Multi-Label Object Categorization Using Histograms of Global Relations

Wail Mustafa*, Hanchen Xiong†, Dirk Kraft*,
Sandor Szedmak†, Justus Piater† and Norbert Krüger*
*Mærsk Mc-Kinney Møller Institute University of Southern Denmark,
Campusvej 55, 5230 Odense C, Denmark. Email: wail@mmmi.sdu.dk
†Institute of Computer Science, University of Innsbruck, Technikerstr.21a, A-6020 Innsbruck, Austria

Abstract

In this paper, we present an object categorization system capable of assigning multiple and related categories for novel objects using multi-label learning. In this system, objects are described using global geometric relations of 3D features. We propose using the Joint SVM method for learning and we investigate the extraction of hierarchical clusters as a higher-level description of objects to assist the learning. We make comparisons with other multi-label learning approaches as well as single-label approaches (including a state-of-the-art methods using different object descriptors).

The experiments are carried out on a dataset of 100 objects belonging to 13 visual and action-related categories. The results indicate that multi-label methods are able to identify the relation between the dependent categories and hence perform categorization accordingly. It is also found that extracting hierarchical clusters does not lead to gain in the system's performance. The results also show that using histograms of global relations to describe objects leads to fast learning in terms of the number of samples required for training.

I. Introduction

Object categorization is important for a variety of tasks, especially when systems are expected to deal with novel objects according to prior knowledge. Categorizing novel objects is useful in several applications such as driver assistance [16] and video surveillance [11]. In robotic applications in particular, categories can be linked to manipulation actions allowing for performing predefined actions on novel objects (see e.g., [19]).

Existing object categorization methods assume that objects belong to single and distinct categories (e.g., 'cup' and 'car') [2], [25] and thus employ single-label learning.

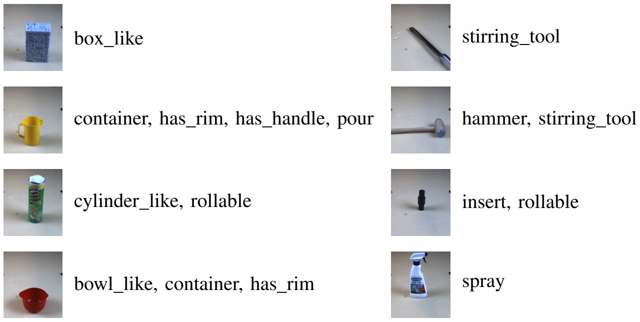


Fig. 1. Examples of labeled objects.

In this work, we consider scenarios in which objects can belong to multiple and related (by overlapping or nesting) categories (Fig. 1) associated with, potentially, different levels of abstraction. Such scenarios are very common in everyday objects. The ability to learn categories of different abstraction levels allows for, e.g., associating manipulation actions to visual patterns rather than designing actions for specific object instances. In the context of robotic manipulation actions, this means that multiple and dependent actions may be proposed as "affordances" for a novel object. For this learning problem, we utilize multi-label classification [32], which is intrinsically able to learn to assign multiple labels per data sample while considering the interdependence of the labels. In contrast, single-label methods—even when configured in 1-versus-all fashion—are expected to perform poorly on dependent categories.

In this work, objects are encoded using global descriptors composed of histograms of relative geometric attributes computed between full 3D features (Fig. 2). The 3D features are extracted using three RGB-D sensors (the three views are fused in 3D space) capturing object shapes rather completely. This description of objects is rich and highly invariant to viewpoint, leading to high performance and fast learning in terms of the number of samples required to train the system.

Mustafa, W.; Hanchen Xiong; Kraft, D.; Szedmak, S.; Piater, J.; Kruger, N., "Multi-label Object Categorization Using Histograms of Global Relations," in 3D Vision (3DV), 2015 International Conference on , vol., no., pp.309-317, 19-22 Oct. 2015 DOI: 10.1109/3DV.2015.42

In this paper, we propose using Joint Support Vector Machines (Joint SVM) as a multi-label classification method [31] exploiting the global 3D description. The Joint SVM, besides being computationally-efficient, was shown to outperform other state-of-the-art methods in image annotation datasets [31]. In addition, we investigate the extraction of higher-level descriptors based on hierarchical clustering as a way of assisting the Joint SVM. This is because the categorization problem addressed in this paper (i.e., the categorization of multiple and related categories) exhibits also a hierarchical nature. The hierarchical clusters are also used to provide categorical-valued descriptors needed for another multi-label method, which is based on homogeneity analysis [29] and used for comparison. Moreover, we make comparisons with a naive classifier introduced in [21] for validation purposes. The comparisons also include single-label approaches operated in 1-versus-all modes. Namely, we include Random Forests, which a state-of-the-art classifier, and Hierarchical Matching Pursuit (HMP) [7] framework, which is a state-of-the-art object categorization method for RGB-D data and it employs SVM as a classifier. The experiments are carried out on a dataset of 100 objects showing the performance of each method on visual and action-related categories.

The work presented in this paper has two main accomplishments. First, we show that multi-label learning methods are able to identify the relation between the dependent categories and hence perform object categorization accordingly. In this regard, Joint SVM is shown to outperform the other methods. However, no gain in performance is obtained by encoding objects in terms of hierarchical clusters. Second, we show that using histograms of global relations as object descriptors leads to learning that is at least as fast as using the HMP method (measured in number of samples required for training).

This paper is structured as follows. Related work is described in Sect. II. Next, we give a description of the visual representation of objects we use including the hierarchical cluster extraction in Sect. III. In Sect. IV, we give a brief description of the Joint SVM method. In Sect. V, we present the methods used for comparison. The experiment and the results are presented and discussed in Sect. VI. Finally, we conclude in Sect. VII.

II. Related Work

Early research on object categorization focused on generic representations that capture object shapes at high levels of abstraction (such as generalized cylinders [6], superquadrics [24], or geons [5]). The difficulty involved in reconstructing such abstractions from real objects has led to the development of solutions that could recognize only exemplar objects (i.e., object recognition) requiring little or no abstraction [9]. Over the years, the gap between the low-level and the high-level abstractions has been narrowed by introducing representations that are invariant to a number of geometrical properties such as view-point, rotation, and scaling. Such representations often make use of local descriptors such as the popular SIFT features [18] and various recently developed 3D features (refer to [1] for an overview).

Belongie et al. [3] proposed representing objects using ‘shape contexts’, which uses relative geometric information within a local neighborhood. Shape contexts were later extended to 3D in [10]. In this paper, we use geometric relations of 3D features introduced by Mustafa et al. in [23], which are similar to shape context but are defined in a global context. They are also similar to the global relative features introduced in [27] but with different underlying set of geometric attributes.

Recently, hierarchical approaches for object representation have shown high performance on large dataset [4]. Notably, Bo et al. [7] introduced a multi-layer network that builds feature hierarchies layer by layer with an increasing receptive field size to capture abstract representations. They show that their method achieves state-of-the-art performance in a large-scale RGB-D dataset of objects [15]. It is worth noting that these results are based on very large training data sets with significant computational cost.

Existing systems typically recognize only one category per object, i.e. single-label learning [7], [19]. In [21], a method that extracts visual clusters based on hierarchical agglomerative clustering [28] was introduced. Building such a hierarchy can be seen as a way to obtain candidates categories of different levels of abstraction where more generic categories are formed at the top of the hierarchy. Moreover, the hierarchy naturally group nested categories (i.e. in category/subcategory fashion). In this work, we investigate the extraction of these categories for assisting the multi-label learning algorithm to recognize multiple and related categorizes of objects.

Recently, multi-label learning has been used in several applications such as text processing, protein function classification and image annotation [32]. In [31], a multi-label classification method, which is based on Support Victor Machines (SVM), was introduced. This classifier, referred to as Joint SVM, was applied in an image annotation benchmark and shown to outperform other state-of-the-art methods. A practical merit of the Joint SVM is that it shares the same computational complexity as one single conventional SVM. In this paper, we use Joint SVM to learn object categories and compare it with other methods. One of the methods used for comparison is an approach based on homogeneity analysis [20]. The approach has shown to learn object-action relations exhibiting a multi-label learning problem [29].

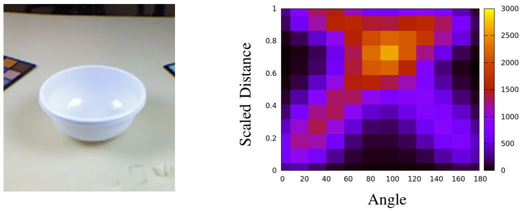


Fig. 2. Object representation using histograms of pair-wise geometric relations.

III. Visual representation of objects

In this section, we present the visual feature extraction and processing for objects (Subsect. III-A). In Subsect. III-B, we describe the extraction process of hierarchical clusters. In Subsect. III-C, we introduce the coding schemes applied to describe objects in terms of the extracted clusters (when used).

A. Histograms of global multi-view 3D relations

In this paper, object shapes are described using histograms of *relations* between pairs of 3D features. From RGB-D data (Kinect sensor), planar 3D surface features, i.e. texlets [14], are extracted. Prior to this, we apply object segmentation. The 3D texlet contains both position and orientation, and provides absolute information (relative to an external reference frame) of objects in 3D space. In this system, we fuse 3D texlets from three Kinect sensors observing the workspace in which objects occur. This fusion allows for a rather complete object information. To describe an object, we compute a set of pairwise relations from all pairs of texlets belonging to the object. One important aspect of relations is that they transform an absolute pose-variant representation into a relative pose-invariant one. The multi-view description of objects allows these global relations to become richer and more invariant to viewpoint. Such properties account potentially for high recognition performance and fast learning.

In this paper, geometric relations were defined by two attributes: angle and scale-invariant distance (i.e, normalized relative to the object size) computed between 3D texlets. The scale invariance of the distance relation is crucial for categorization because what defines a category is usually independent of scale. The final object descriptor is obtained by binning these two relations in 2D histograms, which model the distributions of the relations in fixed-sized feature vectors while considering their co-occurrence. The binning size is set to 12 in both dimensions resulting in a features vector of 144 dimensions. The binning size was chosen according to previous investigations conducted in [22]. Fig. 2 shows an example of an object described with a 2D histogram of angle and scaled distance.

B. Hierarchical clustering extraction

In order to test if the recognition performance improves by describing objects in terms hierarchical clusters, we apply the method introduced in [21]. The method is based on hierarchical clustering analysis [28]. The hierarchical clustering algorithm takes the histogram description of objects discussed above (Subsect. III-A) as an input and builds a hierarchy of clusters (representing candidate categories) in an unsupervised way. Hierarchical clustering naturally allows clusters to overlap and that makes it possible to explicitly extract nested categories (at different levels of abstraction).

The hierarchy is built from the data provided during the training phase (Fig. 3). Objects are then encoded according to the extracted clusters. In the experiments, for adequacy to the classification method, we consider two schemes of coding objects: branch-point coding or level coding. The two schemes provide binary or categorical values, respectively (Subsect. III-C).

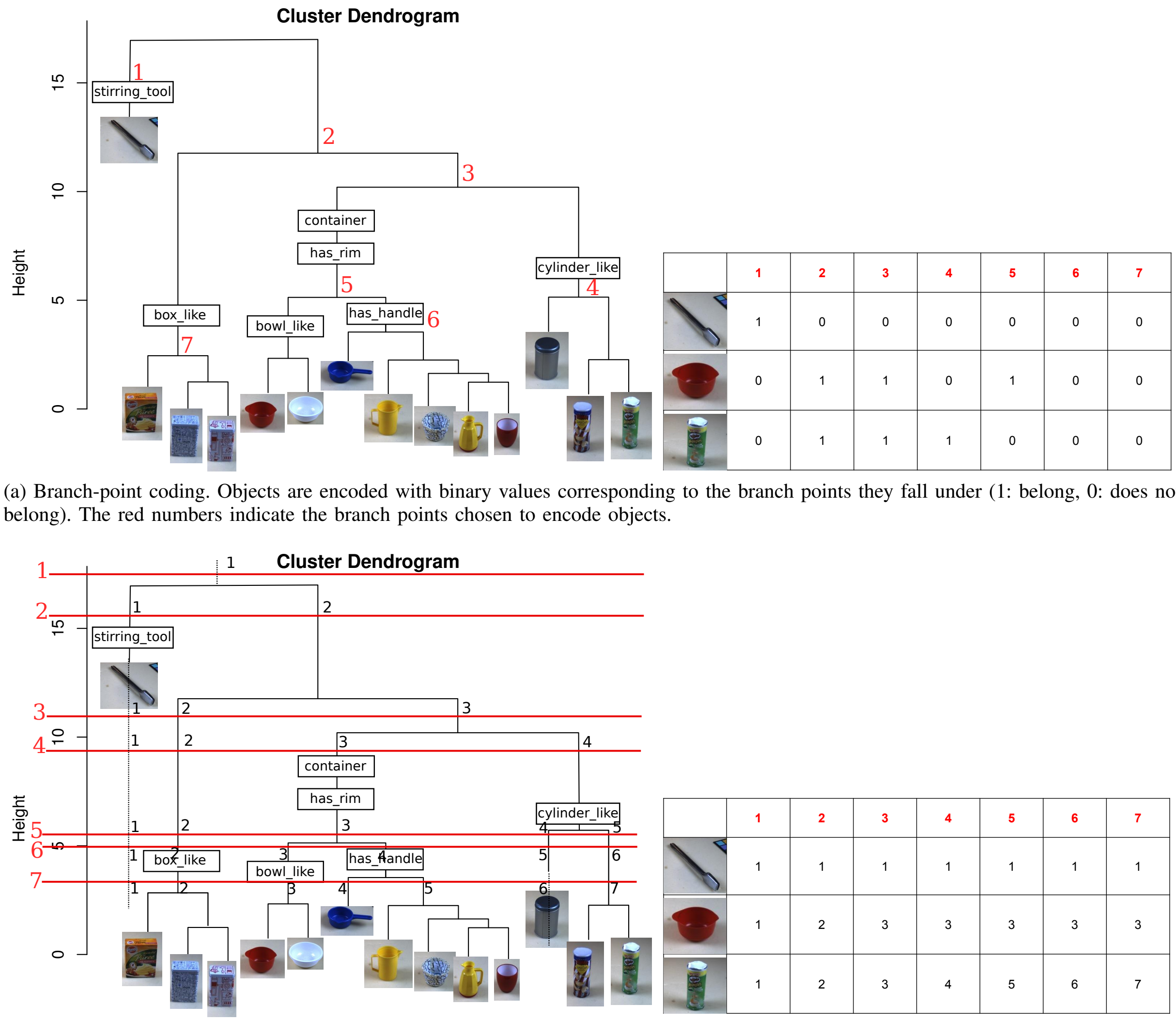
In the prediction phase, we use a method, developed also in [21], that assigns clusters (among the clusters extracted previously during training) to novel objects. Essentially, the method identifies where the novel objects fall in the previously-built hierarchy. The novel objects are then encoded in the same way as for the training objects. The procedure involving these two steps (i.e., generating and predicting clusters) is referred to as *hierarchical cluster extraction*.

To build the hierarchy, the hierarchical clustering algorithm starts by assigning each data sample (object instance in our case) to its own 'atomic' cluster. These atomic clusters are then merged into larger and larger clusters until all clusters are contained in a single cluster at the top of hierarchy (agglomerative approach). The decision on how to merge object samples to form a cluster is based on a dissimilarity measure. We use the Euclidean distance (between pairs of object samples) as dissimilarity measure. Non-atomic clusters (i.e., clusters formed by merging object samples) are merged based on a linkage metric. As a linkage metric, we use Ward's criterion, which aims at minimizing the total within-cluster variance [28]. These two metrics are chosen because they yielded the best results as reported in [21].

C. Encoding Objects using the extracted clusters

From the extracted clusters presented above, objects are encoded according to their associations with these clusters. This coding forms the feature vector fed to the learning algorithm, when the hierarchical clusters are used to describe objects. In the following, we present the two coding schemes: branch-point and level codings.

1) Branch-point coding: In this scheme, the branch points in the hierarchy are regarded as candidate categories.



	1	2	3	4	5	6	7
	1	0	0	0	0	0	0
	0	1	1	0	1	0	0
	0	1	1	1	0	0	0

(a) Branch-point coding. Objects are encoded with binary values corresponding to the branch points they fall under (1: belong, 0: does not belong). The red numbers indicate the branch points chosen to encode objects.

	1	2	3	4	5	6	7
	1	1	1	1	1	1	1
	1	2	3	3	3	3	3
	1	2	3	4	5	6	7

(b) Level coding. Objects are encoded with categorical values corresponding to the branch they fall under (at the specific level). The lines in red indicate the levels chosen to encode objects.

Fig. 3. Hierarchical clustering extraction. An illustrative example on a limited set of objects showing the two schemes used to encode objects.

Hence, objects are encoded with binary digits according to their associations with each point (see Fig. 3a). In the branch-point coding, the structure of the hierarchy is explicitly encoded. Starting from the top, we describe objects using a certain number of branch points. The more points we include the deeper down in the hierarchy we go. The depth of the hierarchy reflects the level of abstraction incorporated in the candidate categories. We found empirically that with about 100 branch points, we reach a non-increasing (steady-state) performance value. In addition, the depth (height) value of the branch points are stored allowing for weighing the feature vectors accordingly.

2) Level coding: This coding scheme was developed particularly to provide a valid input format for the homogeneity analysis approach. This approach works with categorical (discrete and multiple values) features. If we regard each level in the hierarchy as an attribute (Fig. 3b), then the branches within a particular level provide categorical values corresponding to the attribute. Note that at level n, there are n branches (level 1: the top of the hierarchy has 1 branch, level 2 has 2 branches, etc.). In the experiments, we use 100 levels (from level 5 to level 105) resulting in a feature vector of 100 categorical values.

IV. Joint SVM

During the past two decades, support vector machines (SVM) have been popularly employed in various application domains. The successes of SVM mainly originate from its two advantageous components: *maximum margin* and *input kernels*. On one hand, the maximum-margin concept in SVM is an application of the theory of statistical learning [26] on linear binary classification. On the other hand, kernels' role can be considered from two perspectives: *(i)* kernels are designed to capture intrinsic similarities between complex-structured inputs; *(ii)* kernels enable the linear classifier to separate highly non-linear data by mapping them into a reproducing Hilbert space (RKHS). Structural SVM (SSVM) [12] is an extension of SVM for predicting structured outputs. The maximum margin in structural SVM is interpreted as maximizing the score gap between the desired output and the first runner-up. Meanwhile, this results in exponential complexity in solving structural SVM, which limits its applicability to only small scale problems.

Joint SVM was developed with a special focus on the interdependencies within outputs. Essentially, Joint SVM is equivalent to SSVM with a linear output kernel plus a regularization term on the kernel [30]. Therefore, a linear kernel on outputs is automatically learned to capture the interdependencies within outputs. Furthermore, if prior knowledge about the interdependencies is available, a user-specified output kernel can be straightforwardly mounted in Joint SVM as well. In both cases, the computation complexity of Joint SVM is almost the same as a single SVM, in contrary to the exponential complexity in structural SVM.

In the experiments, the Joint SVM takes the histogram descriptors, discussed in Subsect. III-A, as input. Moreover, we investigate the use of the hierarchical clusters (Subsect. III-B) as object descriptors. In this case, objects are encoded using the branch-point scheme see Subsect. III-C and Fig. 3a. As input kernels, we chose polynomial kernels based on initial tests. In addition, we introduce a 'weighted' polynomial kernel to weigh the object according to the depth of the hierarchical clusters. The depth corresponds to the similarity (or the level of abstraction in this case). The estimation of the kernel parameters is embedded in a cross-validation step (which also includes the estimation of the internal parameters of the Joint SVM) performed prior to training.

V. Comparison methods

In this section, we briefly describe the methods used for comparison with Joint SVM in the conducted experiments.

A.Homogeneity analysis

Homogeneity analysis [20] is a multivariate technique for categorical data. Basically, it provides a mapping from multivariate categorical data into a lower-dimensional homogeneous Euclidean space. Based on this method, an approach for learning object action relationship was developed in [29]. The approach performs further reasoning to find the dependencies between objects and categories. In [29], homogeneity analysis was used in a synthetic database in which categories of the test subset are partially known and the goal is to retrieve the missing categories. In this work however, to make it comparable, all categories in the test subset are set to be unknown. When testing this approach, objects are described in terms of the extracted clusters using the level coding scheme (Fig. 3b). Note that these clusters exhibit dependencies between each other due to their hierarchical nature. This may pose a challenge to this approach especially given it is based on linear optimization.

B.Hierarchical cluster matching

Hierarchical cluster matching was introduced in [21] based on the hierarchical cluster extraction. The method performs object category recognition in a supervised fashion (i.e., using labeled data). In [21], this method was shown to outperform other methods (namely, RFs and HMP [7]). The method finds the branches in the hierarchy that best match the categories in the labeled data. The matching is performed using Jaccard's index [17] as a similarity metric. The Jaccard's index rewards the existence of the object in the prospective cluster and also punishes for the absence thereof. This prohibits assigning categories to very specific (at the bottom of the hierarchy) or very generic clusters (at the top of the hierarchy).

C.Random Forests

Random forests (RF) [8] learn a collection of randomized decision trees from different random subsets of the available training data, in a manner similar to *Bagging*. They have been found to be efficient because they combine the simplicity of decision trees with the stability of voting methods. Random Forests is a single-label method, therefore, we apply the method in N-classifier mode (i.e. 1-versus-all) where N refers to the number of categories.

D.Hierarchical Matching Pursuit (HMP)

HMP [7] is a multi-layer sparse coding network that builds feature hierarchies layer by layer with an increasing receptive field size to capture abstract representations from raw RGB-D data. The visual features extracted by the network are used as input to SVM classifiers. In the experiment, the HMP serves a state-of-the-art base-line method, particularly for the visual representation we use. Note that HMP was not designed to combine features from different views in 3D space. Therefore, to make it comparable to the multi-view system used here, we provide all three views to HMP in the training phase. This method is also operated in N-classifier mode (1-versus-all) by training N SVM classifiers corresponding to N categories.

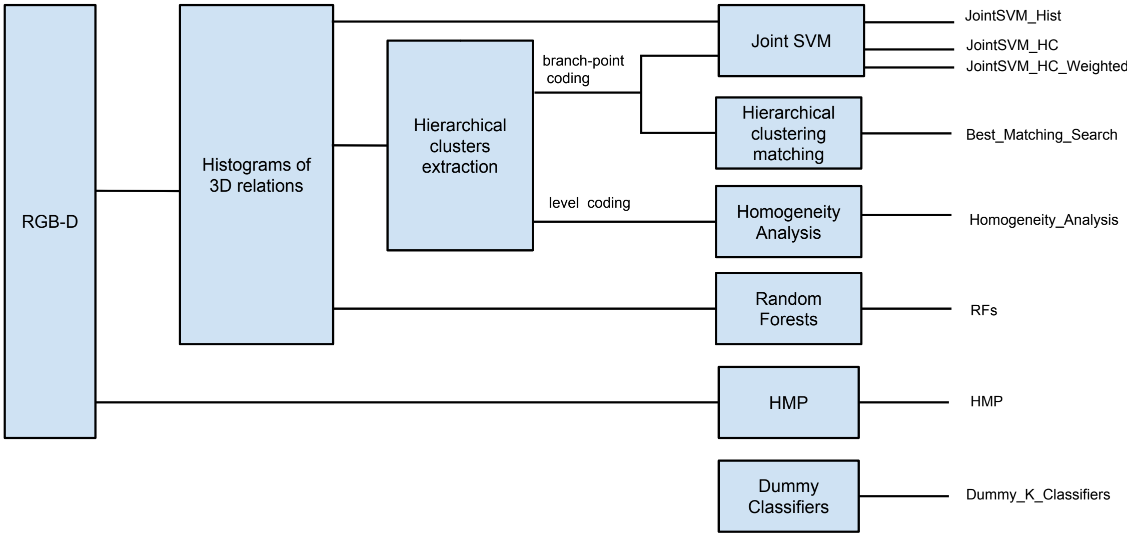


Fig. 4. The methods compared in the experiments and their relations to the visual representations. The labels in the right correspond to the labels shown in the result figure (Fig. 5).

VI. Experiments

In the following experiments, we study the performance of each method for categorizing novel objects. The experiments were performed on a large multi-view object dataset[1] [22]. The dataset contains 100 objects with 30 different samples (random poses) per object (i.e. 3000 object samples from 9000 RGB-D data samples). Object samples are captured using three different Kinect sensors mounted in a close to equilateral triangular configuration (the relative transformations of the calibrated sensors are also provided, allowing for fusing the views in 3D space). The selection of objects covers a wide range including industrial and household objects, some of them taken from the KIT dataset [13].

Object are labeled with human-defined labels with one or multiple (potentially nested) categories (see Fig. I for examples). In all experiments, the dataset is divided into training and test subsets where sampling is performed in a way that prohibits the presence of samples from the same object in both subsets. This prohibition is necessary to ensure the novelty of the test objects. The size of the test subset is set to 100 samples per category whereas the size of the training subset is allowed to vary from two to 100—all samples are randomly chosen. Each experiment is executed 20 times from which the average F1 score and the standard deviation are computed. Note that the same training and test subsets are passed to each method.

A. Methodology for comparison

Fig. 4 shows the different test cases carried out in the experiments. The labels on the far right of the figure represent the following test cases:

- JointSVM_Hist: In this case, objects are described with histograms. For learning, the Joint SVM method is used with polynomial input kernels.
- JointSVM_HC: Objects are described in terms of the extracted hierarchical clusters using the branch-point coding scheme (Fig. 3a). Similar to the case above, polynomial kernels are used as input kernels for the Joint SVM.
- JointSVM_HC_Weighted: In this case, objects are also described using branch-point coding, however, the clusters are weighted according to their depth in the hierarchy. For this, we use weighted polynomial kernels as input kernels for the Joint SVM.
- Homogeneity_Analysis: In this case, objects are described in terms of the extracted clusters but using the level coding scheme (Fig. 3b). For learning, the method based on the homogeneity analysis is used (Subsect. V-A)
- Best_Matching_Search: In this case, we use the hierarchical cluster matching method in which objects are described in terms of the hierarchical clusters (Subsect. V-B).
- RFs: Using the Random Forests method (as 1-versus-all) in which objects are described with histograms.
- HMP: In this case, we use the HMP method (see Subsect. V-D) in which the SVM classifiers are trained in 1-versus-all fashion. The method takes the RGB-D

[1] http://caro.sdu.dk/sdu-dataset/

data as input and extracts its own visual features.

- Dummy_K_Classifiers: These classifiers generate uniformly-distributed random categories. These classifiers are useful as baseline for good performance. This is important given that the definition of the categories involves human judgment and that, presumably, no distinctive features exist for some categories.

B.Result and Discussion

Fig. 5 shows the performance of object categorization on 13 categories. Each sub-figure shows the average F1 score and the standard deviation for a varying number of training samples. The average performance on all categories is also shown in a separate sub-figure.

The results show that using the Joint SVM method (whether objects are described with histograms or with hierarchical clusters) leads to achieving the best performance in all of the test categories. Another observation is that using histograms of global relations leads generally to learning that is at least as fast as using the features extracted by the HMP method. Indeed for the "box_like" category, it is obvious that the use of histogram yields faster learning.

Regarding categorizing nested (depended) categories, e.g., "has_rim" and "container", we observe that using multi-label leaning approaches (Joint SVM, Homogeneity Analysis, Hierarchical Clustering Matching) leads to success (largely outperform the dummy classifier) whereas other methods (HMP, RandomForests) fail. In certain categories however, namely, "pour" and "grasp-open", we observe relatively low performance (comparable with the dummy classifiers) for all methods. This indicates that no distinctive visual features could be found.

The results also show that using Joint SVM with the histograms descriptors (i.e. the JointSVM_Hist case) leads to marginally higher performance than the cases in which objects are described with the hierarchical clusters (i.e. JointSVM_HC, JointSVM_HC_Weighted). This means that no improvement is gained by extracting hierarchical clusters. In addition, the homogeneity analysis approach performs poorly (in terms of F1 score and stability) compared to the Joint SVM. This may be because the homogeneity analysis approach involves linear transformations and thus doesn't, as opposed to Joint SVM, deal with the high dimensionality of the representation

VII. Conclusion

In this paper, we presented multiple object categorization methods capable of assigning multiple and nested categories using multi-label learning approaches exploiting descriptors derived from global 3D features. Specifically, the paper introduced and investigated the use of the Joint SVM method to learn visual categories. We evaluated the advantage of extracting hierarchical clusters to describe objects. Comparisons with other multi-label learning approaches as well as single-label approaches (including a state-of-the-art methods using different object descriptors) were also performed.

The results show that using multi-label learning, we are able to recognize multiple and dependent categories. This indicates that multi-label learning is able to identify and exploit the relations between the dependent categories whereas single-label approaches try to learn discriminatively the individual categories. Another aspect indicated by the results is that using histograms of global relations leads to fast learning in terms of the number of samples required for training. This is important when the training data is limited or hard to obtain.

The results presented in this paper showed a promising approach for categorizing objects belonging to a wide range of categories that may exhibit dependencies between them. One limitation of this work is that the evaluation process involves human subjectivity in defining the categories and labeling the data accordingly. However, this work and the results obtained can be used to establish a foundation for building a system in which objects are labeled automatically according to their everyday functionality. Such labels may be derived from actual robot actions. For example, if a robot is able to roll an arbitrary object, then this object is labeled as "rollable", and so forth. In such kind of scenarios, using the approach proposed in this paper, multiple and dependent actions may be learned by experience allowing for proposing acute actions on novel objects.

Acknowledgment

This work has been funded by the EU project Xperience (FP7-ICT-270273).

References

[1] L. A. Alexandre. 3d descriptors for object and category recognition: a comparative evaluation. In *Workshop on Color-Depth Camera Fusion in Robotics at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vilamoura, Portugal*, volume 1. Citeseer, 2012. 2

[2] A. Andreopoulos and J. K. Tsotsos. 50 years of object recognition: Directions forward. *Computer Vision and Image Understanding*, 117(8):827–891, 2013. 1

[3] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(4):509–522, 2002. 2

[4] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(8):1798–1828, 2013. 2

[5] I. Biederman. Recognition by components: A theory of human image understanding. *Psychological Review*, 94(2):115–147, 1987. 2

[6] T. O. Binford. Visual perception by computer. In *IEEE conference on Systems and Control*, volume 261, page 262, 1971. 2

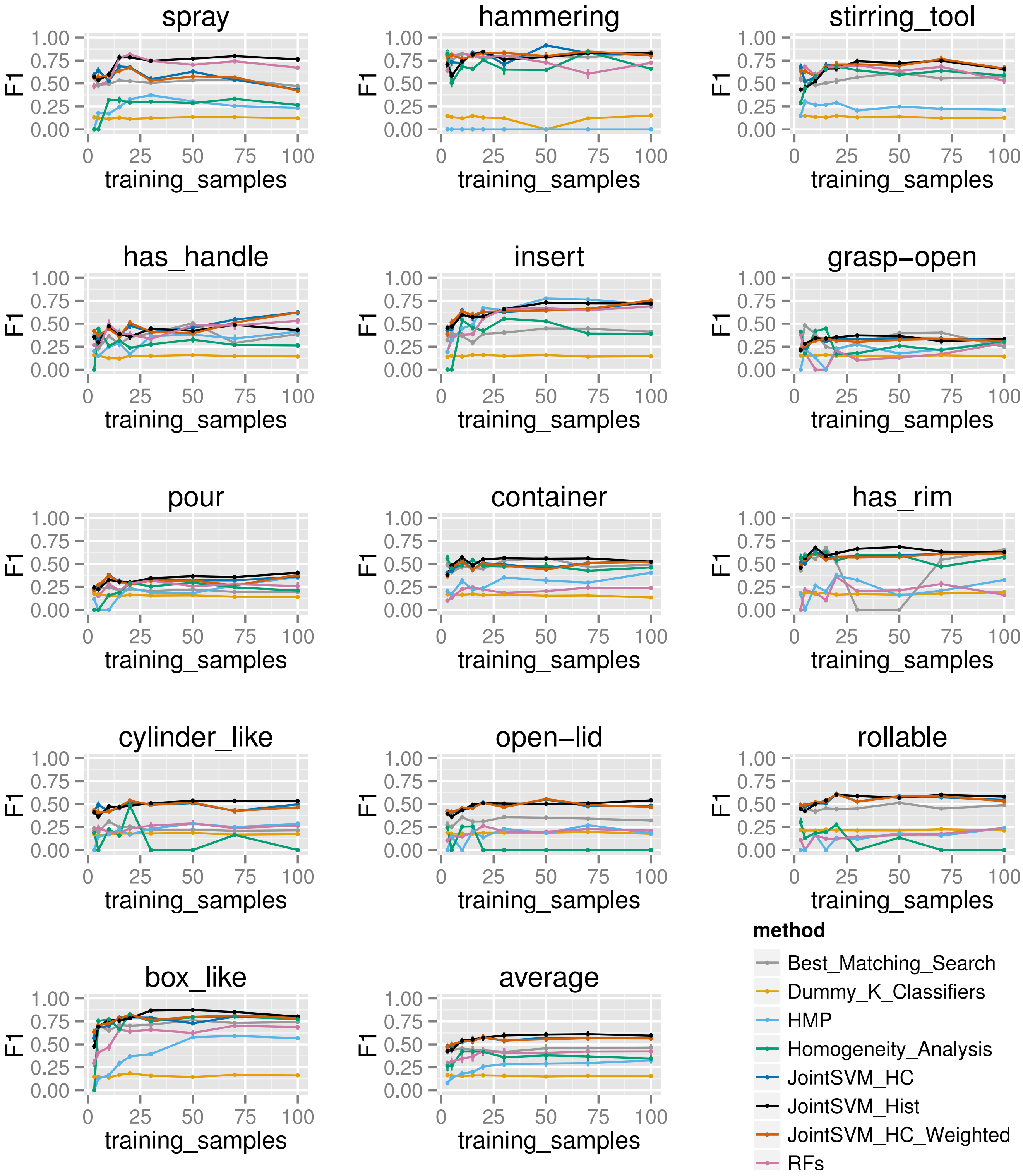


Fig. 5. The performance of object categorization on 13 categories. The average performance is also shown. The labels correspond to the method shown in Fig. 4.

[7] L. Bo, X. Ren, and D. Fox. Unsupervised feature learning for rgb-d based object recognition. In *International Symposium on*

Experimental Robotics (ISER), 2012. 2, 5

[8] L. Breiman. Bagging predictors. *Mach. Learn.*, 24(2):123–140, Aug. 1996. 5

[9] R. Campbell and P. Flynn. A survey of free-form object representation and recognition techniques. *Computer Vision and Image Understanding*, 81(2):166–210, 2001. 2

[10] A. Frome, D. Huber, R. Kolluri, T. Bulow, and J. Malik. Recognizing objects in range data using regional point descriptors. In *Proceedings of the European Conference on Computer Vision (ECCV)*, May 2004. 2

[11] S. Graham and D. Wood. Digitizing surveillance: categorization, space, inequality. *Critical Social Policy*, 23(2):227–248, 2003. 1

[12] T. Joachims, T. Finley, and C.-N. Yu. Cutting-plane training of structural svms. *Machine Learning*, 77(1):27–59, 2009. 5

[13] A. Kasper, Z. Xue, and R. Dillmann. The kit object models database: An object model database for object recognition, localization and manipulation in service robotics. *International Journal of Robotics Research (IJRR)*, 31(8):927–934, 2012. 6

[14] D. Kraft, W. Mustafa, M. Popović, J. Jessen, A. G. Buch, T. R. Savarimuthu, N. Pugeault, and N. Krüger. Using surfaces and surface relations in an early cognitive vision system. *Machine Vision and Applications*, 2015. 3

[15] K. Lai, L. Bo, X. Ren, and D. Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1817–1824, 2011. 2

[16] A. Laika and W. Stechele. A review of different object recognition methods for the application in driver assistance systems. In *Image Analysis for Multimedia Interactive Services, Eighth International Workshop on*, pages 10–10. IEEE, 2007. 1

[17] M. Levandowsky and D. Winter. Distance between sets. *Nature*, 234(5323), 1971. 5

[18] D. Lowe. Object recognition from local scale-invariant features. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1150 –1157 vol.2, 1999. 2

[19] Z.-C. Marton, D. Pangercic, R. B. Rusu, A. Holzbach, and M. Beetz. Hierarchical object geometric categorization and appearance classification for mobile manipulation. In *Humanoid Robots (Humanoids), 2010 10th IEEE-RAS International Conference on*, pages 365–370. IEEE, 2010. 1, 2

[20] G. Michailidis and J. de Leeuw. The gifi system of descriptive multivariate analysis. *Statistical Science*, pages 307–336, 1998. 2, 5

[21] W. Mustafa, D. Kraft, and N. Krüger. Extracting categories by hierarchical clustering using global relational features. In *Pattern Recognition and Image Analysis*, volume 9117 of *Lecture Notes in Computer Science*, pages 541–551. Springer International Publishing, 2015. 2, 3, 5

[22] W. Mustafa, N. Pugeault, A. G. Buch, and N. Krüger. Multi-view object instance recognition in an industrial context. *Robotica*, FirstView:1–22, 8 2015. 3, 6

[23] W. Mustafa, N. Pugeault, and N. Krüger. Multi-view object recognition using view-point invariant shape relations and appearance information. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2013. 2

[24] A. P. Pentland. Perceptual organization and the representation of natural form. *Artificial Intelligence*, 28(3):293–331, 1986. 2

[25] D. K. Prasad. Survey of the problem of object detection in real images. *International Journal of Image Processing (IJIP)*, 6(6):441, 2012. 1

[26] V. Vapnik. *The nature of statistical learning theory*. Springer Science & Business Media, 2000. 5

[27] E. Wahl, U. Hillenbrand, and G. Hirzinger. Surflet-pair-relation histograms: a statistical 3d-shape representation for rapid classification. In *3-D Digital Imaging and Modeling, 2003. 3DIM 2003. Proceedings. Fourth International Conference on*, pages 474–481. IEEE, 2003. 2

[28] J. H. Ward. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301):236–244, 1963. 2, 3

[29] H. Xiong, S. Szedmak, and J. Piater. Homogeneity analysis for object-action relation reasoning in kitchen scenarios. In *Proceedings of the 2Nd Workshop on Machine Learning for Interactive Systems: Bridging the Gap Between Perception, Action and Communication*, MLIS '13, pages 37–44, New York, NY, USA, 2013. ACM. 2, 5

[30] H. Xiong, S. Szedmak, and J. Piater. Implicit Learning of Simpler Output Kernels for Multi-Label Prediction. In *NIPS workshop on Representation and Learning for Complex Outputs*, 2014. 5

[31] H. Xiong, S. Szedmak, and J. Piater. Scalable, accurate image annotation with joint {SVMs} and output kernels. *Neurocomputing*, 169:205 – 214, 2015. Learning for Visual Semantic Understanding in Big Data. Industrial Data Processing and Analysis. Selected papers from the 22nd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2014). Selected papers from the 11th World Congress on Intelligent Control and Automation (WCICA2014). 2

[32] M.-L. Zhang and Z.-H. Zhou. A review on multi-label learning algorithms. *Knowledge and Data Engineering, IEEE Transactions on*, 26(8):1819–1837, Aug 2014. 1, 2