applied sciences

MDPI

Article

Complex Human–Object Interactions Analyzer Using a DCNN and SVM Hybrid Approach

Cho Nilar Phyo [1,*], Thi Thi Zin [2] and Pyke Tin [3]

[1] Interdisciplinary Graduate School of Agriculture and Engineering, University of Miyazaki, Miyazaki 889-2192, Japan
[2] Graduate School of Engineering, University of Miyazaki, Miyazaki 889-2192, Japan; thithi@cc.miyazaki-u.ac.jp
[3] International Relation Center, University of Miyazaki, Miyazaki 889-2192, Japan; pyketin11@gmail.com
* Correspondence: nc16004@student.miyazaki-u.ac.jp; Tel.: +81-70-4447-1332

Received: 31 March 2019; Accepted: 5 May 2019; Published: 7 May 2019

Featured Application: This research can be applied to the abnormal behavior detection system for the elderly by analyzing daily activities.

Abstract: Nowadays, with the emergence of sophisticated electronic devices, human daily activities are becoming more and more complex. On the other hand, research has begun on the use of reliable, cost-effective sensors, patient monitoring systems, and other systems that make daily life more comfortable for the elderly. Moreover, in the field of computer vision, human action recognition (HAR) has drawn much attention as a subject of research because of its potential for numerous cost-effective applications. Although much research has investigated the use of HAR, most has dealt with simple basic actions in a simplified environment; not much work has been done in more complex, real-world environments. Therefore, a need exists for a system that can recognize complex daily activities in a variety of realistic environments. In this paper, we propose a system for recognizing such activities, in which humans interact with various objects, taking into consideration object-oriented activity information, the use of deep convolutional neural networks, and a multi-class support vector machine (multi-class SVM). The experiments are performed on a publicly available cornell activity dataset: CAD-120 which is a dataset of human–object interactions featuring ten high-level daily activities. The outcome results show that the proposed system achieves an accuracy of 93.33%, which is higher than other state-of-the-art methods, and has great potential for applications recognizing complex daily activities.

Keywords: complex human activities recognition; depth sensor; deep learning; multi-class SVM; object usage probability

1. Introduction

The recognition of complex daily activities and human–object interaction plays as an important role in many applications, such as monitoring systems for the elderly, for patients, for human–robot interaction, and other video surveillance systems. For monitoring the elderly living independently, monitoring systems must automatically analyze daily activities and detect abnormal behavior in order to provide assistance health-care services. Although some techniques have been developed for monitoring the elderly using wearable sensors, these devices can be a source of mental and physical discomfort. Therefore, research has concentrated on computer vision-based human action recognition (HAR). In this area, depth sensors have gained much attention because of their reasonable cost and adaptability to variable illumination. Depth sensors, such as Microsoft Kinect [1] and ASUS Xtion

Pro [2], can capture various kinds of data, such as depth images, RGB (red, green, blue) images, infrared images, and skeletal joint information for the human body.

Moreover, humans interact daily with various kinds of objects in different ways, depending on their intentions. Human–object interaction is complex, and recognizing the actions involved is a challenging task. Research into object recognition shows that a deep-learning approach achieves superior performance over other state-of-the-art techniques. Deep learning is also achieving more and more success in HAR research. This paper discusses the application of the deep-learning technology in recognizing human–object interaction. In addition, for improving results, we have built a multi-class support vector machine (multi-class SVM) using the object usage probability (OUP), which is the probability of how many times the objects have been used. Our technique involves a fusion of the results of deep learning and multi-class SVM in the final heuristics involved in interaction recognition (decision fusion). In this study, experiments were performed on the CAD-120 dataset [3] of human–object interaction, which used a depth sensor as an input device for collecting the data. This paper comprises five sections. The first describes a brief overview of the understanding of human–object interaction. In the second section, a case study is analyzed. A new approach for the recognition of human–object interactions is described in the third section. The experimental results of the proposed system are presented in the fourth section. Some discussion and conclusions are drawn in the fifth and sixth sections.

2. Related Works

In the field of computer vision research, various approaches have been proposed to solve the issue of recognizing complex human daily activities. This section describes some related research into the recognition of human–object interaction.

The authors of paper [4] proposed the anticipation of human intentional actions using the affordance of objects and the context of scenes for visualizing possible future actions. The experiments were performed using a Sez3D sensor. This system can predict future action when the frame observation range is between 30% and 60% of the whole action. Moreover, the authors of paper [5], proposed a system of robotic assistants that can anticipate what humans will do next using observations of pose and the surrounding environment, with the purpose of helping people with reactive response. Future actions are represented using the anticipatory temporal conditional random field (ATCRF), which is a model that can maintain a rich context of spatio-temporal relations via object affordances. Alternatively, the authors of paper [6] presented a method for predicting future actions from partially observed RGB-D (red, green, blue, depth) videos. Because of the rich context between humans and the environment while performing actions, the authors used a stochastic grammar model in order to capture the compositional structure of events and to integrate human actions with the corresponding objects and their affordances. In addition, the human–object–object (HOO) interaction affordance learning approach for improving the reliability of object recognition has also been proposed [7]. The relationship between a pair of objects is represented by a Bayesian network, which is then trained for the purpose of improving the reliability of object recognition. Moreover, a system using deep learning based on the affordance model has been proposed in [8] for recognizing human intentions and recommending objects for use. The action–object affordances were modeled using deep structure and gaze information obtained from a Tobii 1750 eye-tracker. This system is used to recognize human intentions and suggest the objects considered useful for the recognized intention.

Moreover, Koppula et al. proposed research on learning human activities and object affordances from RGB-D videos using a structural support vector machine (SSVM) approach [9]. However, the proposed method only achieved an accuracy of 75.0% for high-level activity recognition. In subsequent research by Koppula et al., the spatio-temporal relationship between human poses and objects was modeled using a conditional random field for anticipating activities [10]. In the latter study, the accuracy of detecting high-level activities portrayed in the CAD-120 dataset increased to 83.1%, which is still lower than with our proposed system. In addition, the authors in [11] proposed

a two-layer SVM hidden conditional random field (HCRF) recognition model for recognizing daily activities, specifically for those involving human–object interaction. However, this method relies on learning sub-activities based on the temporal sub-structure of the interaction for recognizing the high-level activities using a hierarchical SVM-HCRF model. Last but not least, a long short-term memory network was developed for recognizing the behavior of baseball players [12]. This network features the fusion of data from multiple sensors.

Some researchers have applied deep learning for recognizing single person actions, such as walking, sitting, standing, etc. Baccouche et al. [13] introduced a method for recognizing human actions by utilizing deep learning of spatio-temporal features. In this work, the authors extended a 2D convolutional neural network architecture to a 3D convolutional neural network (3D-ConvNet) architecture by adding the temporal dimension. Moreover, Liu et al. [14] proposed an approach that can be directly applied to raw depth video sequences for extracting spatio-temporal features using a support vector machine (SVM) for the classification of actions. A HAR based on deep-learning technology using skeleton images of human actions as input data was proposed in [15]. In addition, the authors of [16] developed a system incorporating enhanced images for a skeleton motion history, as well as a HAR system based on images of the relative positions of joints which can work independently on the problem domain.

Most of the HAR systems use RGB-D video, the data of eye-tracking and acceleration sensors, and skeletal tracking data for performing the experiments. To the best of our knowledge, even though much research has been done on HAR, it still remains a challenge for implementing HAR that can accurately recognize activities using a simple and robust approach, especially for the activities involving human–object interaction. Therefore, in this paper, we propose a hybrid approach for recognizing activities of human–object interactions in daily life.

3. Proposed System

In this paper, we propose a system for recognizing complex human–object interaction based on usage information for the objects involved. For this purpose, we apply a deep convolutional neural network (DCNN) over the spatio-temporal features extracted from information on human joints and the objects. We also apply a multi-class support vector machine (multi-class SVM) over the object usage probability (OUP) features in order to apply probability information for humans interacting with each object. The architecture of the proposed system is shown in Figure 1. The proposed system's architecture consists of: (i) input data acquisition, (ii) temporal segmentation, (iii) creating the input data for DCNN, (iv) training and recognizing interactions using DCNN, (v) extracting the object usage probability (OUP), (vi) training and recognizing interactions based on OUP by using a multi-class SVM, and finally (vii) a decision based on fusing the results of DCNN and multi-class SVM to produce the final result for recognizing human–object interactions. The main contributions and significant differences between this proposed system and our previous works [14,15] are as follows:

- Recognition of human–object interactions in which humans interact with different objects in order to complete desired tasks, such as making cereal or microwaving food.
- Creation of input data for a DCNN, which can accurately represent interactions between humans and objects.
- Calculation of object usage probability (OUP) and training OUP using a multi-class SVM to improve the performance of recognizing the human–object interactions.
- Fusing the result of DCNN and multi-class SVM (decision fusion) for generating better and more accurate results.

3.1. Input Data Acquisition

In the proposed system for recognizing complex human–object interaction, we use RGB images, as well as depth and human skeleton tracking data generated by depth cameras. In order to

confirm validity of the proposed system, the experiments were performed on the publicly available human–object interaction dataset of CAD-120 [3]. The structures of skeletal joints and their descriptions used for the experiments are illustrated in Figure 2.

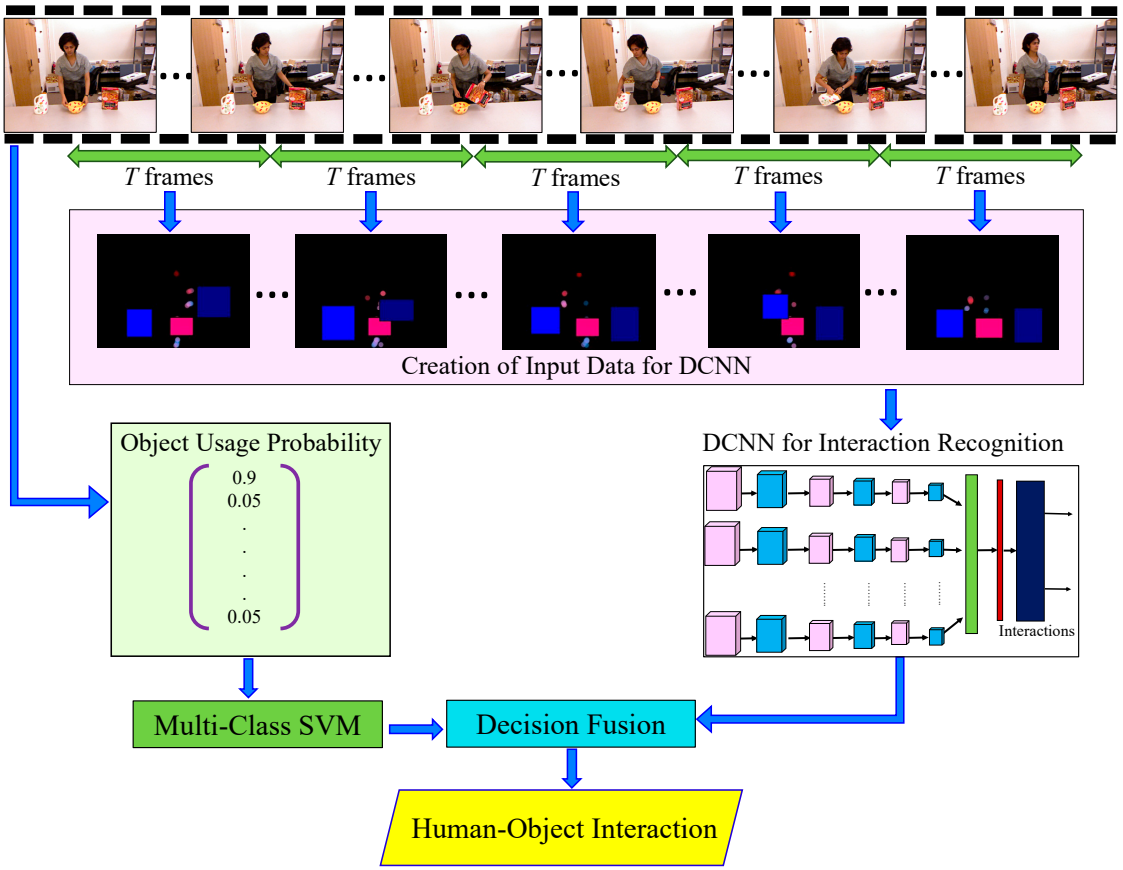
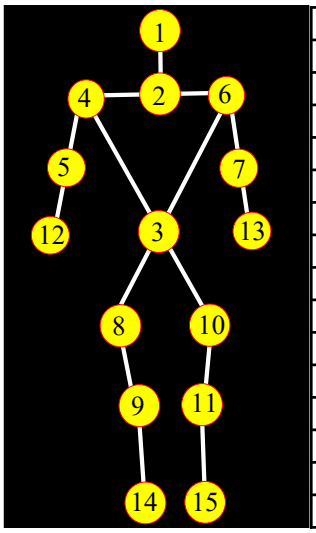


Figure 1. Architecture of a complex human–object interaction recognition system. DCNN: deep convolutional neural network; SVM: support vector machine.



Joint	Name
1	Head
2	Neck
3	Torso
4	Left Shoulder
5	Left Elbow
6	Right Shoulder
7	Right Elbow
8	Left Hip
9	Left Knee
10	Right Hip
11	Right Knee
12	Left Hand
13	Right Hand
14	Left Foot
15	Right Foot

Figure 2. Skeletal joints and their descriptions in the CAD-120 dataset.

3.2. Temporal Segmentation

We performed temporal segmentation over the input data to obtain groups of data representing human motion and object interaction patterns for each activity. This process groups the nearest frames into one segment, allowing extraction of changes in both spatial and temporal dimensions. Temporal segmentation is important because poor temporal segmentation often results in poor results for interaction recognition. For example, if the time duration threshold used in temporal segmentation is too short, features will not be well represented, and if the threshold is too long, distinguishing features within a set of interactions will be difficult to obtain. Here, we use a time duration threshold (a temporal sliding window size) of 15 frames over the input data with a frame rate of 15 fps. Therefore, each segment provides a good representation of changing interaction patterns within 1 second. The input data are uniformly segmented using a fixed temporal sliding window size of W. For action data with a total number of frames N which cannot be divided by 15, we replicate the first frame into R times that can be calculated using Equation (1).

$$R = 15 - (N \bmod 15) \tag{1}$$

where mod refers to the modulus operation for finding the total number of remaining frames R in order to perform uniform segmentation. Figure 3 shows an example of temporal segmentation over the skeletal movement data of the right shoulder, right elbow, and right hand while "taking action using the right hand".

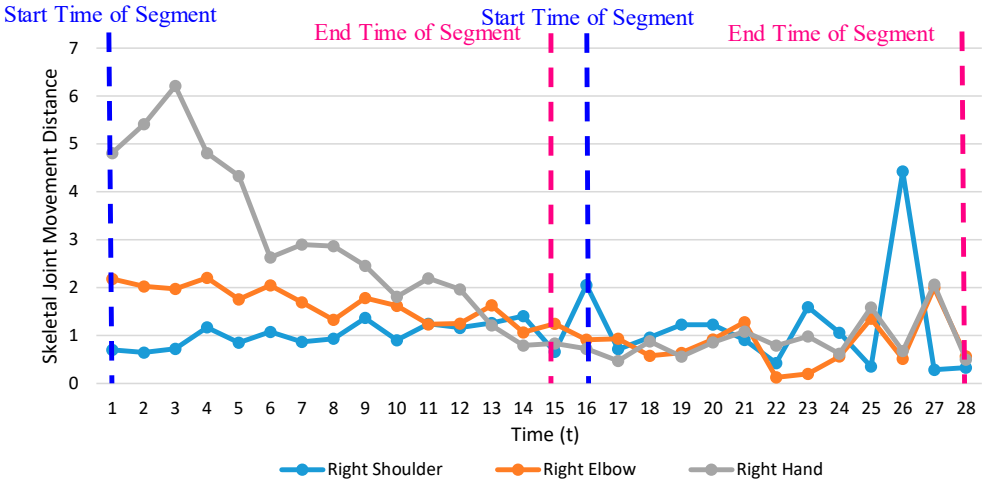


Figure 3. Illustration of temporal segmentation over the skeletal joint movement data.

3.3. Creation of Input Data for a Deep Convolutional Neural Network

3.3.1. Extraction of Skeletal Joint Movement Features

In order to extract spatio-temporal features, we first detected the moving joints within 15 frames. Next, we calculated the Euclidean distance between two consecutive frames for all joints using Equation (2). Then we found the mean and the standard deviation for joint distance as described in Equations (3) and (4).

$$Dist_{i,j} = \sqrt{(x_{i+1,j} - x_{i,j})^2 + (y_{i+1,j} - y_{i,j})^2} \text{ for } i = 1, 2, \ldots, T \text{ and } j = 1, 2, \ldots, S \tag{2}$$

$$m_j = \frac{1}{T}\sum_{i=1}^{T} Dist_{i,j} \quad \text{for } i = 1, 2, \ldots, T \text{ and } j = 1, 2, \ldots, S \tag{3}$$

$$sd_j = \sqrt{\frac{1}{T}\sum_{i=1}^{T}(Dist_{i,j} - m_j)} \quad \text{for } i = 1, 2, \ldots, T \text{ and } j = 1, 2, \ldots, S \tag{4}$$

where the value of S is 15 for total skeletal joints, and the value of T is 14, which represents $W - 1$ (the frame interval within the temporal sliding window W). $Dist_{i,j}$ refers to the Euclidean distance between the locations of joint j in two consecutive frames i and $i + 1$. The symbols of m_j and sd_j represent the mean and the standard deviation for joint j. If the value of $Dist_{i,j}$ is greater than or equal to the value of $(m_j + sd_j \times 0.5)$, then joint j at time i is regarded as a moving joint and its movement frequency ($freq_j$) increases. RGB value markers are predefined as shown in Table 1, and the values of i, j, and $freq_j$ for moving joints are used as indices for selecting RGB color values. Sample data for $Dist_{i,j}$, m_j, sd_j, and $freq_j$ for the sliding window for the action "taking using the right hand" are described in Table 2. After obtaining the locations of moving joints, we assigned a circular shape marker for the location of moving joint j in each frame with time i for creating input images for DCNN. We can see that values for $freq_j$ for the right shoulder, right elbow, and right hand are higher than those for other joints, indicating the validity of this process of detecting moving joints (highlighted with yellow color).

Table 1. RGB values of markers and their indices (i, j, and $freq_j$).

Time (*i*)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	-
R value	18	36	55	73	91	109	128	146	164	182	200	219	237	255	-
Joint (*j*)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
G value	17	34	51	68	85	102	119	136	153	170	187	204	221	238	255
***freq*$_j$**	1	2	3	4	5	6	7	8	9	10	11	12	13	14	-
B value	18	36	55	73	91	109	128	146	164	182	200	219	237	255	-

Table 2. Calculation of joint movement frequency from skeletal joint distance data.

Joint (*j*)	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
$Dist_{1,j}$	0.55	0.69	0.73	0.7	0.89	0.7	2.18	0.74	0.74	0.77	1.1	0.09	4.81	0.72	1.22
$Dist_{2,j}$	0.59	0.64	0.62	0.62	0.57	0.65	2.02	0.63	0.63	0.66	1.03	1.89	5.41	0.64	1.2
$Dist_{3,j}$	0.97	0.77	0.69	0.83	0.67	0.72	1.97	0.56	32.6	0.67	1.04	1.25	6.21	52.8	1.22
$Dist_{4,j}$	1.1	1	0.91	0.86	5.93	1.17	2.2	0.78	10.6	1.11	1.28	7.11	4.81	20.5	1.44
$Dist_{5,j}$	0.86	0.85	0.79	0.89	4.65	0.85	1.75	0.76	5.74	0.7	0.98	4.73	4.33	6.48	1.21
$Dist_{6,j}$	0.78	1.11	1.31	1.15	2.24	1.08	2.05	1.05	9.81	2.06	2.33	2.14	2.63	12.7	2.58
$Dist_{7,j}$	0.81	0.89	1.17	0.95	1.46	0.87	1.69	1.43	2.1	1.47	1.74	1.52	2.9	2.16	1.97
$Dist_{8,j}$	0.95	0.9	0.98	0.84	1.55	0.93	1.33	1.27	8.22	0.99	1.13	0.61	2.87	8.22	1.31
$Dist_{9,j}$	1.45	1.37	1.38	1.32	1.34	1.37	1.78	1.48	3.79	1.32	1.39	0.96	2.46	3.85	1.5
$Dist_{10,j}$	0.94	0.81	0.7	0.81	2	0.9	1.62	0.63	3.33	0.65	0.76	1.96	1.81	3.33	0.93
$Dist_{11,j}$	1.09	0.88	0.73	0.62	2.02	1.24	1.23	0.41	2.64	1.01	1	2.14	2.19	2.7	1.05
$Dist_{12,j}$	0.93	0.74	0.58	0.58	2.69	1.16	1.25	0.31	18.1	1.08	1.06	3.19	1.96	18	1.07
$Dist_{13,j}$	1.26	1	0.67	0.91	2.22	1.26	1.63	0.14	7.84	0.69	0.74	2.66	1.21	7.85	0.81
$Dist_{14,j}$	1.47	1.23	0.82	1.21	1.06	1.41	1.07	0.32	5.25	0.83	0.84	1.7	0.79	5.22	0.95
m_j	0.98	0.92	0.86	0.88	2.09	1.02	1.7	0.75	7.96	1	1.17	2.28	3.17	10.4	1.32
sd_j	0.28	0.21	0.26	0.22	1.52	0.25	0.37	0.42	8.5	0.4	0.42	1.8	1.66	13.7	0.46
$m_j + (sd_j{*}0.5)$	1.12	1.02	0.99	0.99	2.85	1.15	1.88	0.96	12.2	1.2	1.38	3.18	4	17.2	1.55
$freq_j$	3	3	3	3	2	6	5	4	2	3	3	3	5	3	2

The joint movement frequency ($freq_j$) greater than or equal to 5 are described in red color.

3.3.2. Creation of an Object Representation Image

In human–object interaction, humans interact with various kinds of objects in different ways. In real-world applications, the variety of objects with which humans interact is quite large, even though they might be in the same object category. For example, items used for cooking or taking medicine can vary. Therefore, for robustly inserting object information into input images, object representation images (ORI) must be created and combined with input data for a DCNN that provides a good representation of each object category. In this system, we use color values for object representation.

We applied ten color values for the ten object categories because the dataset on which we performed experiments included ten objects with different designs, namely a medicine-box, microwave, remote, milk, plate, cloth, bowl, book, cup, and box. The color representation of each object category is described in Table 3. After defining the color representation for each object category, we added the corresponding color value to each DCNN input image by filling the object regions (within the bounding boxes) provided in the CAD-120 dataset [3] with that color. The result of inserting ORI is shown in Figure 4. This process can provide information on the region connection state (RCS), which can express the graphically connected region for each object. This kind of information is very useful for recognizing human–object interaction, such as the interaction involved in microwaving food or pouring milk into a bowl of cereal. These interactions include overlapping regions between objects. Figure 5 provides illustrations of three kinds of RCS involving a microwave, bowl, milk bottle, cup, and medicine box.

Table 3. Color representation of each object category in the CAD-120 dataset.

Objects	R	G	B	Color	Objects	R	G	B	Color
medicine box	0	255	0		cloth	255	0	255	
microwave	0	255	255		bowl	255	0	128	
remote control	0	128	255		book	255	0	0	
milk	0	0	255		cup	128	0	0	
plate	128	0	255		box	0	0	128	

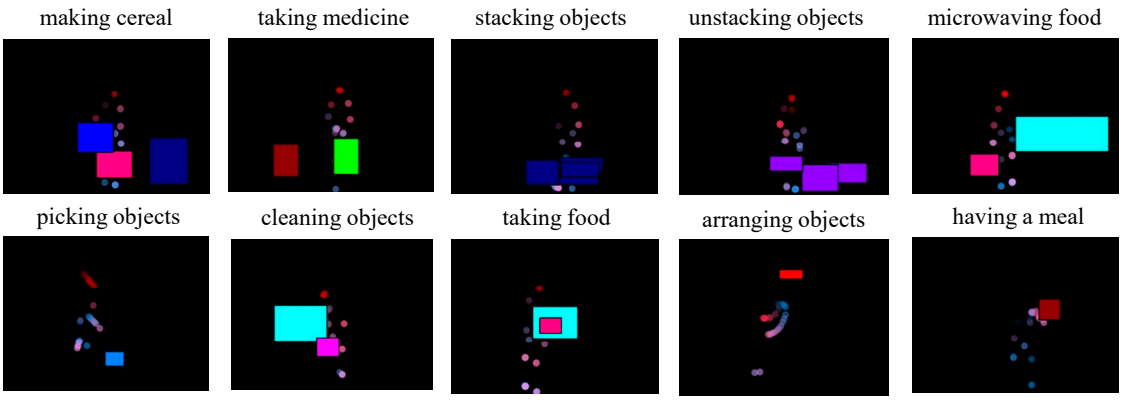
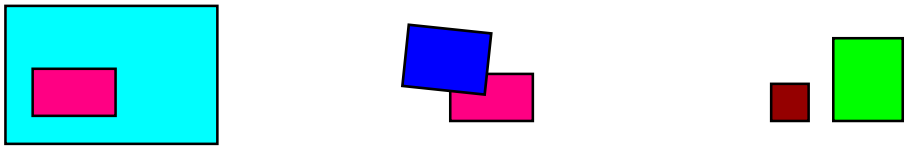


Figure 4. Sample DCNN input images.



(a) Fully Overlapping State (b) Partially Overlapping State (c) Non-overlapping State

Figure 5. Three kinds of region connection states between microwave (cyan), bowl (pink), milk bottle (blue), cup (brown), and medicine box (green).

3.4. Interaction Recognition using a Deep Convolutional Neural Network (DCNN)

Deep convolutional neural networks (DCNN), also known as deep learning, are a kind of neural network that includes grid-type operations of convolution and pooling over grid-type input data such as images. The main function of convolution operations is to extract key information from raw input data through multiplication in a predefined kernel matrix. Pooling operations are used for reducing the dimensions of each DCNN output layer. Pooling can be performed by the mathematical operations of taking averages or maximum values for the data which exist within the defined matrix. In every DCNN layer, a decision must be made on whether the results of convolution or pooling should be produced as output and sent to the next layer. This decision-making process is performed by the

activation function. Various activation functions have been discussed in the literature on DCNN, including sigmoid, hyperbolic tangent, and rectified linear units (ReLU). Each function has its own properties. For example, the output of the sigmoid activation function is within the range of 0 and 1, and the output of the hyperbolic tangent function is from −1 to 1. In the proposed system, the rectified linear unit (ReLU) activation function was applied in all DCNN layers. ReLU produces original input values as outputs if those values are greater than 0 and produces 0 if input values are less than 0.

The most recent DCNN research indicates that DCNN achieves superior performance in visual object and pattern recognition. Therefore, we applied DCNN in the proposed system for recognizing patterns of human motion and object interaction. As shown in Figure 6, the DCNN architecture comprises three convolution (*Conv*) layers and three pooling (*Pool*) layers, followed by one fully connected (*F*) layer and an output (*O*) layer. The total number of output neurons for each DCNN layer was 32, 64, 64, and 10. In addition, each layer was followed by a drop-out (*D*) layer for minimizing the data overfitting problem, using drop-out ratios of 1%, 2%, 3%, and 4%. Convolution operations were performed in three hidden layers using kernel sizes of 7×7, 5×5, and 3×3. For pooling operations, 2×2 kernels were used. Then, one fully connected layer was applied before the output layer in order to combine DCNN features in a one-dimensional vector with a length of 64. The next layer was an output layer with ten neurons for recognizing ten human–object interactions. For transforming the results for DCNN output layers into corresponding probability values (*P_DCNN*), a Soft-Max operation was applied. Weight initialization for all layers was done using the MSRA method, which was developed by Microsoft Research Asia (MSRA) [17]. The stochastic gradient descent algorithm [18] was used for weight updating, and the whole network was constructed using the Matcaffe deep-learning framework [19,20].

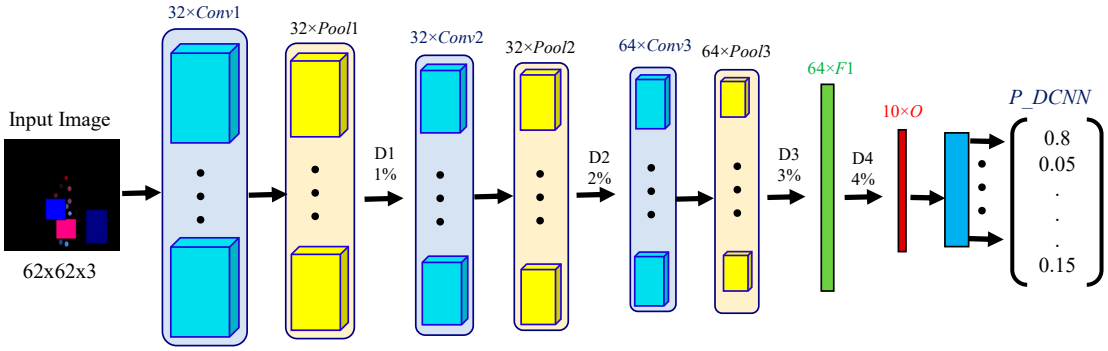


Figure 6. DCNN architecture.

3.5. Extracting the Object Usage Probability

Complex human daily activities consist of many continuous sub-actions. Therefore, the application of fixed-size segmentation can cause a mixing of such continuous sub-actions. However, the proposed system can overcome the problem of mis-recognizing actions due to the mixing features in continuous sub-actions by applying object usage probability (OUP). In daily life, humans use various kinds of objects for accomplishing tasks. Therefore, activeness information for each object involved in a specific set of human–object interactions is very useful in establishing a recognition system for complex human–object interaction. In our proposed system, we consider an object to be in a using state when a human hand reaches for the object and then moves it. Let us denote the total number of frames that include the usage of object o during interaction k as $C_{o,k}$, and the total number of frames in interaction k as TF_k, then the OUP of object o for interaction k can be calculated using Equation (5). The probability that no interaction occurs with objects within interaction k is denoted as $null_k$, which can be calculated

using Equation (6). The usage counts for the sample object and OUP data for ten actions in the CAD-120 dataset are shown in Tables 4 and 5.

$$\mathrm{OUP}_{o,k} = \frac{C_{o,k}}{TF_k} \quad \text{for } k = 1, 2, ..., 10 \text{ and } o = 1, 2, \ldots, 10 \tag{5}$$

$$null_k = 1 - \sum_{o=1}^{10} OUP_{o,k} \quad \text{for } k = 1, 2, ..., 10 \text{ and } o = 1, 2, \ldots, 10 \tag{6}$$

Table 4. Sample data for object usage counts (*C*) for ten actions in the CAD-120 dataset.

Medicine Box	Microwave	Remote Control	Milk	Plate	Cloth	Bowl	Book	Cup	Box	Null	Total Frames (*TF*)	Interaction	
												k	Description
0	0	0	137	0	0	87	0	0	254	42	520	1	making cereal
177	0	0	0	0	0	0	0	131	0	163	471	2	taking medicine
0	0	0	0	0	0	0	0	0	436	113	549	3	stacking objects
0	0	0	0	376	0	0	0	0	0	113	489	4	unstacking objects
0	315	0	0	0	0	0	0	0	248	85	648	5	microwaving food
0	0	0	0	0	0	113	0	0	0	46	159	6	bending
0	235	0	0	0	247	0	0	0	0	111	593	7	cleaning objects
0	179	0	0	0	0	0	0	110	0	115	404	8	taking food
0	0	0	0	0	0	0	0	0	252	110	362	9	arranging objects
0	0	0	0	0	0	0	0	193	0	306	499	10	having breakfast

Table 5. Object usage probability (OUP) sample data for ten actions in the CAD-120 dataset.

Medicine Box	Microwave	Remote Control	Milk	Plate	Cloth	Bowl	Book	Cup	Box	Null	Interaction
0	0	0	0.26	0	0	0.17	0	0	0.49	0.08	making cereal
0.38	0	0	0	0	0	0	0	0.28	0	0.35	taking medicine
0	0	0	0	0	0	0	0	0	0.79	0.21	stacking objects
0	0	0	0	0.77	0	0	0	0	0	0.23	unstacking objects
0	0.49	0	0	0	0	0	0	0	0.38	0.13	microwaving food
0	0	0	0	0	0	0.71	0	0	0	0.29	bending
0	0.4	0	0	0	0.42	0	0	0	0	0.19	cleaning objects
0	0.44	0	0	0	0	0	0	0.27	0	0.28	taking food
0	0	0	0	0	0	0	0	0	0.7	0.3	arranging objects
0	0	0	0	0	0	0	0	0.39	0	0.61	having breakfast

3.6. Training and Recognizing Interactions Based on OUP Using a Multi-Class Support Vector Machine

Support vector machines (SVM) are supervised learning algorithms which are very popular for visual pattern recognition. SVMs were originally designed for binary classification and are also called linear SVMs. The main process involved in a linear SVM is for finding the linear optimal separating hyperplane, which is the decision boundary separating the various classes of data. Linear SVMs use support vectors, which are the training data, and margins are defined by these support vectors. During the training step, a SVM is used to find the maximum margin hyperplane which gives the largest separation between class. For multi-class SVM classification, the one-against-one classification method is used in training for each action. Therefore, a one-against-one class design coded into multi-class SVM yields *M* binary learners for *M* classes. The output of each SVM (separating hyperplane) can be written as

$$y_k(X) = \sum_{v=1}^{N} w_v^k x_v + b \tag{7}$$

where y_k is the output of the SVM for the interaction k, $X = \{x_1, x_2, \ldots, x_n\}$, which is the feature vector containing N features. w_v^k is the weight vector of the SVM of interaction k and features v, and b is a scalar value for bias. In the proposed system, we trained the multi-class SVM using OUP data for implementing the human–object recognition system as shown in Figure 7. In the training phase, the OUP were calculated based on the object information of training video sequences, and then used as feature vectors for training multi-class SVM. In the testing phase, multi-class SVM produced classified interactions together with confidence scores. Later, we fused the confidence scores from the multi-class

SVM, which were the probability values for M possible actions (*P_SVM*), together with the results of DCNN to establish a robust system for recognizing human–object interaction.

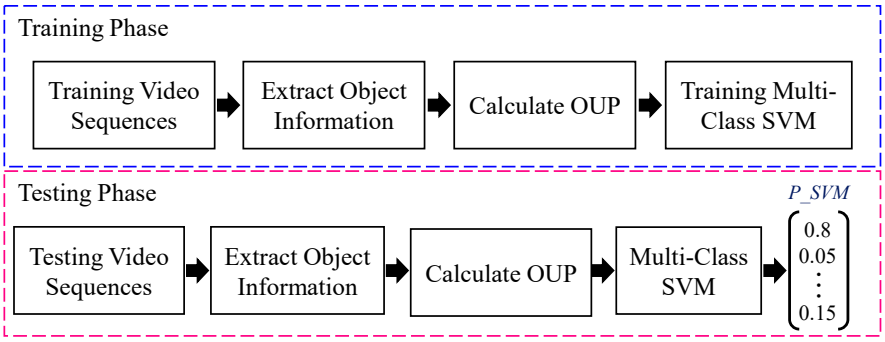


Figure 7. Illustration of training and testing phases of multi-class SVM using object usage probability (OUP) data.

3.7. Decision Fusion and Human–Objects Interaction Recognition

After obtaining probability values for all ten interactions from DCNN and OUP based multi-class SVM, we performed the decision fusion operation for accurately obtaining decision results in the recognition of human–object interaction. The decision fusion process was simply performed by averaging the probability values for all interactions and using the class with the highest probability. The mathematical expression for the decision fusion process is shown in Equations (8) and (9). In this way, mis-recognized actions obtained using DCNN can be correctly recognized by the OUP-based multi-class SVM, and vice versa.

$$\text{Recognized Interaction} = \text{Max}(P_k) \qquad \text{for } k = 1, 2, \ldots, 10 \tag{8}$$

$$P_k = \frac{(P\_DCNN_k + P\_SVM_k)}{2} \qquad \text{for } k = 1, 2, \ldots, 10 \tag{9}$$

where P_DCNN_k, P_SVM_k, and P_k refer to the probability of interaction k obtained using the DCNN, multi-class SVM, and decision fusion process.

4. Experimental Results

The experiments were performed on the CAD-120 human daily activities dataset which consisted of ten high-level activities performed by four different subjects: making cereal, taking medicine, stacking objects, unstacking objects, microwaving food, picking objects, cleaning objects, taking food, arranging objects, and having a meal. The data consisted of RGB images, depth images and skeleton joint coordinates, object locations, and object types for each high-level activity. This data can be downloaded from the following URL: *http://pr.cs.cornell.edu/humanactivities/data.php*. The sample RGB images in the CAD-120 dataset are shown in Figure 8. The performance of the proposed system was evaluated by the leave-one-subject-out-cross-validation method, which uses three subjects' data as training data, and data for the other subject as test data. Therefore, we alternately trained the DCNN and multi-class SVM using data from three different subjects, and used data from the remaining subject as test data. We performed these experiments four times by alternately training and testing using data from four subjects. For accurately recognizing the interactions involved in "stacking objects" and "unstacking objects", we added some rules based on spatial features of the objects, because the obvious difference between those two interactions is the spatial feature of (total width) at the start and end times of the interactions. In the case of "stacking objects", the width of all objects at the beginning is larger than that at the end. However, the opposite is the case for the "unstacking objects", as shown in Figure 9. The graphical form of the result of recognizing "making cereal" interaction is shown in

Figure 10. A detailed confusion matrix for recognizing the actions in the CAD-120 dataset is shown in Table 6.

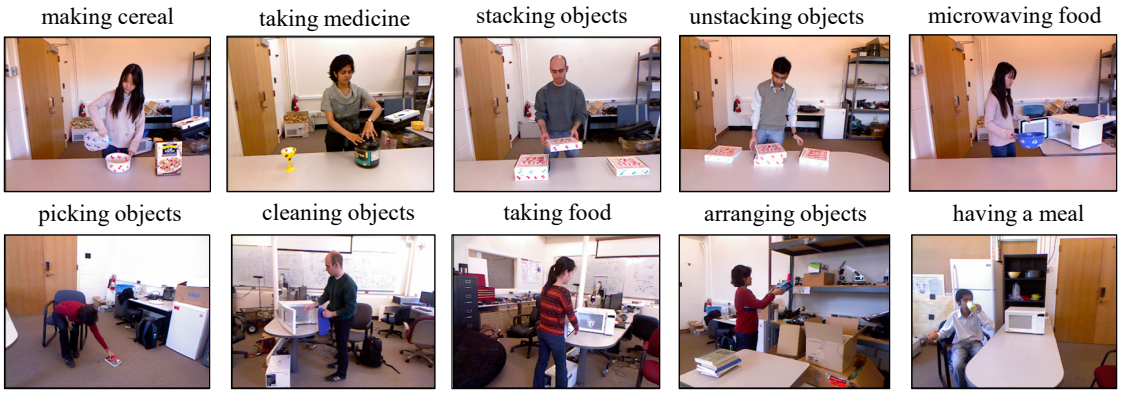


Figure 8. Sample RGB images from the CAD-120 dataset.

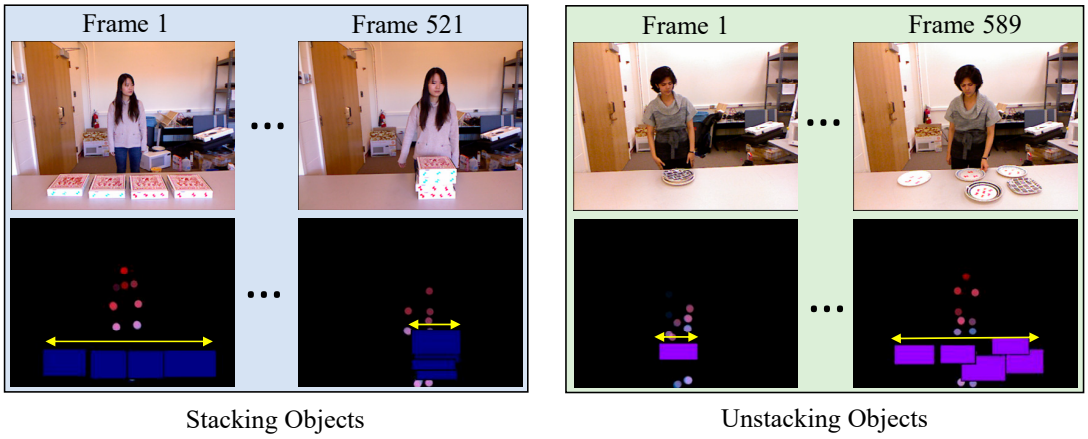


Figure 9. Illustration of spatial features between stacking and unstacking objects (First row: RGB image; Second row: DCNN input image; yellow line indicates total object width).

Table 6. Confusion matrix for recognizing interactions in the CAD-120 dataset.

		Recognized Actions									
		Making Cereal	**Taking Medicine**	**Stacking objects**	**Unstacking Objects**	**Microwaving Food**	**Picking Objects**	**Cleaning Objects**	**Taking Food**	**Arranging Objects**	**Having a Meal**
Actual Actions	making cereal	100	0	0	0	0	0	0	0	0	0
	taking medicine	0	100	0	0	0	0	0	0	0	0
	stacking objects	0	0	91.67	0	0	0	0	0	8.33	0
	unstacking objects	0	0	0	91.67	0	0	0	0	8.33	0
	microwaving food	0	0	0	0	91.67	0	0	8.33	0	0
	picking objects	0	0	0	0	0	83.33	0	0	8.33	8.33
	cleaning objects	0	0	0	0	0	0	100	0	0	0
	taking food	0	0	0	0	8.33	8.33	0	83.33	0	0
	arranging objects	0	0	0	0	0	0	0	0	91.67	0
	having a meal	0	0	0	0	0	0	0	0	0	100

The percentage of correctly recognized actions in the CAD-120 dataset are highlighted in yellow color.

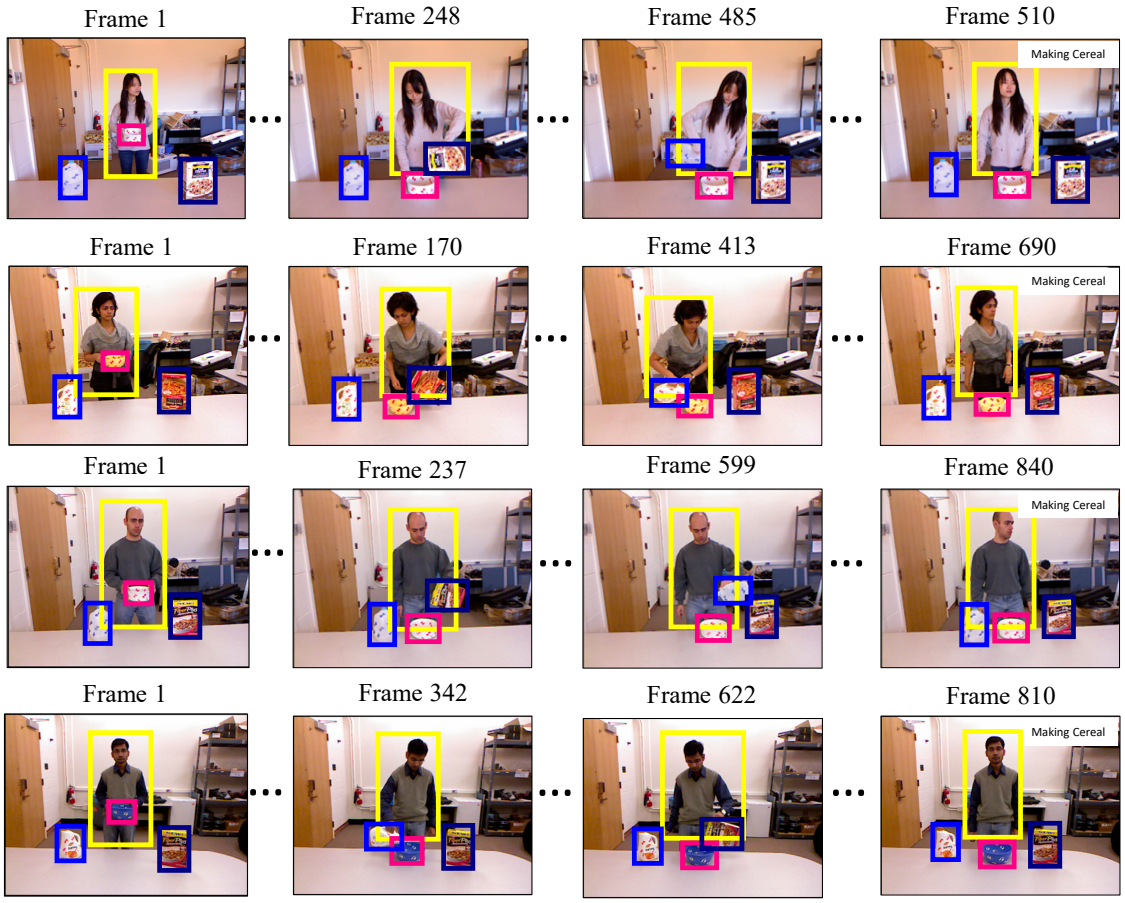


Figure 10. Graphical form of the result of recognizing the making cereal interaction (each row describes the interactions performed by a different subject).

We calculated the overall accuracy (Recognition Rate) using the following Equation (10).

$$\text{Recognition Rate} = \frac{\text{No.of correctly recognized actions}}{\text{No.of total actions in the dataset}} \times 100 \quad (10)$$

As shown in Table 6, the proposed system correctly recognized most actions with high recognition accuracy. The recognition accuracy for taking food from a microwave was low because it closely resembles microwaving food. The interaction involved in picking objects was mistaken for interactions such as those involved in taking food, arranging objects, and having a meal, because object-interaction information was comparatively less for these interactions. As described in Table 7, we also compared our results with those of other state-of-the-art recognition methods that were performed on the same dataset. We can see that the proposed system outperformed the method proposed by Koppula et al. [9] by a significant margin of 12.73%. We also improved accuracy by a margin of 10.23% when compared with the method proposed by Koppula and Saxena [10]. Moreover, our system achieved the comparable accuracy that was 3.03% higher than the method proposed in [11]. Due to the highly insensitive nature of the input data property of deep learning, we believe that our proposed system achieves higher accuracy, and is more robust and efficient in recognizing complex daily activities in real-world applications.

Table 7. Comparison of performance on the CAD-120 dataset.

Method	Recognition Rate
Koppula et al. [9]	80.6%
Koppula and Saxena [10]	83.1%
Selmi et al. [11]	90.30%
Proposed System	93.33%

5. Conclusions

In this paper, we propose a recognition system for complex human–object interactions based on the hybrid approach of combining DCNN and multi-class SVM. We also propose a new feature called object usage probability (OUP) which is highly effective in recognizing human–object interaction. By applying this hybrid approach, the performance of the proposed system has been improved. Moreover, for the purpose of recognizing interactions performed by different people in real-world applications, we use the leave-one-subject-out-cross-validation method for performance evaluation. Using this validation method, after applying a rule based on the spatial features of the objects, the proposed system achieved an overall accuracy of 93.33%, which is higher than that of other state-of-the-art methods.

6. Discussion and Future Work

In the proposed system, we used location information for ground-truth objects in order to create DCNN input images, and to calculate OUP. In the future, we will automate the process of detecting and classifying the objects using deep-learning technology, making the whole system more automatic. In the automatic detection and classification of objects, we will perform experiments with complex backgrounds which have similar color with the associated objects. We will also work on improving recognition accuracy for taking food from a microwave by considering more information that can differentiate this action from the interaction involved in similar activities. The proposed system has only been tested using offline data. Therefore, we will perform an analysis of motion trajectories for each interaction using geometrical and statistical models for recognizing and predicting online action data in combination with DCNN. Finally, we will record a variety of more complex human–object interaction videos under various environmental conditions and more complex backgrounds and perform more experiments.

Author Contributions: Methodology, C.N.P.; Supervision, T.T.Z. and P.T. The major portion of work presented in this paper was carried out by the first author C.N.P., under the supervision of the second author T.T.Z. The third author P.T. provided valuable advice on mathematical concepts. Both T.T.Z. and P.T. gave suggestions for the preparation and revision of this paper; C.N.P. devised the methodology and performed the experiments; all three authors analyzed the experimental results.

Funding: This work is partially supported by the Japan Society for the Promotion of Science (JSPS) KAKENHI (Grant No. 17K08066).

Acknowledgments: The authors would like to express sincere thanks to Hiromitsu Hama who is an emeritus Professor at Osaka City University, Japan for his kind and valuable suggestions and advice during this research. The authors also would like to thank in advance reviewers and editors of this special issue for all suggestions and comments for improving this research paper.

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

References

1. Microsoft Kinect. 2013. Available online: https://developer.microsoft.com/en-us/windows/kinect (accessed on 1 June 2017).
2. ASUS Xtion PRO LIVE. 2013. Available online: https://www.asus.com/3D-Sensor/Xtion_PRO/ (accessed on 28 October 2017).
3. Cornell Activity Dataset. Available online: http://pr.cs.cornell.edu/humanactivities/data.php (accessed on 1 March 2018).

4. Dutta, V.; Zielinska, T. Predicting Human Actions Taking into Account Object Affordances. *J. Intell. Robot. Syst.* **2018**, *93*, 745–761. [CrossRef]
5. Koppula, H.S.; Saxena, A. Anticipating human activities using object affordances for reactive robotic response. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 14–29. [CrossRef] [PubMed]
6. Qi, S.; Huang, S.; Wei, P.; Zhu, S.C. Predicting human activities using stochastic grammar. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 1173–1181.
7. Ren, S.; Sun, Y. Human-object-object-interaction affordance. In Proceedings of the 2013 IEEE Workshop on Robot Vision (WORV), Clearwater Beach, FL, USA, 15–17 January 2013; pp. 1–6.
8. Kim, S.; Kavuri, S.; Lee, M. Intention recognition and object recommendation system using deep auto-encoder based affordance model. In Proceedings of the 1st International Conference on Human-Agent Interaction, II-1-2, Sapporo, Japan, 7–9 August 2013; pp. 1–6.
9. Koppula, H.S.; Gupta, R.; Saxena, A. Learning human activities and object affordances from rgb-d videos. *Int. J. Robot. Res.* **2013**, *32*, 951–970. [CrossRef]
10. Koppula, H.; Saxena, A. Learning spatio-temporal structure from rgb-d videos for human activity detection and anticipation. In Proceedings of the 30th International Conference on Machine Learning, Atlanta, GA, USA, 16–21 June 2013; pp. 792–800.
11. Selmi, M.; El-Yacoubi, M.A. Multimodal sequential modeling and recognition of human activities. In Proceedings of the International Conference on Computers Helping People with Special Needs, Linz, Austria, 13–15 July 2016; pp. 541–548.
12. Sun, S.W.; Mou, T.C.; Fang, C.C.; Chang, P.C.; Hua, K.L.; Shih, H.C. Baseball Player Behavior Classification System Using Long Short-Term Memory with Multimodal Features. *Sensors* **2019**, *19*, 1425. [CrossRef] [PubMed]
13. Baccouche, M.; Mamalet, F.; Wolf, C.; Garcia, C.; Baskurt, A. Sequential deep learning for human action recognition. In Proceedings of the International Workshop on Human Behavior Understanding, Amsterdam, The Netherlands, 16 November 2011; pp. 29–39.
14. Liu, Z.; Zhang, C.; Tian, Y. 3D-based deep convolutional neural network for action recognition with depth sequences. *Image Vis. Comput.* **2016**, *55*, 93–100. [CrossRef]
15. Phyo, C.N.; Zin, T.T.; Tin, P. Skeleton motion history based human action recognition using deep learning. In Proceedings of the 2017 IEEE 6th Global Conference on Consumer Electronic (GCCE 2017), Nagoya, Japan, 24–27 October 2017; pp. 784–785.
16. Phyo, C.N.; Zin, T.T.; Tin, P. Deep Learning for Recognizing Human Activities using Motions of Skeletal Joints. *IEEE Trans. Consum. Electron.* **2019**, *65*, 243–252. [CrossRef]
17. He, K.; Zhang, X.; Ren, S.; Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1026–1034.
18. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
19. Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S.; Long, J.; Girshick, R.; Guadarrama, S.; Darrell, T. Caffe: Convolutional architecture for fast feature embedding. In Proceedings of the 22nd ACM International Conference on Multimedia, Orlando, FL, USA, 3–7 November 2014; pp. 675–678.
20. Caffe. Available online: http://caffe.berkeleyvision.org (accessed on 16 December 2017).