WCCI 2012 IEEE World Congress on Computational Intelligence
June, 10-15, 2012 - Brisbane, Australia IJCNN

Developmental Approach for Interactive Object Discovery

Natalia Lyubova, David Filliat
ENSTA ParisTech - INRIA FLOWERS
75739 Paris Cedex 15, France,
Email: natalia.lyubova@ensta-paristech.fr

***Abstract*—We present a visual system for a humanoid robot that supports an efficient online learning and recognition of various elements of the environment. Taking inspiration from child's perception and following the principles of developmental robotics, our algorithm does not require image databases, predefined objects nor face/skin detectors. The robot explores the visual space from interactions with people and its own experiments. The object detection is based on the hypothesis of coherent motion and appearance during manipulations. A hierarchical object representation is constructed from SURF points and color of superpixels that are grouped in local geometric structures and form the basis of a multiple-view object model. The learning algorithm accumulates the statistics of feature occurrences and identifies objects using a maximum likelihood approach and temporal coherency. The proposed visual system is implemented on the iCub robot and shows 85% average recognition rate for 10 objects after 30 minutes of interaction.**

I. Introduction

For robots to be useful at home, to help people in their activities and domestic services, they will require among many things a very strong capability to detect and recognize objects. As it is difficult to imagine that a robot will know in advance all possible objects, it should be able to learn new objects at any time. Ideally, it should obtain all information needed for that without any complex training stage for the user, but simply from interactions with people or with the object itself.

Most of existing computer vision approaches for object detection and recognition are based on initial image databases, various form of prior knowledge or narrow-purpose detectors such as skin/face detectors. These approaches usually do not allow robots to be autonomous in an open-ended scenario. In contrast, our goal is to develop a system based on incremental online learning following the principles of developmental robotics, where robots start with a few core capabilities and acquire new skills of increasing difficulty [1]. In our application scenario, people show different objects and interact with them in front of a robot in a similar way they would do in front of a child to teach him object. In this situation, our system should be able to learn and recognize any kind of objects, as well as adapt to the environment, background, lighting conditions and camera motion.

The idea of a humanoid robot acquiring knowledge from gradual exploration and interaction with the environment is inspired by the way children learn objects. Indeed, most of perceptual challenges that exist in computer vision are solved in the vision of infants during the first year of their life [2]. Starting with little or no knowledge about the surrounding world, children retrieve all information from light intensities and sounds. They learn from the iterative accumulation of data, associations and results of actions. The self-experimentation starts from the exploration of own body, like an association of the arms with their visual appearance and motion behavior. This process is called 'body babbling' and it is aimed to tune up the predictability of own movements [3]. When a certain progress is reached, the self-learning stage is overlaid by the exploration of the external world, surrounding objects and people [4]. The interaction with objects is important as it enhances its exploration, provides additional physical properties and allows to learn its overall appearance from different perspectives. During object manipulation, infants discover and adjust so-called 'affordances' - actions available in the environment and dependent on individual capabilities [3].

In this paper, we describe a perception system which is to be included in a general approach reproducing the aforementioned developmental steps. This system is able to incrementally detect and learn objects, without supervision. In future work, these objects are going to be classified into own robot's body, human and manipulable objects, using people's feedback and experiments. We therefore designed our system by analogy with the human perceptual learning in terms of input data and general organization, but without trying to reproduce precisely brain inner functioning. Following general principles used by human, we segment the visual space into proto-objects based on visual attention, and learn the corresponding objects, characterizing them by hierarchical visual representation constructed from a set of complementary low-level features, their relative location and motion behavior. The proposed approach is implemented on an iCub robot.

In section 2, we present a short review of related object detection and recognition algorithms. The proposed approach is detailed in section 3. Experimental results are reported in section 4, and last section is devoted to discussion and conclusion.

II. Related work

The computer vision community provides a large amount of object detection and recognition approaches which are mostly based on prior knowledge. This knowledge is often represented as labeled samples, which corresponds to pairs of a sensory input with an interpretation [2]. In the image processing field,

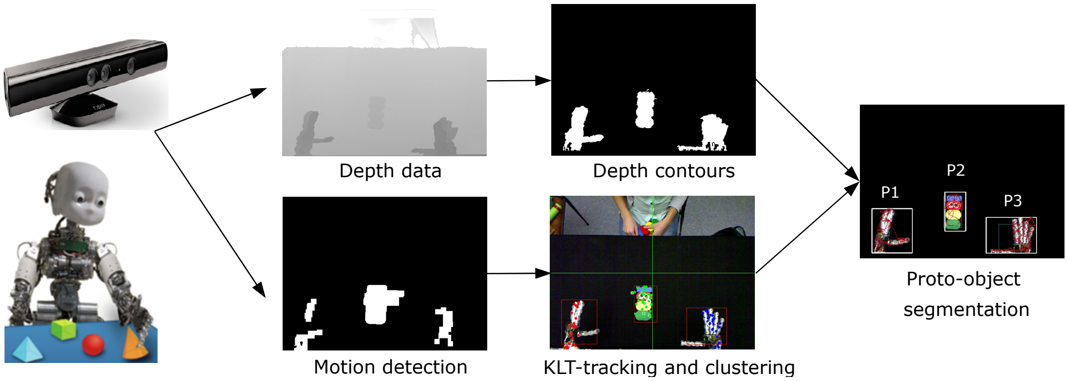

Fig. 1. Proto-object segmentation and intermediate results of the image processing.

prior knowledge takes the form of algorithm choices or image databases, where each object is associated with several images or visual properties encoded by descriptors. In this case, object recognition relies on the similarity with existing database entities; it is fast and reliable, but it is not easily applicable for autonomous learning, where robots should construct object representations adaptable to an environment. This can be achieved by using dedicated interfaces, like the one proposed in [5] that allows a user to provide learning examples to the robot.

Fast and robust real-time object localization is also provided by algorithms detecting artificial markers. As an instance, the ARTag system that is widely used in virtual reality applications, creates and recognizes such markers [6]. However, this requires object tagging. An efficient identification of specific object categories is also possible by numerous narrow-purpose detectors. Existing examples are the Face detector [7] that works with the low-level Haar-like features, and the Human skin detector [8] that processes the color of pixels. The last one is also used to enhance the image segmentation, by subtracting the regions of human hands holding objects [9].

The online object detection algorithm implemented in [10] is close to our goal, since it learns objects by means of robot's actions and object manipulations. Regions of the visual space that have 'objecthood' characteristics are considered as proto-objects [11] and segmented similar to a perceptual grouping of visual information in human cortical neurons. Localization of proto-objects is based on a saliency map. Salient regions often contain most of the information about the visual space [12]. These regions differs from their neighborhood in some of physical properties, like color, intensity, texture, an effect of the spatial orientation or shape. In addition, the ability to stand out from the neighborhood proceeds from dynamic object properties, such as motion trajectory, speed, changes in size or appearance [13].

There are several methods to represent objects. In order to process image faster and more efficiently than operating on pixels intensities, local descriptors are often used to characterize stable image patches at salient positions. Thus, the visual content is encoded to a significantly smaller amount of data. A good descriptor should be balanced between robustness, sparseness and speed. Among the variety of existed descriptors, some of them are computed around key-points, like scale-invariant SIFT [14] and SURF [15], on the extracted boundaries, like straight edge segment EDGE [16], on junctions like scale-invariant SFOP [17], on corners like HARAF [18] and on Good features to track [19]. In order to achieve an efficient object learning and recognition, the completeness (an ability to preserve information) of various feature combinations [20] can be considered, and complementary features, which allow to process various image types, can be chosen.

III. Proposed method

Our algorithm is based on online incremental learning, and it does not require image databases, artificial markers nor face/skin detectors. We acquire all knowledge iteratively during interactions with objects, by analyzing low-level image features. In our setup, we use a robot placed in front of the table, and we take the visual input from a Kinect camera instead of stereo vision, since it is an easy way to obtain the depth information. Our main processing steps include detection of proto-objects as units of attention, characterization of their visual appearance and learning them as perspectives of real objects, which we call views.

A. Proto-object segmentation

Our proto-object detection algorithm is based on visual attention; the main steps and the intermediate results of the image processing are shown in the Fig. 1. According to our scenario, people in front of the robot interact with objects to produce observed motion, which encourages the robot to focus on it. We assume, that the robot is attracted by motion; therefore we compute a saliency map based on the optical flow.

The motion is detected by computing the running average of consecutive images, subtracting it from the current image and thresholding it to a binary mask. Random noise is removed by erosion and dilation operators.

The visual scene is rarely stable, since the optical flow includes both the real motion of external factors in that we are interested in, and the global flow caused by the camera motion. Furthermore, motion artifacts, like blur, accompanied by changes in pixel intensities and color values, make difficult to match key-points, determine the correspondence between consecutive images and identify contours. Thus, we omit the global optical flow and concentrate on the meaningful information from stable images, by analyzing the camera motion from commands sent to robot's head, neck or torso. Additionally, the depth data obtained from a Kinect camera by the RGB Demo software[1], is used to filter the visual input. We discard the regions, which are relatively far from the robot, and we keep pixels within the distance suitable for object location, considering the constraints of the robot's working area.

In the filtered moving areas of the visual space we search for proto-objects and isolate them based on the motion behavior of key-points. We extract the GFT (Good Features to Track) key-points inside moving regions and track them by the KLT algorithm [21], which computes the displacement of the key-points between consecutive images. The tracked key-points are grouped by agglomerative clustering. Initially, each key-point composes its own cluster. Then, we iteratively merge two clusters with a smallest difference in the speed and distance; the process continues until the certain threshold is reached.

Each obtained group of coherent key-points gives an idea about possible objects, but its envelope rarely corresponds to real object contours; it can capture the background or surrounding items. In order to obtain more precise proto-object boundaries, we refine contours extracted from the original RGB image by contours obtained from the depth information. In both cases, the Sobel operator based on the first derivative, is used to detect horizontal and vertical edges. Then, we apply a threshold and define the proto-object borders by continuous contours. These isolated regions are learned and recognized in the following processing.

B. View encoding

The Bag of Visual Words (BoW) approach with incremental dictionaries [22] forms the basis of our object representation model. In general, BoW [23] is used to encode images as a set of visual words represented by quantized descriptors characterizing image patches.

The robot should be able to deal with various kinds of objects, ranging from simple objects with few features, to complex and highly textured objects. For this purpose, a single descriptor is not enough. Thus, we combine complementary features, providing different visual characteristics and maximizing the amount of encoded information. SURF detector is chosen as a good solution for objects with a large amount of details. The key-points neighborhoods are encoded by 64D-vector, but they are isolated and sparse. SURF alone does not perform well for homogeneously colored areas. So, we developed an additional descriptor operating on the level of regularly segmented image regions. The similar adjacent pixels are grouped by a superpixels algorithm [24], that performs the watershed segmentation on LoG (Laplacian of Gaussian) with local extrema as seeds. Then, each superpixel is described by the HSV model (hue, saturation and value) [25]. We chose this color space, because its dimensions are conceptualized in terms of perceptual attributes and they don't change together with light intensity (as happens with RGB values). Our combined descriptor is therefore robust to object texture level, illumination and scale variations.

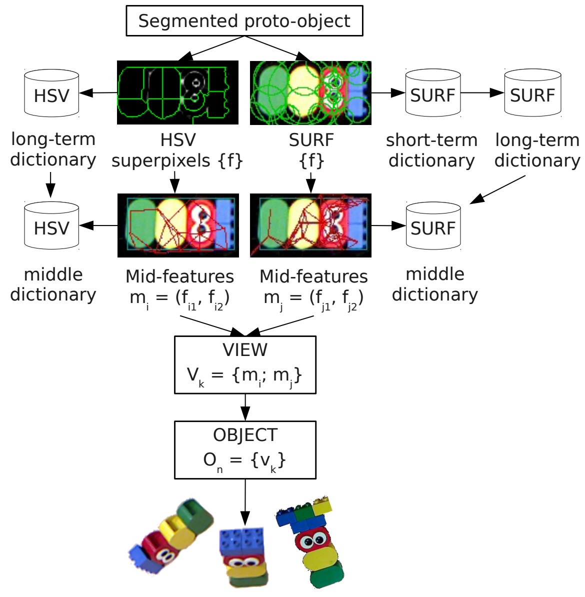

Fig. 2. View encoding and construction of a hierarchical object model.

All extracted descriptors are quantized to visual words and stored in vocabularies, if their dissimilarity with existing items exceeds a given threshold [22]. To avoid rapid and continuous growth of SURF dictionary, we implemented a short- and long-term memory. The short-term stack is filtered according to feature co-occurrence over consecutive frames. The relevant visual words f_i build the long-term vocabulary, that is used in the following processing as a ground level of the hierarchical object representation, as shown in the Fig. 2.

Initial BoW approach does not take into account any spatial relation between visual words inside images. This limitation is resolved in several BoW variations, like the Constellation model [26] that considers the geometrical relationship between image patches, but requires a long computation time. Our goal is an efficient object representation that takes advantage of multiple features, their relative location and coherent motion behavior. We group the closest SURF points and superpixels into mid-features $m_i = (f_{i1}, f_{i2})$, relying on their distance in the visual space. Each feature forms a pair with its 5 closest neighbors. This middle level incorporates local object geometry in terms of relative position of colors and key-points. The mid-features are quantized to visual words, stored in vocabularies and used to encode the appearance of views $V_j = \{m_i\}$.

C. View recognition

The object learning and recognition algorithm is based on a voting method using the TF-IDF (Term-Frequency - Inverse-

[1] Software Kinect RGB Demo v0.6.1 available at http://nicolas.burrus.name

Document Frequency) and maximum likelihood approach. TF-IDF was initially used in text retrieval, where each document was described by a vector of words frequencies [23]. We adapted this theory to describe each view by a vector of mid-features frequencies, as shown in the Fig. 3. The statistical measure evaluates the importance of mid-features with respect to segmented views. The recognition decision depends on a product of the mid-feature frequency and the inverse view frequency:

$$tf-idf(m_i, v_j) = tf(m_i) * idf(m_i), \tag{1}$$

where $tf(m_i)$ is a frequency of the mid-feature m_i, and $idf(m_i)$ is an inverse view frequency for the same mid-feature m_i.

The mid-feature frequency accumulates the occurrence of a mid-feature in a view, and it is calculated as:

$$tf(m_i) = \frac{n_{m_i v_j}}{n_{v_j}}, \tag{2}$$

where $n_{m_i v_j}$ is the number of occurrences of the mid-feature m_i in the view v_j, and n_{v_j} is the total number of mid-features in the view v_j.

The inverse view frequency is related to the presence of a mid-feature in the past. It is used to decrease the weight of mid-features, which present often in different views, and it is equal to:

$$idf(m_i) = log\frac{N}{n_{m_i}}, \tag{3}$$

where n_{m_i} is the number of views in which the mid-feature m_i has been found, and N is the total number of seen views.

During the recognition, we compute the likelihood of the current set of mid-features being one of already learned views, using a voting method. In case of high recognition likelihood, we update the identified view by a set of found mid-features; otherwise, we store this view with a new label. While presenting different objects to the robot, the weights of relevant features for each object grow proportionally to the number of occurrences.

D. Object model

At this level, the learned views describe objects from a single perspective. In order to construct a multi-view model, we accumulate the object appearance from different viewing points $O_n = \{v_j\}$. During manipulations, we track an object using KLT algorithm and associate each detected view v_j with the tracked object label, like in the Fig. 4. If tracking fails, we compute the likelihood of the current view being one of already known objects, using the occurrence frequency of the view among learned objects. In case of low recognition likelihood, we create a new object label. Besides that, the tracking process is also used to facilitate object recognition, as we identify tracked objects without analyzing their visual appearance.

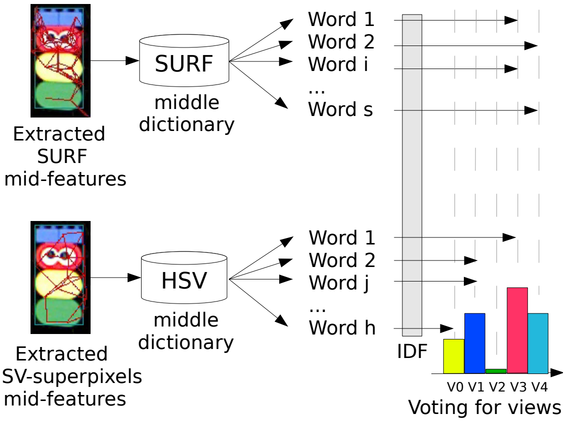

Fig. 3. View recognition through the voting method.

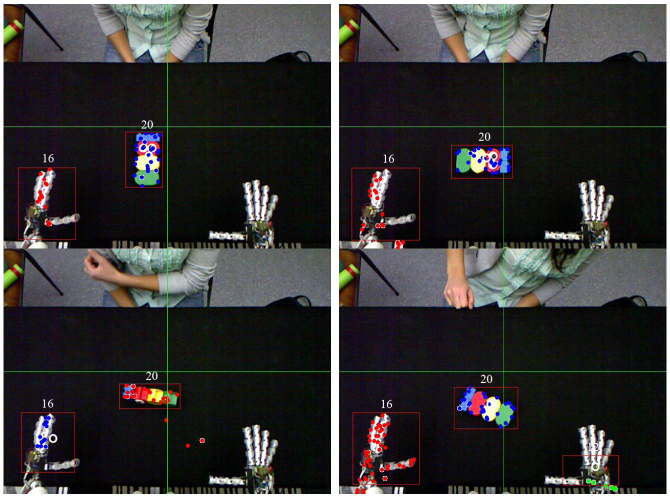

Fig. 4. Object model: object views from different perspectives are recognized as the same object.

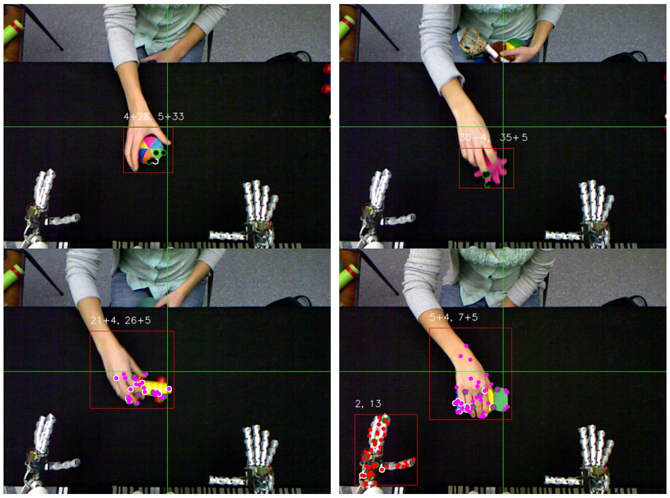

Fig. 5. Connected objects: an object occluded by a human hand on 10, 25, 50, 75% (from left to right, top to bottom) is detected in a single moving blob with a human hand, but recognized as a separate object.

E. Connected objects

Object manipulations introduce additional difficulties in the image processing. When the human or the robot's hand holds an object, they compose a single moving blob (like in the Fig. 5), and multiple occlusions divide an object into parts. This problem requires an object segregation, as it is called in psychology; and this ability is trained up by infants in 5 months of a life [2]. The real borders between objects are distinguished from the similarity of properties and from the 'key events', which imply previously seen images with clear object boundaries. Following this idea of the prior experience, we learn human and robot's hands before presenting objects; then we identify the presence of connected objects into a single moving blob, using a two-stage object recognition demonstrated in the Fig. 6. During the first stage, we identify the most probable object based on the similarity with already known objects. For example, if a human hand has ever appeared before, it will be recognized. Then, we eliminate the features that belong to the identified item from the processing area; and the remaining features are used in the second identical recognition stage to check, if there is enough evidence of a presence of another object inside the same moving blob.

IV. Experimental results

Our perception system is implemented on the iCub robot. The experimental setup was organized as follows: the iCub was localized in front of the table, the Kinect was mounted over the robot's head at a distance of 75 cm from the table, that allows to perceive objects on the table, robot's hands and people in front of the robot.

The vision system was examined on an image sequence prerecorded during 30 minutes. Ten objects, shown in the Fig. 7, were presented to the robot; each object was manipulated by a person for several minutes. An object was considered as detected, when it was segmented as a proto-object. The detection rate was about 98.8%. The detection algorithm includes both stages: motion-based detection and segmentation from depth contours. During the motion-based detection stage, 15% of proto-objects were falsely groupped from KLT-tracked points (that corresponds to the clustering error) and re-segmented correctly by processing depth information.

The object recognition was evaluated a posteriori, by hand-labeling the image sequence. For each object, we defined one major label as the most frequently assigned by the robot, several pure labels, which were never given to other objects, and noisy labels, which were associated with several objects. An object was considered as recognized, when it was assigned to a pure label. The recognition rate, presented in the Table I, was calculated as a ratio of the number of images, where object was recognized, to the total number of images with this object.

The learning process stabilized within several minutes, and labels associated with detected objects remained nearly constant. The recognition rate reached 72% - 93%, depending on objects. An example of the multi-view object recognition is given in the Fig. 4. In 65% of cases, objects were recognized from tracking, in the rest from their appearence. The system distinguished well between two identical octopuses that differ only by color, but some confusions occured between lego-toy and lego-cars that have similar colors and SURF-points (at the place of block attachements).

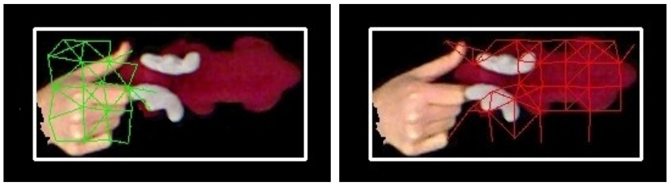

Fig. 6. Two-stage object recognition: mid-features shown by the green color correspond to the human hand recognized on the first stage (left image) and the rest of features shown by the red color correspond to the object recognized on the second stage (right image).

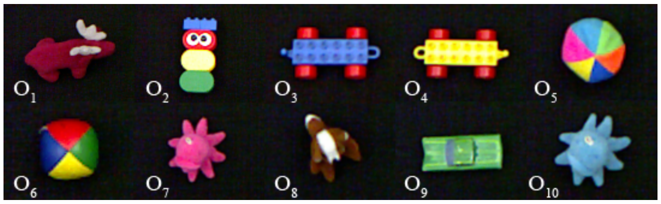

Fig. 7. Experimental objects $O_1 - O_{10}$ from top-left to bottom-right.

TABLE I
Recognition rate

Object	Presented in # images	Recognized by a pure label	Recognized by a major label	Amount of pure labels	Recognition rate, %
O_1	109	99	98	2	91
O_2	139	114	87	2	82
O_3	73	68	68	1	93
O_4	86	71	71	1	83
O_5	54	43	33	3	80
O_6	60	46	39	2	77
O_7	106	99	99	1	93
O_8	36	26	18	2	72
O_9	64	51	32	3	80
O_{10}	53	49	34	2	92
iCub left hand	357	315	267	4	88
iCub right hand	357	325	313	2	91
human hand	61	43	28	5	70

Our system is invariant to scale and rotations, since it uses SURF and color as low-level features. Various manipulations with objects confirmed the resistance of our system to motion artifacts and to the visual clutter. Only rapid actions caused some difficulties, which were resolved by manipulating an object for a longer time. The system was able to track, segment and recognize all 10 objects simultaneously on the same image, shown the Fig. 8. We analyzed the recognition

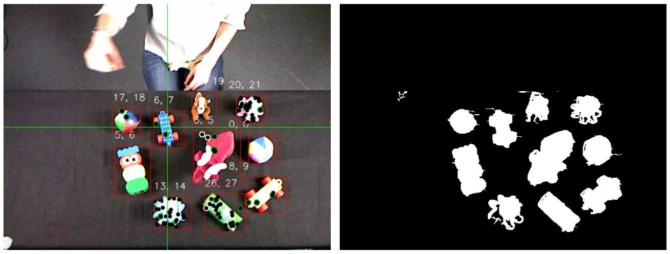

Fig. 8. Simultaneous traking: 10 objects on the same image are successfully segmented (right image) and recognized (left image).

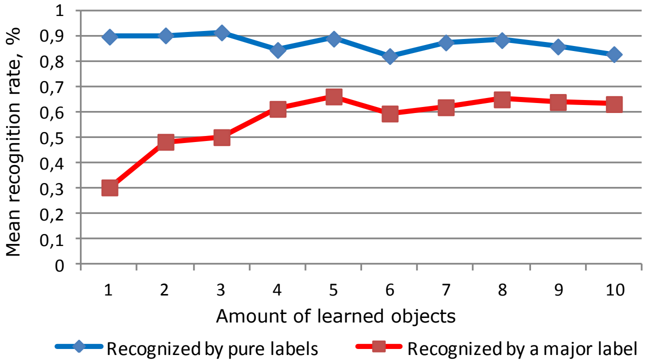

Fig. 9. Mean recognition rate of the system, learning different number of objects.

TABLE II

THE IMPACT OF OCCLUSION ON THE RECOGNITION RATE OF THE SYSTEM

Amount of occlusion, %	Detection rate,% connected objects [+ single objects]	Recognition rate,% connected objects [+ single objects]	Total detection/ recognition rate,%
10	54 [+25]	23 [+2]	79/25
25	63 [+7]	32	70/32
50	47 [+6]	18	53/18
75	38	0	38/0
90	33	0	33/0

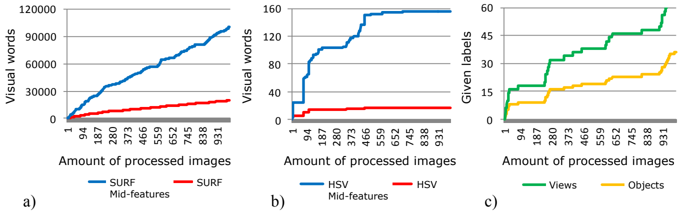

Fig. 10. The growth of the vocabularies: a) SURF and SURF mid−features, b) HSV and HSV mid−features, c) views and objects.

rate of the system with different amount of objects presented during experiments. As we see from the Fig. 9, the recognition rate based on pure labels slightly decreases with a number of objects; the recognition rate based on one major label grows significantly up to 4 objects and then remains nearly stable.

The average processing time was about 0.2 sec per image in case of one object. The use of several objects tends to increase computation time. In the case of 10 objects (Fig. 8) processing time reached 1.5 sec. The recognition of connected objects took 0.36 sec in average. Objects occluded up to 75% were detected in a single moving blob with a human hand and recognized as connected objects due to two-stage object recognition explained in the section III.E and illustrated in the Fig. 5. The increasing amount of occlusion decreases the recognition rate, as detailed in the Table II.

The number of mid-features, labeled views and objects are displayed in the Fig. 10. Once the vocabularies have reached a sufficient amount of knowledge, the robot was able to reliably recognize objects, human and own hands.

V. DISCUSSIONS AND FUTURE WORK

Concluding from our work, the proposed system enables a robot to autonomously explore the visual space in an open-ended scenario, detect and learn objects during interactions with humans.

The solution proposed in this paper is based on online incremental learning. The presented algorithm is able to learn objects while not requiring image databases, nor skin/face detectors. We acquire all knowledge by analyzing the visual space and constructing hierarchical object representations.

In future work, this system will be used as the basis for object categorization through robot experiments and affordance learning. We are going to improve the processing time by optimizing dictionaries. Then we are planning to use the mutual information between the data from robot's motors and the behavior of moving regions in the visual space, by analogy with [27]. In case of high correlation between motors and observed motion, the moving region will be identified as own robot's body, otherwise, as a human. The kinematic model can be used to improve categorization by predicting the position of robot's parts.

ACKNOWLEDGMENT

Our work is performed as a part of MACSi (Modele pour l'apprentissage du comportement sensorimoteur d'iCub) project, and we would like to thank ANR for the funding and all partners for the support.

REFERENCES

[1] J. Weng, J. McClelland, A. Pentland, O. Sporns, I. Stockman, M. Sur, and E. Thelen, "Artificial intelligence: Autonomous mental development by robots and animals," *Science*, vol. 291, pp. 599–600, 2001.

[2] P. Fitzpatrick, A. Needham, L. Natale, and G. Metta, "Shared challenges in object perception for robots and infants," *Infant and Child Development*, vol. 17, no. 1, pp. 7–24, 2008.

[3] F. Kaplan and P.-Y. Oudeyer, "The progress-drive hypothesis: an interpretation of early imitation," in *Models and mechanisms of imitation and social learning: Behavioural, social and communication dimensions*. Cambridge University Press, 2006.

[4] J. Piaget, *Play, dreams and imitation in childhood*. London: Routledge, 1999.

[5] R. Rouanet, P.-Y. Oudeyer, and D. Filliat, "An integrated system for teaching new visually grounded words to a robot for non-expert users using a mobile device," in *IEEE-RAS Int. Conf. on Humanoid Robots*, Tsukuba, Japon, 2009.

[6] M. Fiala, “Artag, a fiducial marker system using digital techniques,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2005, pp. 590–596.

[7] P. Viola and M. J. Jones, “Robust real-time face detection,” *Int. J. Comput. Vision*, vol. 57, pp. 137–154, 2004.

[8] J. Fritsch, S. Lang, M. Kleinehagenbrock, G. A. Fink, and G. Sagerer, “Improving adaptive skin color segmentation by incorporating results from face detection,” in *IEEE Int. Workshop on Robot and Human Interactive Communication*, 2002, pp. 337–343.

[9] D. Beale, P. Iravani, and P. Hall, “Probabilistic models for robot-based object segmentation,” *Robotics and Autonomous Systems*, vol. 59, pp. 1080–1089, 2011.

[10] L. Natale, F. Orabona, F. Berton, G. Metta, and G. Sandini, “From sensorimotor development to object perception,” in *IEEE-RAS Int. Conf. on Humanoid Robots*, 2005, pp. 226–231.

[11] Z. W. Pylyshyn, “Visual indexes, preconceptual objects, and situated vision,” *Cognition*, vol. 80, pp. 127–158, 2001.

[12] D. Walther and C. Koch, “Modeling attention to salient proto-objects,” *Neural Networks*, vol. 19, no. 9, pp. 1395–407, 2006.

[13] H. Wersing, S. Kirstein, M. Götting, H. Brandl, M. Dunn, I. Mikhailova, C. Goerick, J. J. Steil, H. Ritter, and E. Körner, “Online learning of objects in a biologically motivated visual architecture,” *Int. J. Neural Systems*, vol. 17, no. 4, pp. 219–230, 2007.

[14] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int. J. Comput. Vision*, vol. 60, pp. 91–110, 2004.

[15] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, “Speeded-up robust features (surf),” *Comput. Vis. Image Underst.*, vol. 110, pp. 346–359, 2008.

[16] W. Förstner, “A framework for low level feature extraction,” in *European Conf. on Computer Vision (ECCV).* London, UK: Springer-Verlag, 1994, pp. 383–394.

[17] T. Dickscheid, F. Schindler, and W. Förstner, “Detecting interpretable and accurate scale-invariant keypoints,” in *IEEE Int. Conf. on Computer Vision*, 2009, pp. 2256–2263.

[18] K. Mikolajczyk and C. Schmid, “Scale & affine invariant interest point detectors,” *Int. J. Comput. Vision*, vol. 60, pp. 63–86, 2004.

[19] J. Shi and C. Tomasi, “Good features to track,” in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 1994, pp. 593 – 600.

[20] T. Dickscheid, F. Schindler, and W. Förstner, “Coding images with local features,” *Int. J. Comput. Vision*, vol. 94, pp. 154–174, 2011.

[21] C. Tomasi and T. Kanade, “Detection and tracking of point features,” Carnegie Mellon University, Tech. Rep., 1991.

[22] D. Filliat, “A visual bag of words method for interactive qualitative localization and mapping,” in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2007, pp. 3921–3926.

[23] J. Sivic and A. Zisserman, “Video google: Text retrieval approach to object matching in videos,” in *Int. Conf. on Computer Vision*, vol. 2, 2003, pp. 1470–1477.

[24] B. Micusik and J. Kosecka, “Semantic segmentation of street scenes by superpixel co-occurrence and 3d geometry,” in *IEEE Int. Conf. on Computer Visio*, 2009, pp. 625–632.

[25] A. R. Smith, “Color gamut transform pairs,” *SIGGRAPH Comput. Graph.*, vol. 12, pp. 12–19, 1978.

[26] R. Fergus, P. Perona, A. Zisserman, and O. P. U. K, “Object class recognition by unsupervised scale-invariant learning,” in *Conf. on Computer Vision and Pattern Recognition*, 2003, pp. 264–271.

[27] C. Kemp and A. Edsinger, “What can i control?: The development of visual categories for a robots body and the world that it influences,” in *IEEE Int. Conf. on Development and Learning (ICDL), Special Session on Autonomous Mental Development*, 2006.