Generating Knowledge-Enriched Image Annotations for Fine-Grained Visual Classification

Francesca Murabito(✉), Simone Palazzo,
Concetto Spampinato, and Daniela Giordano

Pattern Recognition and Computer Vision (PeRCeiVe Lab),
Department of Electric Electronic and Computer Engineering,
University of Catania, Catania, Italy
{fmurabito,palazzosim,cspampin,dgiordan}@dieei.unict.it

Abstract. Exploiting high-level visual knowledge is the key for a great leap in image classification, in particular, and computer vision, in general. In this paper, we present a tool for generating knowledge-enriched visual annotations and use it to build a benchmarking dataset for a complex classification problem that cannot be solved by learning low and middle-level visual descriptor distributions only. The resulting *VegImage* dataset contains 3,872 images of 24 fruit varieties, over than 60,000 bounding boxes (portraying the different varieties of fruits as well as context objects such as leaves, etc.) and a large knowledge base (over 1,000,000 OWL triples) containing a-priori knowledge about object visual appearance. We also tested existing fine-grained and CNN-based classification methods on this dataset, showing the difficulty of purely visual-based methods in tackling it.

1 Introduction

Object recognition and image classification have been hot research topics in the last two decades. Recently, deep-learning methods have been able to achieve impressive performance on thousands of object classes from the ImageNet dataset. In spite of such progress, classification approaches are still predominantly based on visual features, leveraging the power of statistical machine learning to learn distributions of low and middle-level features. While this has proved to be an effective strategy even for fine-grained classification problems [13,17,31], there are cases where relying on visual appearance only might fail, especially in specialized application domains (such as fruit variety identification). For example, Fig. 1 (left image) shows four different varieties of cherry (namely, bing, black tartarian, burlat and lapin) that cannot be easily identified by only exploiting statistical distribution of visual descriptors. However, objects in the "real-world" do not appear as isolated items, but come in a rich context (the right-hand image in Fig. 1 shows the same cherry varieties in their natural context), which is largely exploited by humans for visual categorization.

S. Battiato et al. (Eds.): ICIAP 2017, Part I, LNCS 10484, pp. 332–344, 2017.
https://doi.org/10.1007/978-3-319-68560-1_30

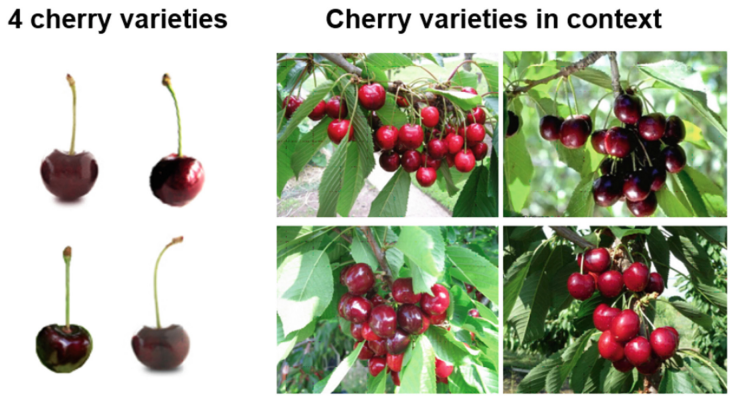

Fig. 1. Example of fine-grained problem tackled in this paper. Left: Four different cheery varieties, namely (left to right, top to bottom), bing, black tartarian, burlat and lapin. Right: The same varieties in their natural context. Information about leaf shape, distance between fruit and tree branches, peduncle length may support the disambiguation between the four object classes.

Our main assumption is that, for a real breakthrough in computer vision, computers need to emulate human visual process by combining perceptive elements (visual descriptors) and cognitive factors (structured knowledge). Such combined perceptive-cognitive knowledge can be then exploited to solve complex visual recognition tasks when low-level visual description fails to express the differences among classes. However, while it is relatively easy to describe visually images, e.g., by identifying variations in shapes, colors, etc., it is more challenging and complex to annotate images according to specific knowledge as the ones depicted in Fig. 1, which only experts, making use of domain knowledge, would be able to do. Nevertheless, domain experts often do not wish to spend time to provide image annotations, so *how can we generate knowledge-enriched visual annotations necessary to train machine learning techniques?*. This paper aims at addressing the above question, specifically through:

- An annotation tool which guides the visual annotation process according to specialized domain knowledge model defined as a formal ontology, and which allows non-experts to generate large-scale domain-specific annotations.
- A knowledge-enriched fine-grained image dataset for fruit variety classification, which is hard to solve with typical visual-oriented approaches (e.g., GoogLeNet, Overfeat, VLFeat PHOW, KDES) without the use of domain knowledge.

2 Related Work

The goal of this paper is three-fold: *(a)* proposing a new semantic annotation tool driven by *(b)* domain knowledge through a formal ontology for *(c)* generating

a fine-grained image dataset enriched with a large knowledge base. The importance of visual world semantics (and of context especially) in automated visual recognition has been long acknowledged [6,14]. Recently, there have been significant advances in modeling rich semantics using contextual relationships [18,25] such as object-object [6,20] and object-attribute [9,16] applied to scene classification [12] or object recognition [27]. In [27], the authors proved that context information is more relevant for object recognition than the intrinsic object representation, especially in cases of poor image quality (e.g., blurred images due to large distances, occlusions, illumination, shadows). However, visual scenes provide richer semantics than object-object or object-attribute relationships, which most of the existing methods do not take into account or do not exploit effectively as they try to solve the recognition problem by brute force. Nevertheless, one of the limitations to a larger use of high-level knowledge in computer vision is the lack of structured resources modeling exhaustively the semantics of our world. Indeed, so far, the largest resource of structured visual knowledge is the ImageNet dataset that, however, captures only limited semantic relations, ignoring, for instance, co-occurrence, dependency, mutual exclusion. The need for exhaustive knowledge is also highlighted by the recent sprout of methods employing high-level knowledge (mainly unstructured) for computer vision tasks: knowledge transfer methods [10,15] and semantic relation graphs [4,21] have been adopted to deal with the limits of traditional multi-class or binary models, which suffer from being overly restrictive or overly relaxed, respectively. Compared to scene graphs, computational ontologies are able to describe deeper scene semantics by defining high-level attributes and imposing constraints about real-world object appearance and their contextual and semantic relations, in an interoperable and generalizable way.

However, including high-level knowledge in the learning loop needs large semantically-annotated visual datasets, whose generation is an expensive process: beside annotating objects in images, other semantic information, such as color, shape, related-objects and their visual properties, etc., needs to be collected. This, especially, holds in specialized application domains (e.g., fruit variety, bird, medical images, etc.) where high precision is necessary to avoid affecting the learning process. In such cases, annotations should be provided by domain experts, who do not have enough time to spend into the process. To tackle this problem, one possible solution is to extract and use domain-knowledge to guide/constrain annotations done by non-expert users. So far, only few ontology-based image annotation tools have been proposed [3,7,19], which are, however, mainly thought for information retrieval rather than for computer vision.

Our proposed tool differs from the above ones and traditional tools [8,22] in that it constrains and guides the annotation process according to specific domain knowledge (codified as a formal ontology) where the visual attributes are inferred through ontology reasoning, thus reducing greatly the knowledge required to carry out the task.

We used our tool to generate knowledge-enriched visual annotations on fruit variety images, thus providing a complex benchmark for fine-grained recognition.

There are several benchmarking datasets for fine-grained classification of birds, stonefly larvae, etc. [5,17,31] but they mainly contain per-instance segmentations and do to provide any semantic visual descriptions of objects and their context. The datasets most similar to ours are the ones for semantic scene labeling [2,24], which, despite including context information, no exhaustive semantic relations are defined.

3 Generating Knowledge-Enriched Visual Annotations

In this section we present a formal OWL ontology encoding specialized knowledge for fruit variety categorization. The combination of such ontology with a tool able to guide and constrain the annotation process allows to minimize expert user intervention, thus providing the chance to create large-scale fine-grained annotations by involving mainly non-expert users.

3.1 The Fruit Ontology

An ontology is a formalism providing, for a specific domain, a common machine-processable vocabulary and a formal agreement about the meaning of the used terms, which include important concepts, their properties, mutual relations and constraints. Basic concepts of a domain correspond to *owl:Class*, whose expressiveness can be enhanced by adding attributes (as *owl:DataProperty*) and relations to other *owl:Class* (as *owl:ObjectProperty*). The vocabulary is designed and validated by human users through axioms expressed in a logic language and the concepts and properties can be enriched using natural language descriptions[1] and links (e.g. *rdfs:seeAlso* property).

We developed a new ontology describing visually the fruit application domain by involving three expert agronomists, who also supported the generation of correct instances for the considered fruit varieties. Figure 2 shows the ontology's VOWL (Visual OWL) representation and some statistics generated using Protégé[2]. We embedded this ontology in an annotation tool to speed up the labeling process, making annotation of domain-specific images accessible for non-expert users (see Sect. 3.2). Before describing the *Fruit Ontology*, let us introduce some terminology to avoid ambiguities. We refer to an *owl:Class* as an *ontology class*, and to an image class (i.e., a fruit variety) as a *dataset class*. Furthermore, we use *target class* to indicate the main object class we want to recognize (in our case *Fruit*), and *context class* for all the object classes that can be considered as part of the context (in our case, *Peduncle*, *Leaf*, *Petiole*) of the *target class*. Typically, *target classes* are objects which are spatially well-defined, easily-recognizable and possibly not a constituent part of a larger object (e.g., a dog rather than its tail, a fruit rather than its peduncle). *Context classes* are, instead, those that either are not classification targets or are more easily identified in relation to a target class. The Fruit Ontology contains two main class

[1] http://www.w3.org/2005/Incubator/mmsem/XGR-image-annotation/.
[2] http://protege.stanford.edu/.

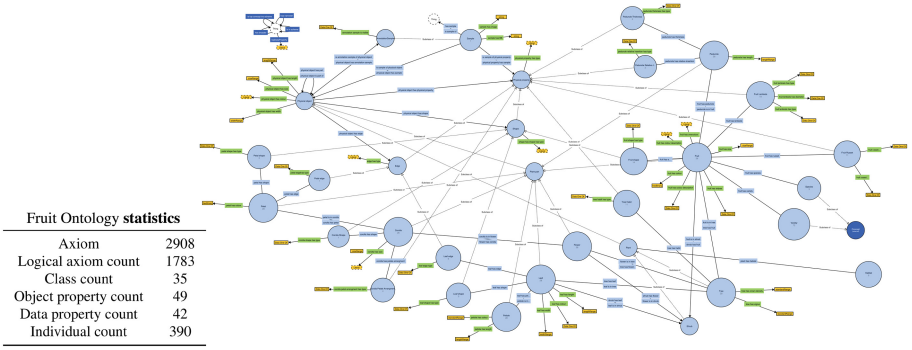

Fig. 2. The fruit ontology VOWL representation. High resolution image: zoom in, too see classes and properties.

categories: the ones for visually describing target and context objects, and the ones needed for the annotation process.

Application Domain Classes and Properties. Domain classes and properties encode *a-priori* and expert knowledge on fruit varieties in terms of both visual appearance (colors, shape, edges, etc.) and their context relations (with *Leaf*, *Peduncle*, *Petiole*, etc.). Three expert agronomists supported us in the ontology design process by identifying for each target class (i.e., *Fruit*), the set of related context classes and the visual features describing their appearance. Both the target and context classes were mapped to ontology classes (*Fruit*, *Leaf*, *Peduncle*, *Petiole* are defined as subclasses of a domain-agnostic *PhysicalObject* class) and were enriched with as many *owl:DataProperty* (e.g., *fruitHasStripes*, *fruitHasColourDescription*, *fruitHasOvercolourDescription*, etc.) as needed to represent their visual appearance. Most of the physical features are defined as classes themselves (e.g. *Shape*, *Edge*, *FruitRusset*, etc., similarly defined as subclasses of *PhysicalProperty*) for defining more articulated visual characteristics (e.g., *fruitRussetHasDistribution*, *fruitRussetHasType*, etc.).

Ontology classes mapping target or context objects only differ in that target classes include the relations *fruitHasSpecies* and *fruitHasVariety* (easily generalizable to other domains) to *Species* and *Variety* ontology classes, which in turn are defined as skos[3] concepts in order to include a taxonomy of varieties (each taxonomy term corresponds to a dataset image class). The *physicalObjectHasPart* and its inverse *physicalObjectIsPartOf* and their specialized sub-properties (e.g., *fruitIsInTree*, *fruitHasPeduncle*) are used by the ontology reasoner to infer, starting from the target class and exploiting property transitivity, all ontology classes (e.g. Leaf, Peduncle, Petiole) related to its context.

[3] http://www.w3.org/2004/02/skos/.

Annotation Tool Specific Classes and Properties. The link between user annotations and entities in the ontology is represented by the *AnnotationSample* class, whose properties *hasBB* (for "bounding box") and *hasImage* specify the location of an annotated object in an image. The *AnnotationSample* class is specifically designed to speed up the annotation phase. For each new annotation, an instance of *AnnotationSample* is created and associated with the corresponding *PhysicalObject* subclass instance; this allows the tool to infer all corresponding *PhysicalObject* subclass instance properties encoded into the ontology without the need to specify manually all its properties. Annotator intervention is needed only in cases a property may assume multiple values (e.g., *Russet* for *Canadian Reinette* apple), from which, however, the tool displays samples (also encoded in the ontology) to simplify the labeling work for non-experts (see right-hand side in the bottom part of Fig. 3).

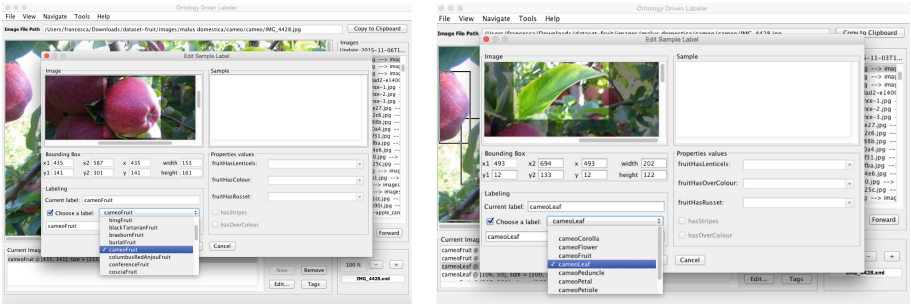

Fig. 3. User interface of the ontology-driven annotation tool. (Left image) Bounding box annotation of a target object (ideally performed by an expert user). (Right image). Annotation of context class objects (e.g., a leaf), with automatically-suggested labels inferred from the one assigned to the bounding box associated to an object of the target class. The right-hand side part is for disambiguating all those properties that can assume multiple values (as per Instance definition) through visual comparison with sample images (also encoded in the ontology instance).

Although the whole annotation schema and representation may seem overly complex (especially if compared to the current "flat-structure" annotations made public by dataset providers), they enable encoding annotations as ontology instances, with one great advantage: the annotation correctness and meaningfulness is implicitly validated, as they have to match the ontology schema.

3.2 The Annotation Tool

The presented ontology-driven annotation tool aims at guiding and constraining users the labeling process within the concepts enforced by the ontology. It basically provides means to draw and assign labels (most of them are automatically inferred through ontology reasoning) to bounding boxes for target and context

classes and to specify attributes for them. Similarly to other annotation tools [8,22], the interface presents the user with an image to work on, together with several tools for browsing through images, zooming in and out, adding, editing and removing annotations. However, unlike those other tools, part of the label assignment responsibility is moved from the user to the tool itself, through a two-phase annotation process. The two phases of the annotation process differ by the degree of expert knowledge required and the amount of annotation work to be carried out. The first annotation phase consists in assigning a dataset image class (e.g., a fruit variety) to each image. This initial task requires expert knowledge necessary to distinguish between dataset classes differing only by subtle details. However, the amount of data to annotate is relatively small, since the user is only asked to draw one bounding box per image and select the corresponding dataset class, thus limiting the expert employment only to a fraction of the whole labeling process. Once annotations have been "bootstrapped" by specifying labels for the bounding boxes containing objects belonging to the target class, the second phase consists of annotating all the other objects present in the image, corresponding (1) to the target class (i.e., the fruit), whose labels are automatically inferred by ontology reasoning, based on the assumption that they are equal to the one provided by experts; and (2) to context classes (i.e., peduncle, leaf, etc.). Annotating bounding boxes of objects related to a context class, while being in general a task which requires expert knowledge, is simplified by the presence of the associated object corresponding to the target class: its label is employed by the tool to infer (through an ontology reasoner[4]) the subset of context class instances which can be used to annotate the current bounding box.

Figure 3 shows how the interface implements the above procedure. Firstly, the (expert) user annotates (left part in Fig. 3) one object related to a target class with the corresponding fine-grained class, by simply drawing a bounding box around the object and selecting its label from a list (dynamically built from the provided domain ontology), e.g. "cameoFruit". Then, the (not necessarily expert) user can continue the process by adding annotations for the other objects in the image, whose labels are inferred based on the target class instance assigned by the expert and on the ontology (right part in Fig. 3). In the example, the inferred labels are "cameoLeaf", "cameoPeduncle", "cameoPetal", since the target class instance was labeled by the expert as "CameoFruit'. As a final consideration, it should be noted that the above process transcends the specific application domain for which the tool is employed, and the concepts to be annotated can be simply configured at setup time by providing a custom ontology and specifying the set of target classes (namely, those for which properties *physicalObjectHasSpecies* or *physicalObjectHasVariety* are defined) and context classes (related to the target classes throughout a series of subproperties of *physicalObjectHasPart* and *physicalObjectIsPartOf*.

[4] http://owlapi.sourceforge.net/.

3.3 The Fruit Image Dataset

The *VegImage* dataset is a collection of 3,872 images of three common fruit species, namely, *malus domestica* (apple), *prunus avium* (cherry) and *pyrus communis* (pear). For each fruit species several fruit varieties were included, 10 for *malus domestica*, 7 for *prunus avium* and 7 for *pyrus communis*. Together with fruit images, we also provide over than 60,000 bounding boxes (depicting the different varieties of fruits, leaves, peduncles, etc.) and a large a knowledge base (over 1,000,000 OWL triples) containing a-priori knowledge about colors, shapes as well as context objects for the considered fruit varieties. A detailed list of fruit varieties is shown in Fig. 4.

Fig. 4. Example images from the fruit image dataset. Numbers in red are number of images per class while in green the number of bounding boxes. (Color figure online)

Dataset Collection. The fruit variety images were mainly downloaded from Google Images, Flickr, ImageNet, Yahoo Images. To increase appearance variability, we also downloaded YouTube documentary videos, from which we manually selected key frames to avoid near duplicates in the dataset. For each of the 27 fruit varieties, about 1,000 images were manually selected to be included in the dataset. Low-quality images or images depicting multiple fruit varieties or people as main subjects were filtered out. After this screening, we asked three expert agronomists to manually check all the resulting images. Thus, we collected up to 500 images for each fruit variety.

Dataset Annotation. We performed a two-stage annotation phase using the tool described in the previous section: (a) **Image labeling**: in this step, the three agronomists annotated each image with a label decided through consensus among them; (b) **Bounding box annotation**: ten non-expert users were asked to draw bounding boxes (a distribution over fruit varieties is given in Fig. 4) for

objects of both target (*Fruit*) and context classes (*Peduncle*, *Leaf* and *Petiole*), and to disambiguate multi-valued attributes defined in the Fruit Ontology (e.g., russet for *Canadian Reinette* apple), which were finally double-checked by the experts, being the only kind of annotations which could be subject to errors.

To test automatically the quality of the generated bounding boxes, we compared them with the ones provided by Selective Search [28]. In detail, for each image we ran selective search (SS) object localization (2,000 object proposals per image) and we computed the maximum Intersection over Union (IoU) index between each annotated bounding box and the ones provided by SS. The average IoU for each fruit class is given in Fig. 5 showing the high-quality of our annotations taking into account also SS failures.

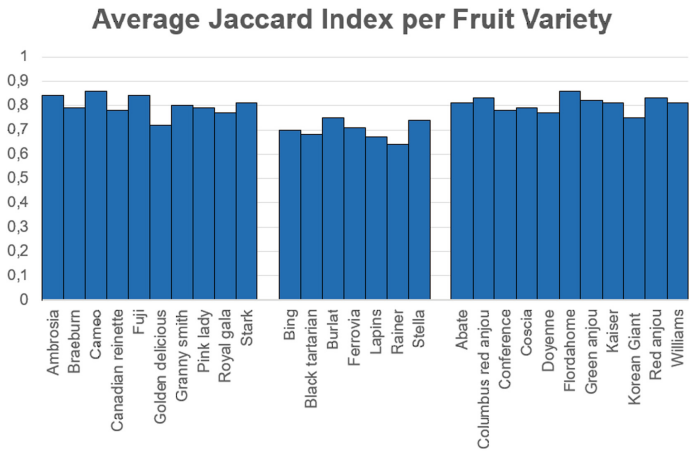

Fig. 5. Average IoU between generated bounding boxes and SS'ones.

Annotation Effort and Times. To test the performance of our annotation tool, we used as evaluation criteria: (1) shifting working time from experts to non-experts, while keeping annotation accuracy high and (2) reducing non-expert annotation time.

Domain experts manually annotated 3,872 fruit images, while over 60,000 bounding boxes were provided by ten non-expert users. Bounding box attributes were inferred automatically by the reasoner (through deductive inference) after the corresponding bounding box class (e.g. *Leaf*) and variety (e.g. *Cameo*) were specified. The annotations of 3,872 fruit images by the three experts took about 13 days (average of 1.3 h per day per expert) for a total of 51 worker hours, while the annotation of 105,284 bounding boxes took about 20 days (average of 4 h per day per annotator). In total, annotating the whole image dataset took 861 worker hours: 810 (about 94% of the total) hours provided by non-experts and the remaining 51 h by experts.

The average annotation time per bounding box for non-experts was 27.7 s, which is impressive given that the Fruit Image Dataset deals with a specific and

complex application domain, and considering that in COCO [11] the annotators spent, on average, about 80 s per bounding box.

As a final note, our tool allows to tackle the issue recently reported in [29], i.e., high-quality annotations on domain-specific applications should be performed, if not by experts, at least by citizen scientists, since unskilled workers perform extremely bad. While this may hold for "traditional" annotation approaches, encoding and incorporating domain knowledge in a tool able to constrain the labeling process is a valid alternative, which allows non-expert annotators to provide high-quality annotations, thus saving significantly expensive resources.

4 Comparison to Existing Datasets for Fine-Grained Recognition

Table 1 compares the proposed Fruit Image dataset with three popular benchmarking datasets for fine-grained image classification: Oxford-IIIT Pet [17], Oxford Flower 102 [13] and Caltech-UCSD Birds [31]. Although the three datasets all have a comparable number of images, the Fruit Image dataset is more complete in the type of annotations it includes, as it contains several examples of images with multiple objects and all objects have associated parts (as context objects) and attributes, beside being enriched with a large knowledge base. Furthermore, although the number of images in the Fruit Image dataset is much smaller than popular image classification (not fine-grained) datasets, e.g., COCO (see Table 1), the number of annotations are comparable, especially since our dataset provides several object annotations per image, completed with bounding box locations, class labels and class-specific attributes. Such achievements would not have been practical if only few experts were asked to perform all annotations; the approach described in Sect. 3.2 allowed us to involve non-experts in a fine-grained annotation process, thus greatly speeding up the whole task.

Table 1. Comparison between popular fine-grained (and not) datasets and our dataset. Key: $\#C$: number of classes; $\#I$: number of images; I/C: average number of images per class; O/I: average number of objects per image; P/O: average number of parts per object; A/O: average number of attributes per object. For our dataset, the O/I value refers to the number of target objects (i.e., fruits), whereas the P/O value counts context objects as object parts; object attributes are the OWL triples, mostly inferred by ontology reasoning, and only a tiny part manually annotated.

	#C	#I	I/C	O/I	P/O	A/O
PET	37	7,349	198.6	1.0	0.0	0.0
Flower_102	102	8,189	80.3	1.0	0.0	0.0
Birds	200	11,788	58.9	1.0	12.0	31.5
COCO	80	123,287	1,541.1	7.3	–	–
Fruit Image	24	3,872	161.3	8.0	1.14	11.0

Table 2. Results obtained by VLFeat PHOW, KDES, OverFeat and GoogleNet on the proposed dataset and on three other fine-grained datasets.

Dataset	Method			
	VLFEAT	KDES	OverFeat	GoogleNet
Oxford-IIIT Pet	39.25%	45.47%	70.48%	86.14%
Oxford Flower 102	56.68%	24.63%	79.02%	90.04%
Caltech-UCSD Birds	14.62%	7.11%	59.2%	70.2%
Fruit Dataset	4.21%	24.4%	26.6%	36.1%

In order to test the complexity of the proposed dataset, we evaluated four state-of-the-art classification methods on these four datasets: VLFeat PHOW [30], KDES [1], OverFeat [23] and GoogleNet [26]. The comparison, in terms of mean classification accuracy (see Table 2) shows that the tested algorithms fail to tackle the proposed dataset. We believe that a cause for this failure is that, unlike current fine-grained datasets, the proposed fruit dataset describes an application domain where class discrimination is strongly based on a context dependency between objects, which needs to be encoded and integrated into the classification methods as *a priori* information.

5 Conclusions

In this paper we present a knowledge-driven annotation tool which exploits specialized domain knowledge to generate semantic fine-grained annotations, greatly reducing the efforts of domain experts, for classification problems that cannot be solved by using only low and middle-level features. The tool was used by three expert agronomists to provide high-level and coarse annotations and by ten non-expert users who provided fine-grained annotations without any knowledge on the application domain. The resulting *VegImage* dataset contains 3,872 images, over than 60,000 bounding boxes, and over than 1,000,000 OWL triples, representing, to the best of our knowledge, one of the most comprehensive resources for fine-grained classification and one the most exhaustive knowledge bases in computer vision. As future work, we are working on building semantic machine learning classifiers integrating classic learning methods with reasoning approaches able to convert a set of detections into an ontology instance describing the application domain to be matched against correct instances as provided by domain experts. The annotation tool, the image dataset, the knowledge base, and the Fruit ontology will be made publicly available.

References

1. Bo, L., Ren, X., Fox, D.: Kernel descriptors for visual recognition. In: NIPS (2010)
2. Brostow, G.J., Fauqueur, J., Cipolla, R.: Semantic object classes in video: a high-definition ground truth database. Pattern Recogn. Lett. **30**(2), 88–97 (2009)

3. Dasiopoulou, S., Giannakidou, E., Litos, G., Malasioti, P., Kompatsiaris, Y.: A survey of semantic image and video annotation tools. In: Paliouras, G., Spyropoulos, C.D., Tsatsaronis, G. (eds.) Knowledge-Driven Multimedia Information Extraction and Ontology Evolution. LNCS, vol. 6050, pp. 196–239. Springer, Heidelberg (2011). doi:10.1007/978-3-642-20795-2_8
4. Deng, J., et al.: Large-scale object classification using label relation graphs. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8689, pp. 48–64. Springer, Cham (2014). doi:10.1007/978-3-319-10590-1_4
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 CVPR, pp. 248–255 (2009)
6. Galleguillos, C., Belongie, S.: Context based object categorization: a critical survey. Comput. Vis. Image Underst. **114**(6), 712–722 (2010)
7. Halaschek-Wiener, C., Golbeck, J., Schain, A., Grove, M., Parsia, B., Hendler, J.A.: PhotoStuff-an image annotation tool for the semantic web. In: 2005 International Semantic Web Conference (2005)
8. Kavasidis, I., Palazzo, S., Salvo, R., Giordano, D., Spampinato, C.: An innovative web-based collaborative platform for video annotation. MTAP **70**(1), 413–432 (2014)
9. Lampert, C., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: CVPR 2009, pp. 951–958 (2009)
10. Lampert, C., Nickisch, H., Harmeling, S.: Attribute-based classification for zero-shot visual object categorization. IEEE PAMI **36**(3), 453–465 (2014)
11. Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). doi:10.1007/978-3-319-10602-1_48
12. Martin, D., Fowlkes, C., Malik, J.: Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE PAMI **26**(5), 530–549 (2004)
13. Nilsback, M.E., Zisserman, A.: Automated flower classification over a large number of classes. In: 2008 Indian Conference on Computer Vision, Graphics and Image Processing (2008)
14. Oliva, A., Torralba, A.: The role of context in object recognition. Trends Cogn. Sci. **11**(12), 520–527 (2007). (Regul. Ed.)
15. Oquab, M., Bottou, L., Laptev, I., Sivic, J.: Learning and transferring mid-level image representations using convolutional neural networks. In: 2014 CVPR (2014)
16. Parikh, D., Grauman, K.: Relative attributes. In: ICCV 2011. vol. 0, pp. 503–510. IEEE Computer Society, Los Alamitos, CA, USA (2011)
17. Parkhi, O.M., Vedaldi, A., Zisserman, A., Jawahar, C.V.: Cats and Dogs. In: CVPR (2012)
18. Patterson, G., Hays, J.: Sun attribute database: discovering, annotating, and recognizing scene attributes. In: 2012 CVPR (2012)
19. Petridis, K., Anastasopoulos, D., Saathoff, C., Timmermann, N., Kompatsiaris, Y., Staab, S.: M-OntoMat-Annotizer: image annotation linking ontologies and multimedia low-level features. In: Gabrys, B., Howlett, R.J., Jain, L.C. (eds.) KES 2006. LNCS, vol. 4253, pp. 633–640. Springer, Heidelberg (2006). doi:10.1007/11893011_80
20. Rabinovich, A., Vedaldi, A., Galleguillos, C., Wiewiora, E., Belongie, S.: Objects in context. In: 2007 ICCV (2007)
21. Ramanathan, V., Li, C., Deng, J., Han, W., Li, Z., Gu, K., Song, Y., Bengio, S., Rossenberg, C., Fei-Fei, L.: Learning semantic relationships for better action retrieval in images. In: 2015 CVPR, June 2015

22. Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T.: LabelMe: a database and web-based tool for image annotation. In: IJCV 2008. vol. 77, pp. 157–173 (2008)
23. Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y.: Overfeat: integrated recognition, localization and detection using convolutional networks. CoRR abs/1312.6229 (2013)
24. Shotton, J., Winn, J., Rother, C., Criminisi, A.: TextonBoost for image understanding: multi-class object recognition and segmentation by jointly modeling texture, layout, and context. In: IJCV 2009, January 2009
25. Sudderth, E., Torralba, A., Freeman, W., Willsky, A.: Learning hierarchical models of scenes, objects, and parts. In: ICCV 2005, vol. 2, pp. 1331–1338, October 2005
26. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: CVPR 2015, pp. 1–9 (2015)
27. Torralba, A.: Contextual priming for object detection. In: IJCV 2003, vol. 53, pp. 169–191 (2003)
28. Uijlings, J., van de Sande, K., Gevers, T., Smeulders, A.: Selective Search for Object Recognition. In: 2013 IJCV (2013). http://www.huppelen.nl/publications/selectiveSearchDraft.pdf
29. Horn, V., et al.: Building a bird recognition app and large scale dataset with citizen scientists: the fine print in fine-grained dataset collection. In: CVPR (2015)
30. Vedaldi, A., Fulkerson, B.: VLFeat: an open and portable library of computer vision algorithms. In: ACM International Conference on Multimedia (2010)
31. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The Caltech-UCSD birds-200-2011 dataset. Technical report CNS-TR-2011-001 (2011)