Human activity recognition from object interaction in domestic scenarios*

Carlos Flores-Vázquez
School of Electrical Engineering
University Catholic of Cuenca
Cuenca, Ecuador
Email: cfloresv@ucacue.edu.ec

Joan Aranda
Institute for Bioengineering of Catalunya
Universitat Politècnica de Catalunya, Barcelona-Tech
Barcelona, Spain
Email: joan.aranda@upc.edu

Abstract—**This paper presents a real time approach to the recognition of human activity based on the interaction between people and objects in domestic settings, specifically in a kitchen. Regarding the procedure, it is based on capturing partial images where the activity takes place using a colour camera, and processing the images to recognize the present objects and its location. For object description and recognition, a histogram on rg chromaticity space has been selected. The interaction with the objects is classified into four types of possible actions; (unchanged, add, remove or move). Activities are defined as receipts, where objects plays the role of ingredients, tools or substitutes. Sensed objects and actions are then used to analyze in real time the probability of the human activity performed at particular moment in a continuous activity sequence.**

Index Terms—**Human Activity, Computer Vision, Human Computer Interaction, Human/Robot Interaction.**

I. Introduction

Robotic assistance in domestic environments imposes special requirements due to the need to adapt to a great diversity of users and to a wide variety of situations. Assumptions on human behavior or sequence of activities cannot be easily specified, as it happens in industrial applications where their structured environment allows a previous planning and predefined response actions that can be programmed.

Assistive robotics needs to identify human activity, aware and provide a proper service. Most of present human activity recognition methods rely on perfectly segmented input sequences with well defined start and end triggers, and they require being finished before proceeding with recognition.

However, proactive assistance by robot companions needs to recognize human activity while it is performed. This work will focus on the recognition of daily actions by taking into account only the manipulated objects and their movements. We propose to identify and locate the objects present in the scene by computer vision, and detect their position changes due to the user manipulation. It is not intended to continuously track the objects, but only register their initial and final positions.

With the application of proactive assistance in mind, we will look for a method capable of assigning probabilities to a set of pre-known activities in real time, with the aim to identify the ongoing activity. This recognition is limited to a kitchen environment and basic activities, such as the ones related to the preparation of the breakfast. The object interaction approach takes into account the importance of these objects being in the field of vision, being brought to it, removed or moved in the scene. These actions are supposed to be carried out by an agent; the user, therefore there is no need to make an analysis of the user trajectories under this approach.

II. State Of The Art

Aggawar and Ryoo [1], carried out an excellent study on the different trends and theories to tackle the study of human activity. They distinguish between two big groups to classify the different existing approaches: *Single-layered* approaches and *Hierarchical* approaches. In addition, they contemplated another type of approaches: *Human-Object Interaction* and *Group Activities*.

In this work, we will apply a *Human-Object Interactions* approach partially extended with some characteristics from the *Syntactic* approaches and *Description-based* approaches, according to taxonomy presented in [1]. Syntactic approaches use grammar syntax such as stochastic context-free grammar (SCFG) to model sequential activities. Essentially they are modeling a high-level activity as a string of atomic-level activities. Description-based approaches represent human activities by describing sub-events of the activities and their temporal, spatial, and logical structures.

Hongeng et al. [2], presented a work in line with the Description-Based methodology, its recognition system has two clearly differentiated modules, the first one is Motion Detection and Tracking" and the second, is "Event Analysis". We agree with use the low cost idea that have a scene view provided by only one camera and segment by subtracting the background, although we differ on the method. They used the intensity variations, however, we apply the *Image Difference* because we believe it is faster and equally reliable on constant illumination conditions.

As for the use of the "Syntactic approaches" method we stand out with the one presented by Moore and Essa [3],

*This research is supported by InHANDS project (Interactive robotics for Human Assistance in Domestic Scenarios), grant P6-L13-AL.INHAND founded by Fundaci La Caixa, inside the Recercaixa research program

C. Flores and J. Aranda are associated with the Institute for Bioengineering of Catalunya and Universitat Politcnica de Catalunya, Barcelona-Tech, Spain. (Corresponding authors e-mail: cfloresv@ucacue.edu.ec).

in which they represent every ”action event” with a unique symbol allowing to represent a sequence of interactions as a string of symbols. Our approach differs on one aspect that is our symbol would turn into a word and an activity would be made of a list of words and not necessarily in order. Thus, our method for activity description and recognition is more related to bag-of-words model (BOW) which is well explained in references [4] and [5]. In this model, a text (such as a sentence or a document) is represented as the bag (multiset) of its words, disregarding grammar and even word order but keeping multiplicity.

BOW is also widely used in computer vision, allowing us to treat an image as a document in which we find words and their repetition in order to recognize the document, using features or words. Liefeng and Sminchescu [4] stated that BOW is one of the most popular methods to represent images, by being conceptually simple and computationally efficient. They support this using BOW together with several types of classifiers for three sets of databases, obtaining satisfactory results.

Ryoo in [5] also considered as an important objective the activity recognition before it finishes, that is during its execution. This way, a probabilistic prediction of these activities can be performed, which matches with the approach presented in this paper.

Lei et al. [6] presented a method for human activity recognition in a kitchen. They demonstrated the ability to identify objects using a Kinect-style camera as the main resource, in combination with RFID tags when needed. Input data for the activity recognition they used split into two basic categories. First, with hand and object tracking, use depth to robustly track the positions of hands and objects, and detect when and where hands interact with the objects (e.g. grasp). Second, with object and action recognition, use both depth (shape) and colour (appearance) to identify objects and to recognize the actions being performed on them. This is the method used when focusing on the actions in their project consists of seven common actions: place (PL), move (MV), chop (CH), mixing (MX), pouring (PR), spooning (SP) and scooping (SC). They proved the reliability of that system defining the preparation of a cake as the activity to recognize. This activity was expressed in terms of 7 objects, 17 actions, about 6000 frames and approximately 200 seconds length.

III. Proposed Approach

We present a method to evaluate the instantaneous probability of a given set of predefined activities in order to to identify the ongoing activity in robotic assistance applications in real time.

This method is based on object presence evolution in the surrounding area of the user, as seen from a standard camera. To get this goal our system needs to recognize and locate the objects and be aware about their movements.

We define the actions that take place with the objects as the followings:

- ADD: It means that the user adds the object to the scene.

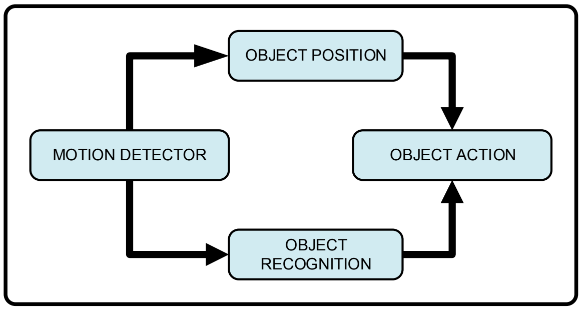

Fig. 1. To build an object system flowchart.

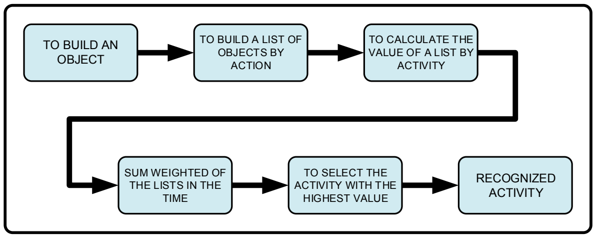
TO BUILD AN OBJECT
TO BUILD A LIST OF OBJECTS BY ACTION
TO CALCULATE THE VALUE OF A LIST BY ACTIVITY
SUM WEIGHTED OF THE LISTS IN THE TIME
TO SELECT THE ACTIVITY WITH THE HIGHEST VALUE
RECOGNIZED ACTIVITY

Fig. 2. Activity recognition system flowchart

- REMOVE: It means that the user removes the object from the scene.
- UNCHANGED: It means that the object is still present in the scene.
- MOVE: It means that the user movs the object in the scene.

The complete system to perform our activity recognition from object interaction is outlined in Fig. 1. It follows a detailed explanation of how this system was conceived.

A. Object recognition and definition

For the object recognition, a motion detector based on image difference is applied to extract the regions of interest (ROI) [7]. Then, a histogram in the rg chromaticity space is generated for each ROI to be used as a descriptor. We choose this space to reduce the problems related to the brightness variation in the scene [8]. Black and white colours can cause singularities in such a colour space, so they are treated specifically and assigned to particular bins. Obtained histograms are then compared against all the models stored in our database by means of Bhattacharyya distance.

In parallel, another process is used to locate the object and finally we establish the action carried out by the user depending on the previous processes (ADD, REMOVE, MOVE, UNCHANGED).

Finally the definition of an object consists of four parameters or characteristics as follows:

1) The IDENTIFICATION NUMBER. (I.D. Number).
2) The COLOUR histogram that defines the model of the object recognized by our system (Colour).

3) The position that consists in the CENTROID coordinates based on the frame of reference specified through the homography (including the known height of the object) [9] (Centroid).
4) The ACTION that defines the object-manipulation last state (ADD, REMOVE, MOVE, UNCHANGED). By default, the ACTION is equal to ”UNDETERMINED”.

B. Object action statement

As presented above, human object interaction is described in this work by 4 options of object-manipulation by the user (Add, Remove, Move and Unchanged).

After building the object with the obtained characteristics from recognition and location procedures (I.D. Number, Colour, and Centroid), action is initially set to the state of ”UNDETERMINED”.

In order to assign the correct state for ”Action”, a comparative analysis between consecutive lists of objects is performed. The list contains the objects appearing in the scene in present time (t) and the list of objects in previous time (t-1). From changes in the list we establish the following actions for the objects: REMOVE, ADD, MOVE and UNCHANGED.

This algorithm is explained in detail as follows:

REMOVE: Is the first action to be considered. It is assigned to those objects that are present in (t-1) and not present in (t), so they must be removed by the user and action is set to REMOVE.

ADD: With the remaining elements in the lists now the algorithm look for those objects in (t) that are not present in (t-1). These elements will be the objects recently added by the user and action is set to ADD.

MOVE AND UNCHANGED: Now, only the objects that coincide with the lists (t) and (t-1) rest UNDETERMINED. The algorithm checks the position of the objects, in other words, it compares their positions in the list (t) in relation to (t-1). If they present difference between positions above a certain threshold, the algorithm considers that the user has moved the object and action is set to MOVE. In the opposite case, action is set to UNCHANGED. It is important to have a small threshold that allows us to detect little movements for cases where the user takes the object and leaves it in almost the same position (In our experiments is set to 5 mm.). Even this little difference must be registered as a movement in our approach.

C. Human activity description

As in syntactic approaches our method uses a syntax to define human activity. Nevertheless, we do not consider a sequential order. We consider sub-events from activities and its temporality but without the spatial consideration and a logical structure.

The methodology that comes closer to the implemented model is BOW (bag of words). BOW represents each local visual feature with the closest word and counts the occurrence frequencies in the image [4]. In this way, every object in the image (with its own characteristics) represents a ”word”, and a specific set of words represents an activity. It is necessary to stress that this set of words is not limited by a specific sequence of the words. The relevancy of each one of these words in a set would allow us to differ between activities.

D. Definition of an activity

Our approach is inspired by a recipe, so for activity definition we will use a list of ingredients, tools and possible substitutes to define an activity.

In the implemented context of a kitchen this components can be better explained as:

- INGREDIENTS: It is the list of ingredients related to the activity described, e.g. for a coffee-activity (Coffee, milk, sugar).
- TOOLS: It is a list of kitchen utensils related to the activity described, e.g. coffee-activity (cup, spoon).
- SUBSTITUTES: It is a list of replacement for both kitchen utensils or ingredients related to the activity described, e.g. coffee-activity (glass).: It is a list of replacement for both kitchen utensils or ingredients related to the activity described, e.g. coffee-activity (glass).
  Every component is associated an index of contribution to the activity membership. This index will be used later during the recognition stage.

E. Evaluation function and activity recognition

As a result of the object recognition process, we obtain all the information needed from the objects to represent in the scene. For activity recognition, the present objects are separated in three different sets depending on the action field, which is the last corresponding action performed by the user: MOVE, ADD or UNCHANGED.

Then to proceed with the calculation of the probability of each activity, first we calculate the value of activity components (list of ingredients, list of utensils and list of substitutes). This value is calculated taking into account the contribution or relevancy of each one of these objects in a particular activity, which is predefined during activity definition. We understand that the same object will have a different value for each of the activities. Even more, the same object could be an ingredient, a tool or a substitute depending on the activity.

$$\begin{bmatrix} V_{A1} \\ . \\ . \\ . \\ V_{An} \end{bmatrix}_M = a \cdot \begin{bmatrix} I_{A1} \\ . \\ . \\ . \\ I_{An} \end{bmatrix}_M + b \cdot \begin{bmatrix} U_{A1} \\ . \\ . \\ . \\ U_{An} \end{bmatrix}_M + c \cdot \begin{bmatrix} S_{A1} \\ . \\ . \\ . \\ S_{An} \end{bmatrix}_M \quad (1)$$

$$\begin{bmatrix} V_{A1} \\ . \\ . \\ . \\ V_{An} \end{bmatrix}_A = a \cdot \begin{bmatrix} I_{A1} \\ . \\ . \\ . \\ I_{An} \end{bmatrix}_A + b \cdot \begin{bmatrix} U_{A1} \\ . \\ . \\ . \\ U_{An} \end{bmatrix}_A + c \cdot \begin{bmatrix} S_{A1} \\ . \\ . \\ . \\ S_{An} \end{bmatrix}_A \quad (2)$$

$$\begin{bmatrix} V_{A1} \\ . \\ . \\ . \\ V_{An} \end{bmatrix}_{Un} = a \cdot \begin{bmatrix} I_{A1} \\ . \\ . \\ . \\ I_{An} \end{bmatrix}_{Un} + b \cdot \begin{bmatrix} U_{A1} \\ . \\ . \\ . \\ U_{An} \end{bmatrix}_{Un} + c \cdot \begin{bmatrix} S_{A1} \\ . \\ . \\ . \\ S_{An} \end{bmatrix}_{Un} \tag{3}$$

- $V_A =$ Value by Activity
- $I_A =$ Value based on the occurrence of the Ingredients by Activity
- $U_A =$ Value based on the occurrence of the Utensils by Activity
- $S_A =$ Value based on the occurrence of the Substitutes by Activity
- $M, A, Un =$ MOVE, ADD, UNCHANGED
- $a, b, c =$ Constants, $a + b + c = 1$.

The constants a, b and c tuned the global influence of ingredients, utensils and substitutes on the activity evaluation lists.

Finally, to obtain the activities probabilities a weighted addition of the values obtained from every list by activity is performed (Eq.4).

$$\begin{bmatrix} \Sigma V_{A1} \\ . \\ . \\ . \\ \Sigma V_{An} \end{bmatrix} = \alpha \cdot \begin{bmatrix} V_{A1} \\ . \\ . \\ . \\ V_{An} \end{bmatrix}_{M} + \beta \cdot \begin{bmatrix} V_{A1} \\ . \\ . \\ . \\ V_{An} \end{bmatrix}_{A} + \gamma \cdot \begin{bmatrix} V_{A1} \\ . \\ . \\ . \\ V_{An} \end{bmatrix}_{Un} \tag{4}$$

- $\Sigma V_A =$ The sum of Value by Activity (activity recognized instantaneously).
- $\alpha, \beta, \gamma =$ Variables depending on the time, $\alpha + \beta + \gamma = 1$.

$$\alpha = \frac{1}{3} + \left(\frac{1}{6} - \gamma\right) \tag{5}$$

$$\beta = \frac{1}{3} + \left(\frac{1}{6} - \gamma\right) \tag{6}$$

$$\gamma = \frac{1}{3} - \left(\frac{ElapsedTime}{AverageTime}\right) \tag{7}$$

- $ElapsedTime =$ The elapsed time from the initiation of activity.
- $AverageTime =$ Average time for the execution of predefined activities.

Initially $[V_A]_M, [V_A]_A, [V_A]_{Un}$ have an equivalent value but thanks to α, β, γ factors, Unchanged objects in the scene gradually lose weight in favor of Add or Moved objects (used ones).

The highest value in (4) indicates the most probable activity in the present moment. However, this instantaneous probability of a given activity is highly dependent on last measure and noise, therefore it is necessary to filter data results. This can be solved by means of the integral value of the results along the time.

This work proposes a sum of ΣV_A in a period of time, so that the recognized activity is the result of the maximum resultant value of the sum of the samples of activity recognized instantaneously. The computation of the period starts from it detects the first scene change and stops realizing it and reset when the movement stops for a specified long period.

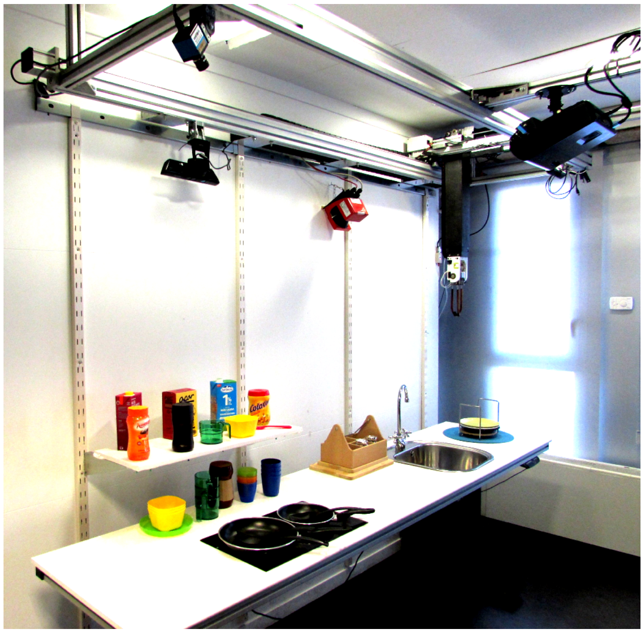

Fig. 3. InHANDS automated kitchen scenario.

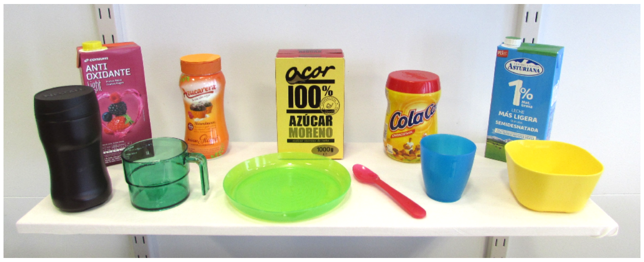

Fig. 4. Selected objects for the experiment.

$$ActivityRecognized = max \left\{ \sum_{1}^{Tsamples} \begin{bmatrix} \Sigma V_{A1} \\ . \\ . \\ . \\ \Sigma V_{An} \end{bmatrix} \right\} \tag{8}$$

- $Tsamples =$ Total samples of instantaneous activity recognized.

IV. Implementation and Results

For experiments on a real domestic scenario we count on the automated kitchen developed under InHands project (Fig.2) [10]. We limit the recognition to a kitchen environment and basic activities, such as the ones related to the preparation of breakfast. We defined four activities to be recognized: the preparation of chocolate milk, coffee with milk, juice and cereal. The selected objects involved in these activities were:

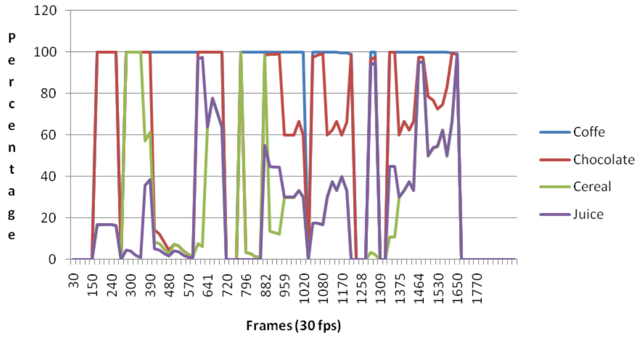

Fig. 5. Instantaneous activity recognized: COFFEE.

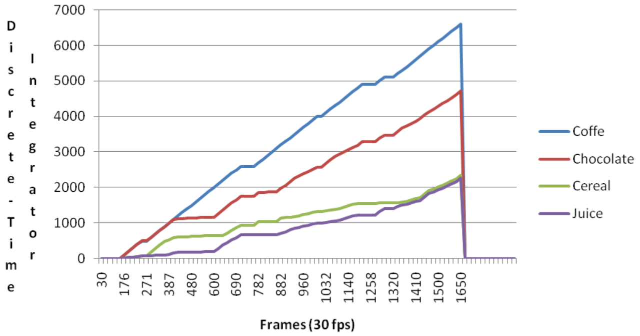

Fig. 6. Final activity recognized: COFFEE.

bowl, cup, glass, plate, spoon, sugar, cereals, coffee, chocolate, juice and milk (Fig. 3).

For evaluating the object recognition process a confusion matrix was used [11]. We took 170 pictures for every object (a total of 1870 images). We take images from 5 different points of view and in different scene locations with different light conditions. A lot of objects give us 100% precision values, however the worst precision was for cereals with only 81%.

For activity recognition evaluation, two kind of tests were developed: isolated activities and continuous performed activities.

The first one was composed on a series of previously segmented image sequences showing the evolution of only one activity. In these examples the activity is isolated without previous or posterior activities. Five different executions of every defined activity were presented to the system. For each execution we obtain the instantaneous activity recognition over time ΣV_A and the Activity Recognized.

Fig. 4 and Fig. 5 show one sample of the tests carried out that belongs to a preparation of a coffee sequence. Fig. 4 corresponds to the instantaneous recognition of activity, which just depend on objects appearing in last picture in the sequence and last actions on them. We can observe the high instability in the output which can be expected principally due to the fact that our defined activities share some common objects. Obviously, our method also suffers from classical misclassification of objects related to partial occlusions or changing lighting conditions in such dynamic scene. As presented in section III, we solve these issues by sum in a period of time the instantaneous recognitions over time. Fig. 5 corresponds to the Activity Recognized, in other words, the activity performed in the interval of time during which there was movement. It can be observed how all activities grew up during the execution feed by instantaneous observations but at different slopes. At the end of the activity execution, the highest scored activity indicates the correct human activity performed in front of the camera. The results were excellent and all the 20 video sequences were perfectly recognized.

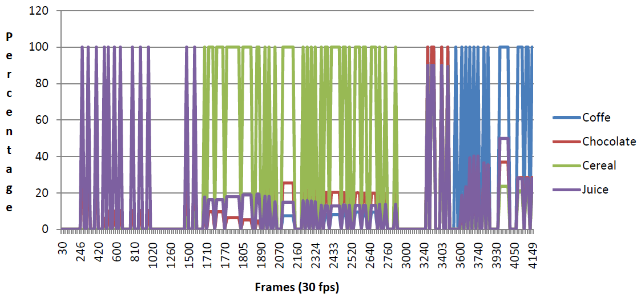

Fig. 7. Instantaneous activity recognized: JUICE - CEREAL - COFFEE.

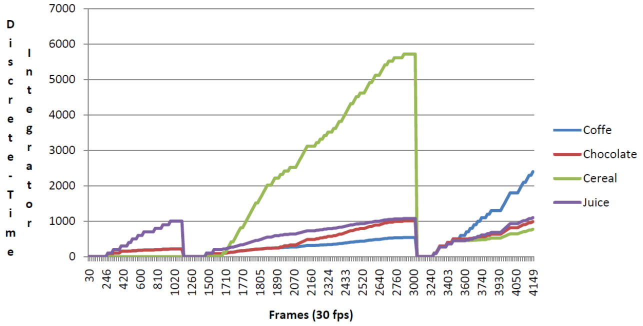

Fig. 8. Final activity recognized: JUICE - CEREAL - COFFEE.

However for a natural robotic interaction neither the beginning nor the end of performed activities must be announced. Therefore, as we wanted to test our method to continuously detect the ongoing activity without the need for previous segmentation of image sequences, a second battery of image sequences were used. These videos contained a continuous sequence of three activities, but also they included the action on objects that do not intervene in the ongoing activity or actions that doesnt belong to any particular activity with the aim to evaluate the robustness of the system. Fig. 8 shows a sample frame of our video process.

The Fig. 6 illustrates the evolution of activity recognition in a continuous way, for a sequence of preparation activities of serving a juice, a bowl of cereals and a cup of coffee. It present previous activities, posterior activities and includes objects that do not participate in the activity. We have to emphasize that in all the tests the recognition was fulfilled with occlusions, to allow completely natural movements by the user.

Fig. 7 shows the same activities recognized by accumulative method. It is similar to a race where activities compete for the

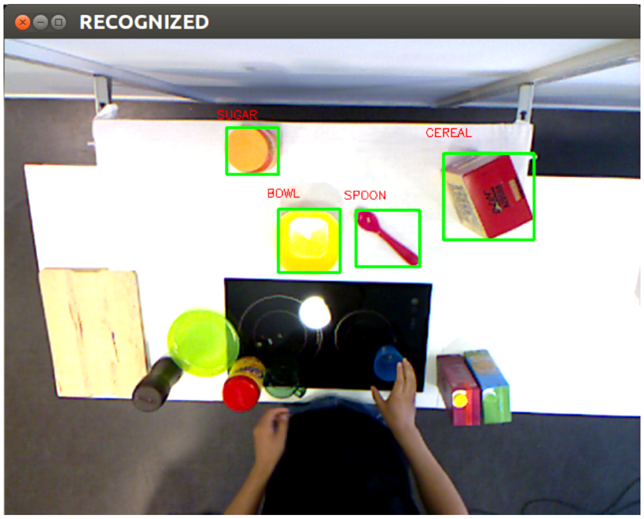

Fig. 9. Sample frame of a processed video sequence.

prize and receive votes (probabilities) from the instantaneous activity detector. The first case corresponds to Juice-activity with correct response. The second case is Cereal-activity with many common objects in the initial frames inducing confusion in the recognition until the activity progresses. The last activity is coffee preparation with a satisfactory performance from the beginning.

V. Conclusion

This paper presents an approach aimed to make possible to recognize human activity only based on the interaction with objects which recognition is performed by means of computer vision techniques that are not intrusive to the user. In addition we achieve almost real time execution with an average time of 0.25 seconds approximately for the whole process in a standard PC.

We have presented a definition for action on objects based on what happen with the objects under the assumption that they are only moved by the user. In this case human object interaction is described by four options of object- manipulation by the user (ADD, REMOVE, MOVE and UNCHANGED).

For the recognition of the activity we have developed a simple structure inspired by a recipe. Hence, we have grouped objects in three classes: ingredients, utensils and possible substitutes. An activity is then defined by the presence of its pre-defined objects lists, demonstrating that it is applicable to the activity recognition process.

Our activity recognition system has been designed to work in a continuous way, without activity segmentation from the test video sequences. In order to evaluate the robustness of the system, these videos include activities previous and posterior to the activities selected, besides other objects that do not directly intervene. It is also emphasized that in all the tests the recognition were fulfilled with occlusions, to allow completely natural movements from the user.

Our proposed method is capable to overcome the common problems in computer vision, brightness and occlusion. The algorithm generally presents a trustworthy behaviour though these are present in some samples. Nevertheless, other activity recognition techniques might complete our project in order to offer higher confidence in the results, such as user movement recognition.

In addition, we have not intentionally established predetermined movements to recognize the activities. With this approach we can obtain a totally flexible and scalable system just by adding extra definitions in base of our structure for recognizing new activities.

A future interesting work would be to develop a statistical study to determine the relevancy of ingredients, utensils and substitutes for the different activities (constants a, b and c of our method). The result would be useful to automatically tune the algorithm.

References

[1] J. K. Aggarwal and M. S. Ryoo, "Human activity analysis: A review," *ACM Computing Surveys (CSUR)*, vol. 43, no. 3, p. 16, 2011.

[2] S. Hongeng, R. Nevatia, and F. Bremond, "Video-based event recognition: activity representation and probabilistic recognition methods," *Computer Vision and Image Understanding*, vol. 96, no. 2, pp. 129–162, 2004.

[3] D. Moore and I. Essa, "Recognizing multitasked activities from video using stochastic context-free grammar," in *AAAI/IAAI*, pp. 770–776, 2002.

[4] L. Bo and C. Sminchisescu, "Efficient match kernel between sets of features for visual recognition," in *Advances in neural information processing systems*, pp. 135–143, 2009.

[5] M. S. Ryoo, "Human activity prediction: Early recognition of ongoing activities from streaming videos," in *2011 International Conference on Computer Vision*, pp. 1036–1043, IEEE, 2011.

[6] J. Lei, X. Ren, and D. Fox, "Fine-grained kitchen activity recognition using rgb-d," in *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pp. 208–211, ACM, 2012.

[7] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, vol. 2, IEEE, 1999.

[8] D. Dennis, C. Tin, and R. Marou, "Color image segmentation." https://theiszm.wordpress.com/tag/color-segmentation/. Accessed: 2014-02-01.

[9] A. Mordvintsev and K. Abid, "Opencv-python tutorials documentation," 2014.

[10] M. Vinagre, J. Aranda, and A. Casals, "An interactive robotic system for human assistance in domestic environments," in *International Conference on Computers for Handicapped Persons*, pp. 152–155, Springer, 2014.

[11] E. R., "Basic evaluation measures for classifier performance." http://webdocs.cs.ualberta.ca/~eisner/measures.html. Accessed: 2014-06-01.