Recognizing Products: A Per-exemplar Multi-label Image Classification Approach

Marian George and Christian Floerkemeier

Department of Computer Science
ETH Zurich, Switzerland

Abstract. Large-scale instance-level image retrieval aims at retrieving specific instances of objects or scenes. Simultaneously retrieving multiple objects in a test image adds to the difficulty of the problem, especially if the objects are visually similar. This paper presents an efficient approach for per-exemplar multi-label image classification, which targets the recognition and localization of products in retail store images. We achieve runtime efficiency through the use of discriminative random forests, deformable dense pixel matching and genetic algorithm optimization. Cross-dataset recognition is performed, where our training images are taken in ideal conditions with only one single training image per product label, while the evaluation set is taken using a mobile phone in real-life scenarios in completely different conditions. In addition, we provide a large novel dataset and labeling tools for products image search, to motivate further research efforts on multi-label retail products image classification. The proposed approach achieves promising results in terms of both accuracy and runtime efficiency on 680 annotated images of our dataset, and 885 test images of GroZi-120 dataset. We make our dataset of 8350 different product images and the 680 test images from retail stores with complete annotations available to the wider community.

1 Introduction

Many image classification techniques try to recognize the object classes present in the image through training multiple binary classifiers or a multi-class classifier using a large number of training images. Improving the performance of such algorithms requires learning the model using as many images as possible that are drawn from the same distribution as the test images [29]. However, in many real-world applications, we face the challenge that the testing images are taken in completely different settings than the training images. For example, in the domain of assisting the visually impaired, or vision for mobile robots, images are very likely to suffer from blur, specularities, unusual viewing angles, a lot of background clutter and very different lighting conditions. Gathering and labeling images that try to mimic the natural environments for which the system is used, is a tedious and very time-consuming task.

Recently, image recognition in the retail products domain has become an interesting research topic due to the remarkable advancements in the capabilities

D. Fleet et al. (Eds.): ECCV 2014, Part II, LNCS 8690, pp. 440–455, 2014.

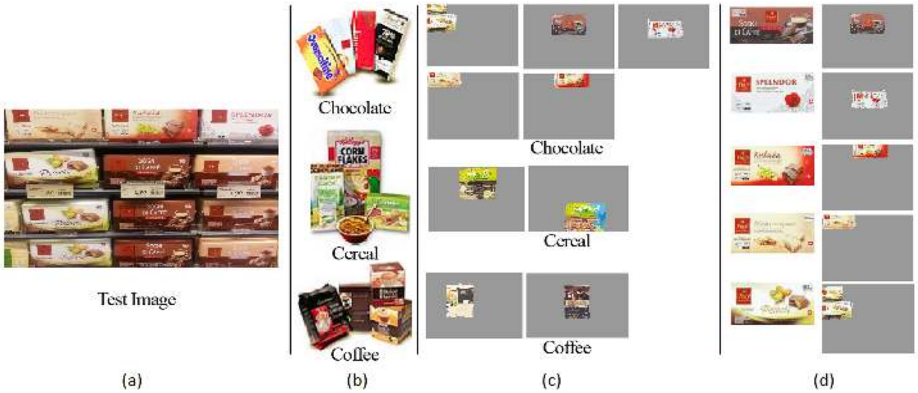


Fig. 1. System overview: (a) Given a test image, (b) we first filter the categories that the test image may belong to, (c) then we match the test image against all images in the filtered categories. (d) An energy function is then optimized given the top-ranked matches to obtain the final list, along with inferred locations, of recognized products.

of mobile phones and mobile vision systems [27,30]. A mobile vision algorithm to recognize products in an image has a wide range of potential applications, ranging from identifying individual products to provide review and price information, to the assisted navigation in supermarkets. Furthermore, mobile retail products recognition can assist the visually impaired in shopping, encouraging them to independently perform daily activities, which promotes their social wellness.

Building a system that parses an image featuring several retail products taken with a smart phone introduces several challenges. This includes the cross-dataset recognition challenges mentioned earlier. Product images available through online shopping websites are taken in ideal studio-like conditions, which are very different from real life images taken with mobile phones in shops, as illustrated in figure 2. These challenges are aggravated when having only one image per product for training and thousands of products (labels) to match against. Due to the increasing number of new products every day, the system also needs to be scalable with no or minimal retraining whenever a new product is introduced. To be applicable in the visually impaired domain, the designed scheme cannot rely on any feedback from the user in improving the retrieved results. The system has to work in a completely autonomous manner. Recognizing grocery products, in particular, is challenging as there are multiple products that have very similar visual appearance except for minor features like the color of the package, or some text describing the product. Finally, runtime efficiency is crucial for mobile vision systems, which makes semantic segmentation or sliding window detection approaches computationally expensive for our problem.

To this end, we designed an efficient per-exemplar multi-label image classification technique (as shown in figure 1) which targets simultaneous recognition and localization of all the individual products in a retail store image. Our algorithm works in a hierarchical manner, where we first filter the possible labels that a test image may contain through ranking the output of a fine-grained classification model. Then, we perform fast dense pixel matching on the images in our filtered list, and rank the individual products by their matching score. Multi-label image classification is then achieved through minimizing an energy function that correlates the matching score, context and recognition localization results through genetic algorithm global optimization. Our proposed approach is evaluated in a per-exemplar cross-dataset settings, where we show promising results in terms of both accuracy and speed. Our main contributions include:

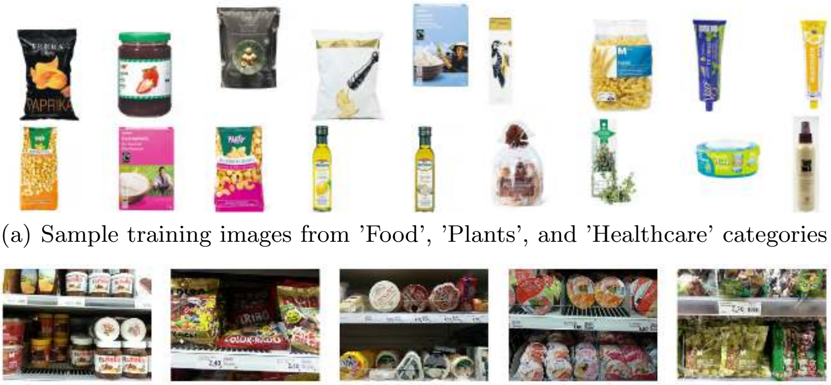
- A large-scale novel retail products dataset, which contains 8350 different products in 80 different fine-grained categories, and 680 test images taken with a smart phone in natural environments with complete annotations and labeling tools.
- A fast technique to simultaneously recognize and localize all retail products in a test image, which scales to thousands of different labels.
- Automatically infer the total number and approximate locations of objects present in a test image in a single optimization round, unlike other more expensive object detection techniques.
- Experimental evaluation of our proposed system with an analysis of its different components, showing promising results.

2 Related Work

General-purpose object detection and recognition techniques [10,11,24] have been extensively used in image classification and retrieval problems. Some approaches [4] use binary classifiers trained on bag-of-words features [4] or compressed Fisher Vectors [25] to obtain a binary decision whether an object class is present or absent in a given test image. Hamming embedding[14] provides binary signatures that refine the matching based on visual words. It has been very successful in instance-level image retrieval domain. Others [11,7] try to detect and localize object classes in the image through extracting dense features like HOG [5] features and apply sliding window detectors or deformable part-based models [11]. However, these techniques require a large number of labeled training examples, which makes them unsuitable for classes with sparse examples.

Other approaches explore fine-grained image classification [3,26] to capture discriminative image regions that distinguish between different classes. In [1], randomization and discrimination are combined in a computationally efficient scheme to achieve fine-grained classification in a large feature space.

Multi-label image classification [17,34,16] differs from multi-class recognition [28] in that a single image is classified using multiple labels. Multi-label classification usually incorporates modelling the correlation between the labels, which significantly boosts the semantic classification performance [16]. In [35], Genetic Algorithm optimization is utilized for filtering the selected features, which are then used for classification. Unlike us, they rely on multi-label training data.



(a) Sample training images from ’Food’, ’Plants’, and ’Healthcare’ categories

(b) Sample testing images from our collected dataset

Fig. 2. Sample images from our collected dataset

There have been relatively limited research attempts to recognize products in images [27,30,15,20,33]. In systems like [15,20], a product image search engine is built. However, these systems deal with recognizing only one product per image, with training images having similar conditions to query images. In [27], cross-dataset single product recognition is targeted through query object segmentation combined with iterative retrieval. They achieve good results in searching an image that contains only a single product. In our work, however, we target multi-products image parsing.

A related dataset to our proposed one is presented in [22], however the dataset size is much smaller than ours with only 120 grocery products in the training set. Each product category represents a single specific product (i.e. no hierarchies or classes of products). In [27], a sports product image dataset is collected. Both the training images and 67 query images contain a single product per image. Other existing object datasets include Caltech 101 [9], Caltech 256 [13], and VOC [8] datasets, which target more general-purpose object category recognition. Fine-grained object datasets include Oxford Flower [23], and Stanford Dogs [18] dataset. We run our experiments on our proposed dataset, as well as the one presented in [22].

In the following sections, we first present our proposed novel dataset in section 3. Then, we detail our approach in section 4. Experimental results and performance evaluation are discussed in section 5. Finally, conclusions and future work are summarized in section 6.

3 Grocery Products Dataset

We built a new supermarket products dataset, which can be used in multi-label, fine-grained cross-dataset object recognition. Our dataset consists of 8350 training images spanning 80 product categories, downloaded from the Web. Each grocery product is represented by exactly one training image, taken in ideal studio conditions with a white background. On the other hand, test images are taken in real-life scenarios using a mobile phone. Each test image contains several products, ranging from 6 to 30 products per image. Test images are taken with different lighting conditions, viewing angles and zoom levels, introducing many challenges to the recognition process.

Training images are organized in hierarchical categories. For example, a Snickers chocolate bar is classified as "Food/Candy/Chocolate". The number of training images in each fine-grained category ranges from 25 to 415 images, with an average of 112 different retail products in each category – one training image for each product. We added an additional label for background regions. The images for the background label represent shelves and price tags, extracted from test images. Examples of training images are shown in figure 2 (a).

Grocery products introduce many challenges to the object recognition problem. First, many products have similar appearance, with only minor differences in the color of the package, size of the package, or some text on the box. Also, non-planar products, like bottles or jars, lower the matching performance, considering that we only have one training image per product. Furthermore, evaluation images contain very little background regions, which makes it a rather challenging task to recognize every single product in the image. This is due to the fact that other regions in the image represent confusing background clutter relative to the specific region of each object.

One of the main goals of this work is to investigate cross-dataset multi-label image classification. Accordingly, our evaluation set is collected in completely different conditions from the training set. A total of 680 images are taken in different grocery stores covering the different classes in the training dataset. Testing images impose additional challenges, like specularities, different viewing angles, rotated or occluded products as shown in figure 2 (b).

We ran our experiments on 27 classes of the "Food" category products in addition to the background class, which represents shelves and price tags, with a total number of 3235 images. Deformable objects, like nuts bags, chips and bakery are included in the "Food" category of our datast. To evaluate the performance of our algorithm, we annotated 680 test images with all the products from the 27 training classes. The ground truth of each test image specifies bounding boxes with a corresponding single product label for each bounding box. A single bounding box covers a group of instances of the same product in a test image as shown in figure 3.

4 Exemplar-Based Multi-label Image Classification

In this section, we describe the design and implementation details of our algorithm. Figure 1 shows an overview of our system. Our proposed technique consists of three main steps. The first two steps filter the best matching products to a given test image through two successive ranking procedures. The third

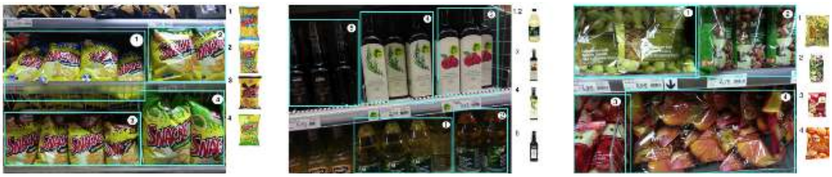


Fig. 3. Sample test images with ground truth annotations from our proposed dataset

step simultaneously localizes and infers the total number of objects present in the test image through globally minimizing an energy function.

Although we are going to use specific algorithms for each step of our pipeline, any other algorithm that fits to a single step can be applied. For example, we used discriminative decision trees for multi-class ranking, but it would suffice to use SVM or k-NN for classification. Similarly, we can use any matching algorithm for the second step of our pipeline like SIFT flow or sparse features matching.

4.1 Multi-class Ranking

To reduce our search space from thousands of possible matches to tens up to a few hundreds of images, we train a classification model using the given training images, and then use a voting scheme, explained below, to retrieve the top-ranked object classes.

For training, we use the recently proposed discriminative random forests [1] technique. The training set contains a single image for each product with a total number of 3235 images in 27 classes. We extract dense SIFT feature descriptors [21] on each image with a spacing of four pixels, with five patch sizes: 8, 12, 16, 24 and 30. Visual vocabulary codebook of 256 code words is then constructed using k-means. Descriptors are assigned to code words using Locality-constrained Linear Coding (LLC) [31].

To retrieve the top ranked object classes for a given test image, we designed a voting algorithm, which first divides the test image into grids with different sizes. We, then, classify each grid region separately using the trained model. We gather votes for each class in the trained model by counting the number of grid segments belonging to that class. For each test image, we return the top k classes.

Our proposed class ranking technique handles two important challenges faced in cross-dataset object recognition, specifically in the products recognition domain. First, each object in the test image is surrounded by many other objects that have very similar features, which can easily confuse the classifier. By dividing the image into patches of different sizes, we limit such confusion. Second, by collecting the total number of votes for grids, we lower the impact of regions in the image suffering from difficult imaging conditions in affecting the final classification decision.

In the experiments section, we detail the parameters used for multi-ranking, and we show how the multi-label ranking performance is improved through gathering votes over grid patches rather than classifying the whole image once.

4.2 Fast Dense Pixel Matching

To achieve simultaneous recognition and localization of specific object instances in each test image, we apply fast dense pixel matching through deformable spatial pyramid matching [19]. No training is required to perform this step. Furthermore, it contributes to the scalability of our system, in such a way that adding new specific objects to the dataset does not require retraining the random forests step, as long as these objects fall under one of the pre-existing classes.

The goal is to rank the images in terms of appearance agreement while enforcing geometrical smoothness between neighboring pixels. The matching objective can be expressed formally by minimizing the energy function [19]:

$$E(t) = \sum_{i} D_i(t_i) + \alpha \sum_{i,j \in N} V_{ij}(t_i, t_j), \tag{1}$$

where Di is a data term which measures the average distance between local descriptors within node i in the first image to those located within a region in the second image after shifting by t_i. V_{ij} is a smoothness term, α is a constant weight, and N denotes pairs of nodes linked by graph edges. The energy function is minimized using loopy belief propagation. Training images are scaled to 200x200 pixels, and test images are scaled to 600x450 pixels. We use the mean difference of dense color SIFT feature descriptors of patch size of 4 as our data term. In all the experiments, the value of α was fixed at 0.005 following [19].

A segmentation mask is obtained specifying the inferred location of every pixel in each matched image with respect to the current test image. The matching costs, along with the segmentation masks are used in the next step of our pipeline to produce the final multi-labeling results as explained in section 4.3.

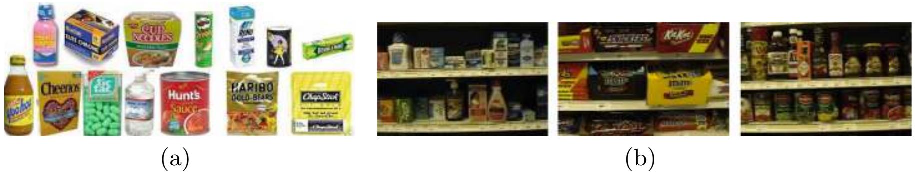
4.3 A Genetic Algorithm-Based Multi-label Image Classification

Once we obtain a ranked list of matching correspondences, we then consider only the top N images, which will be in the range of very few tens of images, to obtain our final multi-labeling results. We formulate our problem in a genetic algorithm (GA) optimization model [12]. The quality of a given solution is determined using a fitness function, which is the objective function to be minimized using GA.

To define our multi-label image classification objective function, let q be our current test image. We want to find the $L \subset N$ images that minimize the following energy function:

$$E(L) = \alpha \sum_{l \in L} D_{lq}(l, q) + \beta U_{Lq}(L, q) + \gamma C_L, \tag{2}$$

where D_{lq} is the data term between image $l \in L$ and the current test image q, U_{Lq} is the uncoverage term, which measures the proportion of pixels not covered



(a) (b)

Fig. 4. Sample (a) training and (b) testing images from Grozi-120 dataset

by any image l in L when the whole set is warped to q. Finally, our context term C_L models the prior knowledge about the co-occurrence of recognized products in the query image. α, β, and γ are weight parameters.

We chose our data term D_{lq} to measure the mean difference between the dense SIFT descriptors between the two images, as defined by [19]. We experimented with adding other features like normalized RGB color histogram. However, the performance was worse with global color, where most products are colorful, and the lighting conditions of test images are very different from training images.

The coverage term U_{Lq} penalizes results that do not cover a big proportion of the test image. If we define S_{lq} to be the set of non-overlapping pixels in q covered by l when warped to q, then U_{Lq} can be defined as:

$$U_{Lq}(L,q) = 1 - \frac{1}{z}\sum_{l \in L} |S_{lq}|, \tag{3}$$

where z is the total number of pixels in the test image q, and $|S_{lq}|$ is the cardinality of the set S_{lq}. This, again, helps in overcoming the challenge of having multiple database images with very similar visual appearance. Such images will all be ranked as top matches, but for only one object in the test image. Just taking the top ranked results, would then yield very poor coverage of the objects present in the test image.

The context term C_L models the prior probability that the labels which appear in the final retrieved set of images occur together. In other multi-label classification approaches, this knowledge is usually inferred from the training images. In our case, this knowledge cannot be obtained from the training images, as each image in our training set contains only a single product. We overcome this problem through utilizing the hierarchical structure of our solution. We model the prior distribution such that images (or labels) which fall under the same category are more likely to occur together than those which fall in different categories. The probability of co-occurrence is higher for more restrictive categories than for broader categories.

$$C_L = 1 - \sum_{l_i, l_j \in L} \tilde{P}(l_i, l_j), \tag{4}$$

where $\tilde{P}(l_i, l_j)$ is the prior distribution over the pairwise co-occurence of labels.

The overall energy function in Eq. 2 is globally minimized using constrained genetic algorithm (GA) [12]. We represent the population of possible solutions as a binary vector of length N, where each element represents the decision of inclusion for each image in the set. We used the "ga" method provided in the Matlab Global Optimization Toolbox. To constrain the type of children that the algorithm creates at each step to be binary, we implemented special creation, crossover, and mutation functions [6].

5 Experimental Results

Datasets: We evaluated the performance of our approach on two datasets:

1. 680 annotated test images from the proposed **"Grocery Products"** dataset, with a total number of 3235 products in 27 leaf node classes. Test images contain products of all subcategories in the "Food" category ranging from 6 to 30 product items per image. Regions in the test images which contain objects that do not belong to the database are given a null label.
2. 885 extracted test images from **GroZi-120** [22] dataset. There is a total of 676 training images representing 120 grocery products. Each product is represented by 2-14 training images with an average of 5.6 images. There are no classes of products (i.e. each class has only one specific product). The originally provided test images were unsuitable, since each image contains a single product item. No shelves images were provided. We, instead, extracted video frames from the provided 29 video files, each representing the whole frame as shown in figure 4. Each test image contains 4-15 grocery product items. Training images are downloaded from the web in ideal conditions, while test images are taken in grocery stores with different conditions.

Implementation Details: We trained 100 trees with a maximum depth of 10. We gathered votes for each test image over 57 patches of 5 different grid sizes. The motivation behind choosing these values is explained in section 5.4. The values of the parameters for the energy function (defined in Eq. 2): α, β and γ are optimized using coordinate descent as detailed in section 5.2.

Evaluation Metrics: We measure the performance of our proposed system using three metrics: mean average precision (mAP), mean average product recall (mAPR), and mean average multi-label classification accuracy (mAMCA) [2]. We chose non-standard measures because standard measures usually address the performance of single-instance retrieval. mAP is measured by computing the average precision over all test images for different values of the number of top matched images (n) that we consider in the matching step, and then the mean is taken over all values of n (ranging from 5 to 70). Averaging helps to capture the joint precision-recall performance. We count groups of specific products in a test image not individual product items (figure 3). We measure the mAPR by computing the average labeling performance (recall) of the retail product items present in an image, and then the mean is computed across all images. To

Table 1. Multi-label image classification performance for baseline labeling, different versions of our system, and state-of-the-art classification and retrieval techniques

Method	mAP(%)	mAMCA(%)	mAPR(%)
Baseline [19]	13.53	11.77	37.33
Full	23.49	21.19	43.13
without global optimization	16.93	15.07	43.36
with ground truth ranking	42.56	38.02	45.63
ground truth ranking without global optimization	30.7	27.8	68.5
FV(1024 dim)	8.62	6.41	20.73
FV(4096 dim)	11.26	9.95	22.14
FV(4096 dim) + RANSAC	12.3	10.1	24.5
HE(k=200000, h_t=22)	4.26	3.96	12.13

compute mAMCA over the test dataset D, suppose Y_x is the set of ground truth labels for test image x, and P_x is the set of prediction labels. We can define the multi-label score for image x as $score(P_x) = \frac{|Y_x \cap P_x|}{|Y_x \cup P_x|}$, and

$$accuracy_D = \frac{1}{|D|} \sum_{x \in D} score(P_x), \tag{5}$$

To analyze the performance of our multi-class ranking approach, we, also, use two measures: mean average recall (mAR) per-class and mean average accuracy (mAA) over the test images. We vary K, i.e. the number of predicted classes from 1 to the total number of classes, and measure the true positive and false positive rates accordingly.

In the next sections, we first perform quantitative and qualitative evaluation of our system (Sec. 5.1). We perform in-depth analysis of our GA optimization in Sec. 5.2. Results on the GroZi-120 dataset are reported in Sec. 5.3. Finally, multi-class ranking and runtime efficiency are discussed in Sec. 5.4 and Sec. 5.5.

5.1 Multi-label Image Classification Performance

To evaluate the performance of our proposed approach, we vary the number of predicted classes of the multi-class ranking (K) from 1 to the total number of classes and report the mAMCA and mAPR values on different variants of our system (see table 1): (1) full system (Full), (2) our system without performing global optimization (i.e. retreive all the n top-ranked images from the dense pixel matching results on the k top-ranked class categories), (3) our system if we have perfect ranking performance of the multi-class ranking step, and (4) ground truth ranking without performing global optimization. We compare the performance of our algorithm to state-of-the-art classification and instance-level image retrieval techniques, Fisher Vectors [25] (FV) and Hamming Embeddings [14] (HE). For FV, we use 1024 and 4096 dim. encodings without PCA. We, also, compare to FV (4096 dim) with Geometric Consistency Checks with RANSAC

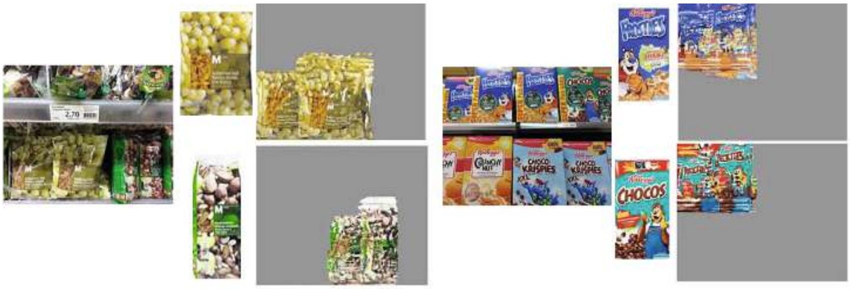


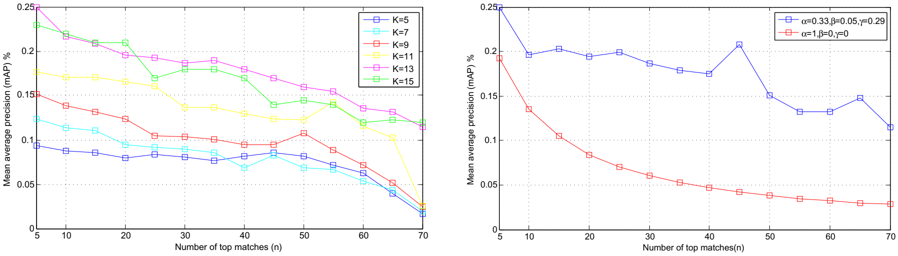
Fig. 5. Examples of two multi-label classification results. Left column shows the test image, then the retrieved products, and finally their inferred locations in the test image.

re-ranking on the top 100 images. For HE, we use $k = 200,000$ visual words for building the bag-of-words histogram representation which was shown to yield good performance and we use a fixed Hamming threshold $h_t = 22$ following [14]. Finally, we compare to the baseline method of ranking all the images by just dense pixel matching score [19] and taking the top n matches.

Our full system achieves 23.49% mAP and 21.19% mAMCA over all the 680 test images, which outperforms the baseline method by over 9%. Our method also significantly performs better than other state-of-the-art approaches. FV and HE are efficient algorithms which achieve impressive precision on other benchmarks. However, for our case, the distribution of the training data from which the GMM model is built (for FV), or the BOF dictionary is built (for HE) is significantly different from the data distribution of the test set. In addition, these methods are better suited for general rather than fine-grained object recognition.

We also show the performance results if we run our dense pixel matching ranking and global optimization steps using the images of the ground truth classes (i.e., we assume that the multi-label ranking step gives a perfect ranking of predicted categories for each test image). This yields a substantial improvement in the mAP and mAMCA, which shows that our system's performance could be further improved by experimenting with different classification techniques. When evaluating the system, the parameters are optimized for maximizing the precision and accuracy of recognition. Accordingly, the recall performance is not much improved given the chosen values of the parameters. Showing the improvement of precision for the same achieved recall values gives an indication of how ground truth ranking can improve the performance of the system.

To verify the impact of our global optimization step, we report results when we pick the top n-ranked images from the matching step as our final multi-label classification result without any global optimization. We have two cases: (1) with random forests model ranking, and (2) with ground truth ranking. We notice that the mAMCA degrades by more than 6%, as more irrelevant images appear in the final result. In case of ground truth ranking, our system still performs better with global optimization.



(a) (b)

Fig. 6. (a) Mean average precision as a function of the total number of matches (n) for different values of the number of filtered classes (K). (b) Mean average precision as a function of the total number of top matches (n) when turning on (and off) the GA optimization. Our GA step significantly yields better performance.

In figure 5, we show sample results from running our full system on different test images to illustrate the effectiveness of our proposed technique. We show the original test image, the inferred labels, and their predicted locations in the test image. Failure cases are mainly due to significant visual resemblance between training images (like the cereal box in the figure), severe specularities, and blurry conditions of test images. We also fail to recognize wrong facing products, which can be addressed with additional training images.

Discussion: Although the absolute value of the results may seem unsatisfactory, we have to consider the challenging settings of our problem. Other fine-grained classification datasets like Caltech-UCSD Birds dataset [32] report a state-of-the-art performance of average accuracy of 19.2%, considering that there are 15 training images per category, and each image contains only one object instance. Whereas for our dataset, we have a single training image per category, and each test image contains an average of 20 objects per image.

5.2 GA Optimization

We analyze the performance of our GA optimization by investigating the mAP when choosing different parameter values for K, n, α, β and γ. We first study the effect of the number of filtered classes K, in the multi-ranking step, and the number of top matches n, in the dense pixel matching step, on the mAP performance of the system. Figure 6 (a) shows the mAP as a function of n for different values of $K = 5, 7, 9, 13, 15$. For each combination of n and K, we obtain the optimal values of α, β and γ which maximize the mAP using coordinate descent optimization. It is shown that increasing the number of classes K generally improves the mAP. However, as K keeps increasing, more noise is added to the filtered set which decreases the mAP. Best performance is obtained for $K = 13$ classes. As expected, mAP decreases as n increases, but at the same time the recall improves. In figure 6 (b), we plot the mAP as a function of n for

Table 2. Performance on Grozi-120. System parameters are optimized to maximize average precision rate.

Method	mAP	mAMCA	mAPR
Baseline [19]	7.62	6.24	16.59
Full	13.21	7.5	9.37
without global optimization	9.54	7.1	17.56
with ground truth ranking	N/A	43.03	43.03
FV(1024 dim)	4.44	5.49	12.50
FV(4096 dim)	7.34	5.74	15.16
FV(4096 dim) + RANSAC	8.13	6.65	15.2
HE(k=200000, h_t=22)	6.32	5.23	10.54

$K = 13$ when turning off the GA optimization, by setting $\alpha = 0$, $\beta = 0$ and $\gamma = 0$, and when turning on the GA optimization by fixing $\alpha = 0.33$, $\beta = 0.05$ and $\gamma = 0.29$ (obtained using coordinate descent). Our GA step significantly improves the mAP performance. Also, our curve is flatter which shows that our method is more tolerant to noise imposed by adding more images in the dense pixel matching step.

5.3 Performance on GroZi-120 Dataset

We ran our experiments on 885 extracted test images (see figure 4). We used the same metrics and compared to the same approaches as in Sec. 5.1. Our system significantly outperforms other methods and the baseline method as shown in table 2. Please note that mAP value for ground truth ranking variant of our system will always have a value of 100.0% because each product category represents a specific product in Grozi-120 dataset. For similar reasons, the ground truth ranking without global optimization setting cannot be applied to Grozi-120 dataset. Figure 7 shows sample results from running our algorithm on Grozi-120. We effectively recognize and infer the locations of the objects in a test image.

We note that our system achieves lower mAP values on the Grozi-120 dataset than on our proposed dataset. This is due to the fact that there are only 5.6 images per product (which represents a class) on average for training which greatly degrades the results of the discriminative random forests. This is verified in the significant improvement of the system performance when using ground truth ranking. Further more, a large proportion of the test images in the Grozi-120 dataset suffer from blurriness. Nevertheless, our system outperforms other approaches. Also, there is no available prior information. We adjusted the prior model to be the $l1$-norm of the total number of recognized products in the image.

5.4 Multi-class Ranking Analysis

To demonstrate the impact of our multi-class ranking scheme, we report the mAA and mAR values using (1) different number of segments (i.e. votes), as

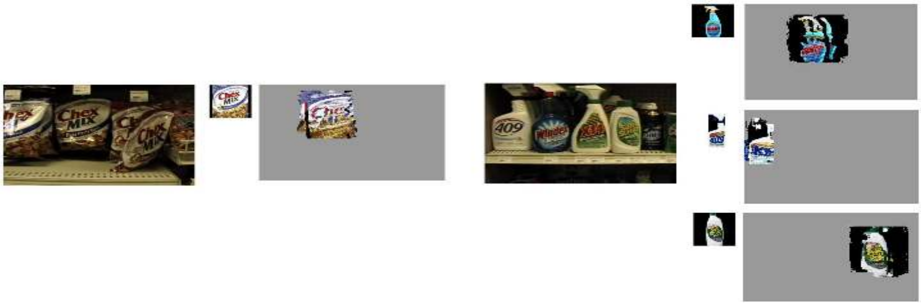


Fig. 7. Examples of two multi-label classification results on Grozi-120. Left column shows the test image, then the retrieved products, and finally their inferred locations.

Table 3. Multi-class ranking analysis. Baseline is the binary classification of images.

	Baseline	1 seg	5 seg	57 seg
mAA	25.56	63.55	62.52	**64.00**
mAR	22.4	57.22	53.32	**58.35**

opposed to (2) using the whole image (i.e. 1 segment) for ranking, and (3) performing binary classification of a test image (baseline). We have experimented with different, empirically chosen, segment sizes. Results in table 3 show that ranking classes through gathering classification votes consistently yields better performance. The impact of regions in the image that suffer from specularities or very wide variation in viewing angles is regularized by considering other patches that have better conditions.

5.5 Runtime Efficiency

Our system consists of 3 steps: (1) Multi-class ranking, (2) fast dense pixel matching, and (3) global optimization. We ran our experiments on a single 2.4G CPU with 4 GB of RAM without code optimization. Step (1) takes an average of 0.2 seconds per test image, not considering feature extraction time. Step (2) takes 0.35 seconds per each matching operation, and finally step (3) converges to an optimal solution in around 1.4 seconds when we consider the top 20 images for optimization. Accordingly, the total runtime of our algorithm is 1.95 seconds, where the time for dense pixel matching is parallelized for n top-ranked images. The most time consuming task is the LLC feature extraction.

6 Conclusions and Future Work

We presented a fast and scalable novel approach to recognize and localize all specific product instances in a retail store image with minimum training. We perform cross-dataset multi-label image classification, where each label is represented by just one instance in our training set. We also propose a new large-scale retail products dataset with thousands of different labels. Experiments showed that our system significantly yields better results than existing state-of-the-art classification and instance-level retrieval methods on both our proposed dataset, and Grozi-120 dataset. Although we apply our proposed method to the grocery products use case, our algorithm is general and can be applied to other multi-label image classification problems. Accordingly, the next step for us is to experiment with applying our system in other domains and compare the performance with previous methods on available benchmarks.

References

1. Bangpeng, Y., Aditya, K., Fei-Fei, L.: Combining randomization and discrimination for fine-grained image categorization. In: CVPR (2011)
2. Boutella, M.R., Luob, J., Shena, X., Brown, C.M.: Learning multi-label scene classification. Pattern Recognition 37(9), 1755–1771 (2004)
3. Branson, S., Wah, C., Schroff, F., Babenko, B., Welinder, P., Perona, P., Belongie, S.: Visual recognition with humans in the loop. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 438–451. Springer, Heidelberg (2010)
4. Csurka, G., Dance, C., Bray, C., Fan, L.: Visual categorization with bags of keypoints. In: Workshop on Statistical Learning in Computer Vision (2004)
5. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR (2005)
6. Deep, K., Singh, K.P., Kansal, M.L., Mohan, C.: A real coded genetic algorithm for solving integer and mixed integer optimization problems. Applied Mathematics and Computation 212(2), 505–518 (2009)
7. Duan, G., Huang, C., Ai, H., Lao, S.: Boosting associated pairing comparison features for pedestrian detection. In: ICCV Workshop on Visual Surveillance (2009)
8. Everingham, M., Gool, L.V., Williams, C., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. IJCV 88(2) (2010)
9. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. In: Workshop on Generative-Model Based Vision, CVPR (2004)
10. Fei-Fei, L., Fergus, R., Torralba, A.: Recognizing and learning object categories. In: ICCV Tutorial (2005)
11. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE T. Pattern Anal. 32(9), 1627–1645 (2010)
12. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley (1989)
13. Griffin, G., Holub, A., Perona, P.: Caltech-256 object category dataset. Technical report, Caltech (2007)
14. Jegou, H., Douze, M., Schmid, C.: Hamming embedding and weak geometric consistency for large scale image search. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 304–317. Springer, Heidelberg (2008)

15. Jing, Y., Baluja, S.: Pagerank for product image search. In: WWW (2008)
16. Jurie, Y.S.F.: Improving image classifcation using semantic attributes. IJCV 100(1), 59–77 (2012)
17. Kang, F., Jin, R., Sukthankar, R.: Correlated label propagation with application to multi-label learning. In: CVPR (2006)
18. Khosla, A., Jayadevaprakash, N., Yao, B., Fei-Fei, L.: Novel dataset for fine-grained image categorization. In: First Workshop on Fine-Grained Visual Categorization, CVPR (2011)
19. Kim, J., Liu, C., Sha, F., Grauman, K.: Deformable spatial pyramid matching for fast dense correspondences. In: CVPR (2013)
20. Lin, X., Gokturk, B., Sumengen, B., Vu, D.: Visual search engine for product images. In: Multimedia Content Access: Algorithms and Systems II (2008)
21. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV 60(2), 91–110 (2004)
22. Merler, M., Galleguillos, C., Belongie, S.: Recognizing groceries in situ using in vitro training data. In: CVPR (2007)
23. Nilsback, M.E., Zisserman, A.: A visual vocabulary for ower classification. In: CVPR (2006)
24. Pandey, M., Lazebnik, S.: Scene recognition and weakly supervised object localization with deformable part-based models. In: ICCV (2011)
25. Perronnin, F., Liu, Y., Sanchez, J., Poirier, H.: Large-scale image retrieval with compressed fisher vectors. In: CVPR (2010)
26. Sharma, G., Jurie, F., Schmid, C.: Discriminative spatial saliency for image classification. In: CVPR (2012)
27. Shen, X., Lin, Z., Brandt, J., Wu, Y.: Mobile product image search by automatic query object extraction. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part IV. LNCS, vol. 7575, pp. 114–127. Springer, Heidelberg (2012)
28. Shotton, J., Winn, J., Rother, C., Criminisi, A.: Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part I, LNCS, vol. 3951, pp. 1–15. Springer, Heidelberg (2006)
29. Torralba, A., Efros, A.: Unbiased look at dataset bias. In: CVPR (2011)
30. Tsai, S.S., Chen, D.M., Chandrasekhar, V., Takacs, G., Cheung, N.M., Vedantham, R., Grzeszczuk, R., Girod, B.: Mobile product recognition. In: ACM Multimedia (ACM MM) (2010)
31. Wang, J., Yang, J., Yu, K., Lv, F., Huang, T., Gong, Y.: Locality-constrained linear coding for image classification. In: CVPR (2010)
32. Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., Perona, P.: Caltech-ucsd birds 200. Technical report cns-tr-201, Caltech (2010)
33. Winlock, T., Christiansen, E., Belongie, S.: Toward real-time grocery detection for the visually impaired. In: CVAVI (2010)
34. Zha, Z., Hua, X., Mei, T., Wang, J., Qi, G., Wang, Z.: Joint multi-label multi-instance learning for image classification. In: CVPR (2008)
35. Zhang, M., Pena, J., Robles, V.: Feature selection for multi-label naive bayes classification. Information Sciences 179(19), 3218–3229 (2009)