Automated generation of training sets for object recognition in robotic applications

Markus Schoeler, Florentin Wörgötter, Mohamad Javad Aein and Tomas Kulvicius
Bernstein Center for Computational Neuroscience (BCCN)
III. Physikalisches Institut - Biophysik, Georg-August University of Göttingen
Email: (mschoeler, worgott, maein, tkulvic)@gwdg.de

***Abstract*—Object recognition plays an important role in robotics, since objects/tools first have to be identified in the scene before they can be manipulated/used. The performance of object recognition largely depends on the training dataset. Usually such training sets are gathered manually by a human operator, a tedious procedure, which ultimately limits the size of the dataset. One reason for manual selection of samples is that results returned by search engines often contain irrelevant images, mainly due to the problem of homographs (words spelled the same but with different meanings). In this paper we present an automated and unsupervised method, coined Trainingset Cleaning by Translation (*TCT*), for generation of training sets which are able to deal with the problem of homographs. For disambiguation, it uses the context provided by a command like "tighten the nut" together with a combination of public image searches, text searches and translation services. We compare our approach against plain Google image search qualitatively as well as in a classification task and demonstrate that our method indeed leads to a task-relevant training set, which results in an improvement of 24.1% in object recognition for 12 ambiguous classes. In addition, we present an application of our method to a real robot scenario.**

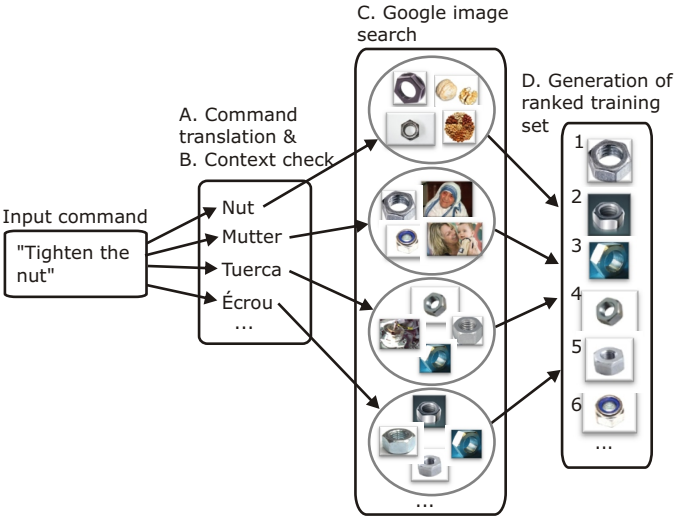

Fig. 1. Flow diagram of the proposed algorithm exemplified on the class "nut" in the context of "tighten". The training set is ranked according to the number of subsets matches occured. Only images which have a match in at least one other subset are further considered.

I. Introduction

In the field of robotics object recognition plays an important role and is crucial for object manipulation tasks, since task specific objects/tools first have to be found and identified correctly before they can be used. To demonstrate, suppose we have a robot-scenario where we tell the robot to "fill the cup with water" as shown in Fig. 6. In order to recognize the bottle and the cup in the scene, the robot has to be trained on these objects beforehand. The training procedure is typically done by off-line training of a classifier with a pre-selected set of classes (images), where images are gathered manually by a human ([1], [2], [3], just to name a few), thus, in a supervised way. Some new approaches make use of Internet searches in order to get information about objects and instructions [4], [5], [6], [7]. Although modern search engines like Google or Yahoo can return a large number of images within milliseconds, not all of the returned images are task/context-relevant, especially due to the problem of homographs (polysemes), i.e., words that are spelled the same but which correspond to different meanings or objects. For example, the word "cup" can correspond to a cup for drinking, the world-cup or bra's cup. "Apple" could mean the fruit, the brand logo or an Apple product. Nut could refer to a hex-nut or the food-nut (see Fig. 2 for an example).

In general, the performance of recognition systems heavily depends on the quality of the training data, thus, only task-relevant images should be collected. This is mostly done by searching for the plain class name or the class name with some context in huge image databases (e.g., Google image search, Bing image search) and by selecting the most-relevant images. As this is especially non-trivial for search terms which are homographs, most recognition methods are trained using manually cleaned or even hand-made training-sets, the creation of which is a time consuming and tedious procedure. Moreover, if a certain task (like "tighten the nut") requires knowledge about an object which is not in the training set, execution is not possible and, even worse, new training images need to be taken or collected and cleaned manually before the robot is able to execute the task.

A lot of research exists on trying to solve this problem of dirty image search results, for example by making use of additional visual cues, e.g., local image patches, edges, texture, color, deformable shapes, just to name a few [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18]. All of these approaches use textual information, too. Either implicitly, by using the first results of text-based image search engines [9], [10], by constructing their own image search engine [12], [13], [18], or explicitly, by making use of image tags and labels as found in photo-sharing websites like Flickr [11], [12], [16]. To our knowledge, all of the above presented approaches achieve an improvement with respect to the quality of the result set. However, none of these methods can automatically cope

with the problem of homographs (polysemes), which would be required in automated robotic applications like [4], [5], [6].

In this paper, in order to address the problem of homographs, we present a method for automatic (without human supervision) generation of task-relevant training sets for object recognition by using the information contained in a language-based command like "cut the apple" or "fill the cup". We ground our approach based on two facts: 1) homographs rarely occur for one word in multiple languages at the same time and 2) context information (action) provided by the command can be used in order to get rid of ambiguous and non-task relevant translations. In order to create such an automatic system we will employ a combination of publicly available image search engines, text search engines and translation services.

The paper is organized as follows. First, we present our algorithm in detail in Section II. Then, in Section III, we show a qualitative comparison for selected classes (Section III-A) and evaluate the performance of our method quantitatively in an object classification task (Section III-B). Additionally we present an implementation of our method in a real-robot scenario (Section III-C). Finally, we conclude our study in Section IV by discussing our approach and comparing it to other existing methods.

II. Proposed Algorithm

The algorithm consists of four sequential steps: 1) command translation, 2) context check, 3) image download and 4) subset matching (see Fig. 1). In this section we will use the example of nut as it will be important for our robot scenario in Sec. III-C. Nut is a homograph and can mean either a hardware-nut or a food-nut. Generating a training set using a plain Google search for "nut" will not work. In this case even humans cannot infer which object nut refers to. However, the command "tighten the nut" or "crack the nut" provides valuable context information for disambiguation which we want the robot to use.

A. Command translation

The first step of the algorithm is to translate the command to different languages ignoring the articles "the","a","an". For this we translated nouns and verbs separately. Note that in our study we used a fixed command syntax: verb/action + noun/object. A more general command syntax would require the usage of grammar analysis methods (i.e., parsers [19]). In this paper we used four languages: English, German, Spanish, French and Portuguese. Here, Portuguese was only used in the case when translations into the other languages resulted in less than three different terms (e.g., orange is the same word in English, French and German). As an example we will show the generation of the German subset. The first three translations for nut, tighten and crack are shown in Table I. "Mutter" and "Schraubenmutter" correspond to the hardware-nut. "Nuss" corresponds to the food-nut. As one can see the double-meaning of nut is not present in German.

TABLE I. First three English to German translations for nut, tighten and crack returned by www.dict.cc.

nut	**tighten**	**crack**
Nuss	anziehen	zerbrechen
Mutter	verschärfen	knacken
Schraubenmutter	straffen	zersplittern

B. Context check

If the translation service returns more than one translation for the noun this step will perform a context check using Google text search. The idea here is that Google will return significantly less results for a phrase which does not make sense like "Nuss anziehen" (tighten the food-nut), compared to a reasonable phrase like "Mutter anziehen" (tighten the hardware-nut). We forced exact matches using the "as_epq=" search parameter in the Google search. Since the order of the words influence the number of results for exact searches, we searched in both orders (noun verb as well as verb noun) and took the maximum number of results as the score. To retrieve the right noun in the specific context the algorithm uses the noun which gets the highest score with any verb combination. Table II shows how the context relevant German translations for "nut" can be reliably determined. The relevant translations in German, French and Spanish for "crack the nut" are Nuss, Noix, Nuez. The translations for "Tighten the nut" are Mutter, Écrou and Tuerca.

TABLE II. Context check for "tighten the nut" and "crack the nut" using the number of exact matches returned by Google text search. The noun with most matches is chosen (marked bold).

"tighten the nut"		"crack the nut"	
Term	**Matches**	**Term**	**Matches**
Nuss anziehen	445	Nuss zerbrechen	114
Nuss verschärfen	5	**Nuss knacken**	**13500**
Nuss straffen	256	Nuss zersplittern	7
Mutter anziehen	**6500**	Mutter zerbrechen	570
Mutter verschärfen	26	Mutter knacken	476
Mutter straffen	6	Mutter zersplittern	1
Schraubenm. anziehen	218	Schraubenm. zerbrechen	3
Schraubenm. verschärfen	4	Schraubenm. knacken	2
Schraubenm. straffen	4	Schraubenm. zersplittern	1

C. Google image search

This step downloads images for all relevant translations. In the "tighten the nut" context it downloads images for Nut, Mutter, Écrou and Tuerca into 4 separate subsets. In the context of "crack the nut" it downloads images for Nut, Nuss, Noix and Nuez.

D. Generation of ranked training set

Task-relevant images can be found in all subsets, whereas images which correspond to irrelevant context can usually be found only in one set. Nut in the hardware context is a good example as it translates to the German word "Mutter" which is also a homograph meaning the hardware-nut as well as "mother" (see Fig. 2). While mother images are only found in the German and food-nut images only in the English subset, images of hardware-nuts are found in all subsets. For similarity matching we used the procedure proposed by Kulvicius et al. [7]. The pseudo-code in Fig. 3 shows how the score is assigned to each image I_i^k : the number of subsets where a match has been found SM_i^k. Only images which have a match in at least one other subset are considered, i.e., $SM_i^k > 0$. Images are then sorted in descending order by the number of subsets they

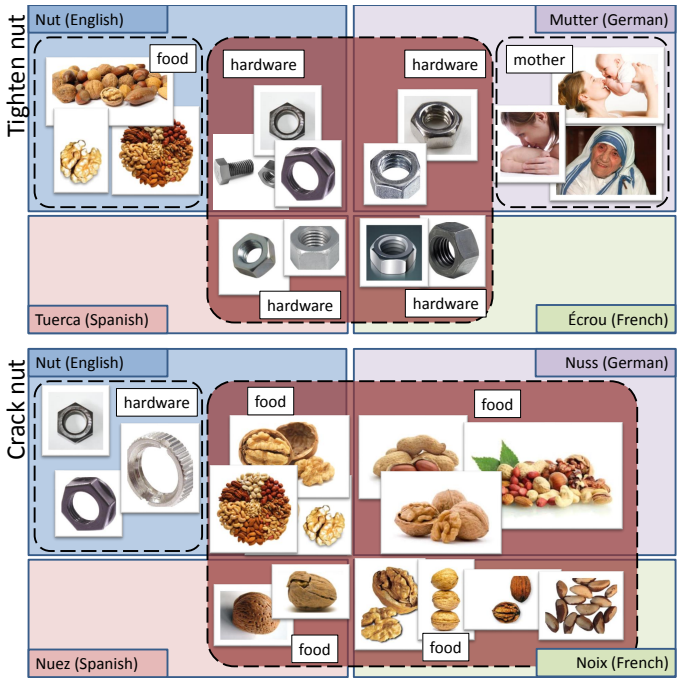

Fig. 2. Example word "nut" which is a homograph in English (food and hardware) and German (mother and hardware). By combining multiple languages and using the context check the proposed algorithm is able to retrieve the task relevant images for "nut" in both tasks ("tighten the nut" and "crack the nut"), the intersection marked with the dark red rectangle)

matched. This assures that most task-relevant images are found at the beginning of the list whereas borderline cases are found at the end.

Get images I_i^k $(k = 1 \dots m, i = 1 \dots n_k)$, where
m is the number of subsearches/languages considered and
n_k is the number of images in subsearch k;
Set similarity threshold θ;
Initialize matches $SM_i^k = 0$.

```
FOR k = 1 to m
    FOR i = 1 to n_k
        FOR l = 1 to m
            IF k! = l
                FOR j = 1 to n_l
                    Compare images I_i^k and I_j^l by
                      calculating similarity s
                    IF s > θ
                        increment(SM_i^k);
                        quit outer loop;
```

Fig. 3. Pseudo-code for the subset matching to determine image relevance.

For similarity calculation the algorithm generates signatures using radially aligned gray-SIFT features as described in previous work [20] (the center is set to the middle of the image). Features are sampled on a dense grid on three scales. A bag-of-visual-words algorithm with 100 visual words is used to generate image signatures. As similarity measure we used the histogram intersection over all visual word bins. The similarity threshold θ was set to 0.7.

III. Results

In order to evaluate the performance of the our algorithm we used 12 homographic classes. All classes, their possible meanings and action contexts are depicted in Table III. All classes have been used in the classification experiment in Sec. III-B. From now on we will denote the class in a specific action context as "class-action" (e.g., nut-crack and nut-tighten). For classification we used the method proposed by [20] which uses a combination of gray-SIFT and CyColor features. Local descriptors are extracted on a dense grid and oriented along the dominant local gradient (the latter using the SURF detector). Three hundred visual words were used for the signature generation. A support-vector-machine with a histogram intersection kernel is used for the machine learning.

TABLE III. The 12 classes used in the evaluation. All classes have multiple meanings (not all are shown). The relevant meaning for the context is marked bold. The last column shows the translations after context check. These are used as the subsets for the image retrieval (see Sec. II-C).

Term-context	Meanings of noun	Translations
apple-cut	**food**, laptop, logo	manzana, pomme, apfel
axe-chop	**hardware**, brand	hacha, hache, axt
bolt-tighten	**hardware**, athlete, movie	tornillo, boulon, bolzen
cup-fill	**drinking**, trophy, bra	taza, tasse
hammer-hit	**hardware**, brand	martillo, marteau
nut-crack	hardware, **food**	nuez,noix, Nuss
nut-tighten	**hardware**, food	tuerca, ecrou, mutter
oil-eat	**food**, mineral-oil	aceite, huile, oel
orange-cut	**food**, color	laranja, naranja
pan-fry	**hardware**, movie, god	sarten, poele, pfanne
peach-eat	**food**, computer character	molocoton, peche, pfirsich
pot-cook	**hardware**, drug	cacerola, casserole, topf
saw-cut	**hardware**, movie	sierra, scie, saege

A. Qualitative comparison

To visualize the qualitative performance of the algorithm Fig. 4 shows the first 10 images retrieved by Google, searching for the plain classname (Google Class only) as well as for the noun together with the action-context (Google Class+Action). Additionally we show the 10 highest ranking images retrieved by our algorithm. The problem of homographs is especially obvious in the case of plain classname searches, since no context is provided which could help to disambiguate. This is why we retrieve the same image sets for "nut" in either context. Consequently bolt and nut in the tighten context show solely irrelevant images except one. Using the action together with the classname does not yield much better results, since images are very affected by image clutter and irrelevant content showing the action instead of the isolated object. In contrast, our algorithm yields a much cleaner image set for all classes.

B. Image classification

Additionally we tested the performance of the algorithm quantitatively in an image classification experiment. We wanted to prove that training a classifier with images obtained by *TCT* results in significantly better classification accuracy as compared to training with uncleaned Google images. For comparison we generated three training sets: One returned from Google search using searches for the plain noun (C200), one with searches for the noun together with the action verb (CA200) and one created by proposed algorithm (TCT). For

Fig. 4. Images retrieved by Google image search and by our algorithm (*TCT*) for 4 example classes. Only the first 10 highest scoring images are shown.

the latter we used all images which got at least one match in another subset $SM_i^k > 0$. The number ranged from 80 (apple-cut) to 381 images (pan-fry).

The sets C200 and CA200 consist of the 200 highest scoring images. For testing we manually created a disjoint set containing only task-relevant images obtained from Google searches using other languages. Fig. 5 shows the confusion matrices for all three training sets. We can observe that using the context for the Google search yields on average better accuracy than the plain search, since it can disambiguate classes. This however comes at the cost of a high fraction of clutter and irrelevant object in the images returned for "Class+Action" which leads to worse results for "cup-fill", "hammer-hit", "nut-tighten" and "oil-eat". Using our algorithm we are able to increase the recognition accuracy by 24.1 % from 45.9 % to 70 %. Even more important: Using the TCT training set, the classifier can tell "nuts" in the context of "crack" from the ones in the context of "tighten". This is a requirement for

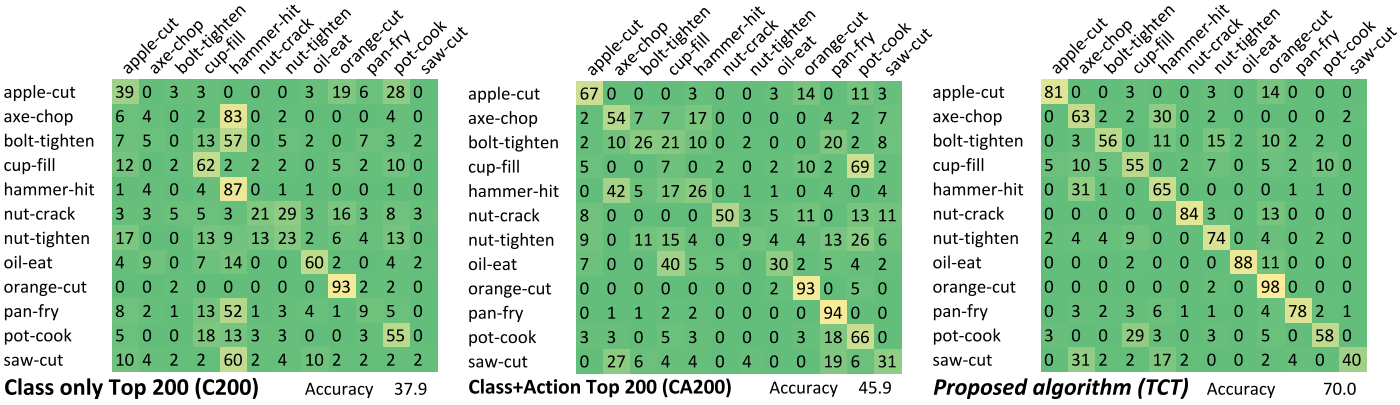

Fig. 5. Confusion Matrix as well as accuracy in percent. Rows correspond to the actual class label and columns to the predicted class labels returned by the classifier. Only if we train the classifier with images returned by *TCT*, we can disambiguate the classes needed in the robot scenario.

the robot scenario in the next section. None of the image sets returned by Google could be used instead. Please note, we also trained a classifier with the 300 highest scoring images (CA300 and C300), but this decreased the classification performance from 45.9 % to 44.7 % for "Class+Action" and from 37.9 % to 33.9 % for "Class only".

C. Robotic application

Last but not least, we applied our method to a robot application where we let a KUKA LWR robot-arm [21] perform three actions (see Fig. 6):

1) "fill the cup" (with water from a bottle)
2) "crack the nut" (with the stone)
3) "tighten the nut"

For each action only one object is task relevant. Since our method was the only one which can discriminate "hardware-nuts" from "food-nuts" we used that one for the training-set generation. In all cases the robot needs to ignore all distractors and choose the right object depending on the action context. Several aspects, like object recognition and robot movement execution, rely on published works and will not be described here in detail. To extract objects from the scene we used the object extraction pipeline of [20] using RGB-D data for segmentation and high resolution images (4928×3264 pixels) for object recognition. We additionally trained a background class which consisted of images of the table, the robot arm as well as the zucchini and the spoon.

For action execution we used the library of manipulation actions from [22], which is based on semantic event chains [23] and modified dynamic movement primitives [24]. Here, specifically, we used pouring, picking-up and putting-down actions. Object positions came directly from the object extraction by averaging all points in the pointcloud belonging to the object. The action "tighten" is a complex action sequence and consists of "pick up", "put on" and "turn". "Put on" and "turn" are difficult actions which require detailed knowledge about the objects and high precision on performing the action (including sensory feedback). As this is not in the focus of this paper, we only required the robot to execute the first step of this action.

In case 1) the robot finds out that cup refers to the coffee-cup and ignores the trophy-cup. Using the context "crack" in case 2) the robot detects the food-nut and ignores the hardware-nut. In case 3) the food-nut is ignored since we generated training images for the context relevant hardware-nut. Note that in our case the commands were typed directly into the computer program with a predefined syntax (action + article + noun). Additionally, we started with the bottle and stone grasped by the robot hand. Consequently, the task for the robot was to find out and recognize which cup and nut the commands refer to and to execute the corresponding action.

In Fig. 6 we show snapshots of the experiment. The robot successfully recognized the cup for filling, the hardware-nut for tightening and the food-nut for cracking. Please refer to the supplementary material for the full video.

IV. Discussion

In this paper we presented a method for automated generation of task-relevant training-sets for object recognition by combining image search engines, text search engines and translation services. The method is useful for obtaining "cleaner results" in image searches. While this is already a valuable property of the algorithm, it is of particular importance in the case of homographs. We showed that the presented approach indeed leads to cleaner search results and better recognition rates as compared to plain Google search. The method was developed with autonomous robotic systems in mind, where a robot has to collect (without human supervision) relevant images from the internet, in order to disambiguate and execute human instructions. In this section we will discuss our approach and how it relates to other existing methods.

In the field of artificial intelligence and computer vision object classification is considered one of the hardest tasks. Due to its importance for many applications, including robotic systems, a lot of effort has been made in order to improve the performance of recognition methods. As shown above, it also highly depends on the quality of the training-set. Generating such training-sets for robotic applications by a human operator is a very time consuming and tedious procedure, which also limits the size of the training set. On the other hand, keeping only the first pages returned by Google [10] limits the size of the training set even more, and worse, will not work

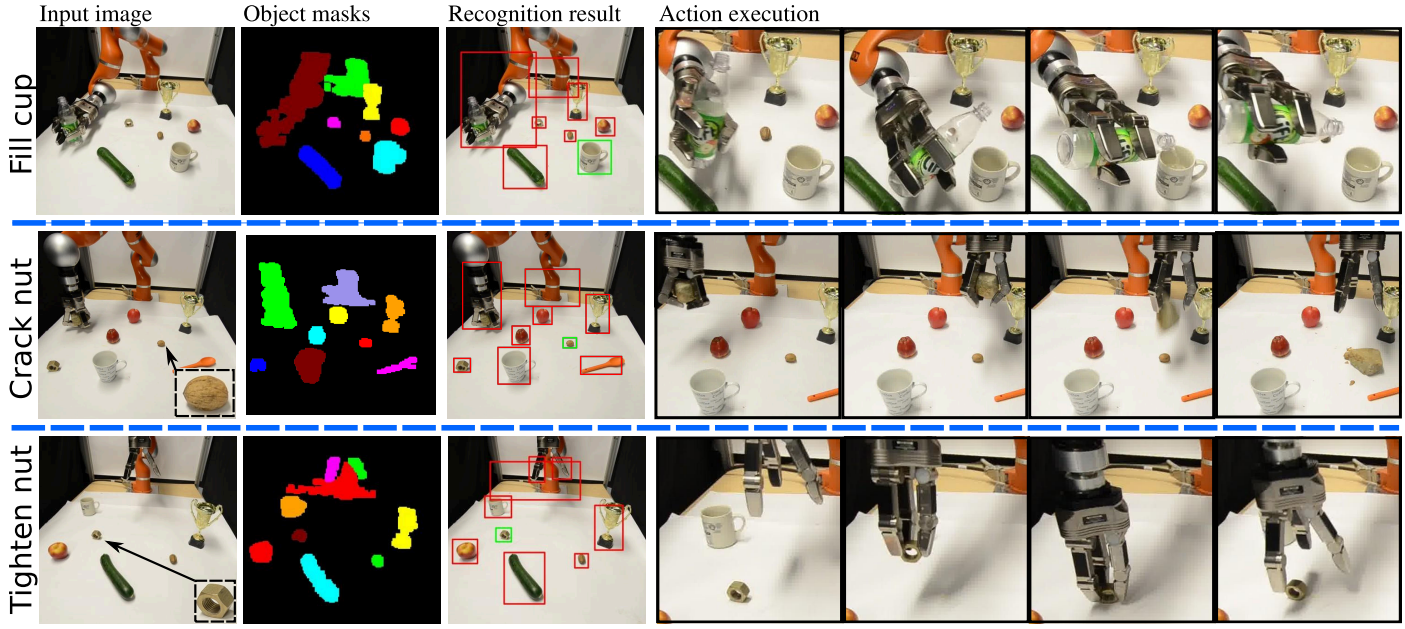

Fig. 6. Three example scenes where the robot had to perform the actions "fill the cup", "crack the nut" and "tighten the nut". The robot starts without knowledge about cups and nuts. In addition to the objects involved in the action we put other items as distractors into the scene. One of them being a different type of cup (the trophy) (see Fig. 4). Even though two items can be referred to by the word nut, only one of them is relevant for the specific action. The robot uses our algorithm to determine the context relevant objects and generates a training set on-the-fly. RGB-D information is used to generate object masks and a high resolution image is used for the classification (see [20] for details). The green box marks the object which gets the highest score from the classifier.

at all for homographic classes (see Fig. 4). The approach presented here provides a solution to solve such problems, based on the additional context information provided by the task (command) and four different subsearches (languages) to automatically retrieve clean training sets. Additionally, adding more languages/subsets (especially with different roots) and several search engines should lead to larger datasets, a more fine-grained relevance score and therefore an even greater improvement in object recognition performance. One could also improve the results by using state-of-the-art image retrieval algorithms like [25] for the image matching.

We have shown that our method performs well as long as the actions allow inference of context, as with fill, crush, crack, pour, cut, screw on, tighten, nail down, and so on. However, performance will drop if actions are used which can be applied to many objects in different contexts, as is generally the case with actions like give, put, move, place, lift and throw. Nevertheless, even humans would experience this problem and would require additional information (if the context is not known beforehand) in cases like "give me the nut".

Using our algorithm for the classifier training we were not only able to boost recognition accuracy by 24.1 % to 70 % (compared to 45.9 % when using images from Google). More importantly using this classifier the robot was able to detect the right objects for "tighten the nut", "crack the nut" and "fill the cup" using the provided context and to successfully execute the command.

Our approach most closely relates to the approaches of Kulvicius et al. [7] and Tamosiunaite et al. [6]. In [7] additional language cues are used in order to perform several sub-searches based on specific context. For example, to generate a task-relevant dataset for the class cup it could use "coffee cup", "tea cup", "full cup", "empty cup", etc. Such context-dependent cues can be obtained from language analysis. However, this requires knowledge about the domain as well as collecting a text-corpora for each specific context. In contrast, in the current approach there is no need for such information, and the context is provided by the action (verb). Similar to our approach, Tamosiunaite et al. [6] make use of language and actions together with Google text search in order to boot-strap in the object domain and to find out which other objects could be used as a replacement. If the command is "cut the cucumber", then the algorithm would return that carrots, potatoes, apples, etc. can be cut, too. Unlike [6], we use the action for a different purpose, i.e., in order to generate the relevant subsets.

As explained above, our approach requires textual (language-based) cues in order to perform image searches. In our study these cues were entered manually in a computer program as a text-command. However, such cues could come from human-robot interaction using natural language communication [26], [27], [28]. Thus robots would obtain language-based commands from humans (e.g., "fill the cup with water"). The other example of language-enabled robots are robots executing instruction sheets based on natural language [4], [5]. The algorithm presented in this paper, as discussed above, is developed having such robotic systems in mind as well.

In summary, we believe that this is a promising approach for automated and unsupervised generation of task-relevant training-sets for object classification/recognition, which has potential for use in many different kinds of robotic applications.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 (Programme and Theme: ICT-2011.2.1, Cognitive Systems and Robotics) under grant agreement no. 600578, ACAT.

REFERENCES

[1] M. Muja, R. B. Rusu, G. Bradski, and D. G. Lowe, "REIN-A fast, robust, scalable REcognition INfrastructure," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.

[2] M. Wail, N. Pugeault, and N. Krger, "Multi-view object recognition using view-point invariant shape relations and appearance information," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[3] Y. Sun, L. Bo, and D. Fox, "Attribute based object identification," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[4] M. Tenorth, U. Klank, D. Pangercic, and M. Beetz, "Web-enabled Robots – Robots that Use the Web as an Information Resource," *Rob. & Automat. Magazine*, vol. 18, no. 2, pp. 58–68, 2011.

[5] M. Beetz, U. Klank, I. Kresse, A. Maldonado, L. Mösenlechner, D. Pangercic, T. Rühr, and M. Tenorth, "Robotic Roommates Making Pancakes," in *IEEE-RAS Int. Conf. on Humanoid Robots*, October, 26–28 2011, pp. 529–536.

[6] M. Tamosiunaite, I. Markelic, T. Kulvicius, and F. Wörgötter, "Generalizing objects by analyzing language," in *IEEE-RAS Int. Conf. Humanoid Robots*, oct. 2011, pp. 557–563.

[7] T. Kulvicius, I. Markelic, M. Tamosiunaite, and F. Wörgötter, "Semantic image search for robotic applications," in *Int. Workshop on Robotics in Alpe-Adria-Danube Region (RAAD2113)*, 2013.

[8] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2003, pp. 264–271.

[9] ——, "A visual category filter for google images," in *Europ. Conf. Computer Vision*, May 2004, pp. 242–256.

[10] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman, "Learning object categories from google's image search," in *IEEE Int. Conf. Computer Vision*, vol. 2, oct. 2005, pp. 1816–1823.

[11] M. Guillaumin, J. Verbeek, and C. Schmid, "Multimodal semi-supervised learning for image classification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[12] T. L. Berg and D. A. Forsyth, "Animals on the web," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006, pp. 1463–1470.

[13] F. Schroff, A. Criminisi, and A. Zisserman, "Harvesting image databases from the web," in *Int. Conf. on Computer Vision*, Oct. 2007, pp. 1 –8.

[14] I. Khan, P. M. Roth, and H. Bischof, "Learning object detectors from weakly-labeled internet images," in *35th OAGM/AAPR Workshop*, 2011.

[15] L. Li, G. Wang, and L. Fei-fei, "Optimol: automatic online picture collection via incremental model learning," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[16] G. Wang, D. Hoiem, and D. Forsyth, "Building text features for object image classification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2009, pp. 1367 –1374.

[17] Y. Jing and S. Baluja, "Visualrank: Applying pagerank to large-scale image search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 11, pp. 1877 –1890, Nov. 2008.

[18] A. D. Holub, P. Moreels, and P. Perona, "Unsupervised clustering for google searches of celebrity images," *IEEE Int. Conf. Automatic Face and Gesture Recognition*, 2008.

[19] D. R. Dowty, L. Karttunen, and A. M. Zwicky, *Natural language parsing: Psychological, computational, and theoretical perspectives*. Cambridge University Press, 2005.

[20] M. Schoeler, S. C. Stein, A. Abramov, J. Papon, and F. Wörgötter, "Fast self-supervised on-line training for object recognition specifically for robotic applications," in *VISAPP*, 2014.

[21] Kuka Robot Systems. [Online]. Available: http://www.kuka-robotics.com

[22] M. J. Aein, E. E. Aksoy, M. Tamosiunaite, J. Papon, A. Ude, and F. Wörgötter, "Toward a library of manipulation actions based on semantic object-action relations," in *IEEE/RSJ International Conference on Intelligent Robots and System (IROS)*, 2013, p. in press.

[23] E. E. Aksoy, A. Abramov, J. Dörr, N. Kejun, B. Dellen, and Wörgötter, "Learning the semantics of object-action relations by observation," *The International Journal of Robotics Research*, vol. 30, no. 10, pp. 1229–1249, 2011.

[24] T. Kulvicius, K. J. Ning, M. Tamosiunaite, and F. Wörgötter, "Joining movement sequences: Modified dynamic movement primitives for robotics applications exemplified on handwriting," *IEEE Trans. Robot.*, vol. 28, no. 1, pp. 145–157, 2012.

[25] S. Paschalakis, K. Iwamoto, N. Sprljan, R. Oami, and M. Bober, "The mpeg-7 video signature tools for content identification," *IEEE Trans. Circuits Syst. Video Technol.*

[26] H. Holzapfel, D. Neubig, and A. Waibel, "A dialogue approach to learning object descriptions and semantic categories," *Robotics and Autonomous Systems*, vol. 56, no. 11, pp. 1004–1013, 2008.

[27] M. Bollini, S. Tellex, T. Thompson, N. Roy, and D. Rus, "Multi-view object recognition using view-point invariant shape relations and appearance information," in *International Symposium on Experimental Robotics (ISER)*, 2012.

[28] R. Deits, S. Tellex, P. Thaker, D. Simeonov, T. Kollar, and N. Roy, "Clarifying commands with information-theoretic human-robot dialog," *Journal of Human-Robot Interaction*, vol. 2, no. 2, pp. 58–79, 2013.