Skeleton based efficient fall detection

Muzaffer Aslan[1*], Yaman Akbulut[2], Abdulkadir Şengür[3], Melih Cevdet İnce[3]
muzaffer.aslan@gmail.com, yamanakbulut@firat.edu.tr, ksengur@firat.edu.tr, mince@firat.edu.tr

HIGHLIGHTS

Construction of depth and skeletal datasets for daily actions with Kinect

Skeletal joint feature extraction

Elderly fall detection based on 3D skeletal data

ABSTRACT

Fall is one of the most important causes of death and injury for the elderly. Real time fall detection has great importance for the safety of the elderly. In this work, we presented a new method for fall detection that is based on Kinect skeleton data. Three-dimensional (3D) skeleton data of "FUKinect-Fall" dataset, which was constructed from 21 volunteers, is firstly reduced to two different (xy and zy) two-dimensional (2D) datasets. Then, coded regions on nested circles are constructed that are located on the determined reference joint. Thus for each action, the average of the coded 19 skeleton points during the action, are used as features. The obtained feature set is classified by using k-Nearest Neighborhood (k-NN) and Support Vector machine (SVM) classifiers. Experimental results show that fall detection is achieved with 97.08% accuracy.

Key Words: Fall detection, Kinect, Skeleton Data, FUKinect-Fall dataset.

2. METHOD

2.1. Feature Extraction from Skeletal Joints

The Microsoft Software Development Kit (MSDK) defines the joint positions by processing depth images. At first, MSDK estimates the joint position for a pixel in depth images and calculates the accuracy level for each pixel. Then, it selects the skeleton from the skeleton labels provided with accuracy levels [22]. The skeleton model of Kinect for human skeleton is shown in Figure 1.

In an action consisting of *T* pieces of frames in which the body is represented by *K* joints, in any frame (t_i) *x, y* and *z* coordinates of any joint (k) $x_k^{(t)}, y_k^{(t)}, z_k^{(t)}$ [21]. Thus; The joint vector of the 20 joints of the three axes is as such $E = [x_1^{(t)}, y_1^{(t)}, z_1^{(t)}, x_2^{(t)}, y_2^{(t)}, z_2^{(t)}, \ \ldots, x_K^{(t)}, y_K^{(t)}, z_K^{(t)}]$.

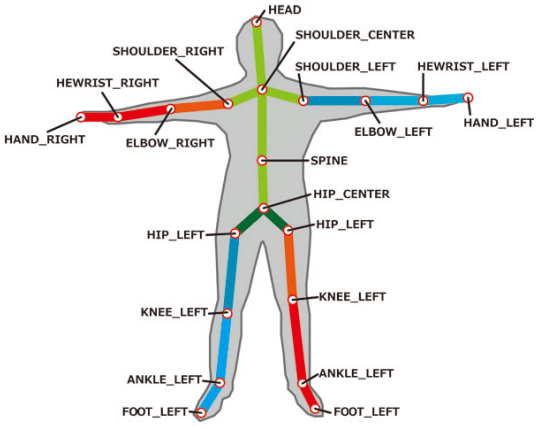

Figure 1. (Kinect skeletal joint structure) [22]

3D model coordinates for action recognition are based on the reduction of two (xy and zy) 2D joint coordinates [23, 24]. The model which is created with an angle of 30° for the selected xy and zy regions in model are shown in Figure 2.

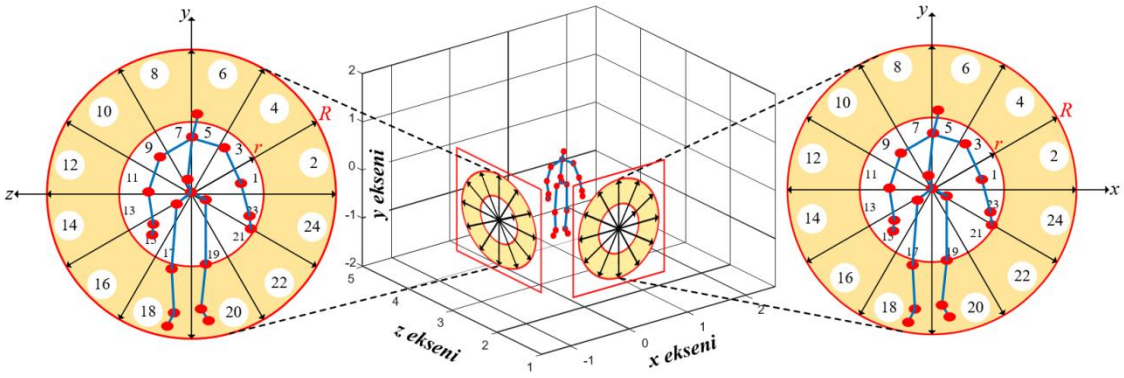

Figure 2. 2D axis model and regions

According to Shotton method, hip centre joint was referenced among joint points [25, 26]. In this way, a pose independent of the scale is obtained. If the joint vector is arranged according to the reference joint, it can be written as such;

$$E = [0, 0, 0,\ x_2 - x_1, y_2 - y_1,\ z_2 - z_1, \dots, x_K - x_1, y_K - y_1,\ z_K - z_1] \quad (1)$$

Angle of each joint according to the reference joint (α_i) and distance (u_i) are respectively given in Eq.2 and Eq.3 [26].

$$\alpha_i = arctan\frac{y_i - y_1}{x_i - x_1} \quad (2)$$

$$u_i = \sqrt{(x_i - x_1)^2 + (y_i - y_1)^2} \quad (3)$$

can be calculated with the relations. Also, it can be calculated with the help of $R_{xy} = u_{max}$ and $r_{xy} = \frac{R}{2} = \frac{u_{max}}{2}$ relations [21]. According to selected xy axis for joint angles in a frame $A_{xy} = [0, \alpha_2, \alpha_3, \alpha_4, \ldots, \alpha_K]$ and distance of all joints to reference joint can be written as $U_{xy} = [0, u_2, u_3, u_4 \ldots, u_K]$ vectors. A_{zy} and U_{zy} vectors can be written by making same process for yz axis [21].

When the axes are divided into regions with a certain θ angle, the number of zones to be generated can be calculated by $n_b = 2 \times (360/\theta)$ [23, 26]. In a frame, depending on the distance and angle of the joints to the reference joint, the fact that in which region it exists is detected. Accordingly, the region vector where the joints are located can be written as $B_{xy} = [0, b_2, b_3, b_4, \ldots, b_K]$. The same operations are performed in the zy axis to obtain the B_{zy} vector. Thus, B_{xy} and B_{zy}matrices are obtained for each joint in all frames [21]. In this way, in which region the 19 joints except the reference joint, the location information in each frame, is determined. Through a movement or action, the feature matrix is obtained by averaging a joint across all frames.

In order to form feature matrix, average of a joint in all frames in an action is taken. It can also be expressed as this mean descriptor. When the feature matrix was created with this method for both axes, it was seen that the temporal information about the actions disappeared. This situation causes the mixing of the actions which are opposite to each other. The temporal hierarchical structure is used to solve this problem [27]. Figure 3 shows the temporal hierarchical structure for the *T* pieces frame in an action. First of all the frames are $t = 1$ to *T* (O_1), then sequentially from $t = 1$ to $\frac{T}{2}$ (O_2), from $\frac{T}{4}$ to $\frac{3T}{4}$ (O_3) and finally from $\frac{T}{2}$ to *T* (O_4) are obtained by taking the average of each joint for each joint. Thus, the number of features has been increased to 4 times.

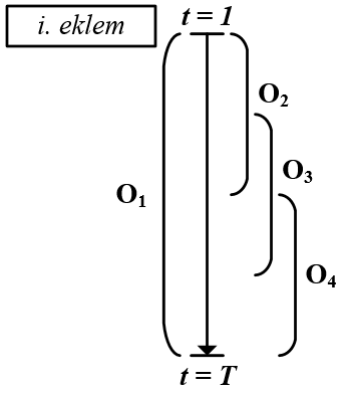

Figure 3. Temporal hierarchical structure of mean descriptor**.** [27]

3. EXPERIMENTAL WORKS

In this section, FUKinect-Fall and UTKinect-Action datasets are used to test the proposed method. The results were evaluated in experimental results section.

3.1. Construction of FUKinect-Fall Dataset

FUKinect-Fall dataset was created using Kinect V1. Kinect camera was placed on a tripot at a height of 95 cm from the floor to create a dataset. The dataset area in Figure 5 was created by taking into consideration the vertical and horizontal viewing angles of the depth and area sensor of 4 × 4 m with the sight boundaries of 0.5 m and 4.5 m within Kinect's vision limits.

Figure 5. Experimental Dataset collection environment

In addition, the actions were performed in the area between 2 m and 3.5 m according to the subjects' physical characteristics such as height, weight and the type of action performed. Figure 6 shows the working area where these actions took place.

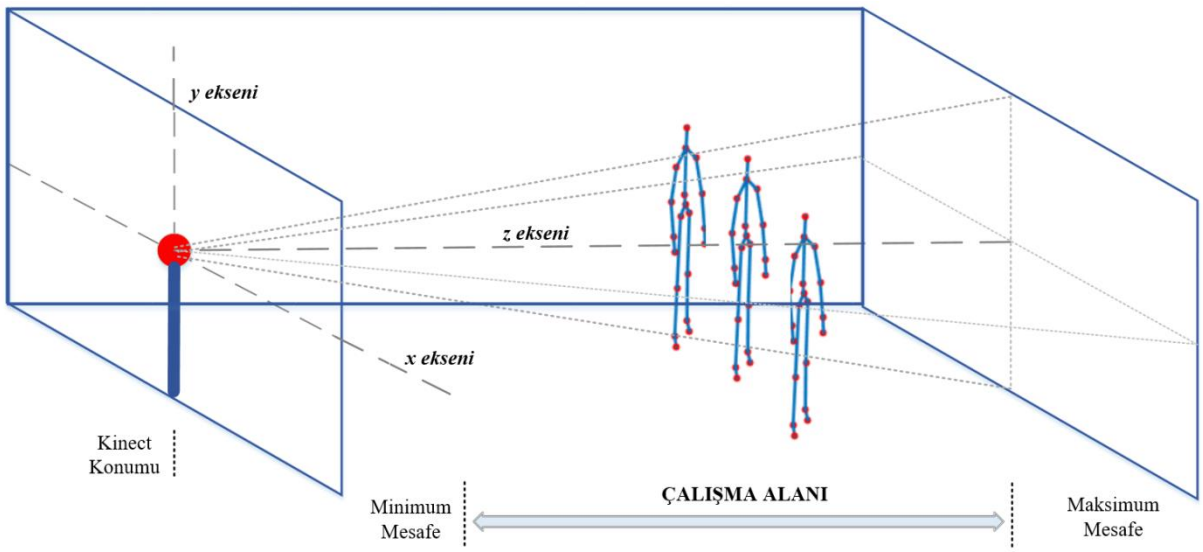

Figure 6. Dataset collection work environment

The dataset includes *walking, bending, sitting, squatting, lying and falling* actions performed by 21 subjects between 19-72 years of age. Each action was repeated 8 times according to the order of action in Figure 7. As shown in Figure 7, the actions 1-4, 2-3, 5-6 are mutually and although the starting point of the actions 7-8 are the same, they continue in the opposite directions after the middle area.

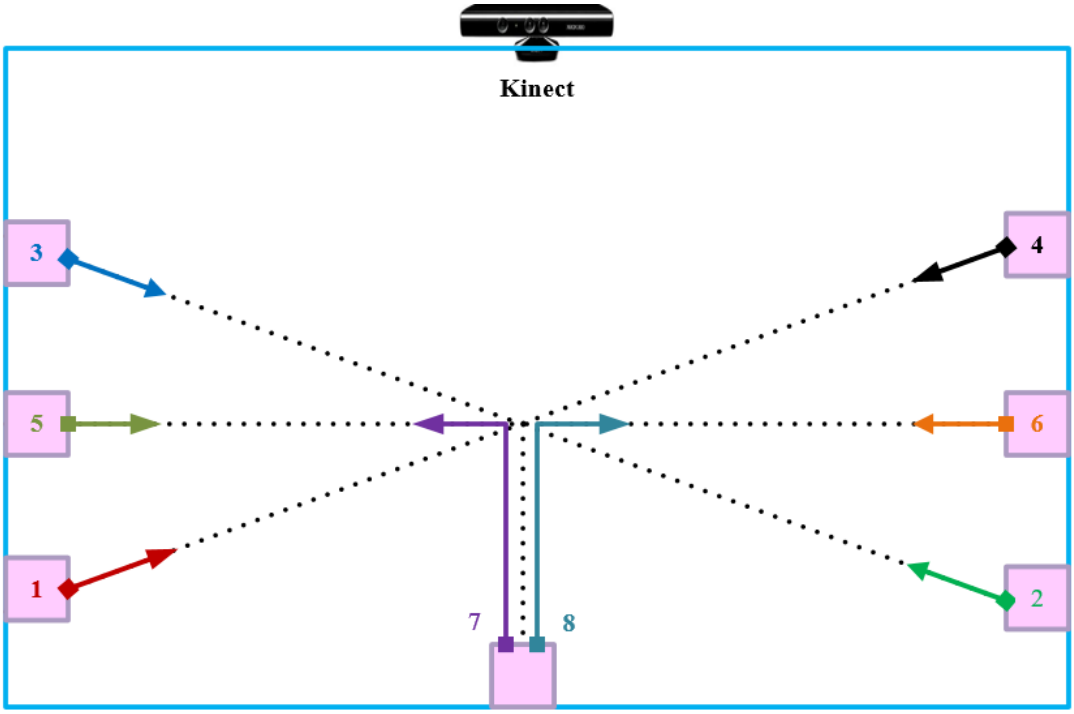

Figure 7. Dataset action directions

A total of 1008 depth videos and 3D coordinates (*x*, *y*, *z*) of 20 joints were recorded in total (6 actions × 8 repeat × 21 subjects). Each video duration is recorded as approximately 4-5 seconds, 320 × 240 resolution and 30 frames per second depending on the action feature. In this study, some skeletal and silhouette images of walking, bending, sitting, squatting, lying and falling are seen in Figure 8.

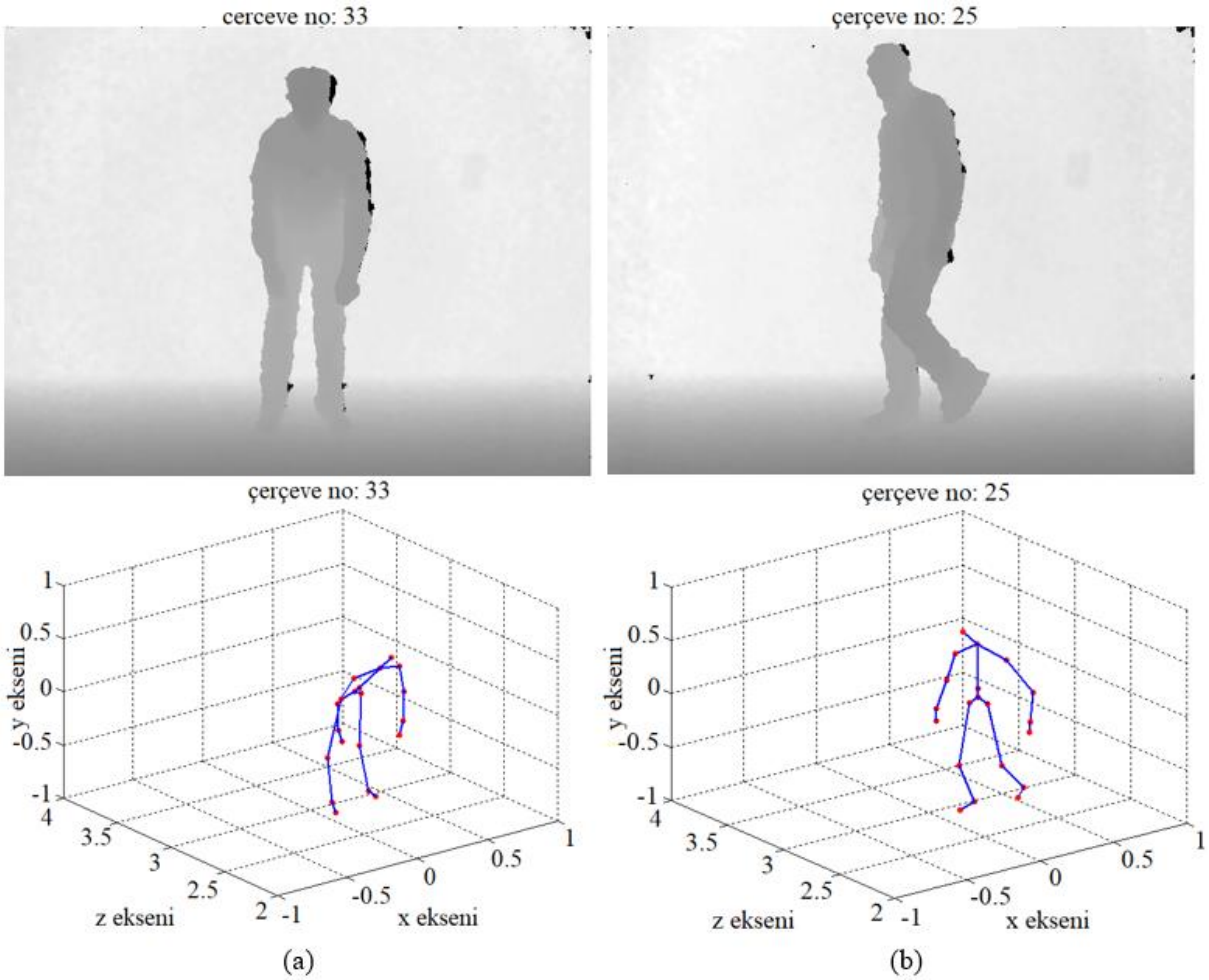

***Figure* 8.** Depth and skeleton illustrations of some actions of the dataset (a) Bending (b) Walking

3.2. Experimental Results

FUKinect-Fall and UTKinect-Action [31] datasets were used in experimental studies. In experimental studies performed with FUKinect-Fall, 2 × (360° / 5°) = 144, 72, 24 and 16 regions were divided respectively with 5°, 10°, 30° and 45° angles according to the skeleton model proposed in Figure 2. Separate attribute matrix created for each region angle. In order to classify the actions, a total of 152 attributes including 19 joints × 4 temporal structures and 2 axes were applied to the input layer of k-NN and SVM classifiers. Firstly, classification of 6 actions was performed with the proposed method. Three separate applications were also carried out in the classification of 6 actions. In the first application, 50% of the attributes were used for training and 50% for the test. In the second application, 75% of the data were used for training and 25% for testing, and in third practice, 90% of the data were used for training and 10% for testing. In the experimental studies, the results of the categorization of 6 actions with k-NN and SVM algorithms are shown in Table 1 and Table 2.

Table 1. Classification results of 6 actions with k-NN

Region Angle	k	1. Application	2. Application	3. Application
5°	1	88,13	88,75	90,63
	3	88,96	90,00	90,63
	5	87,29	91,25	91,67
	7	88,54	91,25	89,58
	9	87,08	90,83	91,67
	11	88,33	90,83	90,63
10°	1	88,13	88,33	88,54
	3	90,00	90,83	88,54
	5	90,63	91,67	91,67
	7	89,38	92,50	91,67
	9	88,33	92,08	91,67
	11	87,71	91,67	92,71
30°	1	82,29	84,17	86,46
	3	83,75	85,00	88,54
	5	85,83	87,08	85,42
	7	85,42	85,42	90,63
	9	83,96	85,42	87,50
	11	83,33	86,67	87,50
45°	1	85,21	86,67	91,67
	3	87,29	89,58	83,33
	5	87,08	88,75	84,38
	7	87,71	88,75	86,46
	9	87,08	89,17	86,46
	11	86,04	87,92	88,38

As shown in Table 1, in three applications, the best classification performance for k-NN was obtained with a 10° angle. In the first application, action classification was performed with 90.63% accuracy for k = 5, this value was 92.50% for k = 7, and 92.71% for k = 11.

In experimental studies conducted with SVM, it is necessary to select the most suitable SVM parameters to achieve high performance. In this context, 10-fold cross validation test which is mostly used in the literature is applied. According to the 10-fold cross validity test results, the experimental parameter was C = 200, the radial based function for kernel function and the sigma value of the function as 15.

Table 2. Classification results of 6 actions with SVM

Region Angle	1. Application	2. Application	3. Application
5°	84,79	88,75	**93,75**
10°	84,38	89,17	92,71
30°	82,79	87,92	90,63
45°	84,38	88,75	89,58

Similarly, when the classification results made with SVM are examined, it is observed that sensitivity is increased with decreasing region angle. Thus, 84.79% classification achievements for the first application, 88.75% for the second application and 93.75% for the third application were obtained.

In addition, the performance of the proposed method was tested with 10-fold cross validation analysis. In this context, the region angle was selected as 5° and k = 7 for the k-NN method in the first application. Also in the second application, for the SVM method, the region angle 5°, the radial based function and sigma value are selected as 15 and the results are shown in Table 3 and 4 respectively. As it can be seen from the results in Table 3, the act of lying out is 20,62% wrongly classified as an act of falling. The worst classification rate was found in lying 75.62% and the most successful classification rate was recorded in walking action with 98.12%.

Table 3. Confusion matrix obtained by the k-NN method and the 10-fold cross-validation test

Reality	Prediction					
	Falling	Bending	Lying	Sitting	Squatting	Walking
Falling	77,50	5,62	12,50	0,63	2,50	1,25
Bending	0,00	91,25	0,00	1,25	1,87	5,63
Lying	20,62	1,88	75,62	1,88	0,00	0,00
Sitting	0,00	1,88	0,00	91,87	3,12	3,13
Squatting	0,63	9,37	0,00	7,50	78,12	4,38
Walking	0,00	1,25	0,00	0,00	0,63	98,12

In Table 4, which shows the results of SVM, it is seen that most of the sitting and squatting activities are mixed with the action of falling. In other words, an average of 4.37% of the sitting and an average of 3.75% of the squatting action are identified as a falling. In addition, the highest accurate (96.25%) action was the action of walking, while the lowest correct classification performance (78.75%) was obtained for the act of lying. The act of lying out was most confused with the act of falling.

Table 4. Confusion matrix obtained by SVM method and 10-fold cross-validation test

Reality	Prediction					
	Falling	Bending	Lying	Sitting	Squatting	Walking
Falling	83,13	1,25	9,37	0,00	1,88	4,37
Bending	0,00	92,5	0,00	0,00	0,63	0,62
Lying	1,88	15,00	78,75	0,63	1,88	4,37
Sitting	4,37	0,00	0,63	88,13	2,50	4,37
Squatting	3,75	0,00	0,00	5,00	84,37	6,88
Walking	3,12	0,00	0,00	0,00	0,63	96,25

Actually, it is more important to differentiate between situations where there is no fall or fall. Here again, three separate applications were carried out, including the classification of 6 actions. The results of the experimental study with k-NN and SVM are given in Table 5 and Table 6, respectively.

Table 5. Classification results of non-falling and falling actions with k-NN

Region Angle	k	1. Application	2. Application	3. Application
5°	1	94,38	94,17	94,79
	3	93,33	93,75	95,83
	5	92,50	91,67	93,75
	7	95,00	95,00	94,79
	9	93,13	93,75	92,71
	11	90,83	92,08	91,67
10°	1	93,54	93,33	96,88
	3	92,50	95,00	95,83
	5	91,88	95,00	94,79
	7	92,92	93,33	94,79
	9	92,50	95,83	93,75
	11	91,67	93,33	92,71
30°	1	93,13	92,08	94,79
	3	93,54	90,42	93,75
	5	91,67	90,00	91,67
	7	92,71	92,08	93,75
	9	91,67	92,08	90,63
	11	90,83	90,00	88,54
45°	1	92,71	96,25	95,83
	3	91,25	95,42	91,67
	5	90,83	95,00	89,58
	7	92,50	96,25	93,75
	9	90,00	94,58	92,71
	11	88,33	93,33	92,71

When the results in Table 5 are examined, it is determined that 95% accuracy classification is used for the first application in the 5° zone angle for the first application, classification with 96.25% accuracy at best 45° angle and in the third application, a classification of 96.88% accuracy was obtained at a 10° angle.

Table 6. Classification results of non-falling and falling actions with SVM

Region Angle	**1.** Application	**2.** Application	**3.** Application
5°	95,00	97,08	97,92
10°	94,37	97,08	97,92
30°	93,96	95,83	96,88
45°	93,96	97,08	95,83

It is seen that in classification of the actions with no fall or fall (walking, bending, sitting, squatting and lying) with SVM the ratio of 95% accuracy in the angle of 5° in the first application and in the second application 97.08% accuracy in the other region angles except than 30° fall action detection is obtained. In addition, in the third application 5°, 10° region angle 97.92% accuracy of the fall action detection is observed.

In order to better compare the performance evaluation of the proposed method, experimental studies were conducted with the UTKinect-Action dataset, which is widely used by many researchers as well as the dataset prepared by us in experimental studies. Table 7 shows the comparison of other methods using this dataset and the proposed method with respect to the average performance in the recognition of the 10 actions in the dataset.

Table 7. Performance results of studies with UTKinect-Action dataset

	Zhu et. [32]	Yang et. [33]	Proposed Method
Accuracy	%87,90	%88,90	**%90,03**

As shown in Table 7, It is observed that it has about 1% better success than the study of Zhu et al. and about 2% better success than the study and Yang et al. This suggests that the proposed method can also be used effectively for action recognition using skeletal data.

4. CONCLUSIONS

In this study, a new method for the classification of daily human actions, especially falling action, was proposed using skeletal data. For this aim, a new set of, data called FUKinect-Fall, which primarily involves falls daily action was created. In the experimental studies of the proposed method with FUKinect-Fall, fall detection with 97.92% accuracy was performed. When compared with the other studies realized only with depth images 4% better results was obtained.

Although our main aim was to determine fall, the performance evaluation of the method could not be performed due to the lack of skeleton data in the existing fall datasets. Performance evaluation of the method with the proposed method using the UTKinect-Action dataset which commonly used skeletal data in recognition of action was performed. Approximately %1 better success in action recognition when compared to studies realized with same data proves the success of the proposed method.

Moreover, FUKinect-Fall dataset including both the skeleton and depth data that can be quite useful for the researchers working on the detection of fall detection was created.