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 the web, 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. Nearly all of the algorithms that address these problems demand large amounts of labeled training examples for each object. Moreover, it is generally necessary that the training examples be acquired under environmental conditions (illumination, pose, etc.) that are representative of what will be observed in the testing data.

In many real-world applications, however, having access to training data for which the appearance is drawn from the same distribution as that of the testing data is not practical. Examples of such applications include assistive vision technology for the blind and object recognition for mobile robots. Consider for example the Semantic Robot Vision Challenge (SRVC)[1], a new research competition designed to push the state-of-the-art in image understanding and automatic acquisition of knowledge from large unstructured databases of images (such as those generally found on the web). Central to the challenge is the problem of training a "laptop-on-wheels" object recognizer using only a textual list of objects (provided shortly before the competition) and access to an internet image search engine. In the case of our application of interest, we seek to design a portable vision system for helping blind people find groceries in a supermarket, with a starting point again consisting of a textual list – in this case a shopping list – and access to publicly available grocery images on the internet.

The web contains an immense collection of structured and unstructured image databases and offers the potential to generate useful models for image recognition and localization. In order to explore this potential in at least one application domain, we built a multimedia database consisting of 120 different grocery items easily retrievable from the web. For every product, two different recordings are available: *in vitro* images extracted from the web (captured under ideal imaging conditions), and *in situ* images extracted from camcorder video collected inside a grocery store. We refer to a grocery product with its corresponding UPC code as an object. 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. In particular, methods such as [5] could be used to discover and label large amounts of *in situ* object examples in photos that currently lack textual annotation using *in vitro* training data from online resources such as Amazon Groceries.

Our goal in this paper is to provide a highly varied real world multimedia database for studying object recognition and detection with the above-described distinction between training and testing data, as well as baseline performance

[1] http://www.cs.cmu.edu/ ∼prybski/SRVC/

1-4244-1180-7/07/$25.00 ©2007 IEEE

SES-1018
11,087,272

Authorized licensed use limited to: Penn State University. Downloaded on June 13,2024 at 18:52:04 UTC from IEEE Xplore. Restrictions apply.

of several approaches used to solve object localization and recognition. In Section 2 we first review the state of the art object detection and recognition algorithms used in our baseline study, and then we discuss 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 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 [20] opened the way to many different approaches with a common ground. Computational complexity, which has always been one major bottleneck of the histogram extraction and comparison based search tasks, has recently been overcome with the introduction of the integral histogram by Fatih Porikli [19]. On the other hand, the choice of which colorspace or chrominance plane to use is still an open issue [14].

In addition to color features, one can employ methods based on on shape [4] 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. The most commonly used interest point detector is probably the Harris corner detector [12], while for the second problem, David Lowe's SIFT [16] descriptor has been shown to outperform the others, as it 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 [21], who introduced a fast and reliable classifier. They suggested using a cascade of classifiers, each of which is a combination of a subset of weak learners selected via Adaboost [8]. The weak learners use Haar-like features, which are responses of filters computed extremely quickly by employing integral images.

2.2. Computer vision databases

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

The Pascal Object Recognition Database Collection [1] 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 [7], 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 [13], 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 database like SOIL-47 may represent an extremely rich *in vitro* training set for the problem we investigate, but it lacks the *in situ* counterpart that is needed for testing.

A more recent computer vision database is the Amsterdam Library of Object Images (ALOI) [9], 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 [15] 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.

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

Authorized licensed use limited to: Penn State University. Downloaded on June 13,2024 at 18:52:04 UTC from IEEE Xplore. Restrictions apply.

the sizes and position of objects inside the images are normalized, as pointed out in [18]. In contrast, we propose a dataset that contains images that present a significant difference in quality (between training and testing) 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[2] is a multimedia database of 120 grocery products. It was created in order to address the problem of learning using training data that differs in quality from the testing data for object recognition and localization tasks. 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 testing 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 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[3].

In order to obtain *in vitro* representations we queried the web using a list of approximately 4000 UPC codes associated with the products of a local grocery store. A Perl script was generated to automatically crawl the web looking for images of the products by using the UPC code. In case the system did not respond positively to a query, a textual description of the product was obtained from the UPC online database[4], and the query was performed again with the new keywords. A manual selection of the downloaded instances was executed in order to eliminate duplicates and images representing products different from the desired one. 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 shown in Figure 5 (a).

	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 unique instances of a given product.

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

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

Figure 3. Sample of *in situ* images for the above products.

Therefore the *in vitro* images become easy to analyze and at the same time, coming from different online vendors and stock photo suppliers, they include a variety of illuminations, sizes and poses. Figure 1 shows the different views of a particular product represented by *in vitro* images. In contrast, *in situ* representations come from natural environments (real world). Figure 3 shows the *in situ* representations of the objects in Figure 2. We went to the grocery store that provided us the UPC list (a campus convenience store), and 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

[2]http://grozi.calit2.net
[3]http://www.froogle.com
[4]http://www.upcdatabase.com

Authorized licensed use limited to: Penn State University. Downloaded on June 13,2024 at 18:52:04 UTC from IEEE Xplore. Restrictions apply.

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. Grozi-120 constitutes a database of mixed multimedia content, with images and videos representing the same objects under different visual conditions, together with their text annotations. The creation of such a dataset involved different multimedia resources, spanning from the web to digital camcorders.

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.

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. We computed histograms of 16 bins per channel, a and b, calculated separately, for a total of 32 bins. Then 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.

SIFT: We computed SIFT [16] 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[5]. 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 (albeit weakly) captured by the descriptors, since they are computed on the masked images.

Boosted Haar-like features: We used the implementation in the Haar training utility of the Intel OpenCV library[6]. Positive samples were synthetically created from the *in vitro* images by applying randomly generated perspective distortions, until 200 positive samples were 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.

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.

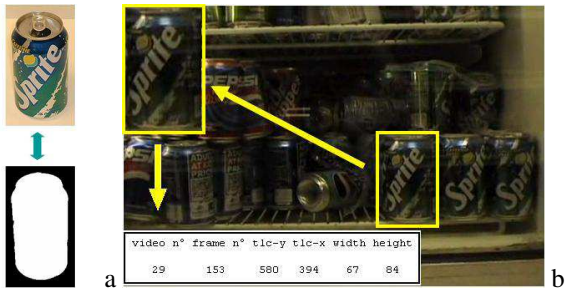

Figure 5. (a) Binary mask applied to web samples. (b) Product image cropped from video frame and stored together with coordinates.

4.2. Recognition

In our recognition study, *in vitro* images were used as training data, while *in situ* images were used as test data. 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. We obtained the best performance using color histogram intersection.

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

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

Authorized licensed use limited to: Penn State University. Downloaded on June 13,2024 at 18:52:04 UTC from IEEE Xplore. Restrictions apply.

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. 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 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 14 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. We took 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 and false positives rates, and a metric based on the average object area recall and the average detected area precision as defined in [17], that gives us the *overall recall* and *overall 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 [3]. 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 [19]. We compute the *in vitro* template histogram in the same way as in recognition. Then, we performed a frame by frame analysis first computing the integral histogram (for the a and b channels) of the whole frame and then moving a window at 5 different scales in raster scan order around the frame, computing the color histograms of different regions. Then those histograms are intersected with the *in vitro* template. 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. Then, we matched the keypoints against the frame features as in [16]. 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. For recognition, we chose 10 *in vitro* image samples for every product for training, corresponding to the highest number of SIFT keypoints, considering that a higher number of keypoints often means an image with lower blurriness. Synthetic images were created when the original number of samples was less than 10. Subsequently we chose 10 *in situ* images per object for testing using the same criteria, but not considering synthetic images. In Figure 6 we report the average performances. It can be noticed how the overall best performing algorithm is SIFT, while the color histogram matching curve (CHM), after an initial flat behavior, rapidly grows towards a 99 percent match rate. Finally, Adaboost offers the worst performance. One of the reasons for

Authorized licensed use limited to: Penn State University. Downloaded on June 13,2024 at 18:52:04 UTC from IEEE Xplore. Restrictions apply.

such a difference relies on the lack of a consistent number of original samples for training, alleviated by the generation of synthetic images, which do not offer effectively diverse characteristics. SIFT and CHM, on the other hand, need only very few samples to extract the necessary information to perform recognition. 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. On the other hand, in Figure 8, CHM benefits from the multimodal histogram obtained from a product presenting colors (yellow, orange and green) very well distinguishable in the ab plane, while SIFT and Haar-like features encounter a lack of distinctive feature points on most of the product's surface.

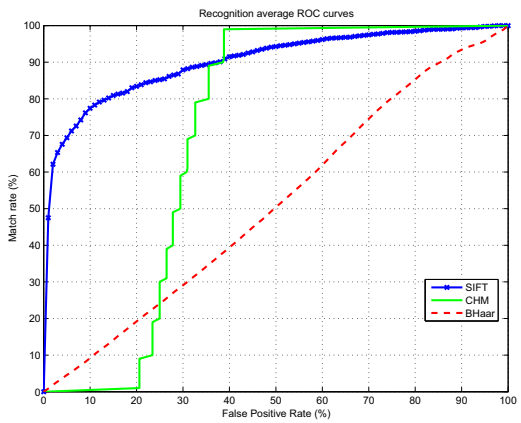

Figure 6. Average ROC curves for all the Products.

In the case of localization we used 100 frames with no instances of any product in the database to test the False Positive rates. Table 2 shows the localization rates in percentages for the three different algorithms. We use a yes/no rule to obtain true and false positives, based on whether the center of the detected box lies or not within ground truth region. Recall is defined as the proportion of the ground truth area covered by the detection boxes. Precision is defined as the proportion of the detection boxes that covers the ground truth area. The best and worst cases are selected considering the combination of all the 4 rates. Figure 9 presents the best and worst cases. Looking the results we can notice that SIFT outperforms the other methods, followed by CHM. For Adaboost the discussion reported for recognition still applies. The high values of the standard deviations in the table show that the methods do not perform uniformly over the set 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. While recognition yields decent results, the average performances in localization are poor.

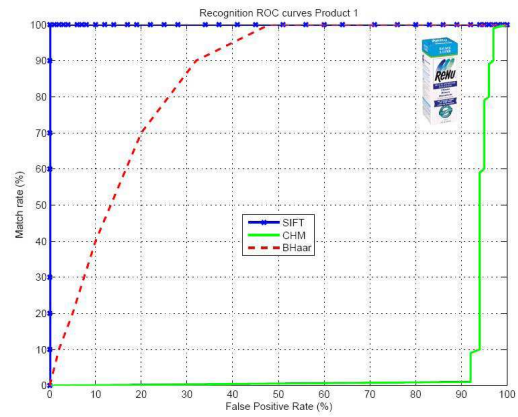

Figure 7. ROC curves for Product 1.

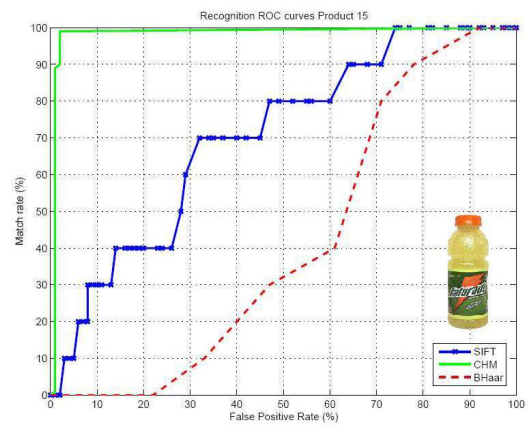

Figure 8. ROC curves for Product 15.

Such a discrepancy is due to the different nature of the 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 the core of the problem: find the correspondence between *in vitro* samples and cluttered, noise corrupted, realistic *in situ* scenes. Figures 10, 11 and 12 are examples of different cases of the localization performance of the 3 algorithms on the dataset. From Figure 10 we see CHM perform 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. On the other hand, the remaining two methods cannot rely on such precious information and they are misled by the text on the

Authorized licensed use limited to: Penn State University. Downloaded on June 13,2024 at 18:52:04 UTC from IEEE Xplore. Restrictions apply.

neighboring products. Figure 11 shows good localization by all the methods on the product where SIFT performs best. The characteristics of the product (clear and distinctive colors, multiple pattern variations) generate 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, the similarity between the two is evident.

CHM	% Recall	% Precision	% TP	% FP
Mean	15	17	18	65
Std Dev.	28	16	35	32
Best (20)	71	82	100	4
Worst(32)	1.7	0.2	0	100

SIFT	% Recall	% Precision	% TP	% FP
Mean	72	18	22	62
Std Dev.	20	17	26	28
Best (34)	14	83	93	25
Worst (9)	26	0.9	0	64

ADA	% Recall	% Precision	% TP	% FP
Mean	15	17	18	65
Std Dev.	13	13	19	24
Best (92)	35	74	50	38
Worst (5)	0.5	0.2	0	92

Table 2. Localization performances of the 3 algorithms. Mean, Standard Deviation, Best Case (product number in brackets) and Worst Case percentages are reported for Overall Recall, Overall Precision, True Positives and False Positives.

Figure 9. Best and Worst localization samples for CHM, SIFT and ADA. The products 32, 9 and 5 correspond to the worst results and 20, 34 and 92 the best.

Finally we present a bad localization example: product 74. CHM is misled by portions of the frame very similar in color to some parts of the *in vitro* image. 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. This together with the reflections on the palstic surface of the Yoohoo bottle prove deceptive for the algorithm. Boosted Haar like features on the other hand, finds (in grayscale) a product which is dark on the top and the bottom, and light in the middle, with some texture inside.

6. Discussion and future work

Our contributions in this paper include (a) a new multimedia database – GroZi-120 – 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) baseline performance figures of three widely used recognition/detection algorithms that highlight the challenge presented by this database. The Grozi-120 database contains both *in situ* and *in vitro* representations of the same products, and it presents a wide range of variability for each product with respect to size, color, rigidity, shape, illumination, viewpoint and quality.

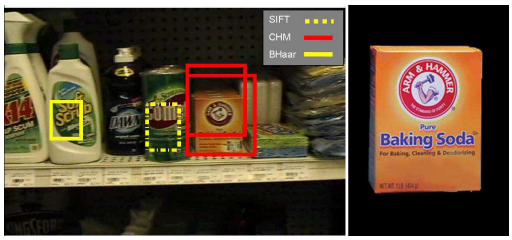

Figure 10. Localization example for Product 52. On the right: *in vitro* image sample.

Figure 11. 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.

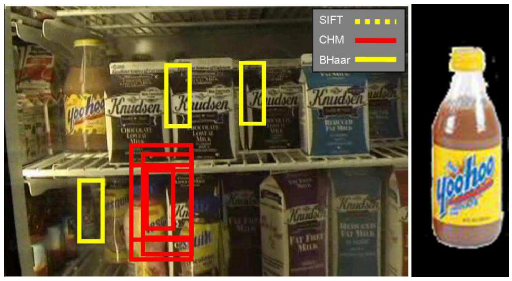

Figure 12. Example of poor localization performance for Product 74. On the right: *in vitro* image sample.

Gathering useful training information from images captured in ideal conditions is linked to the semantic web image retrieval issue, addressed among others by [22]. In this work the authors demonstrate the application of common image search metrics applied to images captured with a camera-equipped mobile device to find matching images on the Web or other general-purpose databases. The results of the testbed suggest the need for more precise and elaborate detection/recognition algorithms to solve the proposed problem. In fact, the aim of our work was to provide a baseline against which to test new methods. Color histogram matching may benefit from the works on color invariants, e.g., [11] and [10]. Another interesting direction may consist in a combination of the grayscale and color based methods, such as [2] and [6]. Our database offers not only a

Authorized licensed use limited to: Penn State University. Downloaded on June 13,2024 at 18:52:04 UTC from IEEE Xplore. Restrictions apply.

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 and multimedia 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 mobile robot navigation like the Semantic Robot Vision Challenge (SRVC) and also in assistive technologies for the visually impaired . 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 means of evaluating different algorithms. Furthermore, we plan to include more objects in the database and expand the number examples per object. We also plan to fuse recognition and detection methods in order to improve the results as well as to make use of context information about physical object proximity, identifying products nearby on the shelf to improve localization of objects in natural scenes.

Acknowledgments

The authors would like to thank Vincent Rabaud, Stephan Steinbach, Anton Escobedo, Al Labotski, Robert Meza and Michael McQuen 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] The pascal object recognition database collection. 2005. 2

[2] A. E. Abdel-Hakim and A. A. Farag. CSIFT: A SIFT descriptor with color invariant characteristics. In *CVPR06*, pages 1978–1983, 2006. 7

[3] 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

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

[5] N. Ben-Haim, B. Babenko, and S. Belongie. Improving web-based image search via content based clustering. *International Workshop on Semantic Learning Applications in Multimedia (SLAM), New York City.*, 2006. 1

[6] T. Burghardt and J. Calic. Analysing animal behaviour in wildlife videos using face detection and tracking. *IEEE Proceedings - Vision, Image, and Signal Processing*, 153(3):305–312, June 2006. 7

[7] 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

[8] 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

[9] 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

[10] J.-M. Geusebroek, R. van den Boomgaard, A. W. M. Smeulders, and A. Dev. Color and scale: The spatial structure of color images. In *ECCV*, pages 331–341, 2000. 7

[11] T. Gevers and H. Stokman. Robust histogram construction from color invariants for object recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(1):113–117, 2004. 7

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

[13] 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

[14] 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

[15] B. Leibe and B. Schiele. Analyzing appearance and contour based methods for object categorization. *IEEE CVPR'03*, June 2003. 2

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

[17] 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

[18] 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. 3

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

[20] M. J. Swain and D. H. Ballard. Color indexing. *Int. J. Comput. Vision*, 7(1):11–32, 1991. 2

[21] P. A. Viola and M. J. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR (1)*, pages 511–518, 2001. 2

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

Authorized licensed use limited to: Penn State University. Downloaded on June 13,2024 at 18:52:04 UTC from IEEE Xplore. Restrictions apply.