Recognizing Groceries *in situ* Using *in vitro* Training Data

Michele Merler
Telecommunications Engineering
University of Trento, Italy
michele.merler@studenti.unitn.it

Carolina Galleguillos
Computer Science & Engineering
University of California, San Diego
cgallegu@cs.ucsd.edu

Serge Belongie
Computer Science & Engineering
University of California, San Diego
sjb@cs.ucsd.edu

Abstract

The problem of using pictures of objects captured under ideal imaging conditions (here referred to as in vitro*) to recognize objects in natural environments* (in situ) *is an emerging area of interest in computer vision and pattern recognition. Examples of tasks in this vein include assistive vision systems for the blind and object recognition for mobile robots; the proliferation of image databases on the web is bound to lead to more examples in the near future. Despite its importance, there is still a need for a freely available database to facilitate study of this kind of training/testing dichotomy. In this work one of our contributions is a new multimedia database of 120 grocery products, GroZi-120. For every product, two different recordings are available:* in vitro *images extracted from online grocery websites, and* in situ *images extracted from camcorder video collected inside a grocery store. As an additional contribution, we present the results of applying three commonly used object recognition/detection algorithms (color histogram matching, SIFT matching, and boosted Haar-like features) to the dataset. Finally, we analyze the successes and failures of these algorithms against product type and imaging conditions, both in terms of recognition rate and localization accuracy, in order to suggest ways forward for further research in this domain.*

1. Introduction

Object detection and recognition are important tasks in computer vision. All the algorithms that address these problems need training data to learn from, and sometimes is difficult to obtain data set of sufficient size or the characteristics of the data are not appropriate due to image resolution, intra-class and inter-class variability, etc. [13]. Therefore there is a need to gather data from different sources like other public datasets, from the web or to create them in a lab or studio. Many applications in computer vision aim at recognizing specific objects rather than general classes such as human or car. Therefore, they rely on having training examples that not only differ from what we can get in real world scenes, but also should allow one to distinguish a specific object within a general class. An example of such an application is the automatic recognition of products in a grocery store used in the field of assistive technologies for the blind. We built a database consisting originally of 120 different objects easily retrievable from the web. We refer to a grocery product with its corresponding UPC code as an object. We also provide a collection of videos, taken from inside a grocery store, that capture real data of the same products. Each product has several different image examples that were extracted from the web and from video captures. The intent of this dataset is to serve as a seed upon which the set of images can grow dynamically by user interaction in the future.

Our goal in this paper is to provide a baseline performance of some of the most common approaches used to solve object localization and recognition in the presence of image quality discrepancy (between *in vitro* and *in situ* data). Therefore in Section 2 we first review three state of the art object detection and recognition algorithms, and then we present existing databases of common use in the computer vision community. In Section 3 we introduce the GroZi-120 database, which is publicly available, with a detailed description of what it contains and how it was created. Section 4 presents the features used and the different models for localization and recognition. Section 5 shows the results on a significant subset of the database and finally in Section 6 we discuss the results and propose ideas for future work.

2. Related work

2.1. Detection/recognition algorithms

Color histogram matching is one of the first algorithms ever applied to detect and recognize objects in images and videos. Swain and Ballard's early work on color object recognition by means of fast matching color histograms by intersecting them [15] opened the way to many different approaches with a common ground. Computational complexity has always been one major bottleneck of the histogram extraction and comparison based search tasks, but this problem has recently been overcome with the introduction of the integral histogram by Fatih Porikli [14]. On the other hand, the choice of which colorspace or chrominance plane to use is still an open issue [8].

In addition to color features, one can employ methods based on on shape [2] and/or gray-scale object descriptors. There are two distinct problems linked to such approaches: interest point detection and description of the distribution of smaller-scale features within the interest point neighborhood. In interest point detection, the most commonly used method is probably the Harris corner detector [6], while for the second problem, David Lowe's SIFT [10] descriptor has been shown to outperform the others, as the SIFT descriptor is invariant to scale and rotation transformations. This property can be explained by the fact that it captures a substantial amount of information about the spatial intensity patterns, while at the same time being robust to small deformations or localization errors.

In the framework of object detection, a major contribution has been offered by Viola and Jones [16], who introduced a fast and reliable classifier. Such classifier is obtained by a cascade of classifiers, each of them resulting from a subset of weak learners combined with Adaboost [4]. The weak learners use Haar-like features, which are responses of filters computed extremely fast because of the integral images.

2.2. Computer vision databases

Over the years many datasets have been introduced in the computer vision community, in order to provide means of evaluation for the different algorithms developed. Here we present a sample of recent ones, summarizing their characteristics. In this way it will appear more clear the innovative contribution offered by the GroZi-120 dataset, which we will introduce in Section 3.

PASCAL Object Recognition Database Collection [12] consists of data gathered for an object recognition in natural scenes challenge that took place in 2005 and 2006. In its final version, the collection contains 5,304 images, provided by Microsoft Research Cambridge and collected from the photo-sharing web-site Flickr, of 10 object classes. All images are annotated with instances of all the classes, for a total of 9,507 labelled objects. The set presents variability of scale, pose, background clutter and degree of occlusion for every object.

Caltech 101, now expanded into Caltech 256 [3], contains 30607 images, grouped into 256 object classes, with a mean of 119 images per class. It also includes a special category for clutter and background. It is widely used for object recognition but is not recommended for object localization.

Another example is SOIL-47 [7], a database of household objects, many of the same shape, viewed over a significant portion of the viewing sphere. The images show mainly multicolored objects, many of them consisting of planar surfaces (boxes) and with generally complex color structure. The database contains 24 objects with approximately planar surfaces and 22 complex scenes. Both objects and scenes are presented against a black background, in absence of clutter. Three different kinds of appearance variation are included: 3D viewpoint, illumination intensity and occlusion/distractors.

A more recent computer vision database is the Amsterdam Library of Object Images (ALOI) [5], a collection of one-thousand small objects. The creators of the dataset systematically varied viewing angle, illumination angle, and illumination color for each object, and additionally captured wide-baseline stereo images. It includes over a hundred images per object, yielding a total of 110,250 images. Again, no clutter or complex background are present.

Finally, the ETH-80 [9] database consists of 80 objects from 8 chosen categories captured in high-resolution color images, with segmentation masks provided for every image. Each object is represented by 41 images from viewpoints spaced equally over the upper viewing hemisphere (at distances of 22.5-26 °).

Apart from the Caltech 256 and the PASCAL dataset, all the reported databases present a uniform one-color background, so objects are easily segmentable from the background (which is usually black and presents no clutter). Moreover, in almost all cases no occlusion is present and the sizes and position of objects inside the images are normalized, as pointed out in [13]. In contrast, we propose a dataset that contains images that present a significant difference in quality (between different types of data) and where individual objects have both cluttered and uniform background. Therefore it can be used for object localization as well as recognition approaches, and it can be used for training and testing purposes, either with images or video captures.

3. GroZi-120 database

The GroZi-120[1] is a multimedia database of 120 grocery products. The objects belonging to it vary in color, size, opacity, shape and rigidity, as can be seen in Figure 2. The dataset introduces variabilities not systematically offered in previous available collections. In fact, many of our images contain multiple instances of the same object, which may present partial occlusion and truncation, as well as size and orientation variations. Furthermore, the location of the product varies considerably from image to image and different objects are found in the same frames. All these

[1] http://grozi.calit2.net

properties are exemplified in Figure 4. Every product has two different representations in the database: one captured *in vitro* and another *in situ*. The *in vitro* images are isolated and captured under ideal imaging conditions (e.g. stock photography studio or a lab) and they can be found on the web, more specifically in grocery web stores such as Froogle[2]. We performed queries by feeding UPC codes of the products, or a description obtained by the UPC online database[3]. In order to make the *in vitro* images usable as a training set for any algorithm, we set their background to transparent and we obtained a binary mask in order to extract only the useful information out of every image, as exemplified in Figure 5 (a). This process is particularly critical for methods based on color histograms. Therefore the *in vitro* images become easy to analyze and at the same time, coming from online vendors and stock photo suppliers, they include a spread variety of illuminations, sizes and poses. Figure 1 shows the different views of a particular product represented by *in vitro* images.

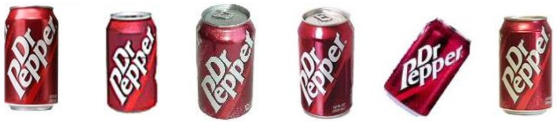


Figure 1. Sample of *in vitro* images of multiple views for a product.

On the other hand, *in situ* representations come from natural environments (real world). Figure 3 shows the *in situ* representations of the objects in Figure 2. We shot 29 videos on the same day at 30 fps, encoded as Divx 5.2.1 with a bitrate of 2000 kbps using a VGA resolution MiniDV camcorder, for a total of around 30 minutes of footage. Such videos include every product present in the *in vitro* part of the dataset. These images were selected every 5 frames and were stored together with their location in the video (video number, frame number, rectangle coordinates), as shown in Figure 5 (b). These images present variations in scale, illumination, reflectance, color, pose and rotation, while the video frames provide a cluttered background. One of the benefits of this data is that it represents the typical low quality of a real world image. Hence, different algorithms for object recognition (where *in vitro* images can be compared to *in situ* images) and object localization (search for products in videos) can be tested on this dataset. Table 1 reports statistics about the dataset.

4. Object detection and recognition algorithms

In order to provide a baseline characterization of the level of difficulty inherent to this problem domain, we tested a selection of popular object detection and recognition approaches and studied their performance on the GroZi-120 dataset. Below we describe the different features used and the approaches implemented in detail.

	in vitro	*in situ*
Total number of images	676	11194
Average number of images per object	5.6	93.3
Min. number of images per object	2	14
Max. number of images per object	14	814

Table 1. General statistics of *in situ* and *in vitro* images for the GroZi-120 database. The reduced number of *in vitro* samples arises from the difficulty to retrieve different instances of the same product not reproducing an image already acquired.

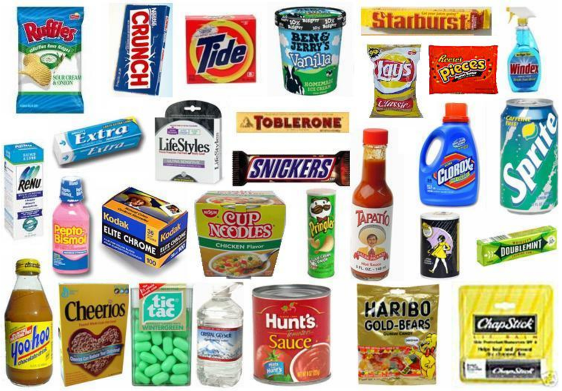


Figure 2. Sample of *in vitro* images for different products.

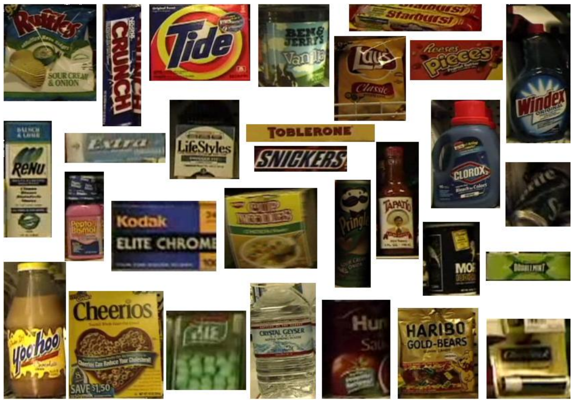


Figure 3. Sample of *in situ* images for different products.

4.1. Features and dissimilarity measures

Color Histogram: We first tested our database with chrominance planes belonging to 3 different color spaces: YCbCr, HSV and Lab. A preliminary study showed that the ab plane from Lab provided the best results. In order to have a compact yet sufficiently descriptive representation,

[2] http://www.froogle.com
[3] http://www.upcdatabase.com

we computed histograms of 16 bins per channel, a and b, calculated separately, for a total of 32 bins. We generated a histogram of the a and b channels for every *in vitro* image in the dataset. Histograms belonging to the same product were subsequently averaged, bin per bin, in order to obtain a final template histogram representative of the object.

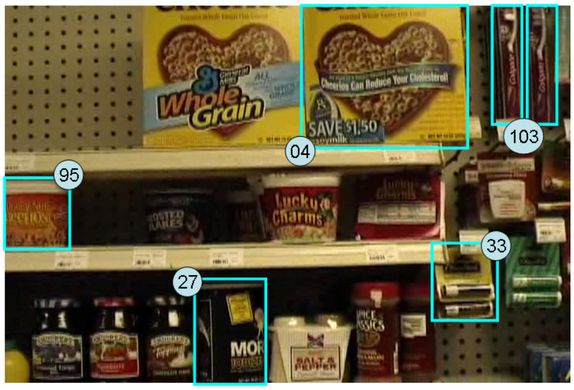
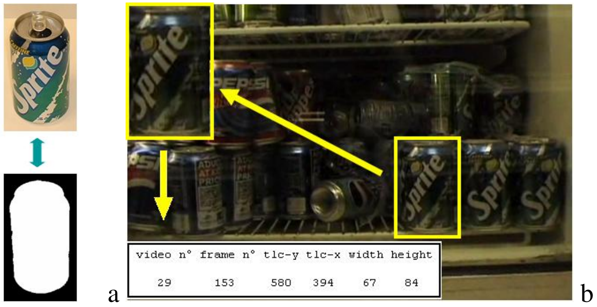


Figure 4. *In situ* video frame sample. There are 2 instances of product 103, product 4 truncated, 2 instances of object 33 (one almost completely occluded by the other), 1 sample of product 27 (rotated out of plane) and 1 instance of product 95.



video n°	frame n°	tlc-y	tlc-x	width	height
29	153	580	394	67	84

Figure 5. (a) Binary mask applied to web samples. (b) Product image cropped from video frame and stored together with coordinates.

SIFT: We computed SIFT [10] keypoints for every *in vitro* image in the data set in order to represent images using scale and rotation invariant descriptors. The keypoints were computed using binaries provided by the UCLA Vision Lab [4]. Each grocery product was represented by a "bag of keypoints" extracted from the *in vitro* images corresponding to a particular object. Therefore the calculated keypoints correspond to the different views of the same object without including background. The background pixels were set to zero when the masks were applied to the images. The product shape information is implicitly captured by the descriptors, since they are computed on the masked images.

Boosted Haar-like features: In the case of Haar like features used as weak classifiers and then boosted through a cascade of Adaboost stages, we used the implementation in the Haar training utility of the Intel OpenCV library[5]. In particular, positive samples were synthetically created from the *in vitro* images by applying randomly generated perspective distortions, until a number of 200 positive samples was obtained, including the *in vitro* instances. The dimensions of such images were obtained by computing the average ratio of all the masked *in vitro* images and then resizing them to have a longest dimension of 50 pixels.

4.2. Recognition

In our recognition study, *in vitro* images were used as training data, while *in situ* images were used as test data. Therefore, recognition consists in a product-to-product match, where an *in vitro* instance is identified in the real world to be the same object. In this context, we isolate the problem of inter-object confusion. The associated issue of product localization is addressed in a later section.

In color histogram matching we computed color histograms for every *in situ* image in the dataset. Then the distances between the *in vitro* image template and *in situ* image histograms were calculated, according to 3 different metrics: Euclidean, χ^2 and histogram intersection (L_1 distance). Once the distances were computed, we calculated the ROC curves shown in Section 5 by integrating a bidimensional histogram of the distances. Such histogram has one axes for the distances in the a channel and the other for those in the b channel. We obtained the best performance using color histogram intersection; this choice of metric is used in the localization study in the next section.

Recognition using SIFT proceeded similarly to the approach used for color histogram matching. For each product we computed the bag of features obtained from its *in vitro* images. Then we matched the features with the keypoints of every *in situ* image in the dataset. We followed Lowe's approach to find the best matches [10]. The distances between samples were represented by the number of matches between the *in vitro* and the *in situ* one. The ROC curves were computed in the same manner as for the color histogram matching.

With regard to training in Adaboost, the cascade was allowed to be 14 stages deep, with a maximum false alarms rate (FAR) of 0.5 per stage. The overall FAR of the final strong classifier is $0.5^{Numstages}$, with a best rate of 6.1035×10^{-5}. Finally, the minimum hit rate was set to 0.995. We note that in some cases the training phase ended

[4] http://vision.ucla.edu/ ∼vedaldi/code/sift/sift.html

[5] http://www.intel.com/technology/computing/opencv/

before reaching the 14th stage, because the rates requested had already been achieved. The ROC curves were obtained by computing the distances to the *in situ* images resized to the dimensions of the training set of the particular product. Such distances were computed for the classifier by taking the weighted sums of the responses of the features selected at every stage and finding their difference from the thresholds of the stage. All the differences were then summed together to obtain a unique value for the distance.

4.3. Localization

Localization experiments were conducted by trying to identify the location of products present in video sequences, using *in vitro* images as the training set. Testing was performed for every product using 20 frames out of the videos containing the product, with its locations manually identified as ground truth, and 100 frames from the same videos not containing any of the objects in the dataset. In this framework, the ground truth information is represented in terms of the manually selected bounding boxes of the object in the selected frames, taking into account the possible presence of multiple instances of the product in the same frame.

We used two different metrics to evaluate localization: a yes/no rule from which we obtain true positives (TP) and false positives (FP) rates, and a metric based on the average object area recall and the average detected box area precision as defined in [11], that gives us the overall Recall and Precision rates. The first is a frame-based metric where a yes is given if the center of the detected box (meaning the best match) lies within the ground truth region and a no otherwise, as presented in [1]. In the second, the recall for an object is defined by the authors as the proportion of its area that is covered by the algorithm's output boxes for every frame and the overall recall is computed as the weighted average recall of all frames. Precision of an output box is defined as the proportion of its area that covers the ground truth objects and overall precision is the weighted average precision of all frames.

The color histogram matching approach relies on the integral histogram computation as in [14]. The *in vitro* histogram of an object is kept as a template in the same way as in recognition. Then, we performed a frame by frame analysis as follow: first, the frame is converted into the Lab colorspace, then the integral histogram (for the a and b channels) of the whole frame is computed. Subsequently a window at 5 different scales is moved in raster scan order around the frame, computing the color histograms of different regions quickly by just accessing 4 elements of the integral histogram. Then those histograms are intersected with the *in vitro* template, and the distances are compared for each color channel against a threshold obtained from the ROC curves of the recognition part so that 99% of the true positives are kept. If the distance is accepted by the system, it is stored to be later compared to all the other best matches in the frame. Finally a maximum of 6 windows are kept, which correspond to the best scores.

The SIFT approach for object localization consisted first in computing the bag of keypoints for every *in vitro* object and for each frame as in recognition. Since the *in vitro* image sizes are different, we normalized the coordinates of all the keypoints by referring them to an image whose dimensions are computed as the average of the sizes of the *in vitro* samples. Then, we matched the keypoints against the frame features as in [10] using a threshold of 1. In order to reduce the number of outliers and locate the object we centered a circle in the *in vitro* average image, with a diameter equal to the average image diagonal. Iterating over the matches we kept the circle containing the maximum number of matches. Subsequently we computed the centroid of the locations of the corresponding matching features in the frame, and also their average distance with respect to the centroid. Using the previous pair of matches, we found the ones inside a new circle with a diameter equal to the average distance in the *in vitro* and frame instances. If the number of frame matches found is greater or equal than the *in vitro* ones found we consider the object as detected, otherwise we use the matches that were not taken into account in the first place and proceed in the same way. If in both cases the condition is not satisfied, we do not consider the object as detected. This approach was performed for different circles so we could handle multiple instances of a product on a frame.

In the Adaboost based method, the classifier obtained from the set of boosted Haar-like features during the training phase is used to decide whether a series of rectangles analyzed at different scales within each frame contains the product of interest. This process is performed by comparing the responses of the filters selected for every stage of the classifier during the training phase to thresholds also selected during the training.

5. Experiments and Results

In this Section we present the results for recognition and localization of the algorithms presented in the previous sections. We selected a subsample of the database to perform such experiments, namely products 1, 18, 27, 34, 38, 51, 52, 74, 96 and 119 shown in Figure 6. Such examples were selected on the basis of number of training samples available (from 3 to 14), variety in color (distinctive and multimodal like 34, uniform or white predominant as for 1), shape and rigidity (51 for example is not rigid), and quality of the representation in the videos (e.g. very poor for 27, as can be noticed in Figure 4).

In order to have the same priors, we chose 10 *in vitro* image samples for every product, corresponding to the high-

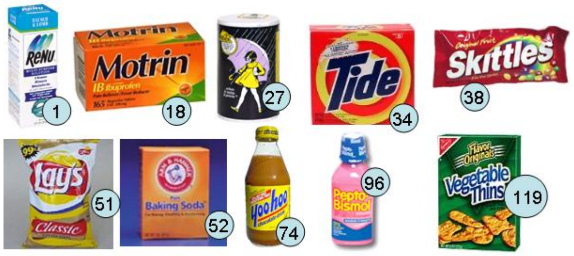


Figure 6. *In vitro* images for the 10 selected experiments out of a total of 120 products

est number of SIFT keypoints, considering that the higher number of keypoints means an image with low blurriness. Synthetic samples were created when the number of samples was less than 10. For recognition we chose 10 *in situ* images per object using the same criteria, but not considering synthetic extras. Figures 7 and 8 present two interesting cases. In the case of product 1, there is a clear disparity between the performances of the different algorithms. While the SIFT performed well, probably due to the distinctiveness of the text and symbols on the product's box, the color histogram curve is affected by the predominance of the white color, which is not particularly distinguishable in the chrominance plane, and performs poorly since it is based on a more global approach. Boosted Haar-like features present an acceptable performance. On the other hand, in Figure 8, CHM benefits from the bimodal histogram obtained from the product (yellow-brown), while SIFT and Haar-like features encounter a lack of distinctive feature points.

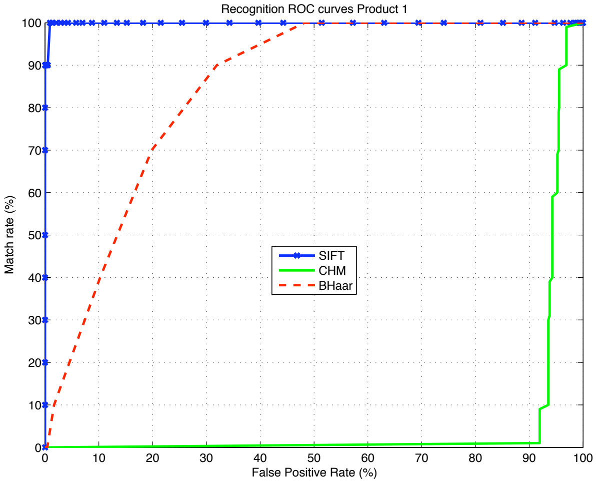


Figure 7. ROC curves for Product 1.

Table 2 shows the localization rates in percentages of the three different algorithms. It can be immediately noticed how the methods do not perform uniformly over the subset of products. This fact is due to the complexity and variety of the dataset, which presents challenges of variable difficulty for any algorithm. Furthermore, from the experiments emerges a discrepancy between the efficacy of the three methods in recognition and localization. Such a discrepancy is due to the different nature of such tasks: in recognition we are still operating in a controlled environment, where although it is true that the samples we are comparing come from the different *in vitro* and *in situ* worlds, it is also true that those instances are segmented from the cluttered background, and therefore easier to analyze. On the other hand, localization must consider the full background, which contributes a significant amount of confusion to the problem. In this sense localization expresses what the core of the problem is: find the correspondence between *in vitro* samples and cluttered, noise corrupted, realistic *in situ* scenes.

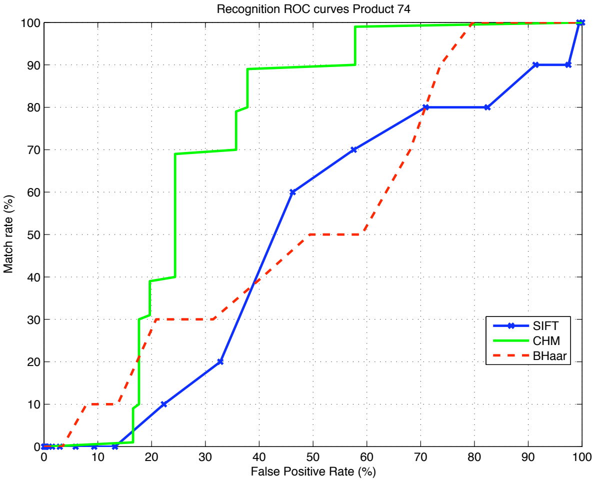


Figure 8. ROC curves for Product 74.

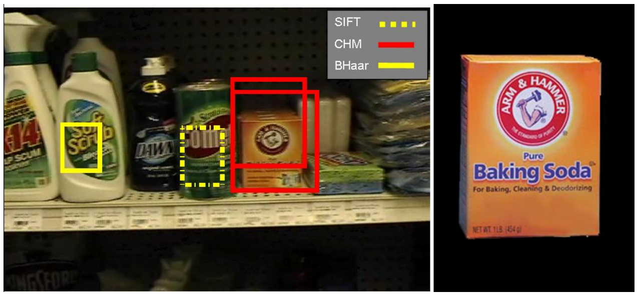


Figure 9. Localization product 52 with *in vitro* image. On the right: disparity of performances between algorithms.

Figures 11, 10 and 9 are examples of different cases of the localization efficiency of the 3 algorithms on the dataset. From Figure 9 we see why CHM performs well for product 52, while SIFT and boosted Haar-like features are not as good. In fact, object 52 has a distinctive orange color, which clearly stands out of the background. The color based approach can easily extract it from the rest of the frame. On

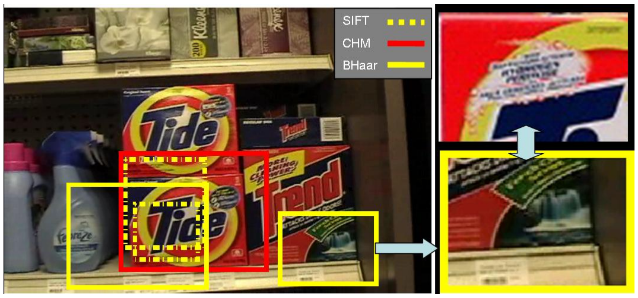


Figure 10. Example of good localization performance by all the methods for product 34. The *in vitro* image detail on the top right resembles the only false positive.

the other hand, the remaining two methods cannot rely on such precious information and they are misled by the text on the neighboring products. Color histogram matching instead does not require great sharpness, since it relies on more global than local information. Figure 10 shows good localization by all the methods. The characteristics of the product (clear and distinctive colors, multiple pattern variations) generates informative features for every algorithm. Only boosted Haar like features are partially misled by a neighbor. Taking a closer look at the misdetection and a fraction of an *in vitro* sample, it is apparent the similarity between the two.

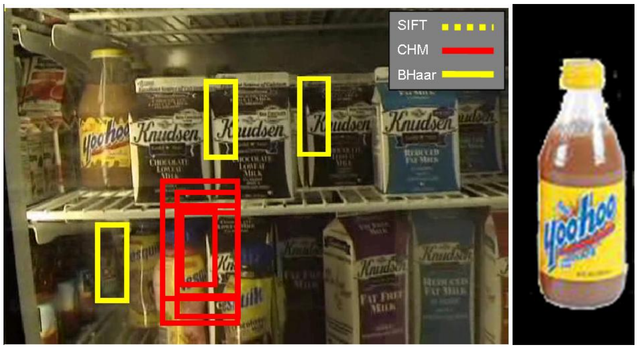


Figure 11. Example of poor localization performance for product 74. On the right: *in vitro* image sample on the right.

Finally we present a bad localization example: product 74. In this case we registered a failure from all the approaches, due to different causes. CHM is misled by portions of the frame very similar in color to some parts of the *in vitro* image. In particular the yellow color parts of the Nesquick bottles and their small blue caps, together with the dark brown of the back of the fridge and the parts of the milk box. For SIFT, the image is too rich with keypoints for the algorithm to be able to succeed, in particular because the product of interest is mainly composed by uniform patterns, so that the keypoints are found on corners in the writing. The abundance of writing labelling the other products in the frame, together with the reflectance on the transparent surface of the Yoohoo bottle prove deceptive for the algorithm. Boosted Haar like features on the other hand, trains (in grayscale) on a product which is dark on the top and the bottom, and light in the middle, with some texture inside. The windows selected in the frame present an exaggeration of the characteristics of the product: white (instead of yellow, which in grayscale corresponds to light grey) plus some pattern in the middle, and almost black on the top and bottom extremes. Again, the cluttered background presents a big variety of misleading regions and difficult to classify.

6. Discussion and future work

Our contributions in this paper include (a) a new multimedia database for studying object recognition *in situ* (i.e., sitting in the real world) using training images from an *in vitro* source (i.e., captured under ideal conditions) and (b) a baseline performance figures of three widely used recognition/detection algorithms that highlight the challenge presented by this database. The dataset contains both *in situ* and *in vitro* representations of the same products, and it presents a wide range of diversity among them in size, color, rigidity, shape, illumination, viewpoint and quality. Gathering useful training information from images captured in ideal conditions is linked to the semantic web image retrieval issue, addressed among others by [17]. In this work the authors demonstrate the avail of common image search metrics applied to images captured with a camera-equipped mobile device to find matching images on the World Wide Web or other general-purpose databases.

Our database offers not only a testbed for such approaches trying to link real world data to clean web or studio images, it can also be seen as an intermediate step or a bridge between the two representations of the same object. In fact, web retrieval can benefit from a set of pre-existing labeled samples, while at the same time dynamically increasing and improving such a set. This type of problem can find applications in assistive technologies for the visually impaired and also in mobile robot navigation. In fact, in the case of a blind or visually impaired person that uses a device that recognizes products in a grocery store, it would be impractical to acquire *in situ* data every time we need to train the system, thus the *in vitro* data captured from the web is a good source of training data.

Therefore we intend as future work to use the *in vitro* data set as a seed to build upon, as the user base continues to use the database and expanding the *in situ* part as a mean of test for different algorithms. We also plan to fuse methods in order to improve the results and we also plan to make use of context information about physical object proximity, identifying products nearby on the shelf to improve localization of objects in natural scenes.

Prod.	Overall Recall			Overall Precision			TP			FP		
	Haar	CHM	SIFT	Haar	CHM	SIFT	Haar	CHM	SIFT	Haar	CHM	SIFT
1	6.3	**11.6**	5.3	**15**	1.3	1.6	**20**	0	0	45	92	**40**
18	4.5	**33**	7.4	1.3	**13**	0.8	5	**20**	3	91	75	**60**
27	**27**	4.8	0	**14**	1	0	**30**	0	0	83	94	**15**
34	22.5	**48.6**	5	29	**54**	11	40	**75**	36	41	**0**	31
38	21	**53**	1.1	8	**41**	3.4	15	**60**	4.1	81	49	**25**
51	9.3	**31.7**	7.7	**41**	37	12	**60**	30	29.5	76	93	**64.5**
52	1.7	**78**	1.5	3.1	**51.6**	5.6	5	**100**	2.4	93	66	**9.9**
74	5	**38.8**	0	5.5	**16.7**	0	5	**20**	0	65	96	**12.7**
96	8.7	**51**	0.4	6.2	**47.4**	0.2	15	**55**	0	53	43	**14.4**
119	10.6	**43.3**	1.8	54	**41.7**	14.4	25	**60**	34.8	33	**9.1**	11.4

Table 2. Localization results (expressed in percentages) for the 10 experiments products (Prod.). Methods analyzed: Haar, Boosted Haar-like features; CHM, Color Histogram Matching; SIFT, SIFT keypoints based matching. The results show the difficulty of the problem. The bold numbers indicate the best performing method on each product, according to the different metrics. We can observe that for product 52 the overall recall and precision are low because they require to have a good overlap between the algorithms output and the ground truth area. Only CHM gives a good performance because it has multiple windows around the ground truth.

Acknowledgments The authors would like to thank Vincent Rabaud, Stephan Steinbach, Anton Escobedo, Al Labotski and Robert Meza for valuable help and feedback, as well as all the people involved in the GroZi project at UCSD. Michele Merler was supported by the California Institute for Telecommunications and Information Technology (Calit2) 2006 Summer Undergraduate Research Scholarship Program. Partial support was also provided by NSF CAREER Grant #0448615 and the Alfred P. Sloan Research Fellowship.

References

[1] F. Bashir and F. Porikli. Performance evaluation of object detection and tracking systems. *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS)*, June 2006. 5

[2] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *PAMI*, pages 509–522, 2002. 2

[3] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. *IEEE. CVPR 2004, Workshop on Generative-Model Based Vision*, 2004. 2

[4] Y. Freund and R. Schapire. Experiments with a new boosting algorithm. *Proceedings of the Thirteenth International Conference on Machine Learning, Morgan Kaufmann.*, page 148156, 1996. 2

[5] J. M. Geusebroek, G. J. Burghouts, and A. W. M. Smeulders. The Amsterdam library of object images. *Int. J. Comput. Vision*, 61(1):103–112, 2005. 2

[6] C. Harris and M. Stephens. A combined edge and corner detector. *4th Alvey Vision Conference*, pages 189–192, 1988. 2

[7] D. Koubaroulis, J. Matas, and J. Kittler. Evaluating colour-based object recognition algorithms using the soil-47 database. *Asian Conference on Computer Vision, 2002.*, 2002. 2

[8] J. Y. Lee and S. I. Yoo. An elliptical boundary model for skin color detection. *Proc of. the International Conference on Imaging Science, Systems, and Technology*, 2002. 2

[9] B. Leibe and B. Schiele. Analyzing appearance and contour based methods for object categorization. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'03)*, June 2003. 2

[10] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision, 60, 2*, pages 91–110, 2004. 2, 4, 5

[11] V. Y. Mariano, J. Min, J.-H. Park, R. Kasturi, D. Mihalcik, H. Li, D. Doermann, and T. Drayer. Performance evaluation of object detection algorithms. *ICPR*, 03:30965, 2002. 5

[12] Pascal. The pascal object recognition database collection. 2005. 2

[13] J. Ponce, T. L. Berg, M. Everingham, D. A. Forsyth, M. Hebert, S. Lazebnik, M. Marszalek, C. Schmid, B. C. Russell, A. Torralba, C. K. I. Williams, J. Zhang, and A. Zisserman. *Dataset Issues in Object Recognition*. Toward Category-Level Object Recognition, Springer-Verlag Lecture Notes in Computer Science., 2006. 1, 2

[14] F. Porikli. Integral histogram: A fast way to extract histograms in cartesian spaces. *CVPR Vol. 1*, pages 829–836, 2005. 2, 5

[15] M. J. Swain and D. H. Ballard. Color indexing. *Int. J. Comput. Vision*, 7(1):11–32, 1991. 1

[16] P. Viola and M. Jones. Robust real-time object detection. *International Journal of Computer Vision*, 2002. 2

[17] T. Yeh, K. Tollmar, and T. Darrell. Searching the web with mobile images for location recognition. *CVPR*, 02:76–81, 2004. 7