IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 19, 2019, accepted December 23, 2019, date of publication January 3, 2020, date of current version January 10, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2963933

Prognosing Human Activity Using Actions Forecast and Structured Database

VIBEKANANDA DUTTA, AND TERESA ZIELINSKA, (Senior Member, IEEE)
Institute of Aeronautics and Applied Mechanics, Warsaw University of Technology, 00-665 Warsaw, Poland

Corresponding author: Vibekananda Dutta (vibek@meil.pw.edu.pl)

The initial stage of the work was supported by "HERITAGE" EU program (Grant Agreement 2012-2648/001-001 EM Action 2 Partnership) and in the later stages, the work was supported by the Preludium 11 (Grant No. 2016/21/ N/ST7/ 01614) funded by National Science Center (NCN), Poland.

ABSTRACT The goal of this work is to forecast human activities that may require robot assistance. Each activity consists of consecutive actions. Each action is bounded by initial and final state and is created by the motion trajectory. The states are defined in the training phase. The vision and depth sensors are used for data collection. The data are processed and the structured database is built. This base is used for making prediction. The method allows us to forecast the trajectories of nominally possible motion goals (prognosing of an action). The probability functions support the selection of possible motion goal. Then the possible motion trajectory is created which predicts the ongoing action. The activity is predicted on the basis of already completed action sequences and using knowledge about possible sequences stored in the database. The core of the reasoning process are: the probability functions, the action graphs (describing the activities) and the structured database. The approach was evaluated using four datasets: CAD 60, CAD-120, WUT-17, and WUT-18. The efficiency of the presented solution compared to the other existing state-of-the-art methods is also investigated.

INDEX TERMS Human activity, human-object relation, probability distribution, action prediction, structured database.

I. INTRODUCTION

The actions and activities recognition is needed for the personal robots taking care of the elderly, children or persons with some dysfunctions. If the person starts to do something but due to dementia, motion or force limits, or due to the lack of skills is not able to finish it (e.g. when trying to grasp a bottle of water and drink) s/he needs the robot support. That is why a robot must,,understand'' ongoing actions. Due to those requirement, inferring human activities using visual information plays a significant role in human-robot interaction, content-based video analysis, and intelligent surveillance. Publicly available inexpensive RGB-D sensors and increasing computational power allow us to process a large amount of input data. Therefore, a large number of potential applications [1] for commercial use, as well as for scientific research are expected. Forecasting human activities for safe human-robot collaboration tasks in crowded environments needs the interpretation of the observed actions. These actions include the type of an action, the object of interest, who is involved, and what might the future action be. For example, considering the case of a person standing in the kitchen and wishing to drink water (see Fig.1). A robot assisting in this task must be able to recognize and forecast the next action of this person in order to help by placing the glass on the kitchen table, if necessary. The robot either assists a person if it is requested to do so or it predicts the need to assist based on the perception provided by visual information [2].

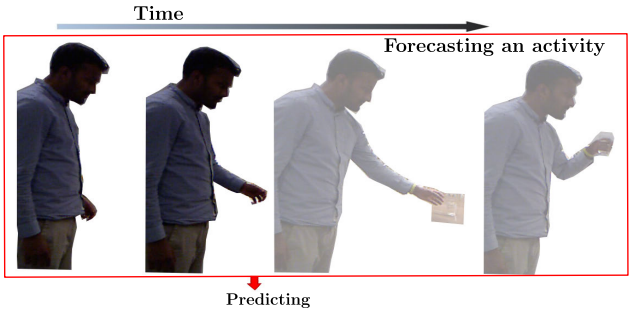


FIGURE 1. A pictorial illustration of a human activity prediction: a robot needs to infer ongoing human actions and make a decision based on partial observation.

A lot of efforts have been placed in the area of recognizing and prognosing human activities using videos in both: 2D and

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

3D scenes [3]–[5]. Situation contexts and semantics are often used for prediction which involves typical human-object interactions [6], [7]. The features of the scene elements and the human attributes (age, illness, habits) create semantics. Therefore, inferring human activities requires to address not only actions but also the involved objects and their dynamic in spatio-temporal arrangements which are changing during the actions [8].

In this paper, we present the method of forecasting human activities by prognosing sequential actions. We used two RGB-D sensors for recording the datasets (WUT-18) in an office and a home environment. The proposed approach was investigated in a supervised setting. The method aims at solving the short/ long-duration prediction problem. The low-level features, temporal segmentation of the recorded video clips, the spatio-temporal human-object relations, a data-gathering process, and the structured database are the key elements of this method. To demonstrate its effectiveness, evaluations were done using four experimental datasets covering a wide range of activities.

Comparing to our previous work [9], this paper focuses on the following aspects:

- the universal formalization of the problem is proposed,
- the definition of spatial and temporal features is formulated,
- data processing method is detailed,
- the definition of structured database is proposed,
- a series of probability functions and the reasoning (summarized by the structured graph) is detailed,
- the publicly available Warsaw University of Technology (WUT-17, WUT-18) datasets consisting of 11 short and long term activities were produced and released,
- the comparison of the proposed method with the results obtained using other state-of-the-art methods was made.

Starting our research on actions prediction we used the software developed by the authors of [5] and [10]. Following our ideas the software was further modified and adjusted. Moreover, we adapted some concepts presented in [11] and [12]. The additional parameter, namely the edge preference, was introduced. This parameter is applied in established by us probability functions used for actions prediction. The correctness of these functions was justified. The experiments confirmed that proposed amendments are resulting in competitive performance comparing to other prediction methods. Another contribution is the definition and implementation of structured database used for the activities' prediction. Such database allows fast, directed by the seen objects access to the parameters needed for actions prediction. The conclusion about ongoing activity is finally reached by comparing the sequence of performed actions with the sequences stored in the database.

The remaining part of the paper discusses this contribution in details. In section II we discuss the other works in the area of forecasting human activities. Section III focuses on formalization of the problem in brief and section IV describes the details of the data gathering process and the concept of structured database. Section V presents the testing stage and experimental results. The paper ends with the conclusions and future work suggestions.

II. RELATED WORKS

Activity recognition has a long history with the past research focusing on recognizing human actions from video sequences taken by a single camera [13]. Recently, the research has concentrated on detecting human activities using RGB-D data [14]–[17]. Forecasting human activities has attracted the attention of many computer vision and robotics communities. The recognition of complex human activities uses spatial and/or temporal descriptions of motion trajectories. A lot of efforts has been devoted to recognizing human activities using still images and videos in both: 2D and 3D scenes. The general methodology uses the observation of a human body or a hand motion and associates it with the activity [18]–[20]. Guo *et al.* [21] outlined methods for human activity recognition using still images and categorized them depending on a type of features that are considered. In [22], the authors summarized the different methods of human activity recognition using 3D motion capture data with the main focus on the depth information. In many scenarios, predicting the intended actions is very desirable. It is also significant to differentiate between recognition and prediction. An activity recognition concerns an ongoing activity and observation of the current stage. Activity forecasting (prediction) predicts the intention (future) when observing few previous action segments [23]. Ryoo [3] pointed out that the activity prediction requires the recognition of unfinished action by observing its early stage. Cao *et al.* [24] expanded this work with recognizing human activities based on a partially observed action. An unobserved sequence can arise at any time due to temporal gaps in the sensing (i.e., scene obstruction).

In [23], the whole activity was concluded by observing only a few actions. Yang and Tian [25] addressed an activity classification approach based on a Naive-Bayes-Nearest-Neighbor Classifier where only a part of an action is observed. In Gehrig *et al.* [26], a framework for activity recognition which combines a description of intention, activity, and motion, was proposed. Kim *et al.* [27] presented the method that can be applied for the whole activity prediction, their work focusing on temporal segmentation using activity partitioning. They applied event transition segments and event transition probabilities. Furthermore, work [5] incorporated object affordances to anticipate human activities for reactive responses. Work [9] addressed the so-called, modified object affordance with spatio-temporal human-object relation taking into account selected features. This research showed that careful selection of features and proper data gathering approaches are crucial for forecast performances, cancelling the need for a sophisticated learning algorithm. Kitani *et al.* [28] addressed the prediction task as a decision-making process and proposed the semantics for the scene labelling.

In [29], a recursive process of motion recognition and synthesis based on the hidden Markov models is used enabling a robot to understand the human behaviour for proper reactions. The proposed framework is based on learning the interactions between the two-person through observation, and by that generating human-like motions for the robot. Takano and Nakamura [11] proposed an approach establishing a fundamental framework using the so-called motion symbols and the motion labels extracted using established stochastic translation model. The,,distances'' between the labelled motions are calculated using the probabilities. The label space summarizes motion similarities. The label space concept allows also the motion recognition. Recently the authors of [30] have described the approach dealing with understanding human daily activities through the so-called Interaction Unit analysis that enables decomposition of activities into a sequence of units. Each of these units is associated with a behavioural factor. The recognition of human behaviour is performed by Dynamic Bayesian Network (DBS) that operates on top of the Interaction Unit, offering quantification of the behavioural factors and formulation of the human's behavioural model.

A limited number of studies [4], [5], [9] addresses relevant factors such as spatial features, human-object interactions, data gathering, and the database. Although activity recognition and forecasting methods have shown good performance for small datasets with controlled background settings, it is still challenging to generalize them for real-time and uncontrolled settings due to large computational complexity. Our previous work [9] presented some situations with higher computational need, however, still on a limited scale. Visual information may be redundant, and therefore it is not necessary to consider each observation in the data processing. Such a situation motivated us to propose the method which uses a series of probabilistic functions for the prediction of sequential actions.

III. PROBLEM DEFINATION

A. GENERAL OVERVIEW

A human activity is a state of doing. Since a human activity is a broader concept, for the sake of simplicity, in this work, we considered only human activities involving objects manipulation. Our method of activity prediction consists of two phases: (a) the first one is the training phase (data gathering, processing and storing), and (b) the second one is the testing phase, alternatively called as the ''prediction phase.''

In the training phase, human activities are recorded using two Intel Senz3D cameras. Next, the records are temporally segmented in terms of atomic actions. The boundaries between atomic action are defined by the initial and final states (section III-B). Inside each action, human pose and the object features are observed, recognized and quantized. The created database relates the actions sequence and the objects to the corresponding activities. We extract the spatio-temporal features (i.e., the features that describe the relationship between a human and object of interest in the scene) using video clips. In the training process, we obtain the motion parameters evaluating the mean value and variance of some distances. The forecasting (prediction) problem is a sequence of prognoses basis on partial observation. The proposed approach uses probability functions for the prognosis. A maximum likelihood is taken into account for inferring the future motion trajectories.

B. FORMAL DESCRIPTION

In this part, we provide the formal description of the activities prediction system. The scene is observed and the objects in the human vicinity are identified (recognized). The objects are used as the discriminates for indicating which actions will be nominally taken by the human being.

First we explain the applied notations:

- the capital letter (i.e. S, A, O) denotes a set,
- the small letter (i.e. s, a, o) denotes an element of a set,
- the upper script denotes the assignation, e.g. S^{AC} means that the set S is assigned to AC,
- the lower script marks the concrete element.

Each action a_i is an elementary transformation of the human state. For each action, several possible initial and final states exist. Therefore for each a_i action we assign a pair $(S_{in}^{a_i},\ S_{fin}^{a_i})$, where $S_{in}^{a_i}$ is a set of possible initial states and $S_{fin}^{a_i}$ is a set of possible final states. An action a_i is the transition:

$$a_i : s_k \longrightarrow s_m \ = < a_i : s_k,\ s_m > \tag{1}$$

where $s_k \in S_{in}^{a_i}$, $s_m \in S_{fin}^{a_i}$, what can be also expressed as $(S_k^{a_i}, S_m^{a_i})$. It must be noted that any s_k can be not only the initial (or final) state of an action a_i but of any action as well, therefore in final description we use the upper script ^{a_i} if the state concerns the action a_i and neglect this upper script when considering any of possible states not relating them to concrete action.

For example:

$a_i = move\_hand$
$S_{in}^{a_i} = \{hand\_free,\ bottle\_in\_hand,\ hand\_on\_table\}$
$S_{fin}^{a_i} = \{hand\_over\_cup,\ hand\_free\}$

The action can be performed involving some objects. In the example above, the object can be *a bottle*, *a table*, *a cup*. Potentially involved object o^{a_i} belongs to O^{a_i} ($o^{a_i} \epsilon\ O^{a_i}$). Finally the action is described by:

$$a_i = a_i(S_{in}^{a_i},\ S_{fin}^{a_i},\ O^{a_i}) \tag{2}$$

For the sake of simplicity of the further notation if the specific object o_p is involved in an action a_i we denote it as $a_i(o_p)$. Realization of an activity AC_k is achieved by the sequence of actions $a_i,\ a_m,\ a_k, \ldots$ performed by a human being for fulfilling some goal. Naturally, the actions are separated by states. The activity starts and ends with particular state. It means that with each activity is associated the pair $(s_{in}^{AC_k},\ s_{fin}^{AC_k}) \epsilon\ S^{AC_k}$.

Lets us denote by A^{AC_k} all possible actions which can build the activity AC_k. Considering the above:

$$AC_k = AC_k(A^{AC_k}, O^{AC_k}) \tag{3}$$

For $a_i \in A^{AC_k}$ basis on equation 2 and equation 3 $O^{a_i} \supset O^{AC_k}$. Let us $seq\_A^{AC_k}$ denotes one out of several possible sequences of actions building an activity:

$$if\ a_i \in seq\_A^{AC_k}\ then\ a_i \in A^{AC_k} \tag{4}$$

When prognosing the activity we consider the scenario (observed scene). The scenario delivers the vocabulary. First, the objects are identified making the first element of vocabulary. Based on the equation 3 having the set O of the seen objects, the possible AC_k can be defined and then respectively the corresponding actions (equation 4), as well as initial and final states (equation 2) can be determined.

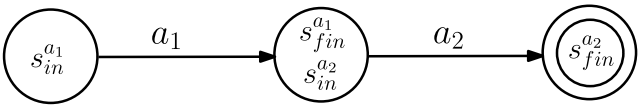


FIGURE 2. The figure represents a graph structure for sequence of two actions making an activity.

Let us consider an example of *activating of a computer* (see Fig. 2), where the activity is summarized by: (a) *scenario*, (b) *the vocabulary*.

1) SCENARIO

The scene is observed in an office environment where a computer and a person are present. The person is by the table and is expected to activate the computer.

2) THE VOCABULARY FOR THE ACTIVITY

- **Activity:** $AC_1 = activate\_computer$
- **Initial state:** $s_{in}^{AC_1} = screen\_black$
- **Final state:** $s_{fin}^{AC_1} = scren\_bright$
- **Involved object:** ${o_1}^{AC_1} = computer$

The activity is build out of two actions (Fig. 2) as it is described below.

3) THE VOCABULARY FOR BUILDING THE ACTIONS

Actions = {reaching, pressing}

Action $a_1^{AC_1}$ = reaching	Action $a_2^{AC_1}$ = pressing
Initial state $s_{in}^{a_1} = HNO$[1]	Initial state $s_{in}^{a_2} = SCB$[2]
Final state $s_{fin}^{a_1} = HO$[3]	Final state $s_{fin}^{a_2} = SB$[4]
Involved object $o_1^{a_1} = CP$[5]	Involved object $o_1^{a_2} = CP$

C. OUTLINE OF THE METHOD

In the proposed method, the states $s_k \in S_{in}^{a_i}$, $s_m \in S_{fin}^{a_i}$ separating the actions are indicated by the human expert (the person who makes the training) in the data-gathering phase. A set of video records for each activity is collected. Each record is cut according to the set of states indicated by the trainee $S_{in}^{a_i}$ and $S_{fin}^{a_i}$. It means that each segment starts with $s_k \in S_{in}^{a_i}$, and ends with $s_m \in S_{fin}^{a_i}$. One segment represents an action a_i. For the sequence of consecutive actions a_i and a_{i+1}, the final state $s_m \in S_{fin}^{a_i}$ of a previous action makes naturally an initial state $s_m \in S_{in}^{a_{i+1}}$ of the next action. The recorded segments are processed further in order to collect human pose features and object features. Obtaining such data is needed for estimating the spatio-temporal attributes.

When building the base of the structured database our method takes as an input a set of features consisting of spatio-temporal attributes (d^{Ho}, θ, e) which are obtained in the first stage (data gathering). For each action, we collect such data for the records repeated, M times ($M \geq 60$). Collected data are used for evaluating the mean value $\mu_{d^{Ho}}, \mu_\theta, \mu_e$ and variance $\sigma^2_{d^{Ho}}, \sigma^2_\theta, \sigma^2_e$ respectively. Those values are applied later as the probability function parameters. The probability functions support the selection of possible motion goal and the possible motion trajectory is created which makes an activity prediction. The activity graph is built out of $seq\_A^{AC_k}$. During the experiment the method allows to forecast the trajectories to nominally possible motion goals (predicting an action) and next, using the action graphs, to prognose of an activity. The human states are defined by the nodes and the nominally possible actions are represented by the edges respectively. Section IV describes the data gathering process and the creation of the structured database. The testing stage uses the graphs and the probability functions are described in section V.

IV. DATA GATHERING AND BUILDING THE STRUCTURED DATABASE

A. EXPERIMENTAL SETUP FOR DATA COLLECTION

Although a few relatively large RGB-D datasets are available, to facilitate experiments with larger and diverse datasets, the new datasets (WUT-17) were introduced [31]. Recently we have produced one more data set called WUT-18 which uses multi-view settings. Our method is dedicated to be implemented in the environment with properly distributed sensors. Since the required sensors are inexpensive, we assume that the relevant space will be well observed by as many sensors as needed to obtain the proper view of a human and objects. It must be also noted that the human takes specific postures when doing a specific activity. For example, it is rather unusual that the person will be reaching the bottle placed on the table without some turning towards it. However, if it is the custom of the user the sensors placement and the data collection phase must consider it. Therefore in our approach, we avoid the disturbances and occlusions analysis as it is commonly ignored by the other researchers. However, one question still remains to be investigated. It refers to the problem what is more efficient - either the disturbances rejection and occlusions adjustment or to design the sensing system carefully. It seems that with the fast progress in sensing technologies, the second option would be a better choice.

[1]HNO: Hand next to object.
[2]SCB: Screen black.
[3]HO: Hand on an object.
[4]SB: Screen bright.
[5]CP: Computer.

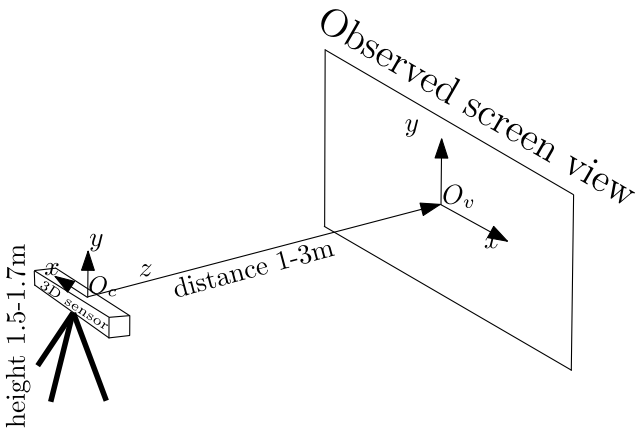


FIGURE 3. Graphical illustration of camera setup.

Two fixed viewpoint RGB-D vision sensors were placed on tripods with adjustable height $1.5 - 1.7m$. We applied Senz3D RGB-D [32], [33] which is the sensor released by the Intel RealSense is one of the most advanced 3D depth-sensing cameras, which provides information about the depth and the RGB images as well. The sensor is equipped with an RGB camera and an infrared camera. The depth range applied for human observation is $1-3m$. Customized programming tools were developed for data extraction from the raw images. The software was developed using the C++ language and robot operating system (ROS).

We used down-sampled images (640×480 pixels) at $60fps$ frame rate since real-time decoding and display of multiple streams of a high-resolution video is a bottleneck problem. The applied system has the ability to provide the $3D$ visual data and allows tracking $3D$ skeleton points and object position. During the experiments, the orientation of the cameras was fixed [9]. We used several objects on which the manipulations were performed. In addition, the recording included a wide variability of the activities performed by different persons using both: left and right hand with different time duration as well as the speed. Two cameras were delivering independently two images in which (after proper preprocessing) the x, y, z coordinates of the points of interest were provided ($\{x_{c_1}, y_{c_1}, z_{c_1}\}$, $\{x_{c_2}, y_{c_2}, z_{c_2}\}$) in each camera plane. We applied two coordinate systems with the origin placed in the center of each camera frame respectively.

B. DATA PREPROCESSING

The first step of the human activities prognosing is preprocessing the recorded observations. Proper preprocessing speeds up the training method. In preprocessing, the temporal segmentation and features extraction are made as it was introduced in our previous work [9]. Here we are summarizing this step using a more formal description.

1) TEMPORAL SEGMENTATION AND FEATURE EXTRACTION

The goal of video segmentation is to turn the recorded data into a set of parts. Each segment is a sequence of video frames together with depth values which have no significant inter-frame difference in terms of motion contents.

Each activity AC_k was performed M times (in our case $M = max(m) \geq 60$) by 6 different users. Let's $F_m^{AC_k}$ denotes $m\text{-}th$ record of an AC_k activity. Each record $F_m^{AC_k}$ consists of f number of frames (where f can vary from case to case), what is denoted by $F_{m_f}^{AC_k}$. As it has already been mentioned, we do the temporal segmentation by partitioning an activity into a group of actions a_i. $F_{m_f}^{AC_k}$ is cut into smaller parts by the human expert (see Fig. 4). Each part represents an action. Therefore, the part associated with a_i in $F_{m_f}^{AC_k}$ is denoted by $\left(F_{m_f}^{AC_k}\right)^{a_i}$. The part representing $i\text{-}th$ action (action a_i) is again cut into segments automatically. Such segmentation process is required for obtaining the data used for motion prognosis. As each action can start when the human hand is at the different distance from the object of interest and can be realised with different speed, the segmentation helps to establish a proper dataset. The number of video segments starting from some (selected by us) moment till the end of an action is used for extracting the relevant data.

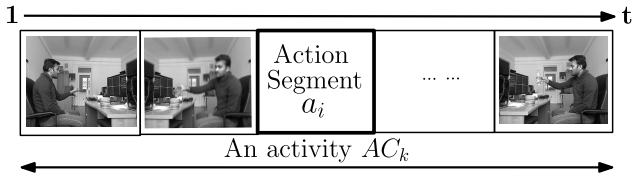


FIGURE 4. Graphical illustration of an activity segmentation into actions. A human expert divides the record producing the sequence of actions.

The second step of preprocessing is the extracting of relevant features. We require not only to track movements focusing on those body parts, which are mostly involved when executing the activity but also on the objects on which the actions are performed. Therefore, features selection and extraction is a significant step in the activity forecasting. Finding an appropriate set of features is problem-oriented. We extract three groups of features: (a) human position matrix H, (b) object position o_c^{ft}, and (c) features describing human-object interaction: distance d^{Ho}, angle θ, and edge e.

The human position matrix (human feature) H consists of $3D$ positions of some relevant points of a human body. To extract such features, we use the data delivered by *senz3D* RGB-D sensor. The data provides visual data and the real-time position of body points. In this work, we consider three points: *center of torso (CT)*, *right hand (RH)* and, *left hand (LH) position*. Torso position is needed in order to conclude about the whole body movement, e.g., walking towards the door before the hand starts manipulation. Observation of both hands is needed to make a decision about which hand is involved in an action. Some noisy points (i.e., hip center, right wrist, left wrist, right ankle, left ankle) due to their closeness to the other points (i.e., right hand, left hand, right foot, left foot) are not considered, however more points are used for

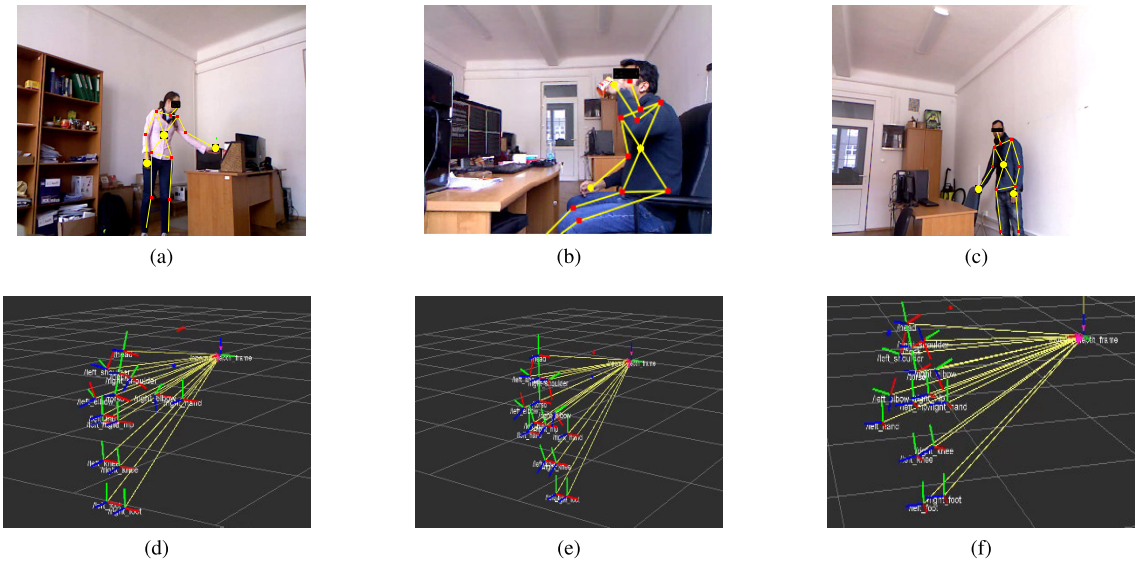


FIGURE 5. The figures are representing human posture estimation obtained from the Senz3D camera. The illustration concerns following following actions: (a) reaching an object, (b) drinking coffee, (c) pulling the chair. The bottom figures show the extracted sketch diagram representing the positions of the body points for the posture above: (d) $Sk_{reaching}$, (e) $Sk_{drinking}$, and (f) $Sk_{pulling}$.

body visualization. The feature matrix H is expressed by:

$$H = [P_{CT}, P_{RH}, P_{LH}] \tag{5}$$

where each $P_r \in \Re^3$, is the vector containing the $3D$ coordinates (x_p, y_p, z) of the $r-th$ point $(r = \{CT, RH, LH\})$. x_p, y_p are the positions of the points expressed first in pixel coordinates, z is the depth value for a pixel (x_p,y_p). We transform all the values to the real world coordinates:

$$x = \frac{z}{f}(x_p - x_0 + \delta_x) \tag{6}$$

$$y = \frac{z}{f}(y_p - y_0 + \delta_y) \tag{7}$$

where (x_0, y_0) is the image center, δ_x and δ_y are parameters correcting the lens distortion.

The feature o_c^{ft} represents a vector containing the (x_c, y_c, z_c) coordinates of the object center expressed in the world coordinates. We perform both, object detection and tracking for two object categories: (a) larger objects (*door*, *table*, *box*, *whiteboard*, etc.), (b) smaller objects (*marker*, *bottle*, *cup*, etc.). To effectively detect and track objects in real time we follow the $tracking - by - detection$ paradigm described in [9].

Larger objects are labelled by QR codes which can be properly recognized in the observed scene using label-based object detection method [34]. For the smaller objects, we use the "Lucas-Kanade Descriptor" (KLD) method [35] which is based on the search of an object which picture is stored in the database.

Besides of H and o_c^{ft} the third group of features is the spatio-temporal vector which contains: (a) a temporal *distance* d^{Ho} between human hand and the object of interest, (b) *angle* θ between human hand and the object of interest. (c) *edge* e which is the normalized distance obtained as the

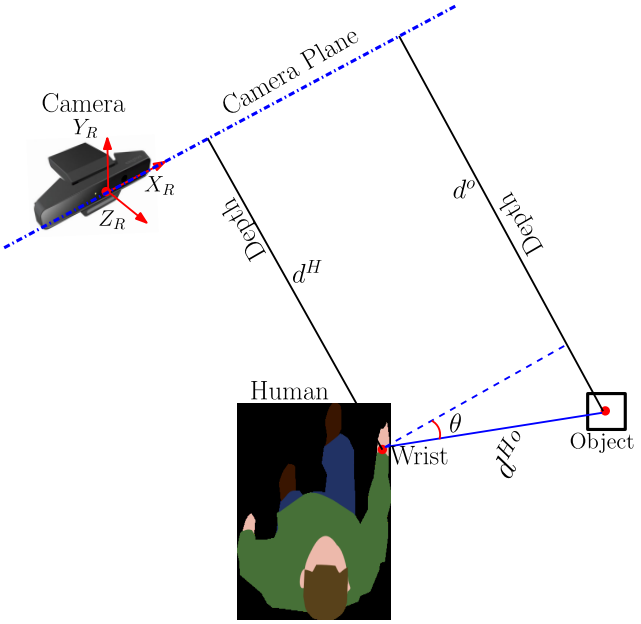


FIGURE 6. The figure illustrates the scenario for obtaining the distance d^{Ho}, and angle θ.

distance d^H from the camera to the human hands normalized by the distance d^{Ho} between the human hand and the object of interest, as it is shown in the Fig. 6. The edge e is expressed by.

$$e = \frac{d^H}{d^{Ho}} \tag{8}$$

C. DATA PROCESSING AND THE DATABASE

The temporal values of relevant features are used to obtain their mean μ_i and variance σ_i^2 as it is listed in Table 1.

(a) (b) (c)

FIGURE 7. An example of object detection and recognition. The figure illustrates the scenario with detection and tracking of an artificial landmark and a cup. The following images are representing an action such as: (a) beginning of an action, (b) ongoing action, and (c) end of an action.

TABLE 1. Considered data.

Data
Distance between human hand and the object (d^{Ho})
Angle between human hand and the object (θ)
Depth from the camera towards the human hand (d^{H})

Next, 50 out of 60 records for each action are selected for training, and the remaining 10 are selected for testing. Selection of such a specific amount of records for both training and testing was justified and explained later in the section V-C.3 (data test).

For each recorded frame we store the distance d^{Ho} between the hand wrist position and the object of interest (object to be manipulated) from the beginning till the end of an action. Similarly, the edge e is calculated using equation 8 and stored together with θ. For each object which can be manipulated (considering all nominally possible actions) the values of $\mu_{d^{Ho}}, \mu_\theta, \mu_e, \sigma^2_{d^{Ho}}, \sigma^2_\theta, \sigma^2_e$ are calculated. Once those parameters are obtained, we create the database consisting of activities and corresponding actions taking into account the objects. The database for each activity contains all nominally possible actions. In particular, each action sequence is defined taking into account the object/objects involved in the activity. The obtained parameters (μ_i, σ_i^2) are used as the parameters of the probability functions (see Appendix). The set of parameters was selected taking into account the probability approach. These parameters are considered in probability functions used for action forecasting. The probability of an action concerns the object of interest and the easiness of reaching/manipulating it. Therefore we call it the object affordance. The object affordance in our case results from the edge, angular and distance preferences. In the testing phase, probability functions are used for concluding the future trajectory. Bezier curve is used to prognose the trajectory from the current location to the predicted location.

The graphs representing the activities reflected in the database using the equation 4. Fig. 8 shows the overall structure of the database used in our work. Considering the different objects can be involved in different actions, we split the action sequences taking into account the involved object. Let us use as the example of an activity AC_i performed on the objects w and z, or w or $(\vee)$ z, which can be realized using the following sequences:

$$seq\_A^{AC_k} = \begin{cases} a_n(w), a_k(w), a_p(w), a_b(z), a_c(z), a_d(z), \text{ or} \\ (a_n(w), a_k(w), a_p(w), a_d(w) \vee \\ a_b(z), a_c(z), a_d(z)) \end{cases} \quad (9)$$

The example described by equation 9 is shown on the left side of Fig. 8.

The database is divided into two main parts. The first part consists of activities definition. Each activity is described by the all possible action sequences taking into account the involved objects. The second part (right side of the Fig. 8) contains the parameters established during data gathering phase based on calculated quantities (μ_i, σ_i^2). Those quantities are obtained using the segments cut from data records. A detailed description of this stage is given in [9]. For each addition of new actions or, objects the model requires to be additionally,,trained'' and the additional parameters must be obtained.

The specific action involving the specific object can be performed by several trajectories. During prognosing we must forecast the most possible trajectory for each action. The parameters from the second part of the database are used for the trajectories prediction to identify the target location. The referred part of the database is organized as follows: there are segments associated with as many objects as many were indicated in the first part of the database describing the activities. Each object description consists of the list of all possible actions which can be performed involving such an object. For each earlier obtained parameters (μ_i, σ_i^2) probability functions are given. In the testing phase these probability functions are used for prognosing the future trajectories.

D. ACCESSING THE DATA BASE

Once the objects are recognized in the sensor field of view, the right part of the database is accessed (Fig. 8). Let's say objects z and w are noticed. The parameters of all probability functions assigned to those objects are accessed (see Fig. 8 right side). Next, on the basis of the equation 10, actions

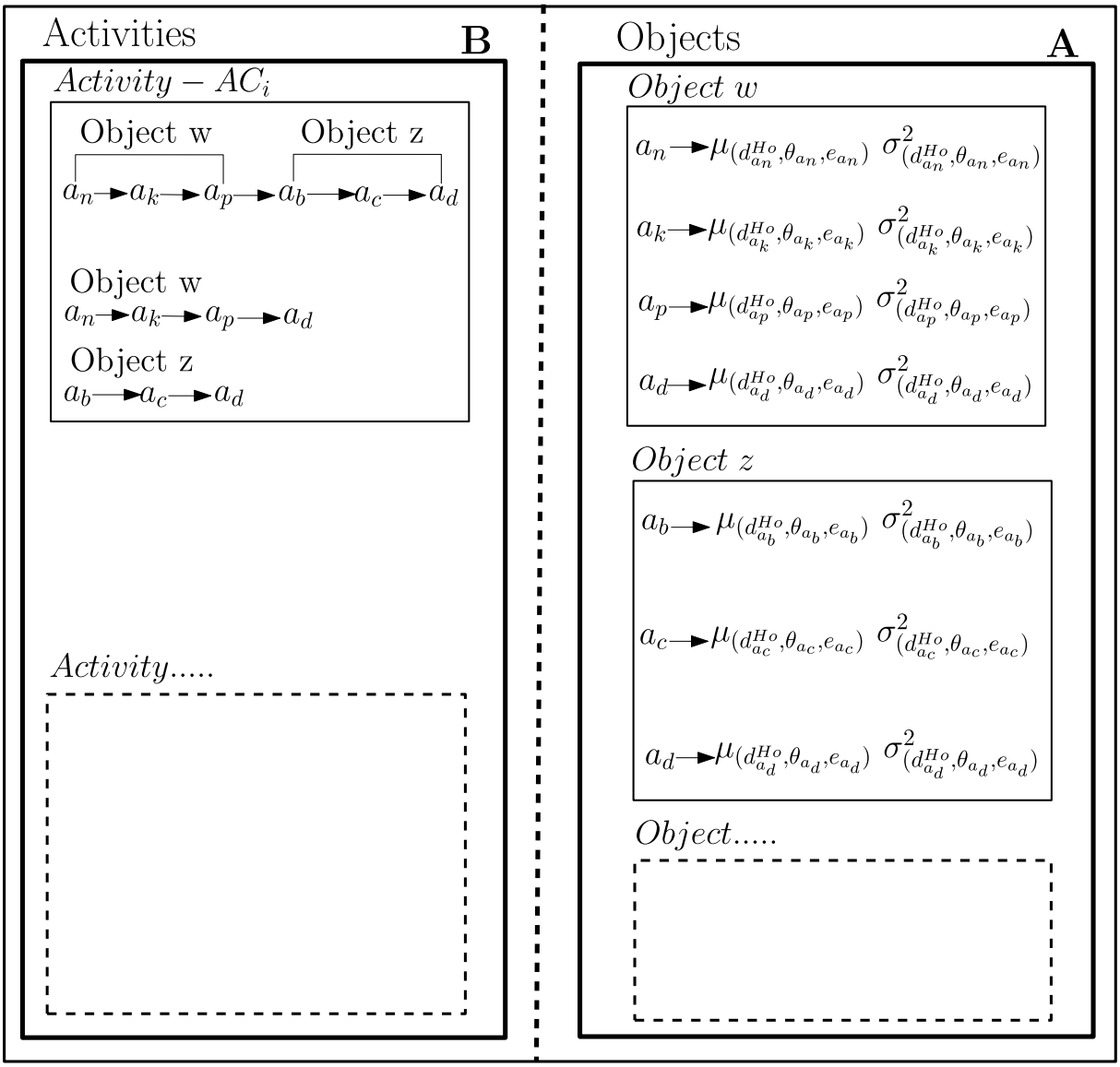


FIGURE 8. Graphical illustration of the database: A-right side of database (first part), B-left side of database (second part).

with the highest probability are forecasted and the possible future trajectories to the goal of interests are obtained as it is described in V-B. Once the action is indicated for the concrete object the first part of the database is used for predicting the ongoing activity. As it can be seen in Fig. 8 (left side) the combination of an object and action is associated with an activity.

If the same pair object-action occurs in several activities then all those activities will be considered. If for some activities some specific sequences and objects are only possible the conclusion about ongoing activity will be made faster. The left part of the database is used for predicting the activities. At the beginning of reasoning, all possible activities containing the pair object-action are considered as possible. At some point, the passed activities sequence becomes particular which concludes the ongoing activity. The activity prediction time results from the,,particularity'' of the object-action pair and the actions sequence.

Fig. 9 gives the graphical representation of the database concept reflecting the equation 1 described in section III-A. The graph is built out of $seq\_A^{AC_k}$ as it is defined in equation 4. The nodes represents the human states and the edges illustrates the elementary transformation (i.e. action). The example graph is made out of 7 activities which are considered in this work. In the testing phase, the method allows us to forecast the trajectories to nominally possible motion goals (prognosing an action) and next, an activity is prognosed using the sequence of the completed actions. Expanding the graph means the proper update of the database with properly feeding it with objects, action sequences and the probability function parameters collected during the training phase.

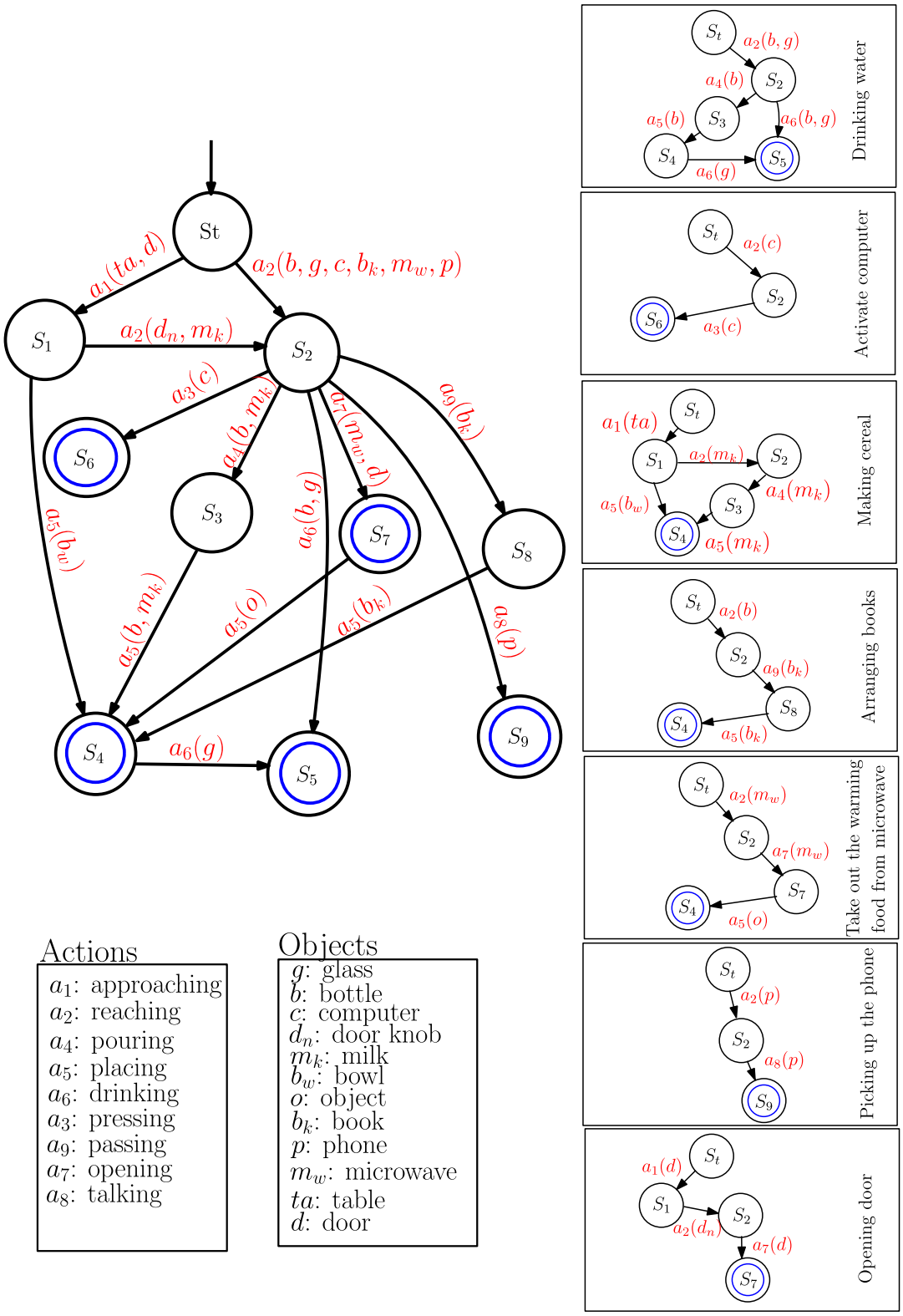


FIGURE 9. This figure illustrates the activity graph made out of 7 actions. The possible sequences of actions are defined in training stage.

V. TESTING STAGE: VERIFICATION OF THE METHOD

In this section, we discuss the testing stage of inferring human activities. The inputs are the depth information and video data. We track the human motion using "Skeltrack" application to obtain the locations of the key points of the human skeleton (feature H). We recognize the objects of

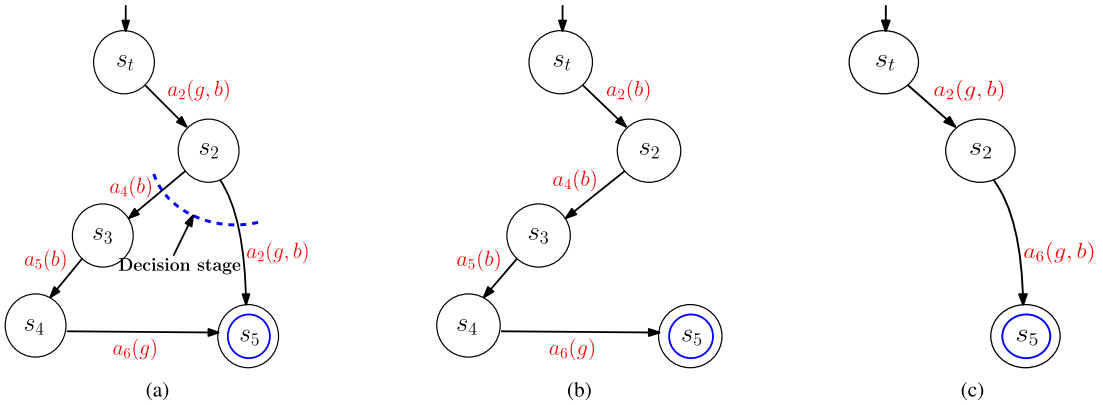


FIGURE 10. Graphical representation of an activity with different possible sequences associated with specific object of interest (in this example – bottle): (a) complete graph, (b) graph representing the drinking activity associated with both glass and bottle (c) another graph representing the drinking activity associated with either the glass or the bottle.

interest being used in the activity and track them (feature o_c^{ft}). We obtain d^{Ho}, θ, and e respectively using the above features and the depth information.

In the testing phase the possible activities with all nominally possible action sequences are indicated for each recognized object (Fig. 8 - right side). Next, for each action, probability functions are used to forecast the motion trajectories. These functions consist of: *distance preference*, *angular preference* and *edge preference*. Action a_i for which the probability is biggest is selected using the action selection function described in section V (part A).

Let's refer to the example of concrete activities which are described by graph presented in Fig. 9. This graph describes 7 activities: (a) *drinking water*, (b) *activating computer*, (c) *making cereal*, (d) *arranging books*, (e) *taking out the warming food from microwave*, (f) *picking up the phone*, (g) *opening the door*. Now we address only the first activity,,drinking water'' (upper part of the Fig. 9). In this example are possible 3 sequences:

- $\{a_2(b), a_4(b), a_5(b), a_6(g)\}$ performed on two objects (bottle, glass) as it is shown in Fig. 10b,
- $\{a_2(b), a_6(b)\}$ (see Fig. 10c) performed on one object (bottle),
- $\{a_2(g), a_6(g)\}$ (see Fig. 10c) involves one object (glass).

A. ACTION SELECTION FUNCTION

Once the object is recognized during the testing phase the set of actions associated with this object is considered (right side of the Fig. 8). Selection of an action means that the current parameters θ_{a_i}, $d_{a_i}^{Ho}$, e_{a_i} are justified as a valid initial quantities (at the moment) for this action. Then the probability is calculated considering the actions associated with the object. Such action $P(a_i)$ is selected for which the probability is biggest.

$$P(a_i) = \max_{a_i} \begin{cases} (P(e_{a_i}) \cdot P(\theta_{a_i})) & \text{for } d^{Ho} > 20cm \\ (P(d_{a_i}^{Ho}) \cdot P(\theta_{a_i})) & \text{for } d^{Ho} \leq 20cm \end{cases} \quad (10)$$

The threshold $20cm$ was selected heuristically noticing that when the hand is farther than $20cm$ from all the objects any object can the targeted. For the distance not bigger than 20 cm the motion towards that object will be completed and the equation 10 in this case is used for selecting the trajectory described in section V (part B).

$P(e_{a_i})$ is the edge preference function, $P(\theta_{a_i})$ is the angular preference function, and $P(d_{a_i}^{Ho})$ is the distance preference function. Selection of those functions together with its validation was presented in detail in our publication [9]. The summary in Appendix provides the overview of those selected functions.

B. PROGNOSING THE TRAJECTORY

A forecasted trajectory is produced using the parameterized cubic equation of Bezier curve [9]. Each fragment of such a curve lies inside the outline established by the so-called control points. The curve shape is influenced by these points. The Bezier curve is a polynomial of p_n, where $p_n \in < 0, 1 >$:

$$t_j = (1-p_n)^3 t_0 + 3(1-p_n)^2 p_n t_1 + 3(1-p_n) p_n^2 t_2 + p_n^3 t_3, \quad (11)$$

$t_j = \{x_j, y_j, z_j\}, j = 0, 1, 2, 3$. Applied Bezier curve is parameterized by a set of four points: the starting and final point of the trajectory (t_0 and t_3), and two control points (t_1 and t_2). In our case, t_0 is the current position of the hand. The point t_3 is the end point of the action indicated by the probability function (equation 10) and it is the object position. For prognosing of hand trajectory associated with the selected action two control points (future position of the hand) are used. The control points coordinates t_1 and t_2 are obtained from the number of observations (records) in data gathering phase. For this purpose the full records collected in data gathering phase are used. Point t_1 is the hand coordinate at 40% since the starting point t_0 (when the action was selected) and point t_2 concern 80% of the record since t_0. Point t_1 and t_2 are the points taken from the previously recorded trajectory which has its beginning close to the t_0.

C. RESULTS

The proposed approach was validated using two methods: (a) a comparison with the other methods (model test) (b) quality of prognosis depending on the amount of observation data (data test).

We applied our method on four datasets: (a) CAD-60, (b) CAD-120, (c) WUT-17, (d) WUT-18. The details of the data sets are given in section V-C.1, and the implementation details are summarized in section V-C.2. In section V-C.3, we present the experimental results and the performance analysis of the proposed method.

1) DATASETS

We created publicly available data sets (named as WUT-17, WUT-18) of the following daily activities: *drinking water*, *opening a door*, *arranging a book on the shelf* , (Fig. 11). These activities were performed by 6 participants in 3 different settings (a) *an office*, (b) *a living room*, (c) *a kitchen*. The participates had neither prior knowledge of the purpose of the study nor instructions how to perform each activity. The data sets were collected under RGB-D settings, at the rate of 60*fps*. The cameras range for human observations was fixed and covered the range.

The Cornell Activity Dataset CAD-60 [36] is composed of 12 different activities (see Fig. 11) performed in 5 different environments: (a) *an office*, (b) *a kitchen*, (c) *a bedroom*, (d) *a bathroom*, and (e) *a living room*. The activities are performed by 4 people. The data set is a collection of RGB images, depth information, and skeleton data with 15 points. The activities are: *rinsing a mouth*, *brushing teeth*, wearing contact lens, *talking on the phone*, *drinking water*, microwaving food, etc.

We also considered CAD-120 [37] - one of the most used publicly available data set of complex human activities in daily life situation. The data set consists of 120 RGB-D videos of 10 long activities: *arranging objects*, *having meal*, *making cereal*, *picking objects*, etc. as shown in the Fig. 11. These activities are performed by 4 different participants repeating each action three to four times.

2) IMPLEMENTATION DETAILS

The method was implemented using *Intel Core i*7 3.10GHz machine with 16 GB of RAM, with 64-bit Linux operating system. With such implementation the method shows low memory consumption justifying its efficiency.

3) EXPERIMENTAL EVALUATION

Two different tests were performed. The first one called the *model test*, gives the performance of our approach comparing it with other baseline algorithms. The second one was the *data test*, giving the performance accuracy related to the properties of the training set. In this case, it was investigated how much the prediction is influenced by the number of data samples considered in the training stage.

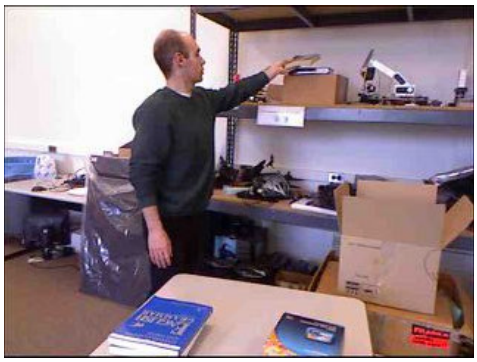
a: MODEL TEST

It is not possible to compare the different methods for the activities prediction as the implementation of the methods described in the literature needs a huge amount of work. Additionally, not all implementation details are disclosed. That is why, we make the comparison considering only the action level.

To conduct this study, we compared our approach with the methods using: (a) Hidden Markov Model (HMM), (b) Linear Support Vector Machine (LSVM), (c) the method developed by Kappula, Gupta, Saxena (KGS), (d) the method with Anticipatory Temporal Conditional Random Field (ATCRF).

In HMM-based approach [11], human full-body motions are encoded as a set of parameters of HMM. This method stochastically determines the activity category to which it belongs. The authors applied commonly used HMM with 40 hidden states and the corrosponding hyperparameters were: (a) maximum number of iterations was 20, (b) the number of training samples were 50 records, and (c) the value of learning rate was fixed to 0.01. To optimize the model evaluation process, the authors used traditional approach Baum-Welch (BW) algorithm based on expectation-maximization (EM) algorithm to find the maximum likelihood given a set of observed feature vectors. The Gaussian Mixture Model (GMM) was used to establish emission probability. In LSVM based approach, the linear kernel was used as the single classifier, the action features (spatio-temporal motion features) are fed to a single SVM classifier for labelling (recognizing) the actions categories [12]. The linear kernel function based approach offers reduced calculation demand comparing to the other methods when many features or many training samples are being considered. The main hyperparameters of the SVM classifier were the kernel function and the regularization parameter C. This parameter is often termed as soft margin constant C and in this experiment it was fixed to 100. For large values of C, the optimization chooses a smaller-margin hyperplane and does a better job of getting all the training samples classified correctly. To find the best solution of the problem optimization algorithm was used and Particle Swarm Optimization (PSO) was the choice. In KGS method [10], the action transition probabilities and object affordances are obtained using the training data. The observed frames are first labelled using the Markov Random Field (MRF) model. To learn the model parameters the authors used the cutting plane method [38] and sloved the optimization problem using a graph-cut method (i.e., quadratic pseudo-Boolean optimization) [39]. The anticipated actions and object affordances for the future are predicted based on the transition probabilities given the inferred labelling of the last frame. ATCRF [5] method samples the future nodes of sub-activities and object affordances (both in temporal segments and frames in each temporal segment) as described in [5] and uses a fixed temporal structure. The authors took the advantage of a large-margin approach [40] to learn the parameter vector from the labelled training

(a) Drinking (CAD-60)



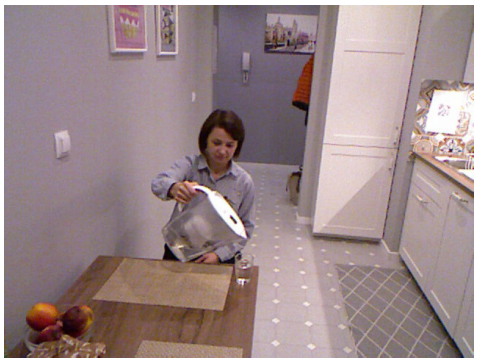
(b) Placing (CAD-60)

(c) Reaching (CAD-120)

(d) Moving (CAD-120)

(e) Reaching (WUT-17)

(f) Placing (WUT-17)

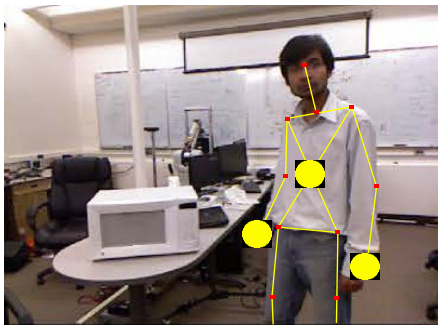


(g) Pouring (WUT-18)

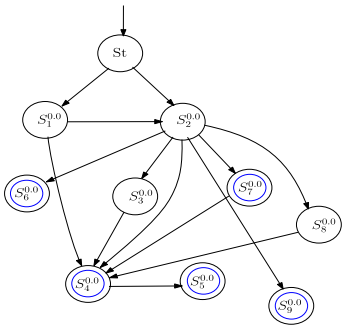
(h) Approaching (WUT-18)

FIGURE 11. **Example images of *drinking*, *placing*, *reaching*, *moving*, *pouring*, *approaching* actions from four data sets such as CAD-60 [5], CAD-120 [5], WUT-17, WUT-18.**

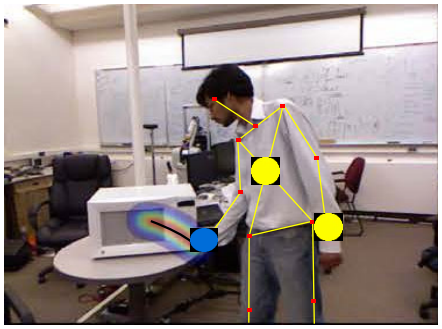
examples and used the graph-cut optimization method. The prediction accuracy was tested on four datasets (WUT-17, WUT-18, CAD60, CAD120) by comparing each predicted action with the observed action and averaging the results over the entire tests performed by the *new person*. We followed the train-test split described in [9]. Table 2 shows the



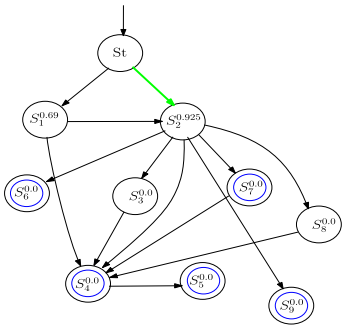
(a) Static situation



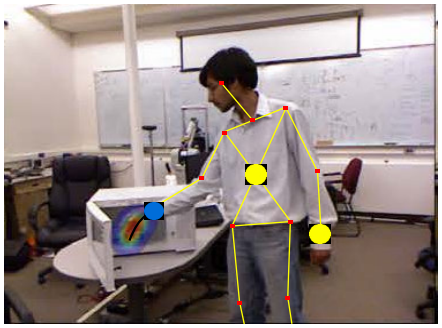
(b) Grh_{static}



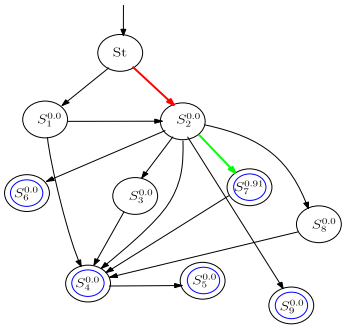
(c) Approaching



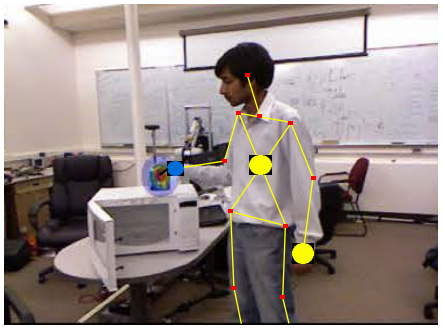
(d) $Grh_{approaching}$



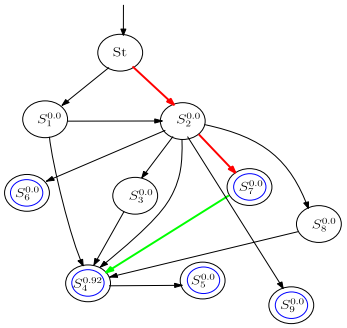
(e) Reaching



(f) $Grh_{recahing}$



(g) Placing



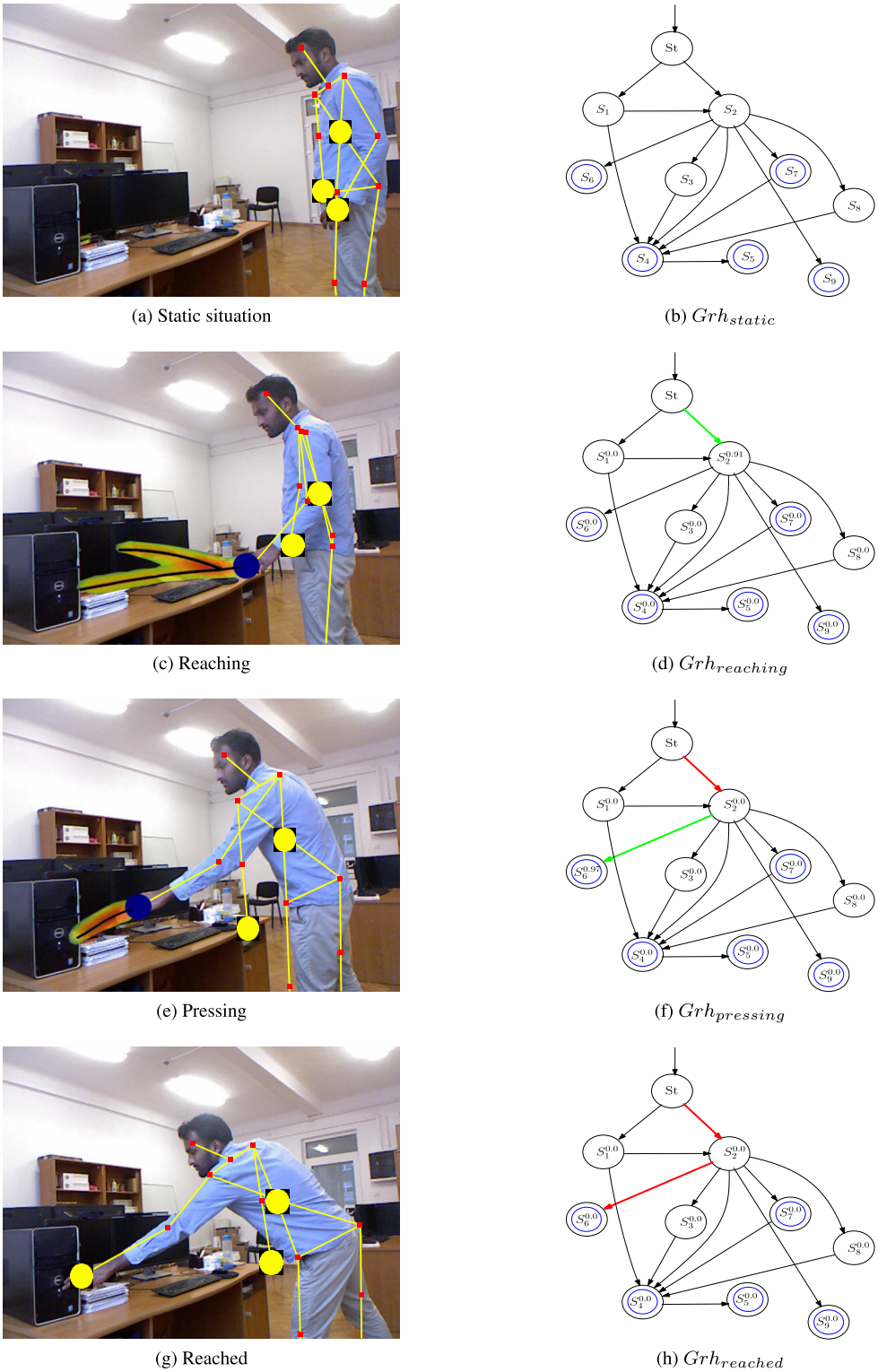
(h) $Grh_{placing}$

FIGURE 12. Experimental result of forecasting an activity "remove food from the microwave" using CAD-120 dataset. The figures in the first column show the following actions: (a) static situation (start), (c) approaching, (e) reaching, (g) placing. The second column illustrates how the progress in an activity is marked in the graph: (b) Grh_{static}, (d) $Grh_{approaching}$, (f) $Grh_{recahing}$, (h) $Grh_{placing}$ respectively.

performances comparison. The following indicator for accuracy evaluation was applied

$$P_{acc} = \frac{N_{corr}}{N_{tol}} \times 100\%, \tag{12}$$

where N_{corr} is the number of correctly predicted actions, N_{tol} is a total number of actions. We achieved good performance in predicting the activities. However, the experiment with WUT-17 dataset shows poorer accuracy due to



(a) Static situation

(b) Grh_{static}

(c) Reaching

(d) $Grh_{reaching}$

(e) Pressing

(f) $Grh_{pressing}$

(g) Reached

(h) $Grh_{reached}$

FIGURE 13. Experimental result of forecasting an activity "activating computer" using WUT-18 dataset. The figures in the first column show the following actions: (a) static situation (start), (c) reaching, (e) pressing, (g) reached. The second column illustrates how the progress in an activity is marked in the graph: (b) Grh_{static}, (d) $Grh_{reaching}$, (f) $Grh_{pressing}$, (h) $Grh_{reached}$ respectively.

background noise, low light conditions, and single camera view.

The achieved performance is compared to those reported in [5], [41], [42] (Table 2). The best accuracy of prediction is 93.02%. Fig. 12-13 show the actions sequences and the predicted motion trajectories with the heat maps around them. In the bottom part of the Fig. 12-13, the graphs representing the investigated activities are introduced in Fig. 9. The edge

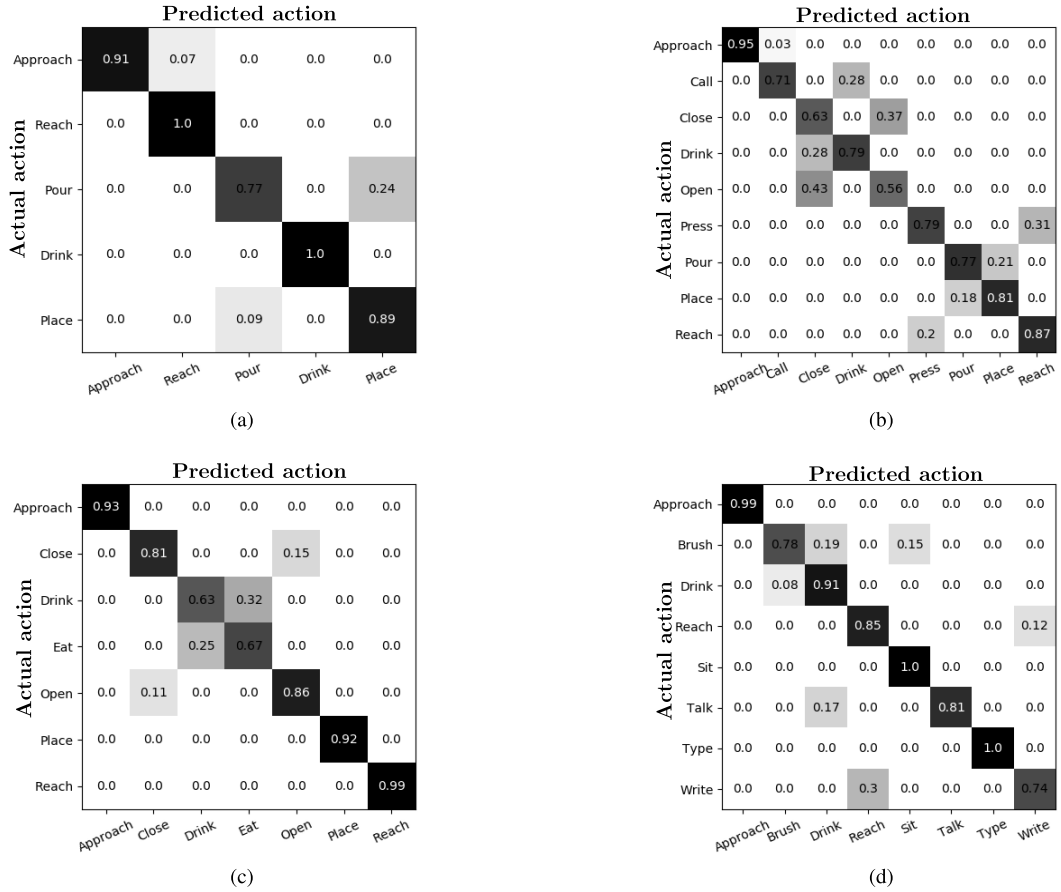


FIGURE 14. Error matrices of action prognosis on the test video records of both WUT and CAD datasets. Figure 14a and 14b show the confusion matrix of prognosis accuracy for WUT-17 and WUT-18 datasets. The confusion matrix of prediction accuracy for CAD-60 and CAD-120 test datasets are shown in Fig. 14c and 14d.

TABLE 2. Performance accuracy. The comparison of other baseline algorithms against our method on CAD-60, CAD-120, WUT-17, WUT-18 datasets.

Methods	Prediction accuracy (%)			
	CAD-60	CAD-120	WUT-17	WUT-18
HMM [41]	81	75.08	80.06	71.39
LSVM [42]	80	78.03	74.81	72.1
KGS [10]	83.08	79.89	84	77.34
ATCRF [5]	87	**91.29**	**92.89**	86.1
Ours	**92.6**	89.06	89	**93.02**

denoted by *green* color indicates the prognosing of an action. The edge represented by *red* color denotes an action already performed.

b: DATA TEST

The data test aimed at evaluating the performance that the proposed method can achieve while varying the number of training sets. Each dataset was divided into four subsets and each of the listed below experiment was repeated 5 times. At each run, we randomly selected the data for the training sets. Since the data is of a temporal sequential nature, we do not perform cross-validation for this data but average the results from multiple runs.

- *Experiment-1:* Half of the samples (50%) from the dataset is used for training and the rest is used for testing.
- *Experiment-2:* 70% of the data samples of each activity is used for training and the rest is used for testing.
- *Experiment-3:* 80% of the data samples of each activity is used for training and the rest is used for testing.
- *Experiment-4:* 95% of the data samples of each activity is used for training and the rest is used for testing.

TABLE 3. Performance accuracy (%) of the proposed method on four different experimental settings using data sets CAD-60, CAD-120, WUT-17 and WUT-18.

Experiments	Dataset			
	CAD-60	CAD-120	WUT-17	WUT-18
Experiment-1	71	68	73	71.67
Experiment-2	79	72.8	80.09	78.87
Experiment-3	87.81	84.34	87.9	89
Experiment-4	**91.1**	**90.89**	**94.27**	**96.10**

Results of the data test are given in Table 3. The performance is better for *experiment-3* and *experiment-4* as it is shown in the Table 3. In *experiment-1*, the results are worse indicating that 50% of data set (50% of motion record) is not enough for making the prognosis.

TABLE 4. Precision (%) and Recall (%) of the performance of our proposed method on four different activities.

Activity	No. of users	No. of objects	Actions	New Person	
				Precision (Pr)	Recall (Re)
Making Cereal	1	3	Approaching	**97.8**	96.3
			Reaching	97.0	**96.8**
			Pouring	88.1	87.6
Microwaving food	1	2	Reaching	**89**	**88.3**
			Opening	76.9	71.6
			Placing	86.7	85.1
Activating computer	1	4	Reaching	**88.7**	86.1
			Pressing	83.4	**87.2**
Arranging books	1	3	Reaching	**90.0**	85.6
			Passing	81.3	80.4
			Placing	87.0	**85.8**

Fig. 14 illustrates the outcomes using error matrix which shows the interpretable aspect of the proposed method and its ability of making the correct prognosis. Note that Fig. 14b and 14c indicate few errors, such as *closing* action sometimes was miss-classified as an *opening* action, the reason is that the movements range of the hand in both scenarios is minimal. Moreover, in various scenario the *pouring* action was predicted as a *placing* action and *talking* action was miss-classified as a *drinking* action and so on due to the problem with light sensitive object recognition. During the experimental evaluation we also detected that the proposed method performs poorly when the actions are not finished and repeated. Such a situation occurs when the change of an action arises close to the final stage and the same action is again initiated and started from the initial stage.

Following [9], we applied the binary scores TP, TN, FP, and FN respectively. TP, TN, FP, and FN denote *true positive, true negative, false positive and false negative* respectively. The following measures were used to evaluate the precision $(Pr) = \frac{TP}{TP+FP}$, recall $(Re) = \frac{TP}{TP+FN}$. We tested four activities for which the precision and recall values are listed in Table 4.

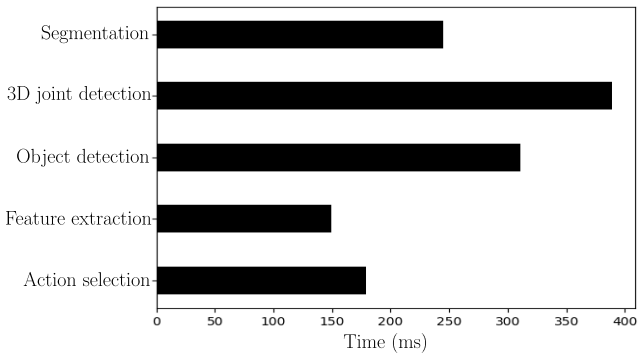


FIGURE 15. Average processing time for the data analysis steps.

In Fig. 15, we illustrate the average processing time for each step in the testing process. The most time consuming step is the step responsible for skeleton points detection, objects detection, and tracking. The features extraction and action selection take overall less processing time.

c: GENERAL NOTE

The CAD datasets were recorded using first generation of Kinect sensor (Kinect V1) with resolution 320 × 240. Since, Kinect V1 sometimes provides poor readings for skeleton poses due to poor depth resolution, the achieved joints detection accuracy for CAD datasets was up to 81%. Joints detection accuracy using WUT datasets was better (up to 89%) because of good quality of applied recordind system. The WUT dataset was recorded using Intel RealSense camera working in a range 1m - 3m, the applied resolution of images was 640 × 480.

VI. CONCLUSION AND FUTURE WORK

In this paper, we proposed and described the method for human activity prediction considering the spatio-temporal human-object relations. We used the sensory system capable of collecting visual and depth information. Such information was used to obtain the relevant attributes which were later stored in the structuralized database. In our approach, the activities are described by the sequences of actions concerning the involved objects. The list of the objects which can be recognized by our system together with the actions performed on them make the first part of our database. The parameters used in probability functions indicating the possible actions are stored in the second part of the database. Those data concern all considered activities and objects. The choice of probability functions was experimentally justified. The relevant details of the methods were introduced covering the training and testing stage. The performed experiments proved that the proposed method allows to forecast an activity regardless of different scenarios and the speed at which the actions were executed. The performance was tested off-line using real-life scenarios with four datasets.

There are some limitations of our method: our data only includes cases in which the person was not occluded by partial body view or by other people; our method may not be robust in such situations. We designed the sensing system

carefully for avoiding the disturbances and occlusions. However, in broader sense, when occlusions occur it may lead to a poor performance in prognoses. Some actions when the hand is repeatedly moving forwards and backwards without finishing an action degrades the performance of the prognosis. Because of that such cases were not considered and not included in the database. In this work, we only considered the parameters for finished actions. It must be also added that we do not optimized the real-time performance of our system. This work refers to the activity prediction based on the benchmark datasets. The concept of structured database orients the search what speeds up the access to the needed data, however, addition of many new actions and their corresponding parameters to the database might slow down the accessing the needed data therefore declining the forecasting speed. Therefore, the forecasting speed should be further improved using distributed calculations and stronger computers with a more efficient operating systems.

In the future, we intend to study the prediction process in real-time implementation considering the robot assistants. More data should be also collected considering more complex activities and the method should be tested to a bigger extent. The database should be expanded by more cases of everyday human activity. Obviously, the addition of a new data will increase the size of the database. Therefore, it could be useful to investigate the problem of more advanced structuring of the database for speeding up the search method further. The example of this investigation can be the division of the second part of the database into the parts with data for the objects grouped by their size or by their application (e.g. object in an office environment, in-home environment). Then the data search for parameters of probability functions will concern only the part for relevant objects. Besides refining the database, more studies should be done on activities prediction with deeper insight into the forecasting efficiency taking into account the different possible scenarios where occlusion occurs or when the object tracking accuracy drops.

APPENDIX

A. DISTANCE PREFERENCE

The *distance preference* $P(d^{Ho})$ is a probability function which takes into account current human-object distance. It uses the current value of d^{Ho} comparing it to the mean value $\mu_{d^{Ho}}$ and the variance $\sigma^2_{d^{Ho}}$ obtained during training phase for each segment of motion. The distance preference is described by a normal *Gaussian* distribution parameterized by mean $\mu_{d^{Ho}}$ and variance $\sigma^2_{d^{Ho}}$.

$$P(d^{Ho}) = \mathcal{N}\left(d^{Ho}; \mu_{d^{Ho}}, \sigma^2_{d^{Ho}}\right) = \frac{1}{\sqrt{2\pi\sigma^2_{d^{Ho}}}} \exp^{-\frac{1}{2}\left(\frac{d^{Ho}-\mu_{d^{Ho}}}{\sigma^2_{d^{Ho}}}\right)}$$

The standard statistical test was applied to check whether the data are consistent with the selected distribution. A common test in such case is a *Shapiro Wilk* normality test, it has good performance for the smaller amount of samples as it was in our case [9].

B. ANGULAR PREFERENCE

The *angular preference* $P(\theta)$ a probability function based on circular distribution, where the data are expressed in an angular scale, the parameters are current value of θ, mean value μ_θ and the variance σ^2_θ. Angular position towards the object is relevant for certain actions. The angular preference is defined by the modified *wrapped normal* distribution, as it was described in [43]. The angular preference is described by:

$$P(\theta) = \mathcal{N}\left(\theta; \mu_\theta, \sigma^2_\theta\right) = \frac{1}{2\pi}\left(1+2\sum_{k=1}^{K}\left(\exp^{-\frac{\sigma^2_\theta}{2}}\right)^{k^2}\cos(k(\theta-\mu_\theta))\right)$$

The distribution covers the area $[0, 2\pi]$. For this distribution, we used *goodness-of-fit* test based on *Watson's* U^2 to justify the applied distribution. The *goodness-of-fit* test enables us conclude that the conisdered functions is sufficient and more complex relations are not needed. The details are given in [9].

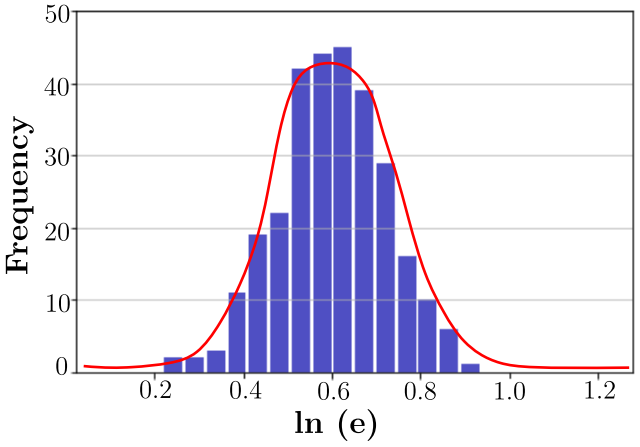


FIGURE 16. The figure shows the histogram plot of the data justifying the *log-normal* distribution. The variable edge e is log-normally distributed.

C. EDGE PREFERENCE

During the interactive actions where the human and the object are separated by a distance the *edge preference* $P(e)$ is considered. The likelihood of the next position of the hand is the position depends on the normalized distance - the edge (Eq. 8). This function considers the human hand - object distance related to the hand distance to the sensor plane (Fig. 6) in this paper. The edge preference is described by *log-normal* distribution function:

$$P(e) = \mathcal{N}\left(e; \mu_e, \sigma^2_e\right) = \frac{1}{e\sqrt{2\pi}\sigma_e} \exp^{-\left(\frac{(\ln(e)-\mu_e)^2}{2\sigma^2_e}\right)}, \quad e > 0$$

To validate the correctness of applied function we followed the statistical test described in [9]. We analyzed the data for a *log-normal* distribution transforming the data (i.e., e) using the logarithm. Transformed data have the Gaussian normal distribution. Here also the *Shapiro Wilk* normality test was applied to justify the distribution. The plot is given in Fig. 16 proofs that data are following the normal distribution.

REFERENCES

[1] V. Krüger, D. Kragic, A. Ude, and C. Geib, "The meaning of action: A review on action recognition and mapping," *Adv. Robot.*, vol. 21, no. 13, pp. 1473–1501, 2007.

[2] A. Chrungoo, S. S. Manimaran, and B. Ravindran, "Activity recognition for natural human robot interaction," in *Proc. Int. Conf. Social Robot.*, 2014, pp. 84–94.

[3] M. Ryoo, T. J. Fuchs, L. Xia, J. K. Aggarwal, and L. Matthies, "Robot-centric activity prediction from first-person videos: What will they do to me?" in *Proc. 10th IEEE/ACM Int. Conf. Hum.-Robot Interact.*, Mar. 2015, pp. 295–302.

[4] V. Dutta and T. Zielinska, "Predicting the intention of human activities for real-time human-robot interaction (HRI)," in *Proc. Int. Conf. Social Robot.*, 2016, pp. 723–734.

[5] H. S. Koppula and A. Saxena, "Anticipating human activities using object affordances for reactive robotic response," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 1, pp. 14–29, Jan. 2016.

[6] W. Choi, K. Shahid, and S. Savarese, "Learning context for collective activity recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 3273–3280.

[7] A. Gupta, A. Kembhavi, and L. Davis, "Observing human–object interactions: Using spatial and functional compatibility for recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 10, pp. 1775–1789, Oct. 2009.

[8] V. Dutta and T. Zielinska, "Activities prediction using structured data base," in *Proc. 12th IEEE Int. Workshop Robot Motion Control (RoMoCo)*, Jul. 2019, pp. 80–85.

[9] V. Dutta and T. Zielinska, "Predicting human actions taking into account object affordances," *J. Intell. Robot. Syst.*, vol. 93, nos. 3–4, pp. 745–761, Mar. 2019.

[10] H. S. Koppula, R. Gupta, and A. Saxena, "Learning human activities and object affordances from RGB–D videos," *Int. J. Robot. Res.*, vol. 32, no. 8, pp. 951–970, Jul. 2013.

[11] W. Takano and Y. Nakamura, "Construction of a space of motion labels from their mapping to full-body motion symbols," *Adv. Robot.*, vol. 29, no. 2, pp. 115–126, Jan. 2015.

[12] D. Xu, X. Xiao, X. Wang, and J. Wang, "Human action recognition based on Kinect and PSO-SVM by representing 3D skeletons as points in lie group," in *Proc. IEEE Int. Conf. Audio, Lang. Image Process. (ICALIP)*, Jul. 2016, pp. 568–573.

[13] L. Xia, "Recognizing human activity using RGBD data," Ph.D. dissertation, Univ. Texas Austin, Austin, TX, USA, 2014.

[14] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3D points," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2010, pp. 9–14.

[15] B. Ni, G. Wang, and P. Moulin, "RGBD-HuDaAct: A color-depth video database for human daily activity recognition," in *Proc. Consum. Depth Cameras Comput. Vis.*, 2013, pp. 193–208.

[16] J. Sung, C. Ponce, B. Selman, and A. Saxena, "Unstructured human activity detection from RGBD images," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2012, pp. 842–849.

[17] H. Zhang and L. E. Parker, "4-dimensional local spatio-temporal features for human activity recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2011, pp. 2044–2049.

[18] J. Yamato, J. Ohya, and K. Ishii, "Recognizing human action in time-sequential images using hidden Markov model," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jan. 1992, pp. 379–385.

[19] A. Gritai, A. Basharat, and M. Shah, "Geometric constraints on 2D action models for tracking human body," in *Proc. 19th IEEE Int. Conf. Pattern Recognit. (ICPR)*, Dec. 2008, pp. 1–4.

[20] P. Huang, A. Hilton, and J. Starck, "Human motion synthesis from 3D video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 1478–1485.

[21] G. Guo and A. Lai, "A survey on still image based human action recognition," *Pattern Recognit.*, vol. 47, no. 10, pp. 3343–3361, Oct. 2014.

[22] J. Aggarwal and L. Xia, "Human activity recognition from 3D data: A review," *Pattern Recognit. Lett.*, vol. 48, pp. 70–80, Oct. 2014.

[23] L.-J. Li and L. Fei-Fei, "What, where and who? Classifying events by scene and object recognition," in *Proc. 11th IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2007, pp. 1–8.

[24] Y. Cao, D. Barrett, A. Barbu, S. Narayanaswamy, H. Yu, A. Michaux, Y. Lin, S. Dickinson, J. Mark Siskind, and S. Wang, "Recognize human activities from partially observed videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2658–2665.

[25] X. Yang and Y. L. Tian, "Eigenjoints-based action recognition using Naïve-Bayes-nearest-neighbor," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2012, pp. 14–19.

[26] D. Gehrig, P. Krauthausen, L. Rybok, H. Kuehne, U. D. Hanebeck, T. Schultz, and R. Stiefelhagen, "Combined intention, activity, and motion recognition for a humanoid household robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2011, pp. 4819–4825.

[27] Y. Kim, J. Chen, M.-C. Chang, X. Wang, E. M. Provost, and S. Lyu, "Modeling transition patterns between events for temporal human action segmentation and classification," in *Proc. 11th IEEE Int. Conf. Workshops Autom. Face Gesture Recognit. (FG)*, vol. 1, May 2015, pp. 1–8.

[28] K. M. Kitani, B. D. Ziebart, J. A. Bagnell, and M. Hebert, "Activity forecasting," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 201–214.

[29] W. Takano, T. Jodan, and Y. Nakamura, "Recursive process of motion recognition and generation for action-based interaction," *Adv. Robot.*, vol. 29, no. 4, pp. 287–299, Feb. 2015.

[30] I. Kostavelis, M. Vasileiadis, E. Skartados, A. Kargakos, D. Giakoumis, C.-S. Bouganis, and D. Tzovaras, "Understanding of human behavior with a robotic agent through daily activity analysis," *Int. J. Social Robot.*, vol. 11, no. 3, pp. 437–462, Jun. 2019.

[31] V. Dutta. (2017). *WUT-17*. Accessed: Mar. 11, 2017 [Online]. Available: https://ztmir.meil.pw.edu.pl/web/Pracownicy/mgr-Vibekananda-Dutta

[32] C. R.-D. Camera. *Senz3D*. Accessed: 2017. [Online]. Available: https://us.creative.com/p/web-cameras/creative-senz3d/

[33] F. L. Siena, B. Byrom, P. Watts, and P. Breedon, "Utilising the Intel realsense camera for measuring health outcomes in clinical research," *J. Med. Syst.*, vol. 42, no. 3, p. 53, 2018.

[34] V. Dutta, "Mobile robot applied to QR landmark localization based on the keystone effect," *Mechatronics and Robotics Engineering for Advanced and Intelligent Manufacturing*. Cham, Switzerland: Springer, 2017, pp. 45–60.

[35] CVLAB. (2016). *Robust 3D Tracking With Descriptor Fields*. Accessed: Nov. 19, 2016. [Online]. Available: https://cvlab.epfl.ch/research/page-90554-en-html/page-107683-en-html/

[36] H. S. Koppula and A. Saxena, "Anticipating human activities for reactive robotic response," in *Proc. IROS*, 2013, p. 2071.

[37] Y. Jiang and A. Saxena, "Modeling high-dimensional humans for activity anticipation using Gaussian process latent CRFs," in *Robotics: Science and Systems*. Berlin, Germany: Robotics, Science and Systems Foundation, 2014, pp. 1–8.

[38] T. Joachims, T. Finley, and C.-N.-J. Yu, "Cutting-plane training of structural SVMs," *Mach. Learn.*, vol. 77, no. 1, pp. 27–59, Oct. 2009.

[39] C. Rother, V. Kolmogorov, V. Lempitsky, and M. Szummer, "Optimizing binary MRFs via extended roof duality," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.

[40] B. Taskar, V. Chatalbashev, and D. Koller, "Learning associative Markov networks," in *Proc. 21st Int. Conf. Mach. Learn.*, 2004, p. 102.

[41] L. E. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state Markov chains," *Ann. Math. Statist.*, vol. 37, no. 6, pp. 1554–1563, Dec. 1966.

[42] H. Wu, W. Pan, X. Xiong, and S. Xu, "Human activity recognition based on the combined SVM&HMM," in *Proc. IEEE Int. Conf. Inf. Autom. (ICIA)*, Jul. 2014, pp. 219–224.

[43] V. Dutta and T. Zielinska, "Action prediction based on physically grounded object affordances in human-object interactions," in *Proc. 11th IEEE Int. Workshop Robot Motion Control (RoMoCo)*, Jul. 2017, pp. 47–52.

VIBEKANANDA DUTTA received the M.Sc. degree in computer science, with specialization in artificial intelligence, from the Central University of Rajasthan, Rajasthan, India, in 2012, and the Ph.D. degree in automation and robotics from the Warsaw University of Technology, Warsaw, Poland, in 2019. His researches focus on the human–robot interaction, computer vision, and mobile robots.

TERESA ZIELINSKA received the Ph.D., D.Sc., and M.Sc. degrees in engineering. She has authored or coauthored of over 300 research publications. She focuses her research interests on novel robotic systems, walking machines, humanoids, and mobile robots. She also works on real-time control systems and interfacing methods for no-conventional robotic applications. She is the Secretary General of IFToMM. She is a member of the editorial board of several international journals devoted to the robotics and mechanics.