IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 9, 2019, accepted January 9, 2020, date of publication January 21, 2020, date of current version January 31, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2968529

A Hybrid Deep Model Using Deep Learning and Dense Optical Flow Approaches for Human Activity Recognition

SENEM TANBERK[1], ZEYNEP HILAL KILIMCI[1,4], DILEK BILGIN TÜKEL[1,2], MITAT UYSAL[1], AND SELIM AKYOKUŞ[3]

[1]Department of Computer Engineering, Doğuş University, 34660 Istanbul, Turkey
[2]Future Systems, Altinay Robot Technologies, 34957 Istanbul, Turkey
[3]Department of Computer Engineering, Istanbul Medipol University, 34810 Istanbul, Turkey
[4]Department of Information Systems Engineering, Kocaeli University, 41001 Izmit, Turkey

Corresponding author: Zeynep Hilal Kilimci (hkilimci@dogus.edu.tr)

ABSTRACT Human activity recognition is a challenging problem with many applications including visual surveillance, human-computer interactions, autonomous driving and entertainment. In this study, we propose a hybrid deep model to understand and interpret videos focusing on human activity recognition. The proposed architecture is constructed combining dense optical flow approach and auxiliary movement information in video datasets using deep learning methodologies. To the best of our knowledge, this is the first study based on a novel combination of 3D-convolutional neural networks (3D-CNNs) fed by optical flow and long short-term memory networks (LSTM) fed by auxiliary information over video frames for the purpose of human activity recognition. The contributions of this paper are sixfold. First, a 3D-CNN, also called multiple frames is employed to determine the motion vectors. With the same purpose, the 3D-CNN is secondly used for dense optical flow, which is the distribution of apparent velocities of movement in captured imagery data in video frames. Third, the LSTM is employed as auxiliary information in video to recognize hand-tracking and objects. Fourth, the support vector machine algorithm is utilized for the task of classification of videos. Fifth, a wide range of comparative experiments are conducted on two newly generated chess datasets, namely the magnetic wall chess board video dataset (MCDS), and standard chess board video dataset (CDS) to demonstrate the contributions of the proposed study. Finally, the experimental results reveal that the proposed hybrid deep model exhibits remarkable performance compared to the state-of-the-art studies.

I. INTRODUCTION

Board games have been played for centuries in all cultures and societies. There are elegant rules, deep strategies, and numerous tactical possibilities involved in such games. Chess is one of the most popular board game in the world. The World Chess Federation (FIDE) informs that 605 million people play chess [1]. Chess has always been a challenge for computer hardware designers and software developers. For this purpose, robot systems interacting with people in complex environments need the capability to correctly interpret and respond to human behaviors. Entertainment is a natural way of integrating social robots into our lives. Playing chess with a social robot requires the recognition of human behavior using computer vision. Thus, chess is an attractive topic for research on human-machine interaction. It is necessary to produce solutions in areas such as image processing, strategy establishment, and others to solve chess problems.

Human activity recognition (HAR) is a challenging task that provides recognition of activities in complex interactions without verbal communication. HAR has numerous potential applications such as in devices to determine the communication between humans and the environment, surveillance systems, video understanding applications including online advertising, or video retrieval and video surveillance. Depending on their complexity, human activities are categorized as gestures, atomic actions, human-to-object or human-to-human interactions, group actions, behaviors and events [2]. A software system that handles HAR is required, which should perform three functions: background

The associate editor coordinating the review of this manuscript and approving it for publication was Shuhan Shen.

subtraction to separate the parts of the image that are invariant overtime; human tracking in which the system locates human motion; and human action and object detection, in which the system is able to localize human activity in an image [3]. In order to localize action recognition, there are local representations such as the pipeline of interest point detection, local descriptor extraction and aggregation of local descriptors.

Local descriptors based on pixel values and optical flow are used for HAR tasks. Local representations for action recognition on Space-Time Interest Points (STIPs) follow the pipeline of interest point detection, local descriptor extraction and aggregation of local descriptors. Klaser *et al.* [4] suggest the histogram of gradient orientations as a motion descriptor. Optical flow is the distribution of apparent velocities of movement in captured imagery data. The motions are extracted by comparing two images, which might be considered between two images captured at two different times (temporal) or two images captured at exactly the same time but using two different cameras with known camera parameters (static optical flow). Generally, optical flow is a 2D vector where each vector is a displacement vector showing the movement of pixels from the first frame to the second in the perpendicular image axes. The Farnebäck method [5] is a two-frame motion estimation algorithm that uses polynomial expansion, where a neighborhood of each image pixel is approximated by a polynomial. In this work, we focus on quadratic polynomials, which give the local signal model represented in a local coordinate system.

The application of deep learning (DL) algorithms has become very popular in recent years in different research fields such as image processing, natural language processing, speech recognition, and machine translation. Deep neural network models are also now being also proposed for human activity recognition field. DL methods are preferred by researchers because they provide better predictions and results compared with traditional machine learning algorithms. Deep learning models are mainly used to provide automatic feature extraction by training complex features with minimal external support to obtain meaningful representation of data through deep neural networks. Furthermore, deep learning methods are also employed for the purpose of classification tasks in many fields. Convolutional neural networks (CNNs), recurrent neural networks (RNNs), long short-term memory networks (LSTMs) and deep belief networks (DBNs) are well-known architectures. In this work, a novel technique is presented for recognizing human activity from videos using deep learning algorithms.

In this study, the main purpose is to design a hybrid video-based HAR classification system for complex interactions inspired by study in [6]. To the best of our knowledge, this is the first study to consolidate the optical flow and auxiliary movement information in video by using deep learning models. Furthermore, the hybrid deep model classifies videos and generates video subtitles in addition to the consolidation process. Because it can determine many complex activities such as playing chess, playing checkers, playing peg solitaire, or playing cards, the proposed system is referred to as flexible. It can also be called as extendable because of adding any number of features as auxiliary information in long-short term memory networks. The system is simulated for recognizing the playing of chess in order to demonstrate the efficiency of the proposed model. For this purpose, a 3D-CNN model for the optical flow part and the LSTM algorithm for auxiliary information in videos are employed in the feature extraction step. Hand tracking, movement based on chessboard recognition, and chess pawn recognition are evaluated as auxiliary information in LSTM. After that, Optical Flow based extracted features with the CNN and Hand Tracking - Chessboard Recognitions - Chess Pawn Recognition based extracted features are consolidated in our proposed system. After the feature extraction step, Support Vector Machine (SVM) algorithm is utilized for video classification purposes. To demonstrate the effectiveness of the proposed system, two types of chess playing datasets are employed in the experiments.

The rest of this paper is organized as follows: Section 2 gives a summary of related work about human activity recognition, optical flow, and video classification. Section 3 explains deep learning methods used in the experiments. Section 4 describes the proposed framework. Sections 5 and 6 present the experimental setup, and experimental results, and the conclusions, respectively.

II. RELATED WORK

This section provides a brief summary of deep learning models related to human activity recognition, optical flow, and video classification in the literature.

The traditional approach for video classification [7]–[10] has three phases: First, visual features in videos are extracted densely [11], or sparsely [12], [13]. Next, fixed-sized video-level description is obtained. Lastly, a classifier like SVM is used to get video classes. In [14], the authors are inspired by CNNs [15] to replace all three phases with a single neural network. They model videos with convolutional neural networks in a very similar way to model images of CNNs for video classification. CNNs are directly applied to the multiple frames. Experiments are carried out on the UCF-101 action recognition dataset. They apply video based deep learning techniques with CNNs to analyze videos containing human motions and observe significant performance improvements compared to the baseline model.

Simonyan and Zisserman [16] propose a deep video classification model to incorporate spatial and temporal information based on ConvNets. Two-stream CNNs are employed in the proposed approach. One stream takes RGB image frames as input while the other stream takes stacked optical flow by computing. Standard action recognition datasets, namely UCF-101 and HMDB-51, are evaluated and tested to show the contributions of their approach. Despite limited training data, they demonstrate that the proposed deep model achieves remarkable classification performance. RNNs are

also considered for video-based HAR. In [17], Arif et al. combine 3D-CNN and LSTM for human activity recognition task effectively. They introduce a 3D-Conv-based model and its iterative training method to integrate distinctive information in a video into motion maps. They use the LSTM encoder/decoder to obtain video level representations to help recognize complex frame-to-frame hidden sequential patterns for video classification. They test the proposed model with benchmark datasets and achieve a promising performance. In [18], video classification and captioning is implemented with 3D-CNN and LSTM networks in one architecture. The authors explore methods and architectures to understand how to categorize and caption videos, automatically. Experimental results demonstrate that the proposed multi-task architecture exhibits significant advantages for complex video captioning models. With this architecture, the image captioning dataset can be trained in the first stage, and then pre-trained weights can be used during the video captioning model training phase. The proposed architecture gives better results than end-to-end training on video captioning datasets and saves training time.

In [19], a system framework is presented to recognize human activities in videos by an SVM multi-class classifier with a binary tree architecture. Hierarchical classification is introduced and multiple SVMs are aggregated to recognize multiple actions. Each SVM in the multi-class classifier is trained, separately. The proposed multi-class SVM model is applied to both a home-brewed activity dataset and Schüldt's public dataset. Experimental results demonstrate that the usage of the proposed architecture yields exceptional identification performance. In [11], dense optical flow trajectories are employed for action recognition tasks. Wang et. al. propose an approach to describe videos by dense trajectories inspired by the success of dense sampling in image classification. They introduce a novel descriptor that is robust to camera motion based on motion boundary histograms. The proposed descriptor model outperforms state-of-the-art descriptors. The KTH, YouTube, Hollywood2, and UCF sports datasets are employed to demonstrate the efficiency of the proposed model in the experiments. The works reported in [20] and [13] are also examples of optical flow based approaches. In [21], 3D CNNs and SVM are used to recognize human action in videos. First, 3D CNNs are used to extract spatial and temporal features from consecutive video frames. The SVM approach is then used in order to classify videos. The proposed architecture is trained on the KTH action recognition dataset. Experimental results show that the proposed architecture achieves significant classification performance.

Depending on the structure of the deep learning network, the main representative works can be summarized as in [22] as methods based on two-stream convolutional networks, those based on 3D convolutional networks, and those based on long short-term memory (LSTM). In a two-stream convolutional network, the optical flow information is calculated from the image sequence. The image and optical flow sequence are respectively used as the input to the two convolutional neural networks (CNNs) during the model training process. Fusion occurs in the last classification layer of the network. The inputs of the two-stream network are a single-frame image and a multi optical flow frame image stack, and the network applies 2D image convolution. In contrast, a 3D convolution network regards the video as a 3D space-time structure, and uses a 3D convolution method to learn human action features.

Our work differs from the above mentioned studies in that this is the first work of employing a hybrid model for the purpose of human activity recognition. Unlike the state-of-the-art studies, a novel hybrid model is constructed blending deep learning models including 3D convolutional neural networks, dense optical flow, and LSTM to enhance the classification success of the proposed system. The details of the proposed framework can be found in Section 3.

III. METHODOLOGIES USED IN PROPOSED FRAMEWORK

This section introduces the optical flow approach, deep learning techniques such as convolutional neural networks (CNNs), and long short-term memory networks (LSTMs), support vector machines (SVMs) used in the experiments.

A. FEATURE EXTRACTION

1) CONVOLUTIONAL NEURAL NETWORKS (CNNs)

CNN is widely known and commonly used deep learning model in image recognition, image classification, video classification, visual recognition, and natural language processing [14]–[18], [21]–[25]. CNN is common as a class of deep learning networks. CNN is also a feedforward neural network with an input and output layer, and hidden layers. Hidden layers occur from convolutional layers combined with pooling layers. The convolution layer is the most significant part among the blocks of a CNN. A convolution filter is performed in the convolutional layer to input data to produce a feature map to consolidate information with data on the filter. Multiple filters are implemented to input data to get a stack of feature maps that transforms the final output of the convolutional layer. Local dependencies are obtained with a convolution operation in the regions of original data. Moreover, a supplementary activation function such as a rectified linear unit is applied to feature maps for the purpose of associating non-linearity with the CNN. Next, the number of samples in each feature map is diminished by a pooling layer, which holds the most significant information. In this way, the training time decreased while the dimensionality and over-fitting processes are reduced with the usage of the pooling layer. Max-pooling is a widely-used type of pooling function that obtains the largest value in a specified neighborhood window. CNN architectures are based on a series of convolutional layers blended with pooling layers, followed by a number of fully connected layers.

In this work, a 3D-CNN version is employed in the experiments. Each input feature map is convolved employing a shifting window with a K x K kernel in order to generate the

one pixel in one output feature map [26]. 3D convolutional layers are also used to capture the motion information from multiple stacked frames [6]. The value of the kth 3D feature map for the first convolutional layer is given in Equation (1) and Equation (2):

$$v_1^k = \sigma(W_1^k * x + b_1^k) \quad (1)$$
$$v_j^k = \sigma(W_j^k v_i + b_j^k) \quad (2)$$

Here W_1 represents the filter weights, x is the input frame, b_1 is the bias, $*$ is the 3D convolution operation and σ is the activation function used in the current convolutional layer [21].

2) LONG SHORT-TERM MEMORY NETWORKS (LSTMs)

Recurrent neural network (RNN) is a type of neural network in which the output from the previous step is sent as input to the current step. The need of RNN is arised with the requirement of remembering the previous steps. This procedure is provided with the help of a hidden layer as mentioned in [27], [28]. The most significant feature of an RNN is located in the hidden state, which remembers some information about a sequence. In this way, an RNN has a memory that recollects all information about what is computed. Unlike other neural networks, RNN concentrates on decreasing the complexity of parameters. RNN implements the same task on all inputs or hidden layers to generate the output. In this way, the usage of the same parameters for each input diminishes the complexity of the parameters.

Long short-term memory network (LSTM) another popularly employed technique, which are designed as a special type of recurrent neural networks to solve vanishing gradient issues of RNNs [29], [30]. LSTM preserves the error to back-propagate through deeper layers and proceeds to learn over many time steps. Actually, LSTMs are improved to obtain long-distance dependencies within sequence data. The contextual semantics of information are retained and stored for the purpose of obtaining long dependencies between data with the LSTM approach. For this purpose, special memory units are exploited to stock information for dependencies in a long range context in LSTM. Each LSTM unit contains input, forget, and output gates to check which fragments of information to remember, forget and pass on to the next step. In this way, LSTM gains the capability to make decisions about what to store, and when to permit reads, writes and deletions by via gates that pass or block information through the LSTM unit. In this work, LSTM is employed to obtain auxiliary information from hand tracking, and movement based on chessboard recognition. In summary, 3D-CNN model for the optical flow approach and an LSTM algorithm for auxiliary information in video are employed in the feature extraction step.

3) OPTICAL FLOW (OF)

The state-of-the art studies focus on object detection for detecting instances of objects belonging to a certain class and semantic segmentation for pixel-wise classification. Most applications of these models merely exhibit the relationships of objects within the same frame (x,y) paying no attention to the time information (t) in real-time videos. This means that each frame is evaluated independently, as if the images are entirely irrelevant for each other at each epoch. This can be a problem if there is a relationship between consecutive frames when the motion of vehicles across frames is being tracked in order to forecast current velocity and estimate the position in the next frame. To eliminate this issue, the concept of optical flow was proposed by James J. Gibson in the 1940s [31].

Optical flow is employed in many research areas comprising techniques for image processing and control of navigation including motion detection, object segmentation, time-to-contact information, expansion calculations, luminance, motion compensated encoding, and stereo disparity measurement [32], [33]. Basically, optical flow is the modeling of the pronounced movement of surfaces, edges, and objects in a visual occurrence induced by the relative motion between an observer and a scene [34], [35]. Optical flow is also known as the distribution of obvious velocities of the motion of brightness model in an image. There are two main versions of the optical flow approach which are sparse and dense. Sparse optical flow employs the flow vectors of some features such as a few pixels depicting the edges or corners of an object within the frame while dense optical flow utilizes the flow vectors of the entire frame (i.e. all pixels) up to one flow vector per pixel. Because of this, dense optical flow demonstrates higher accuracy while it has computationally expensive cost compared to the sparse optical flow as expected. In this study, we concentrate on dense optical flow for these reasons.

An optical frame vector is determined for every pixel of each frame in the dense optical flow approach. As mentioned above, more accurate results are obtained with this approach while the computation time is slower compared to the other method. Furthermore, denser results are obtained which are suitable for applications such as learning structures from motion and video segmentation in terms of the usage of dense optical flow. There are different implementations of dense optical flow. In this study, we concentrate on the Farnebäck method which is the most commonly implemented one, using OpenCV, an open source library of computer vision algorithms. All the points in the frame are calculated in the Farnebäck's approach. It is based on Gunnar Farnebäck's algorithm, proposed to implement two-frame motion estimation based on polynomial expansion [5].

B. VIDEO CLASSIFICATION

1) SUPPORT VECTOR MACHINES (SVMs)

Support vector machine (SVM) is actually a binary classifier that separates an N-dimensional space with n features into two regions related to two classes [36]. The n-dimensional hyperplane divides two regions in such a way that the hyperplane has the largest distance from training vectors of two classes called support vectors. SVM is also employed for the purpose of non-linear classification using a method called

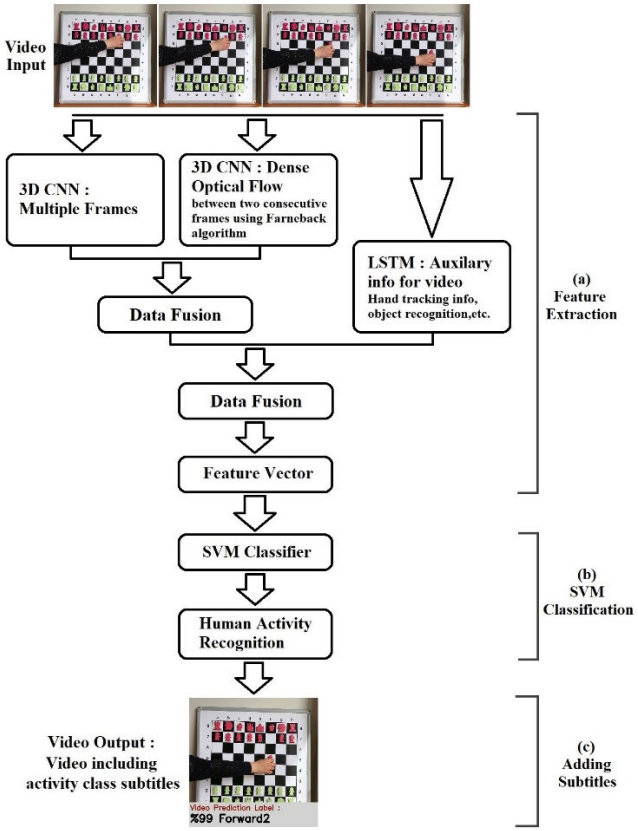


FIGURE 1. Flowchart of the proposed system.

the kernel trick that implicitly maps input instances into high-dimensional feature spaces that can be separated linearly. In an SVM, the usage of different kernel functions facilitates the construction of a set of diverse classifiers with different decision boundaries. In this work, a radial basis function (RBF) kernel is used for the classification task. In summary, the objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space where N is the number of features that distinctly classify the data points.

For a given dataset $A\{(x_1, y_1), (x_2, y_2), \ldots (x_n, y_n)\}$ where $x_i \epsilon R^n$; $y_i \epsilon \pm 1$, and i represents a label associated with each action in our dataset, the set of all hyperplanes is defined in Equation (3) and Equation(4).

$$w.x_i + b \geq +1; \quad y_i = +1 \tag{3}$$

$$w.x_i + b \leq -1; \quad y_i = -1 \tag{4}$$

$$Minimize: \quad \| w \| \quad subject\ to\ y_i\,(w.x + b) \geq 1 \tag{5}$$

Maximizing the distance between the hyperplanes requires minimizing ‖w‖. Therefore, this is an optimization problem and written as in Equation (5).

IV. PROPOSED SYSTEM

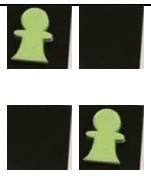
A hybrid deep model is proposed to evaluate human activity recognition in videos as seen in Figure 1. For this purpose, there is a need to determine motion vectors located in frames t and t−1. 3D-CNN which is called multiple frames, 3D-CNN with dense optical flow, and LSTM model are combined to construct the feature vectors. To the best of our knowledge, this is the first attempt to recognize human activity using the combination of multiple frames, optical flow, and LSTM. As a first step, features are extracted with 3D-CNN, 3D-CNN with dense optical flow, and LSTM from video and located in arrays. In 3D-CNN, multiple frames are evaluated as input in frame t by decreasing the resolution due to the size limitation. Moreover, the other inputs are gathered from the combination of 3D-CNN, and dense optical flow in frame t−1 by using the Farnebäck algorithm. As a last step, LSTM is employed as another component of the proposed system in the feature extraction step to provide auxiliary information for hand-tracking and objects. The LSTM is fed a model with a single shot detector (SSD) network architecture for hand tracking purposes and another neural network embedded in TensorFlow Object Detection API [37] for object recognition. After obtaining features from each separate model, the features collected from 3D-CNN model and the features gathered from the 3D-CNN with the dense optical flow model are concatenated. The new feature vector is then enhanced with the new features acquired from the LSTM model. In summary, a novel feature vector is constructed using 3D-CNN, 3D-CNN with dense optical flow, and LSTM as shown in Figure 1.

In order to recognize human activity, the classification task is performed by using support vector machine (SVM) algorithm. Before implementing the SVM, the classes of motions, which are called activity classes in this study, are specified in the magnetic wall chess board dataset (MCDS) and the standard chess board dataset (CDS). The activity classes of the proposed system are Forward1, Forward2, DiagonalLeft, DiagonalRight, and Trash as observed in Table 1. We then focus on the classification task to determine the class of motion or activity when the motion whose previous class label is actually known is encountered in the video. In this way, the classification performance of the proposed system is evaluated with precision, accuracy, recall, and F-measure as the evaluation metrics to demonstrate the contribution of our work.

V. EXPERIMENTS

Extensive comparative experiments are carried out on the Ubuntu 16.04 system with an Intel Core i7 machine and NVIDIA GTX1080 Ti GPU. The needed libraries are developed with Python. Magnetic wall chess board dataset (MCDS) and standard chess board dataset (CDS) are the datasets used in the experiments. We randomly divide the datasets into two parts whereby 70% of the data are used for training and 30% for testing. First, videos are gathered with 1080×1080 resolution format, but each frame is resized to 50×50 resolution due to size limitations. The OpenCV library is employed to read and write videos in experiments in the feature extraction step. A total of 100 frames for each video instance are extracted using TensorFlow Object Detection

TABLE 1. Examples of activity classes for magnetic wall chess board dataset (MCDS) with each corresponding video frame.

Activity Class	Frame Before Motion	Starting Frame of Movement	End Frame of Movement	Frame After Motion	Aux : Crop First/Last Cell
Forward1					
Forward2					
DiagonalLeft					
DiagonalRight					
Trash					

API [37] for custom objects including pawn recognition, patterns in chess boards [38], and real time hand detection with single shot detector (SSD) network architecture in TensorFlow [39], [40]. Furthermore, the Keras functional api is utilized to construct the proposed hybrid deep model in the feature extraction step. In order to handle overfitting, we used a dropout technique including both SpatialDropout3D and Dropout.

A. DATASETS

The human activity recognition is performed by generating datasets with data collected from mobile phones. For this purpose, three different datasets are employed, including a pawn dataset for a magnetic wall chess board in object recognition, a video dataset for a magnetic wall chess board, and a video dataset for a standard chess board. After collecting videos, VivaVideo for formatting and Bandicut for splitting videos into meaningful parts are employed in the preprocessing phase. Data preparation is conducted as described in detail below:

Data Preparation for Magnetic Wall Chess Board Video Dataset (MCDS): The dataset is preprocessed before the feature extraction step by applying the following phases:

1. Split all the videos into basic activity folders as DiagonalLeft, DiagonalRight, Forward1, and Forward2. These activity folders are class labels at the same time.
2. Extract as.jpg format of each frame for each video into a tmp_xxx folder.
3. Summarize the videos as their classes, frame numbers for some statuses, etc. in a CSV file.
4. Based on frame numbers in the CSV file:
   - 4.1. Split all the videos into all activity folders: DiagonalLeft, DiagonalRight, Forward1, Forward2, Trash.
   - 4.2. Extract as.jpg format for the empty chessboard for each video.
   - 4.3. Extract as.jpg format for the chessboard image before movement starts for each video.
   - 4.4. Extract as.jpg format for chessboard image after movement ends for each video.
   - 4.5. Crop as.jpg for the chessboard's first processed cell and second processed cell before movement starts for each video.
   - 4.6 Crop as.jpg for the chessboard's first processed cell and second processed cell after movement ends for each video.

TABLE 2. Confusion matrix.

	Class 1 Predicted	Class 2 Predicted
Class 1 Actual	TP (true positive)	FN (false negative)
Class 2 Actual	FP (false positive)	TN (true negative)

5. The feature extraction step is run.

Data Preparation for Standard Chess Board Video Dataset (CDS): The dataset is processed before applying feature extraction as follows:

1. Split all the videos into activity folders as DiagonalLeft, DiagonalRight, Forward1, Forward2. These activity folters are class labels at the same time.
2. Extract as.jpg format of each frame for each video into a tmp_xxx folder.
3. Summarize the videos as their classes, frame numbers for some statuses, etc. in a CSV file.
4. Extract as.jpg format for the empty chessboard for each video based on frame numbers in the CSV file.
5. The feature extraction step is run.

B. EXPERIMENTAL RESULTS

The classification performance of the proposed system is evaluated with precision, accuracy, recall, and F-measure as evaluation metrics using the confusion matrix seen in Table 2 to demonstrate the contribution of our work.

Precision, Recall, Classification Rate/Accuracy, and F-measure are calculated as follows:

$$\begin{aligned} Precision &= \frac{TP}{TP + FP} \\ Accuracy &= \frac{TP + TN}{TP + TN + FP + FN} \\ Recall &= \frac{TP}{TP + FN} \\ F\text{-}measure &= \frac{2 * Recall * Precision}{Recall + Precision} \end{aligned} \tag{6}$$

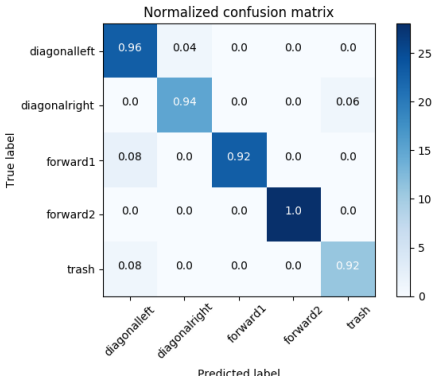
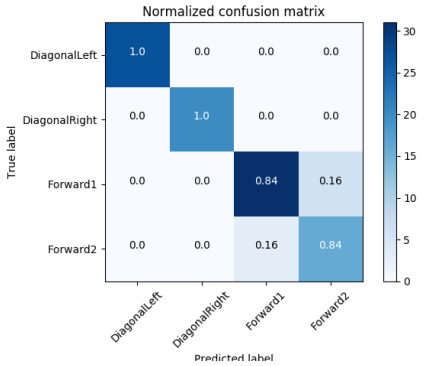
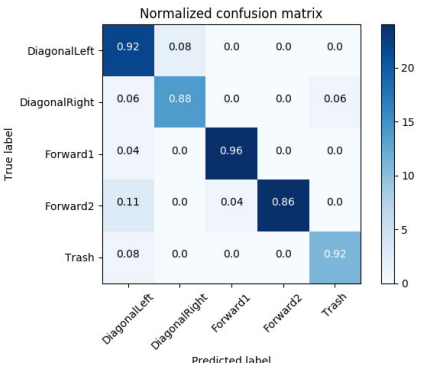
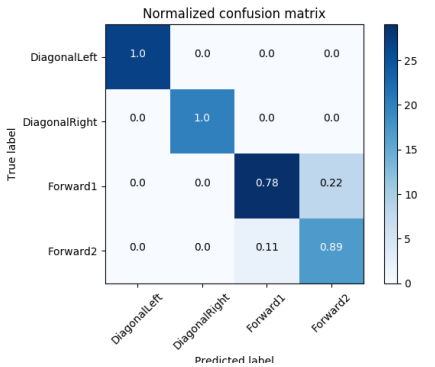
In all tables, abbreviations are employed for deep models as follows: HDM: Flexible hybrid deep model which is the proposed model in this work, CNN-OF: Convolutional neural network model combined with dense optical flow which is a sub-model of HDM, HDM – MCDS: The classification performance of HDM on the MCDS dataset, HDM – CDS: The classification performance of HDM on the CDS dataset, CNN-OF – MCDS: The classification performance of CNN-OF on the MCDS dataset, CNN-OF – CDS: The classification performance of CNN-OF on the CDS dataset. The best classification results acquired are indicated in bold in the tables. First, we analyze the classification performance of the proposed system in terms of precision, recall, and F-measure results for different human activities on magnetic wall chess board (MCDS) and standard chess board (CDS) datasets using HDM and CNN-OF.

TABLE 3. Precision, recall, and F-measure results of the proposed HDM model and CNN-OF for different activities on the MCDS and CDS datasets.

		MCDS			CDS		
Model	**Activity Class**	**Precision**	**Recall**	**F-measure**	**Precision**	**Recall**	**F-measure**
HDM	DiagonalLeft	0.88	0.96	0.918	1.00	1.00	1.00
	DiagonalRight	0.94	0.94	0.94	1.00	1.00	1.00
	Forward1	1.00	0.92	0.958	0.91	0.84	0.874
	Forward2	1.00	1.00	1.00	0.73	0.84	0.781
	Trash	0.92	0.92	0.92	NA	NA	NA
CNN-OF	DiagonalLeft	0.79	0.92	0.85	1.00	1.00	1.00
	DiagonalRight	0.88	0.88	0.88	1.00	1.00	1.00
	Forward1	0.96	0.96	0.96	0.94	0.78	0.853
	Forward2	1.00	0.86	0.925	0.68	0.89	0.771
	Trash	0.92	0.92	0.92	NA	NA	NA

Table 3 demonstrates the classification results of different activities in terms of precision, recall and F-measure for the proposed models on the MCDS and CDS datasets. Forward2 movement as a human activity is recognized with 100% classification success with the proposed HDM model on the MCDS dataset when considering the F-measure results. It is followed by Forward1 with 95.8%, DiagonalRight with 94.0%, Trash with 92.0%, and DiagonalLeft with 91.8% for the MCDS dataset. With the CNN-OF model, the best classification success is achieved by Forward1 movement with 96.0% performance on the MCDS dataset. The classification success of human activities for the CNN-OF model on the MCDS dataset can be summarized as follows: Forward1 > Forward2 > Trash > DiagonalRight > DiagonalLeft. As a result, usage of the HDM model can provide 100% classification success on the MCDS dataset. On the other hand, DiagonalLeft and DiagonalRight movements are recognized with 100% classification performance with the HDM method on the CDS dataset. The classification performances for the CDS dataset are ordered as follows: DiagonalLeft = DiagonalRight > Forward1 > Forward2 > Trash. In the CNN-OF model, the best classification success is shared between DiagonalLeft and DiagonalRight with 100% classification performance as observed before in the HDM model. However, with the CNN-OF model, the Forward1 and

TABLE 4. Confusion matrix results in term of recall metric.

Model	MCDS	CDS
HDM		
CNN-OF		

Forward2 movements exhibit 2% and 1% lower classification success, respectively when compared to the HDM model. As seen in Table 3, the proposed HDM model generally exhibits remarkable classification success compared to the CNN-OF method in terms of F-measure results.

Table 4 demonstrates confusion matrix results for both MCDS and CDS datasets by comparing the HDM and CNN-OF models in terms of recall results. In this table, the x-axis denotes the predicted labels while the y-axis represents the actual labels.

- HDM – MCDS: Maximum confusion is observed between Forward1 and DiagonalLeft movements whereas the best results are achieved by the Forward2 class.
- HDM – CDS: Maximum confusion is seen between Forward1 and Forward2 activities while DiagonalLeft and DiagonalRight movements yield the best classification results.
- CNN-OF – MCDS: Forward2 and DiagonalLeft represent the maximum confusion while the Forward1 and Trash activity classes yield the best results.
- CNN-OF – CDS: Between Forward1 and Forward2 movements, maximum confusion is clearly observed while DiagonalLeft and DiagonalRight activities exhibit the best classification performances.

Some classes with similar activities are more readily confused with each other. A possible reason for them interfering with each other in this way is the similarity of features and representations among the activities.

Table 5 displays the overall accuracy results of the HDM and CNN-OF models on the MCDS and CDS datasets. In order to compare our proposed model with the CNN-OF model trained with our new datasets, the overall accuracy of the proposed system is calculated using Equation (6). The proposed method achieves 95.2% accuracy on the MCDS and 91.3% accuracy on the CDS dataset while the CNN-OF model exhibits 90.5% accuracy on MCDS and 90.3% accuracy on CDS. This means that the newly proposed hybrid deep model outperforms CNN-OF with nearly 5% enhancement on the MCDS dataset and with 1% improvement on the CDS dataset. The performance of the proposed HDM model on the MCDS dataset is marked by almost 4% more improvement compared to the CDS dataset. The main reason for this result is that there is no auxiliary information contained in the CNN-OF model, and the auxiliary information of the MCDS dataset used in the HDM model is more than that of the CDS dataset. As seen in Table 5, the usage of the HDM model improves the classification performance of the system thanks to the auxiliary information.

Table 6 provides the accuracy results of the proposed HDM model and CNN-OF for different activities on the MCDS and CDS datasets with accuracy results for each class.

TABLE 5. Overall accuracy results of HDM and CNN-OF models on MCDS and CDS datasets.

Model	MCDS	CDS
HDM	**0.952**	**0.913**
CNN-OF	0.905	0.903

TABLE 6. Accuracy results of the proposed HDM model and CNN-OF for different activities on MCDS and CDS datasets.

Model	Activity Class	MCDS Accuracy	CDS Accuracy
HDM	DiagonalLeft	0.962	1.000
	DiagonalRight	0.980	1.000
	Forward1	0.980	0.913
	Forward2	1.000	0.913
	Trash	0.980	NA
	OverAll Accuracy	**0.952**	**0.913**
CNN-OF	DiagonalLeft	0.922	1.000
	DiagonalRight	0.960	1.000
	Forward1	0.979	0.903
	Forward2	0.960	0.903
	Trash	0.979	NA
	OverAll Accuracy	**0.905**	**0.903**

• DiagonalLeft: HDM outperforms CNN-OF with approximately 4% enhancement on the MCDS dataset for DiagonalLeft class. Both models show the same classification success on the CDS dataset for DiagonalLeft class.

• DiagonalRight: HDM represents almost 2% improvement compared to CNN-OF on the MCDS dataset for the DiagonalRight class. Both models exhibit the same classification performance on the CDS dataset for DiagonalRight class.

• Forward1: The classification performances of the models are almost the same for the MCDS dataset for the Forward1 class. For the CDS dataset, the performance difference between the two models is nearly 1% for the Forward1 class.

• Forward2: HDM surpasses from CNN-OF by 4% enrichment in classification accuracy on the MCDS dataset for the Forward2 class. On CDS dataset, the classification improvement of the HDM model reaches 1% compared to CNN-OF for the Forward1 class.

TABLE 7. Video classification result for DiagonalLeft movement.

TABLE 8. Video classification result for Forward2.

Sample results of video output are demonstrated in Table 7 and 8 for the DiagonalLeft and Forward2 classes as inspired by [41]. Moreover, human activity class names with their precision rates are embedded in video as subtitle. Thus, new video is generated with class subtitle as output video.

It is difficult to compare the performance of our proposed model with other studies in the literature because of the lack of works with similar combinations of deep learning architectures in the feature extraction step as in the HDM model, and also due to the newly produced datasets. Although the datasets employed in the experiments are different, the performance of the CNN-OF model can be compared with the state-of-the-art study in [21]. For the CNN-OF model which is the same as that in [21], the classification accuracies of our system suggest almost the same classification performance with 90.5% accuracy on MCDS and 90.3% accuracy on CDS while the study presented in [21] yields 90.34% accuracy results. For the CNN-OF model, the performance of the system is consistent because of the similarity of the accuracy results in both studies (ours and [21]) even if the datasets are different. On the other hand, our proposed HDM model outperforms the results of [21], [32] which yield accuracies of 90.34% of and 86.10%, respectively, while our model exhibits 95.2% accuracy on the MCDS dataset and 91.3% accuracy on the CDS dataset.

VI. CONCLUSION

Human activity recognition is a challenging problem with many applications in fields such as visual surveillance, human-computer interaction, autonomous driving and entertainment. To overcome this issue, there are many possible motion estimation approaches.

In this study, it is proposed to construct a hybrid deep model for the purpose of HAR. The proposed architecture is built combining a dense optical flow approach and auxiliary movement information in videos using deep learning methodologies. First, deep learning models,namely 3D convolutional neural network (3D-CNN), 3D-CNN with optical flow, long short-term memory network (LSTM) are combined to determine the motion vectors. Classification task for videos are then processed by support vector machine algorithm. A wide range of comparative experiments are conducted on two newly generated chess datasets, namely magnetic wall chess board video dataset (the MCDS), and standard chess board video dataset (CDS) to demonstrate the contributions of the proposed study. Finally, the experimental results represent that the proposed hybrid deep model exhibits considerable classification success compared to the state-of-the-art studies. Furthermore, to the best of our knowledge, this is the first study based on a novel combination of 3D-CNN, 3D-CNN with optical flow, and LSTM over video frames to recognize human activity.

In conclusion, the experimental results demonstrate that the proposed architecture represents significant advantages for recognizing and classifying human activities in videos. First, the proposed hybrid deep model is flexible, extendable and customizable as it is able to determine many complex activities in various video datasets including playing chess, playing checkers, playing peg solitaire, and playing cards. Second, any number of features can be easily consolidated as auxiliary information for the proposed architecture. In addition to these advantages, the proposed hybrid deep model architecture allows the connection of other deep learning models to our proposed model as auxiliary features for purposes such as object recognition, hand tracking, and so on.

As future work, we plan to enrich the set of features by employing other deep learning techniques. Furthermore, heuristic optimization based algorithms can also be used to enrich the feature set for the purpose of improving the classification performance of HAR.

REFERENCES

[1] A. Jankovic and I. Novak, "Chess as a powerful eduacational tool for successful people," in *Proc. 7th Int. OFEL Conf. Governance, Manage. Entrepreneurship: Embracing Diversity Org.*, Dubrovnik, Croatia, 2019, pp. 425–441.

[2] M. Vrigkas, C. Nikou, and I. A. Kakadiaris, "A review of human activity recognition methods," *Frontiers Robot. AI*, vol. 2, Nov. 2015, Art. no. 28.

[3] A. Darwishalzughaibi, H. Ahmed Hakami, and Z. Chaczko, "Review of human motion detection based on background subtraction techniques," *Int. J. Comput. Appl.*, vol. 122, no. 13, pp. 1–5, Jul. 2015.

[4] A. Klaser, M. Marszałek, and C. Schmid, "A spatio-temporal descriptor based on 3D-gradients," in *Proc. 19th Brit. Mach. Vis. Conf. (BMVC)*, Leeds, U.K., vol. 275, 2008, pp. 1–10.

[5] G. Farnebäck, "Two-frame motion estimation based on polynomial expansion," in *Proc. Scand. Conf. Image Anal.*, vol. 2749, 2003, pp. 363–370.

[6] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 221–231, Jan. 2013.

[7] H. Wang, M. M. Ullah, A. Klaser, I. Laptev, and C. Schmid, "Evaluation of local spatio-temporal features for action recognition," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2009, pp. 124.1–124.11.

[8] J. Liu, J. Luo, and M. Shah, "Recognizing realistic actions from videos in the wildRecognizing realistic actions from videos 'in the wild,'" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1996–2003.

[9] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, vol. 2, Oct. 2003, pp. 1470–1477.

[10] J. C. Niebles, C.-W. Chen, and L. Fei-Fei, "Modeling temporal structure of decomposable motion segments for activity classification," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 392–405.

[11] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *Proc. CVPR*, Jun. 2011, pp. 3169–3176.

[12] I. Laptev, "On space-time interest points," *Int. J. Comput. Vis.*, vol. 64, nos. 2–3, pp. 107–123, Sep. 2005.

[13] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *Proc. IEEE Int. Workshop Vis. Surveill. Perform. Eval. Tracking Surveill.*, Jan. 2006, pp. 65–72.

[14] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1725–1732.

[15] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[16] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 1, 2014, pp. 568–576.

[17] S. Arif, J. Wang, T. Ul Hassan, and Z. Fei, "3D-CNN-based fused feature maps with LSTM applied to action recognition," *Future Internet*, vol. 11, no. 2, p. 42, Feb. 2019.

[18] J. Sun, J. Wang, and T. C. Yeh. (2017). *Video Understanding: From Video Classification to Captioning*. Stanford University. Accessed: Jun. 27, 2018. [Online]. Available: http://cs231n.stanford.edu/reports/2017/pdfs/709.pdf

[19] H. Qian, Y. Mao, W. Xiang, and Z. Wang, "Recognition of human activities using SVM multi-class classifier," *Pattern Recognit. Lett.*, vol. 31, no. 2, pp. 100–111, Jan. 2010.

[20] I. Laptev and T. Lindeberg, "Space-time interest points," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, vol. 1, Oct. 2003, pp. 432–439.

[21] M. Latah, "Human action recognition using support vector machines and 3D convolutional neural networks," *Int. J. Adv. Intell. Inform.*, vol. 3, no. 1, p. 47, Jul. 2017.

[22] H.-B. Zhang, Y.-X. Zhang, B. Zhong, Q. Lei, L. Yang, J.-X. Du, and D.-S. Chen, "A comprehensive survey of vision-based human action recognition methods," *Sensors*, vol. 19, no. 5, p. 1005, Feb. 2019.

[23] A. Karpathy, "Connecting Images and Natural Language," Ph.D. dissertation, Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA, 2016.

[24] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[25] Z. Wu, T. Yao, Y. Fu, and Y.-G. Jiang, "Deep learning for video classification and captioning," Feb. 2018, *arXiv:1609.06782*. [Online]. Available: https://arxiv.org/abs/1609.06782

[26] C. Zhang, P. Li, and G. Sun, "Optimizing fpga-based accelerator design for deep convolutional neural networks," in *Proc. ACM/SIGDA Int. Symp. Field-Program. Gate Arrays*, 2015, pp. 161–170.

[27] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," May 2015, *arXiv:1506.00019*. [Online]. Available: https://arxiv.org/abs/1506.00019

[28] J. L. Elman, "Finding structure in time," *Cognit. Sci.*, vol. 14, no. 2, pp. 179–211, Mar. 1990.

[29] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017.

[30] D. Kent and F. M. Salem, "Performance of three slim variants of the long short-term memory (LSTM) layer," Jan. 2019, *arXiv:1901.00525*. [Online]. Available: https://arxiv.org/abs/1901.00525

[31] J. J. Gibson, *The Perception of the Visual World*. London, U.K.: Houghton Mifflin, 1950.

[32] K. R. Aires, A. M. Santana, and A. A. D. D. Medeiros, "Optical flow using color information: Preliminary results," in *Proc. ACM Symp. Appl. Comput.*, vol. 3, 2008, pp. 1607–1611.
[33] S. S. Beauchemin and J. L. Barron, "The computation of optical flow," *ACM Comput. Surv.*, vol. 27, no. 3, pp. 433–466, Sep. 1995.
[34] A. Burton and J. Radford, *Thinking in Perspective: Critical Essays in the Study of Thought Processes*. Evanston, IL, USA: Routledge, 1978.
[35] D. H. Warren and E. R. Strelow, *Electronic Spatial Sensing for the Blind: Contributions From Perception*, 1st ed. Dordrecht, The Netherlands: Springer, 1985.
[36] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining Knowl. Discovery*, vol. 2, no. 2, pp. 121–167, 1998.
[37] J. Francis. *O'Reilly Online Learning*. Accessed: Jun. 27, 2018. [Online]. Available: https://www.oreilly.com/ideas/object-detection-with-tensorflow
[38] *Open Source Computer Vision Library*. Accessed: May 25, 2019. [Online]. Available: https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_calib3d/py_calibration/py_calibration.html
[39] V. Dibia. *How to Build a Real-Time Hand-Detector Using Neural Networks (SSD) on Tensorflow*. Accessed: Jun. 27, 2018. [Online]. Available: https://towardsdatascience.com/how-to-build-a-real-time-hand-detector-using-neural-networks-ssd-on-tensorflow-d6bac0e4b2ce
[40] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
[41] P. Saikia and K. Das, "Head gesture recognition using optical flow based classification with reinforcement of GMM based background subtraction," Aug. 2013, *arXiv:1308.0890*. [Online]. Available: https://arxiv.org/abs/1308.0890

SENEM TANBERK was born in Düzce, Turkey, in 1976. She received the B.S. degree in control and computer engineering from Istanbul Technical University, in 1998, and the M.S. degree in mechatronics from Marmara University, in 2012. She is currently pursuing the Ph.D. degree in computer engineering with Doğuş University, Turkey.

She is also an IT Professional in telecommunications, finance, and banking sector for over 15 years in Istanbul, Turkey. She has special expertise in Telco pricing and billing, revenue and financial reports, DWH support, software design and development, and advanced PL/SQL (Oracle) perfection. She is also an Oracle PL/SQL Trainer in Telco. She has been a Research and Development Consultant in a private company, Istanbul. Her research interests include deep learning, machine learning, big data, data science, robotic programming, embedded programming, simulation programming, and graphical programming.

ZEYNEP HILAL KILIMCI was born in Erzurum, in 1985. She graduated from the Computer Engineering Department, Doğuş University, in 2008, the M.Sc. degree from the Computer Engineering Department, Doğuş University, in 2013, and the Ph.D. degree from the Computer Engineering Department, Kocaeli University, in 2018.

From 2009 to 2011, she worked as a Software Engineer in data mining with the CRM Department and the Data Warehouse Department, DenizBank. In 2011, she started to work as a Research Assistant with the Computer Engineering Department, Doğuş University, where she working as an Assistant Professor, in 2018 and 2020. She is currently working an Assistant Professor with Kocaeli University. Her research fields include text mining, machine learning, ensemble learning, deep learning, reinforcement learning, and artificial intelligence.

DILEK BILGIN TÜKEL was born in Adana, Turkey, in 1965. She received the B.S. degree in electrical engineering, in 1987, the M.S. degree from Boğaziçi University, Istanbul, in 1990, and the Ph.D. degree in mechanical engineering from Katholieke Universiteit Leuven, Heverlee, Belgium, in 1997.

She has been an Assistant Professor with the Software Engineering Department, Doğuş University, and the Research and Development Leader with Altinay Robot Technologies Company, Istanbul, Turkey. Her research interests include modeling and control of robotic systems, industrial automation systems, AR/VR, and the IoT.

MITAT UYSAL was born in Istanbul, Turkey, in 1954. He received the B.S. and M.S. degrees in mechanical engineering and the Ph.D. degree in system analysis from the Technical University of Istanbul, in 1984.

From 1977 to 1984, he was a Research Assistant with the System Analysis Department, ITU. Since 1984, he has been an Assistant Professor with the System Analysis Department, Technical University of Istanbul. He is currently a Professor with the Software Engineering Department, Doğuş University, Istanbul. He is the author of 30 books on computer applications and more than 70 articles. His areas of interest include numerical analysis, optimization, data mining and algorithms, and software engineering.

SELIM AKYOKUŞ received the B.S. and M.S. degrees from the Computer Engineering Department, Middle East Technical University, in 1982 and 1985, respectively, and the Ph.D. degree in computer and information science with Syracuse University, in 1992.

He worked at Yildiz Technical University, Istanbul, and Doğuş University, Istanbul. He is currently a Professor of computer engineering with Istanbul Medipol University, Turkey. His research interests include data science, data mining and knowledge discovery, text mining, database and knowledge base systems, the Internet computing and web technologies, semantic web, software engineering, and programming languages.