Teaching iCub to recognize objects using deep Convolutional Neural Networks

G. Pasquale[1,2,3] **C. Ciliberto**[2,4] **F. Odone**[3] **L. Rosasco**[2,3,4] **L. Natale**[1]

[1] iCub Facility,
Istituto Italiano
di Tecnologia

[2] Laboratory for
Computational and
Statistical Learning,
Istituto Italiano
di Tecnologia

[3] Dipartimento di
Informatica, Bioingegneria,
Robotica e
Ingegneria dei Sistemi
Universitá degli Studi di Genova

[4] Poggio Lab,
Massachusetts
Institute of Technology

Abstract

Providing robots with accurate and robust visual recognition capabilities in the real-world today is a challenge which prevents the use of autonomous agents for concrete applications. Indeed, the majority of tasks, as manipulation and interaction with other agents, critically depends on the ability to visually recognize the entities involved in a scene. At the same time, computer vision systems based on deep Convolutional Neural Networks (CNNs) are marking a breakthrough in fields as large-scale image classification and retrieval. In this work we investigate how latest results on deep learning can advance the visual recognition capabilities of a robotic platform (the iCub humanoid robot) in a real-world scenario. We benchmark the performance of the resulting system on a new dataset of images depicting 28 objects, named iCubWorld28, that we plan on releasing. As in the spirit of the iCubWorld dataset series, this has been collected in a framework reflecting the typical iCub's daily visual experience. Moreover, in this release we provide four different acquisition sessions, to test incremental learning capabilities over multiple days. Our study addresses the question: how many objects can the iCub recognize today?

1 INTRODUCTION

Nowadays, the major bottle neck preventing the use of robots for real-world tasks, from simple manipulations up to navigation or interaction with humans, is the lack of good visual recognition systems. Concurrently, computer vision systems have witnessed tremendous progress, especially in the context of object recognition. It has been largely demonstrated that multi-layer CNNs can learn powerful hierarchical visual representations able to cope with large-scale problems. An important reason for the rapid evolution of this field was the availability of public datasets on which to train and benchmark the performance of new solutions (e.g., ImageNet LSVRC [1]). These datasets are essentially tailored to image retrieval problems and indeed this is the kind of task on which the performance of many vision systems have been ultimately tested. It is then worth asking whether these new developments can impact robotics systems, where the real-world setting differs from the typical retrieval scenario. Indeed the nature of the learning problem is affected by the amount and type of visual data (usually few video frames rather than millions of independent images), the availability of valuable contextual information and the time constraints, especially during the training phase.

The iCub humanoid [2] (Fig. 1) offers an ideal platform to conduct the above inquiry. In particular, in this work we investigate to which extent we can exploit latest deep CNNs to provide the robotic system with a visual recognition system enabling it to gather and learn, in a human-like fashion, the visual information necessary to interact with the nearby surrounding environment. We consider this problem within the Human-Robot Interaction (HRI) scenario introduced in [3]: a human teacher shows a new object to the robot, verbally annotating it; the robot focuses on the unknown object and learns it, becoming able to recognise it among others.

Appearing in Proceedings of the 4^{th} Workshop on Machine Learning for Interactive Systems (MLIS) 2015, Lille, France. JMLR: W&CP volume 43. Copyright 2015 by the authors.

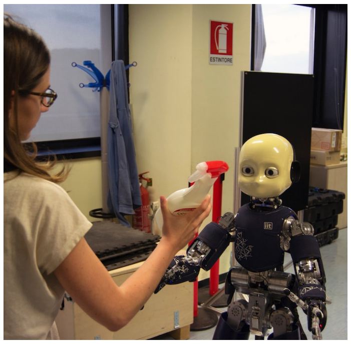

Figure 1: Snap-shot of the setup used in this work.

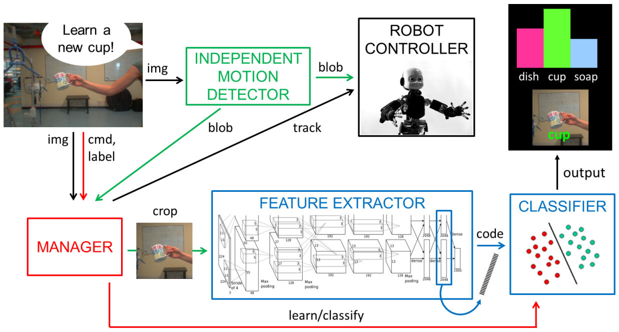

Figure 2: The visual recognition system currently used on the iCub. The application is implemented in the YARP [15] middleware, so that each module runs independently, communicating data with the others.

Within this scenario, we aim to determine how far we are from achieving human-like object identification capabilities, by asking the question: *"How many objects can the iCub actually recognize?"*. Indeed, in realistic robotic applications we need reliable perceptual systems that, at least for limited sets of objects, are virtually infallible. In order to investigate such question in a reproducible way, we perform our study on a dataset collected in the same setting: a human taught 28 different objects to the iCub for 4 days, leading to the acquisition of a new release of the iCubWorld[1] dataset, named iCubWorld28, that we will make available to the community as the previous ones. We propose a possible approach to quantify the confidence within which the conclusions drawn on the present benchmark are expected to hold in real-world scenarios.

2 METHODS

In the following we detail the visual recognition pipeline adopted in the robotic system (Sec. 2.1) and the acquisition setup of iCubWorld28 (Sec. 2.2).

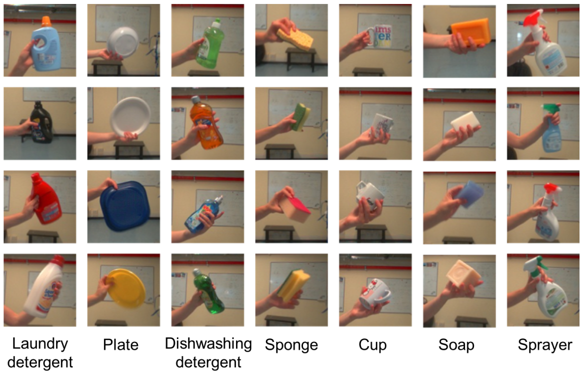

Figure 3: Example images from one of the 4 datasets comprising iCubWorld28.

2.1 Recognition Pipeline

Modern computer vision algorithms for image classification are composed of multiple layers alternating convolution, pooling and non-linear mappings, that are trained to learn a suitable representation for the visual data. Once the representation map has been learned, each image is encoded into a vector in the new space. If the representation is "good", namely, *invariant* to transformations that do not affect the actual object class, while being *discriminative* with respect to other classes, a linear predictor (such as an SVM [4] or RLS [5]) can perform the required classification task.

Visual representation Lately, the availability of powerful GPUs has allowed to train deep CNNs on very large datasets. In particular, models trained on the ImageNet dataset [1] exhibited very good generalisation properties and proved to be effective as off-the-shelf representation extractors on a variety of other datasets and recognition tasks [6, 7, 8, 9]. This approach is particularly appealing and is the one evaluated in this paper. Indeed, training such complex architectures from scratch requires very large numbers of examples and high computational effort. These are non-trivial problems that make training deep CNNs impractical in robotics settings. Therefore we employed one of the CNN models provided in the Caffe library [10], *BVLC Reference CaffeNet*, based on the network architecture proposed in [11] and originally trained on the ImageNet dataset [1]. Following a strategy suggested in recent works [6, 9, 8], the CNN takes images as input and returns their corresponding vector representations as the output of one (the highest) convolutional layer (see Fig. 2).

Learning In this work we rely on the GURLS [12] machine learning library to perform RLS. As observed from previous work on the iCub [3], RLS exhibits com-

[1] http://www.iit.it/en/projects/data-sets.html

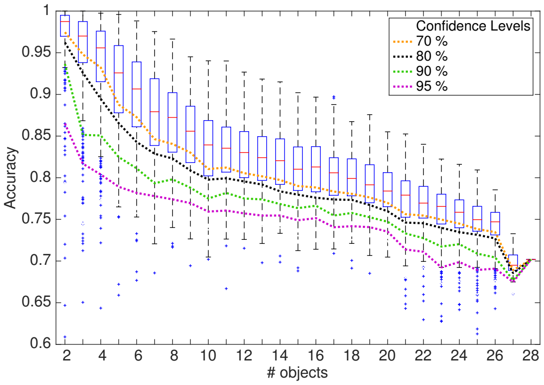

Figure 4: Box plots of the accuracies measured for predictors trained on random subsets of $t = 2, ...28$ objects (whiskers with maximum 1.5 IQR), with superimposed (dotted curves) the minimum accuracy guaranteed within a fixed confidence level.

parable or even better results than the liblinear [13] SVM library. Moreover, the rank-one update rule for matrix inversion [14], implemented in the GURLS library provides a natural variant of the classic RLS algorithm to the setting in which training data is provided incrementally to the system.

2.2 Acquisition Setup

The acquisition protocol followed to collect iCubWorld28 is analogous to the one used for previous releases [3]. A human supervisor stands in front of the iCub and shows a novel object to it, while verbally giving the instruction to start learning the new class (Fig. 1). The robot localizes and tracks the object through a motion detection routine [16] (Fig. 2). In the meanwhile, cropped images are fed to the representation module (Sec. 2.1) that encodes them into vectors. The latter are used to train the classifier for a certain period or until the operator issues the instruction to finish the training. At test time, the human asks the robot to recognize an object; the iCub tracks it for some seconds (or until it is told to stop the recognition) and outputs at each frame the predicted class.

Within this application, we collected the iCubWorld28 dataset, comprising images of 28 objects organised into 7 categories (Fig. 3). For each object we acquired train and test sets of 220 images each, during sessions of 20s in which the human operator was moving and rotating the object randomly in front of the robot. To assess the incremental learning performance of the system (see Sec. 3.2) we repeated this acquisition protocol for 4 days, ending up with 4 datasets (Day 1, to 4) of more than 12k images each.

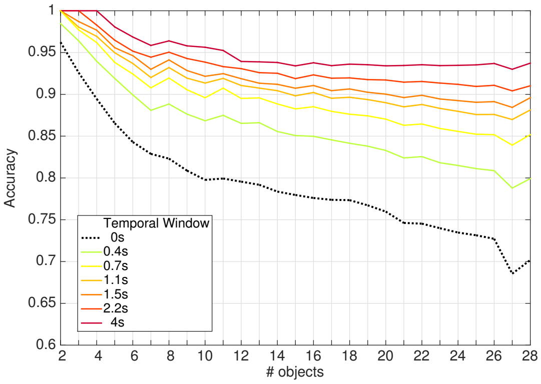

Figure 5: Improvement of the classification accuracy with an increasingly temporal filtering window. Confidence level fixed to 80% (see Fig. 4).

3 RESULTS

In this Section we evaluate the considered system on the acquired iCubWorld28 dataset, according to the criteria pointed out in Sec. 1.

We first introduce a way to measure the confidence with which the classification performance achieved on our benchmark is expected to generalize during a generic run of the application described in Sec. 2.2. To this end, we observe that the classification performance of a robotic recognition system should not vary dramatically when we change the set of objects on which it is trained/tested. Therefore, to offer insights on the expected recognition capabilities of the iCub for any choice of objects – at least among our dataset of 28 objects – in Fig. 4 we report, in the form of box plots, the empirical probability distributions of the accuracy achieved by the considered pipeline on increasingly larger object identification tasks. We randomly extracted many subsets of increasing size (∼ 400 couples, triplets, and so on) from the pool of 28 objects in the dataset. We trained/tested a classifier on each subset to the task of object identification, considering the achieved accuracy as an observation for estimating these distributions (we show the results for Day 4).

We superimpose on the box plots the minimum accuracy that a classifier trained on a number of random objects is guaranteed to achieve within a specified confidence level (dotted lines). We computed this as the minimum accuracy value for which the fraction of observations in the estimated distribution was higher than the specified confidence threshold. This result and its visualisation is of particular use from a practical perspective since it can be employed as a reference "data sheet" to train the iCub. Indeed, depending on

		TEST Accuracy (%)				
		Day 1	**Day** 2	**Day** 3	**Day** 4	**Average**
TRAIN	**Day** 1	67.7	41.9	37.2	67.2	53.5
	Day 2	40.1	67.8	35.4	66.8	57.5
	Day 3	62.0	63.5	66.4	64.9	64.2
	Day 4	62.9	64.1	65.3	67.1	64.8
	All Days	73.4	71.0	68.1	68.9	70.3

Table 1: Accuracy of predictors trained on single days compared with a predictor trained on all days together.

the desired confidence level, and the number of objects we want the robot to discriminate, Fig. 4 informs us what is the approximate level of accuracy that we can expect to achieve with the classifier that we will train.

3.1 Exploiting Contextual Information

The reported classification performances are clearly not comparable to the human-level accuracy that we expect on the problem considered. However, so far we have not been considering that the robotic setting offers a great deal of contextual information that must be leveraged in order to meet the strict requirements imposed by the interaction scenario. In particular, in the real-world the natural assumption holds that the class of an object does not change while the robot observes it from multiple points of view. Therefore, given a stream of frames acquired from different viewpoints around the object of interest, it may be more reasonable to consider, as the correct prediction for the object at a given frame, instead of the frame label, the most frequent label appeared during the last w frames. In Fig. 5 we report the effect of this label-filtering approach on the results reported in Fig. 4: in particular, it can be seen that the confidence curve associated to 80% (black), remarkably improves when increasing the duration of the temporal windows from 0 (instantaneous) to 4 seconds (from green to red), corresponding to a range of w between 1 and 50 frames.

3.2 Incremental Learning: A week (almost) with iCub

In this Section we consider another important source of contextual information commonly available in humanoid robotics and that yet it is not clear how to integrate in recognition systems employed in the field. Indeed, during its lifetime the robot undergoes repeated training sessions in different situations, eventually involving the same objects. In this regard, a humanoid recognition system should implement life-long learning methods to continuously update its internal models as new observations are encountered.

Here we provide a preliminary but promising evaluation of how much novel training evidence improves the performance of identification of known objects. We recall that iCubWorld28 is a dataset collected during 4 separate days and that for each day training and test sets were acquired. While experiments discussed so far were performed on a single day (Day 4), we now consider also the remaining 3 days and, to reduce the amount of computations, we focus only on the task of identifying the 28 objects in the dataset. We compare the performance of 4 predictors, each trained on a different day (taking the first 100 examples per class) with the performance of another predictor, trained on a mixed training set composed by all days (taking the first 25 examples per class from the training set of each day). Table 1 reports the classification accuracy of the five predictors tested separately on each day and on the union of the test sets of all days. The predictor trained on the mixed dataset clearly outperforms the others when tested on a global test set. Remarkably this is also true when performance is tested on data acquired on a single day, for which the single classifiers had been trained specifically.

	Confidence				
	98%	90%	80%	70%	50%
# Objects	2	4	6	7	14

Table 2: The maximum number of objects that iCub is able to recognize with 0.98 accuracy.

3.3 How many objects can iCub recognize?

We now come back to the original question regarding the maximum number of objects that iCub recognizes with the described recognition system. Table 2 empirically answers this question, in terms of the maximum number of objects on which we achieved human-level classification accuracy – fixed for reference to 0.98 – on iCubWorld28, within decreasing confidence levels.

4 DISCUSSION

In this paper we tested the current visual recognition capabilities of the iCub humanoid robot. Our study addressed the generic question "*How many objects can the iCub recognize?*", which was then formulated more accurately as the problem of determining the maximum number of objects that current visual recognition systems can recognize with virtually perfect accuracy.

We identified a natural HRI application as a testbed for our investigation of the visual recognition problem. In order to foster the reproducibility of our experiments we collected a novel dataset within this scenario, iCubWorld28. We approached the problem by first defining a performance measure that would allow us to operatively quantify our confidence that results observed on iCubWorld28 would then generalize to the real application. We then identified multiple contextual aspects that can be leveraged to improve the recognition capabilities of the robotic system.

Following these principles we were able to provide a preliminary answer to the original question: on the one hand, modern visual representation architectures as CNNs are finally able to address visual recognition in robotic settings; on the other hand however the problem is still challenging and far from being solved.

Current research focuses on extending this study to object categorization tasks, through an ongoing acquisition of a new iCubWorld. This comprises more object instances per category and more acquisition sessions under different conditions, in order to better assess the benefit of incremental learning. We are then improving this HRI application by introducing online active learning techniques that allow the robot itself to decide when the training/test examples acquired are enough to provide a prediction with a required confidence level. This goes also in the direction of minimizing the need for training samples, in order to efficiently learn deep representations in robotic settings while accounting for contextual information.

Acknowledgements

The research leading to these results has received funding from the European FP7 ICT project No. 270273 (Xperience). We acknowledge NVIDIA Corporation for the donation of the Tesla K40 GPU used for this research.

References

[1] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, “Imagenet large scale visual recognition challenge,” 2014.

[2] G. Metta, G. Sandini, D. Vernon, L. Natale, and F. Nori, “The icub humanoid robot: an open platform for research in embodied cognition,” in *8th Work. on Performance Metrics for Intelligent Systems*, 2008, website: http://www.icub.org.

[3] C. Ciliberto, S. Fanello, M. Santoro, L. Natale, G. Metta, and L. Rosasco, “On the impact of learning hierarchical representations for visual recognition in robotics,” in *IROS*, 2013.

[4] B. Schölkopf and A. J. Smola, *Learning with kernels: Support vector machines, regularization, optimization, and beyond.* MIT press, 2002.

[5] T. Hastie, R. Tibshirani, J. Friedman, T. Hastie, J. Friedman, and R. Tibshirani, *The elements of statistical learning.* Springer, 2009, vol. 2, no. 1.

[6] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, “Decaf: A deep convolutional activation feature for generic visual recognition,” *arXiv preprint arXiv:1310.1531*, 2013.

[7] M. D. Zeiler and R. Fergus, “Visualizing and Understanding Convolutional Networks,” *ArXiv e-prints*, Nov. 2013.

[8] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, “Return of the devil in the details: Delving deep into convolutional nets,” in *British Machine Vision Conference*, 2014.

[9] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, “CNN Features off-the-shelf: an Astounding Baseline for Recognition,” *ArXiv e-prints*, Mar. 2014.

[10] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding,” *arXiv preprint arXiv:1408.5093*, 2014.

[11] A. Krizhevsky, S. Ilya, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems (NIPS)*, 2012.

[12] A. Tacchetti, P. K. Mallapragada, M. Santoro, and L. Rosasco, “Gurls: A least squares library for supervised learning,” *Journal of Machine Learning Research*, vol. 14, pp. 3201–3205, 2013. [Online]. Available: http://jmlr.org/papers/v14/tacchetti13a.html

[13] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, “Liblinear: A library for large linear classification,” *Journal of Machine Learning Research*, vol. 9, 2008.

[14] G. H. Golub and C. F. Van Loan, *Matrix computations.* JHU Press, 2012, vol. 3.

[15] G. Metta, P. Fitzpatrick, and L. Natale, “Yarp: Yet another robot platform,” *International Journal on Advanced Robotics Systems*, 2006.

[16] C. Ciliberto, S. R. Fanello, L. Natale, and G. Metta, “A heteroscedastic approach to indipendent motion detection for actuated visual sensors,” in *IROS*, 2012.