Spatio-temporal lstm with trust gates for 3d human action recognition. In Proceedings of the European conference on computer vision, Amsterdam, The Netherlands, 8–16 October 2016. [CrossRef]
57. Song, S.; Lan, C.; Xing, J.; Zeng, W.; Liu, J. An end-to-end spatio-temporal attention model for human action recognition from skeleton data. In Proceedings of the AAAI conference on artificial intelligence, San Francisco, CA, USA, 4–9 February 2017.
58. Li, C.; Zhong, Q.; Xie, D.; Pu, S. Skeleton-based action recognition with convolutional neural networks. In Proceedings of the 2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), Hong Kong, 10–14 July 2017.
59. Si, C.; Jing, Y.; Wang, W.; Wang, L.; Tan, T. Skeleton-based action recognition with hierarchical spatial reasoning and temporal stack learning network. *Pattern Recognit.* **2020**, *107*, 107511. [CrossRef]
60. Zhang, S.; Liu, X.; Xiao, J. On geometric features for skeleton-based action recognition using multilayer lstm networks. In Proceedings of the 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), Santa Rosa, CA, USA, 24–31 March 2017. [CrossRef]
61. Du, Y.; Fu, Y.; Wang, L. Skeleton based action recognition with convolutional neural network. In Proceedings of the 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR), Kualalumpur Westin Hotel Kuala Lumpur, Malaysia, 3–6 November 2015. [CrossRef]

62. Liu, J.; Akhtar, N.; Mian, A. Skepxels: Spatio-temporal Image Representation of Human Skeleton Joints for Action Recognition. In Proceedings of the CVPR Workshops, Long Beach, CA, USA, 16–20 June 2019.
63. Ren, J.; Reyes, N.H.; Barczak, A.L.C.; Scogings, C.; Liu, M. An investigation of skeleton-based optical flow-guided features for 3D action recognition using a multi-stream CNN model. In Proceedings of the IEEE 3rd International Conference on Image, Vision and Computing (ICIVC), Chongqing, China, 27–29 June 2018. [CrossRef]
64. Li, B.; Dai, Y.; Cheng, X.; Chen, H.; Lin, Y.; He, M. Skeleton based action recognition using translation-scale invariant image mapping and multi-scale deep CNN. In Proceedings of the 2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), Hong Kong, China, 10–14 July 2017. [CrossRef]
65. Chen, Y.; Wang, L.; Li, C.; Hou, Y.; Li, W. ConvNets-based action recognition from skeleton motion maps. *Multimed. Tools Appl.* **2020**, *79*, 1707–1725. [CrossRef]
66. Li, C.; Hou, Y.; Wang, P.; Li, W. Joint distance maps based action recognition with convolutional neural networks. *IEEE Signal Process. Lett.* **2017**, *24*, 624–628. [CrossRef]
67. Hou, Y.; Li, Z.; Wang, P.; Li, W. Skeleton optical spectra-based action recognition using convolutional neural networks. *IEEE Trans. Circuits Syst. Video Technol.* **2016**, *28*, 807–811. [CrossRef]
68. Wang, P.; Li, Z.; Hou, Y.; Li, W. Action recognition based on joint trajectory maps using convolutional neural networks. In Proceedings of the 24th ACM international conference on Multimedia, Amsterdam, The Netherlands, 15–19 October 2016. [CrossRef]