Automatic produce classification from images using color, texture and appearance cues

Anderson Rocha Daniel C. Hauagge Jacques Wainer Siome Goldenstein

Institute of Computing
University of Campinas (Unicamp)
13084–851, Campinas, SP – Brazil
{anderson.rocha, wainer, siome}@ic.unicamp.br
hauagge@liv.ic.unicamp.br

Abstract

We propose a system to solve a multi-class produce categorization problem. For that, we use statistical color, texture, and structural appearance descriptors (bag-of--features). As the best combination setup is not known for our problem, we combine several individual features from the state-of-the-art in many different ways to assess how they interact to improve the overall accuracy of the system. We validate the system using an image data set collected on our local fruits and vegetables distribution center.

1. Introduction

Recognizing different kinds of vegetables and fruits is a recurrent task in supermarkets, where the cashier must be able to point out not only the species of a particular fruit (i.e., banana, apple, pear) but also it's variety (i.e., Golden Delicious, Jonagold, Fuji), which will determine its price. The use of barcodes has mostly ended this problem for packaged products but given that consumers want to pick their produce, they can not be packaged, and thus must be weighted. A common solution to this problem is issuing codes for each kind of fruit/vegetable; which has problems given that the memorization is hard, leading to errors in pricing.

As an aid to the cashier, many supermarkets issue a small book with pictures and codes; the problem with this solution is that flipping over the booklet is time-consuming. In this paper, we review several image descriptors in order to propose a system to solve the problem by adapting a camera to the supermarket scale that identifies fruits and vegetables based on color, texture, and appearance cues.

Formally, we state our problem in the following manner: given an image of fruits or vegetables, in arbitrary position and number, the system must return a list of possible candidates of the form (species, variety). Sometimes, the object can be inside a plastic bag that can add specular reflections and shifts to the hue.

Given that the big variety and the impossibility of predicting which kinds of fruit/vegetables are sold, training must be done on site by someone with little or no technical knowledge. Therefore, the system must be able to achieve a high level of precision with only a few training examples (e.g, up to 30 images). Another desirable characteristic would be continuous learning. On one hand, more training data would be generated as the system commits mistakes and the cashier corrects them. On the other hand, in this semi-supervised scenario, eventually the operator will commit mistakes and the learning algorithm must be robust to noisy training data.

Here, we combine local and global features into a linear discriminant classifier. We have used global color histograms, local texture, and correlation descriptors with distinctive fruit parts, learned from an image data set collected from the internet. Using those images comprising a multitude of classes, we endow our system the capacity of identifying parts that are specific to the species being analyzed (the apple's calyx, for example).

The contribution and novelty in this paper are twofold. The first is that we evaluate several image descriptors in the literature and point out the best ones to solve our multi-class fruits/vegetables categorization problem. Important questions about such descriptors are: which ones require less training? Is it necessary to use complex approaches such as bag-of-features or constellation models? How do the descriptors perform when increasing the number of training examples? What combinations and parameters of the descriptors provide better effectiveness? How do the descriptors behave under the curse-of-dimensionality? In the experiments we show answers for such questions. We also show that, in some situations, it may be worthier using simpler solutions that yield faster results than more complex and time-consuming approaches. The second contribution and novelty is that we create an image data set collected from our local fruits and vegetables distribution center and make it public. In general, there are a few well-documented image data sets freely available for testing algorithm performance in image categorization and content-based image retrieval tasks. In this context, we provide an image data set with 11 categories of fruits and more than 2,000 images collected on site with all its creation details. We have collected such image data set in several runs during five months. This added variability to the data sets with respect to environmental changes (e.g., weather, illumination).

In Section 2, we give a brief overview of previous work in object recognition and image categorization. In Section 3, we present the different kinds of image descriptors we have used in this paper and describe their best combination for our problem. In Section 4, we report results for the image data set we have created. Finally, in Section 5, we draw conclusions and future directions.

2. Literature review

Recently, there has been a lot of activity in the area of *Image Categorization*. Previous approaches have considered patterns in color, edge and texture properties [13, 17, 19]; low- and middle-level features to distinguish broad classes of images [5, 14, 15]; In addition, Heidemann [9] has presented an approach to establish image categories automatically using histograms, colors and shape descriptors with an unsupervised learning method.

VeggieVision [4] was the first attempt of a supermarket produce recognition system. However, as this system was presented long time ago, it does not take advantage of recent achievements of the literature. For instance, *VeggieVision* has 95% accuracy in some scenarios but to achieve such result it uses the top four responses.

In general, we can view our problem as a special instance of object's categorization. Turk and Pentland [18] employed principal component analysis and measured the reconstruction error of projecting the image to a subspace and returning to the original image space. We believe this is ill suited for our purpose because it depends heavily on illumination, pose and shape.

Viola and Jones [20] have presented an approach with localization speed and precision in recognition employing a cascade of classifiers composed of simple features and trained with the Ada-Boost algorithm. The main drawback of this approach for our problem is that it would require same format objects previously aligned. Other issue relates to the very costly training which often requires thousands of images.

Recently, Agarwal et al. [1] and Jurie and Triggs [10] have adopted approaches that break down the categorization problem to the recognition of specific parts that are characteristic of each object class. These techniques, generally called bag-of-features [8, 12, 16], have shown promising results even though they do not try to model spatial constraints among features.

Weber [21] takes into account spatial constraints using a generative constellation model. The algorithm can cope with occlusion in a very elegant manner, albeit very costly (exponential in the number of parts). A further development made by Fei-Fei et al. [6] introduced prior knowledge into the estimation of the distribution, thus reducing the number of training examples to around 10 images while preserving a good recognition rate. Notwithstanding, even with this improvement, the problem of exponential growth with the number of parts persists, which makes it unpractical for our problem, which requires speed for on-line operation.

Another interesting technique was proposed by Malik [2]. In this work, feature points are found in a gradient image. The points are connected by a joining path and a match is signalized if the found contour is similar enough to the one in the database. A serious drawback of this method for our problem is that it requires a nonlinear optimization step to find the best contour; besides that it relies too heavily on the silhouette cues, which are not a very informative feature for fruits like oranges, lemons and melons.

3. Materials and methods

In general, image categorization relies on combinations of statistical, structural and spectral approaches. In statistical approaches, we describe the objects using global and local descriptors such as mean, variance, and entropy. In structural approaches, we represent the object's appearance using well-known primitives such as patches of important parts of the object. Finally, in spectral approaches, we describe the objects using some spectral space representation such as Fourier spectrum [7]. In this paper, we analyze statistical color and texture descriptors as well as structural appearance descriptors to categorize fruits and vegetables in a multi-class scenario.

As the best combination of features was not known for our problem, we have combined the individual features in several ways to assess how they interact to improve the overall accuracy. Feature description combination is of particular interest in the literature and has demonstrated important results over the last years [11]. However, the approach presented in [11] employs EXIF information which are not relevant for our produce classification system.

In the following, we present the statistical and structural descriptors we have analyzed and used in this paper, as well as the data set we have created for the validation process.

3.1. Image acquisition

We have created two image data sets for this work: *Internet* and *Supermarket Produce* data sets. The *Internet* data set, comprises images retrieved from the internet using Google Images[1]. We used several fruit names from different languages, such as English, Spanish, Portuguese, German, and French. The *Internet* data set initially comprised about 70,000 images, which were visually inspected, to finally render about 6,000 images. We used this data set to build a visual vocabulary of parts of fruits, to be used when extracting appearance descriptors.

The *Supermarket Produce* data set is one of our contributions in this paper. In Section 3.2, we present its detailed description.

3.2. Supermarket Produce data set

The *Supermarket Produce* data set is one of our contributions in this paper[2]. In general, there are a few well-documented image data sets available for image categorization and content-based image retrieval tasks for testing algorithm performance. ALOI[3] and Caltech[4] are two examples of such data sets. In this paper, we provide an image data set with 11 categories of fruits comprising more than 2,000 images collected on site.

The *Supermarket Produce* data set is the result of five months of on site collecting in our local fruits and vegetables distribution center.

We have used a Canon PowerShot P1 camera, at a resolution of $1,024 \times 768$ pixels. For the experiments in this paper, we down-sampled the images to 640×480. For all images, we have used a white background. We acquired images from 11 different categories: Plum (264), Agata Potato (113), Cashew (210), Kiwi (171), Fuji Apple (212), Granny-Smith Apple (155), Watermelon (192), Honeydew Melon (145), Nectarine (247), Williams Pear (159), and Diamond Peach (211); totalizing 2,078 images. Figure 1 depicts some of the classes of our image data set.

The data set comprises images with different illumination parameters even within the same category. All the images were stored in the RGB color-space with 8 bits per pixel. We continue to increase the number of images and categories in this data set[5].

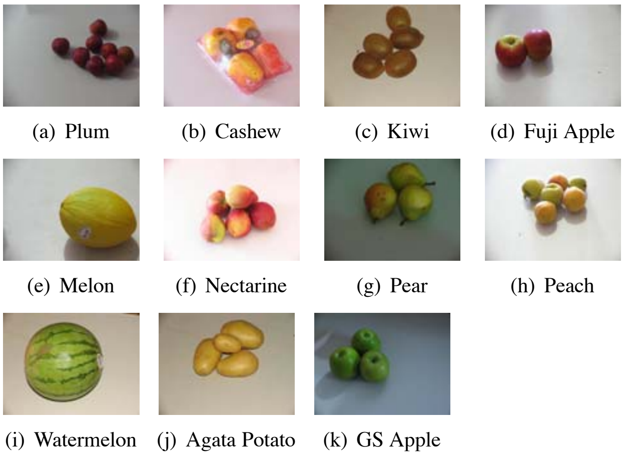
(a) Plum (b) Cashew (c) Kiwi (d) Fuji Apple
(e) Melon (f) Nectarine (g) Pear (h) Peach
(i) Watermelon (j) Agata Potato (k) GS Apple

Figure 1. Supermarket Produce data set.

We have gathered images in different times of the day and in different days for the same image category. These features increase the data set variability and represent a more realistic scenario. Image 2 shows an example of Kiwi category with different illumination parameters. The illumination changes are due to daylight exposure only and are not artificially tampered.

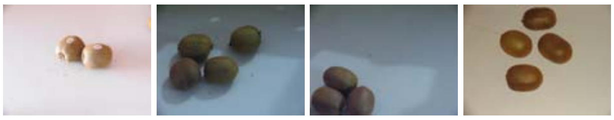

Figure 2. Kiwi Category. Illumination differences within categories.

The *Supermarket Produce* data set also comprises differences in pose and in the number of elements within an image. Figure 3 shows examples of the Cashew category. Note that there are variations in the pose of the Cashew's plastic repository. In addition, Figure 4 show the variability in the number of elements within an image. Sometimes, some fruits are apart from one another.

Sometimes the elements are within a plastic bag which adds specular reflections to the analyzed image. Furthermore, the presence of shadows (e.g., second and third images of Figure 2) and cropping/occlusions (e.g., Figure 5) makes the data set more realistic.

[1] `http://images.google.com`

[2] Freely available from `http://www.liv.ic.unicamp.br/∼undersun/pub/`

[3] `http://staff.science.uva.nl/∼aloi`

[4] `http://www.vision.caltech.edu/Image_Datasets/`

[5] Our current configuration is 2,633 images from 15 different categories. The newest added categories are: Asterix Potato, Onion, Orange, and Taiti Lime.

Figure 3. Pose differences. Cashew category.

Figure 4. Variability on the number of elements. Plum category.

3.3. Image descriptors

In this section, we analyze statistical color, texture, and structural appearance descriptors (bag-of-features) in order to propose a system to solve a multi-class fruits/vegetables categorization problem. As the best setup of features is not known for our problem, we combine the individual features in several ways to assess how they interact to improve the overall accuracy of the system.

3.3.1. Unser's descriptors Unser [19] has showed that the sum and difference of two random variables with same variances are de-correlated and define the principal axes of their associated joint probability function. Hence, the author introduces sum s and difference d histograms as an alternative to the usual co-occurrence matrices for image texture description.

The non-normalized sum and difference associated with a relative displacement (δ_1, δ_2 for an image I, are define as

$$\begin{aligned} s_{k,l} &= I_{k,l} + I_{k+\delta_1,l+\delta_2}, \quad (1) \\ d_{k,l} &= I_{k,l} - I_{k+\delta_1,l+\delta_2}. \quad (2) \end{aligned}$$

The sum and difference histograms over the domain D are defined in a manner similar to the spatial level co-occurence or dependence matrix definition:

$$h_s(i;\delta_1,\delta_2) = h_s(i) = Card\{(k,l) \in D, s_{k,l} = i\}, \quad (3)$$

Figure 5. Examples of cropping and partial occlusion.

$$h_d(j;\delta_1,\delta_2) = h_d(j) = Card\{(k,l) \in D, d_{k,l} = j\}. \quad (4)$$

In addition to the histograms, we use some associated global measures: mean (μ), contrast (C_n), homogeneity (H_g), energy (E_n), variance (σ^2), correlation (C_r), and entropy (H_n)) over the histograms for each color channel.

3.3.2. Color Coherence Vectors (CCVs) Zabih et al. [13] have presented an approach to compare images based on color coherence vectors. They define color's coherence as the degree to which pixels of that color are members of large similarly-colored regions. They refer to these significant regions as coherent regions. Coherent pixels are part of some sizable contiguous region, while incoherent pixels are not.

In order to compute the CCVs, the method blurs and discretizes the image's color-space to eliminate small variations between neighboring pixels. Afterwards, it finds the connected components in the image aiming to classify the pixels within a given color bucket as either coherent or incoherent.

3.3.3. Border/Interior (BIC) Stehling et al. [17] have presented the border/interior pixel classification (BIC), a compact approach to describe images. BIC relies on the RGB color-space uniformly quantized in $4 \times 4 \times 4 = 64$ colors. After the quantization, the image pixels are classified as *border* or *interior*. A pixel is classified as *interior* if its 4-neighbors (top, bottom, left, and right) have the same quantized color. Otherwise, it is classified as *border*.

After the image pixels are classified, two color histograms are computed: one for border pixels and another for interior pixels. The two histograms are stored as a single histogram with 128 bins. BIC compares the histograms using the *dLog* distance function [17]

$$dLog(q,d) = \sum_{i=0}^{i<M} \|f(q[i]) - f(d[i])\| \quad (5)$$

$$f(x) = \begin{cases} 0, & \text{if } x = 0 \\ 1, & \text{if } 0 < x < 1 \\ \lceil \log_2 x \rceil + 1, & \text{otherwise} \end{cases} \quad (6)$$

where q and d are two histograms with M bins each. The $q[i]$ value represents the ith bin of histogram q, and $d[i]$ represents the ith bin of histogram d.

3.3.4. Appearance descriptors To describe appearance, we use a vocabulary of parts, similar to Agarwal [1] and Jurie [10]. We sample the patches sparsely and use Lowe's feature points detector to find the interest points.

To find the representative parts among the many patches we use two algorithms: K-means, and a slightly modified version of the algorithm proposed by [1]. The second algorithm is a bottom-up clustering procedure. We introduced the modification (Equation 7) on the similarity metric between two clusters. Agarwal et al. [1] have proposed to join the clusters if their average similarity is above a threshold. In contrast, we connect two clusters if the similarity of any two patches, one from each cluster, is above the threshold. Similarity between two patches is simply their normalized correlation.

We create the feature vectors as a binary vector: 1's signalize that a particular feature is present in the image. When convenient, we show the name of the algorithm used and the size of the used feature vector. For instance, K-Means-99 refers to the use of K-Means algorithm on a feature-space of 99 dimensions. Often, the more complex an image, the more feature regions a feature point detector provides for it.

We introduced the modification in Equation 7 because we observed that frequently some clusters (C_i) were growing much faster than the others and they were incorporating patches that were not alike. We believe that this is because a bigger cluster was able to swallow smaller ones with dissimilar patches due to the average similarity measure, which allowed compensation of differences between two clusters. In fact, we noticed that more evenly-sized clusters were formed after this modification. The similarity between two groups of patches C_1 and C_2 is given by smallest distance among two patches (p_i, p_j), one from each group. We have used patches of 11×11 pixels

$$sim(C_1, C_2) = \min\{sim(p_i, p_j) | p_i \in C_1, p_j \in C_2\}. \quad (7)$$

We use two approaches for the vocabulary of parts. In the first one, we use some images from *Supermarket Produce* data set in the training stage. In the second one, in a quest to incorporate prior knowledge into the system, we put together images collected from the internet, resulting from queries to Google Images. We refer to these combinations pointing out the algorithm used and the source of the patches (e.g., Agarwal-Base, denotes we apply Agarwal algorithm using training patches from *Supermarket Produce* data set).

4. Experiments and results

In our experiments, we have used a Bagging Ensemble of Linear Discriminant Analysis (BLDA) with 17 iterations [3]. In Bagging (Bootstrap aggregation) ensemble, we repeatedly apply an inductive learner to bootstrap samples on the training set. We use the training set to generate bootstrap samples using random sampling with replacement. Once several hypotheses (i.e., base learners) have been generated on such bootstraps, we determine the aggregate classifier by majority voting among the base learners. The final classifier evaluates test samples by querying each of the base classifiers on the sample and then outputting their majority opinion.

Some can argue that Support Vector Machines (SVMs) can be more suited for categorization. However, in general, SVMs are more computational costly than LDA. Furthermore, for our specific problem here, when using bagging ensembles with LDA, the two approaches are comparable in effectiveness but BLDA is more efficient.

We select the training images using sampling without replacement from the pool of each image class. If we are training with 10 images per class, we use the remaining ones for testing. We repeat this procedure 10 times, and report (μ) and standard deviation (σ)

4.1. Average classification rates

In this paper, we show the results in terms of average classification rates because in a real application, it would be straightforward to figure out that 96% accuracy for 100 packages in the supermarket would mean they will get the correct produce classification in roughly 96 packages.

In Figure 6, we show the six best combinations of descriptors. The x-axis represents the number of images per class in the training set and the y-axis represents the accuracy in the testing set. Unser + BIC combination has produced the best results. This is interesting because BIC is a descriptor that analyzes color and shape in the sense that it codifies the object's border and interior. In addition, Unser is a texture descriptor that analyzes the relationship of the pixels and their neighbors. For 32 examples of each class in the training set, we have an average accuracy of about $93.3\% \pm 2.7\%$ while with 96 images of each class in the training set, we have an average accuracy of about $96.6\% \pm 0.1\%$. This shows us that the more images we use in the training, the better the overall accuracy. However, the payload is higher. If we triplicate the number of elements in the training set ($32 \rightarrow 96$) and, consequently, increase the training complexity, we improve the accuracy in 3 percentual points only (less than 2σ).

The inclusion of CCV descriptor does not yield an improvement on the classification rate of Unser + BIC combination, resulting an accuracy of about $91.7\% \pm 1.7\%$. However, Unser + BIC + CCV combination provides smaller variations in the average accuracy per class (Section 4.2).

The worst result of the top-six selected results is the combination of Unser + CCV + Agarwal Google, which combines the texture and color descriptors Unser and CCV with the Agarwal's appearance descriptors that use images from Google to create the initial vocabulary of patches. The inclusion of Agarwal in the combination of Unser and CCV does not provide a significant increase in the classification accuracy. And this shows us that for this particular problem the use of a more complex approach does not yield an improvement on the overall classification effectiveness. In Figure 7, we show the six worst combinations of the descriptors. The Unser descriptor provides the best result of the six-worst combinations. For this case, using 48 images

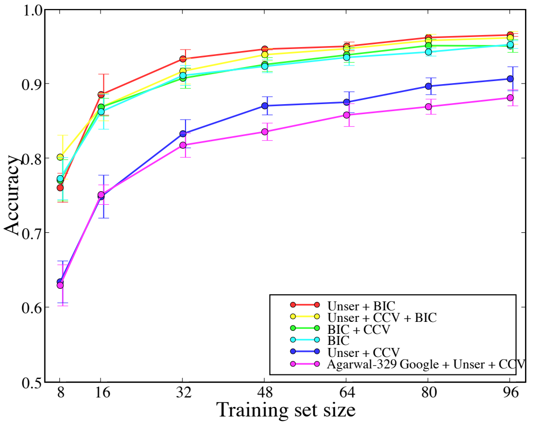

Figure 6. Six best combined classifiers.

per class in the training set, we have an average accuracy of about 81.8% ± 0.9%. The worst result is using Agarwal-Base, which uses the Agarwal's appearance descriptor and 10% of images selected from the *Supermarket Produce* data set to create the initial vocabulary of patches[6].

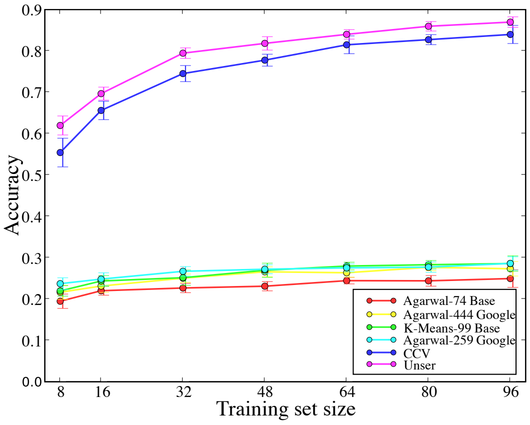

Figure 7. Six worst combined classifiers.

4.2. Average accuracy per class

In Figure 8, we plot the average accuracy per class of 10 executions of the BIC + CCV + Unser descriptors' combination. Although the best average accuracy showed in Figure 6 is the BIC + Unser combination, the BIC + CCV + Unser combination provides smaller variations (smaller standard deviations) in the accuracy per class. The more images in the training set, the better the average accuracy and the lower the standard deviations. If we use too much images in the training set, the overall process becomes more complex compromising the efficiency.

Agata Potato provides the best average classification accuracy per class (99.21% ± 1.6%) while Williams Pear is the class with the worst average classification accuracy (89.6% ± 2.2%) for 48 images per class in the training set.

4.3. Agarwal and K-means appearance descriptors' influence

The inclusion of a more complex approach such as Agarwal appearance descriptor for this particular problem does not yield an improvement on the classification accuracy. For instance, in Figure 9, we show that Agarwal appearance descriptors does not improve Unser + CCV combination's effectiveness.

The same is true when we apply K-means to cluster the patches selected from Google images and use them as representative appearance patches of fruits/vegetables images. The results are statistically the same given the standard deviation of the 10 validations we have performed. We believe that the Agarwal appearance descriptor does not provide a significant increase in the classification given that the used patches do not represent well all the images classes we are trying to classify. Further investigation must be done in this direction to validate the inclusion of Agarwal appearance descriptor or any other similar model.

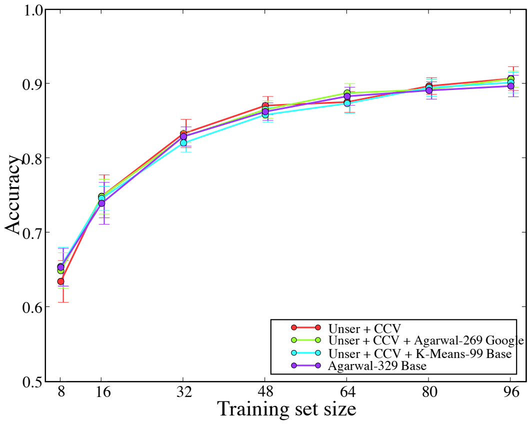

Figure 9. Agarwal vs. K-means contribution to Unser + CCV.

[6] The images used in Agarwal's vocabulary are removed from the base and they are not used in the further training and testing tasks.

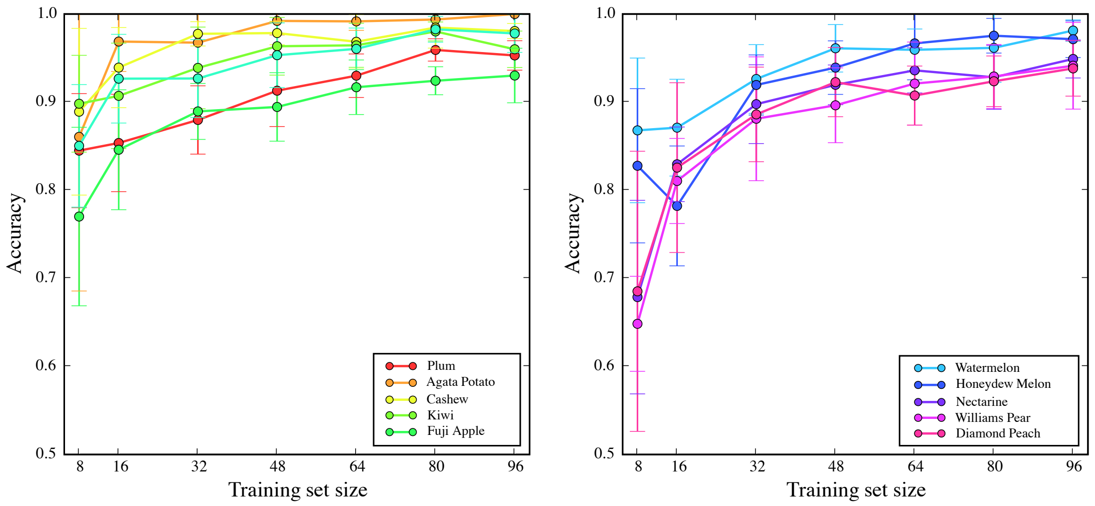

Figure 8. Average accuracy per class.

4.4. Most difficult images to classify

We have tested several combinations of different image descriptors for different sizes of training sets. Figure 10 depicts the top 20 misclassified images in our experiments considering all tested descriptor's combinations.

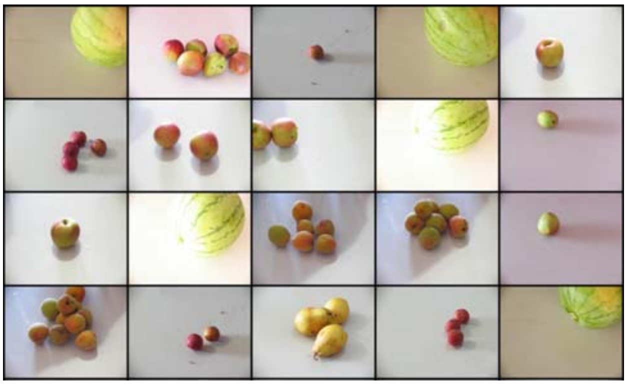

Figure 10. Top 20 misclassified images in our experiments.

Some of the images were overexposed, some are in a dirty background and some are blurred. There are some examples that, even for a human, are difficult to point out what class they belong to (e.g., the 10th image from top to bottom, left to right). The pre-processing would bring a positive impact to the recognition rates. We did not perform it because we focused in the description power without pre-processing, leaving space for further work from the community.

4.5. Average time

The average time for the image descriptors feature extraction and classification on an AMD 64 bits machine with 2GB of RAM using BLDA with 17 iterations is less than 1 second, even when they are combined. However, the more examples in the training set the more time consuming are the combinations in the training stage. To train a multi-class classifier using *Bagging* + *LDA* with 17 iterations, 96 training examples per class, and a BIC + CCV combined feature vector, it is necessary about one hour. If online training is required (c.f., Sec. 1), this can be a problem.

Some of the analyzed descriptors in this paper have high-dimensionality. For instance, the combination of the image descriptors BIC + CCV + Unser results in a feature vector of $128 + 64 + 189 = 381$ characteristics. This high-dimensionality is not so important when we use LDA because such approach naturally reduces the dimensions to a linear combination of them. Notwithstanding, due to the curse of dimensionality [3], it is an issue that must be handled. In this context, it is better to use combinations of low-dimension descriptors. Finally, the use of complex appearance descriptors such as Agarwal's approach can impact the training set without yielding significant improvements on the overall effectiveness of the classification system.

5. Conclusion and Future Work

In this paper, we have presented a complete and well-documented fruit/vegetables image data set suitable for content-based image retrieval, object recognition, and image categorization tasks. We hope this data set will endure beyond this paper as a common comparison set for researchers working in this space.

Furthermore, we have analyzed an extensive set of image descriptors and their combinations in the context of fruit/vegetables multi-class categorization. Such analysis is a milestone for other researchers interested in such problem. They can delve into this matter without the requirement of testing such huge combination of descriptors again.

We have found that Unser + BIC image descriptors combination provides the best average accuracies in the multi-class classification task. However, BIC + CCV + Unser combination provides a better combination when considering intra-class variability. Such combination yields smaller variations in the average accuracy per class.

Whether or not more complex approaches such as Agarwal's appearance descriptors and K-means associated with color and texture descriptors increase the classification rate is still an open problem. It would be unfair to conclude they do not help in the classification given that, their success is highly based on their patches representation. Nevertheless, it is fact that such approaches are computational demanding and perhaps not advisable in some scenarios.

The choice of the best image descriptors combination is difficult given that it must consider: (1) the average accuracy, (2) the average classification accuracy per class, (3) background's clutter, and (4) the final number of features in the combined description vector.

Adding more classes to the presented pool needs to be tested, mainly if the added classes have substantial overlapping with one another. However, our experiments showed that the tested approaches scale with the number of classes.

Further work include the improvement of the fruits/vegetables representative patches, and the analysis of other appearance and texture image descriptors. Furthermore, we are interested in the incorporation of spatial constraints among the local descriptors. We want to create the conditions for a semi-supervised approach that leads us to continuous learning, taking advantage of misclassified examples.

6. Acknowledgments

We thank people at our local fruits and vegetables distribution center (CEASA). In special, we thank Francisco Domingos of Agro Comercial Bella Vista Ltda. Finally, we thank FAPESP (Proc. 05/58103-3) and CNPq for their financial support.

References

[1] S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation. *TPAMI*, 26(11):1475–1490, November 2004.

[2] A. Berg, T. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondences. In *CVPR*, volume 1, pages 26–33, 2005.

[3] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 1 edition, 2006.

[4] R. M. Bolle, J. H. Connell, N. Haas, R. Mohan, and G. Taubin. Veggievision: A produce recognition system. In *WACV*, Sarasota, USA, 1996.

[5] F. Cutzu, R. Hammoud, and A. Leykin. Distinguishing paintings from photographs. *CVIU*, 100(3):249–273, March 2005.

[6] L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *TPAMI*, 28(4):594–611, April 2006.

[7] R. Gonzalez and R. Woods. *Digital Image Processing*. Prentice-Hall, 3 edition, 2007.

[8] K. Grauman and T. Darrel. Efficient image matching with distributions of local invariant features. In *CVPR*, pages 627–634, 2005.

[9] G. Heidemann. Unsupervised image categorization. *IVC*, 23(10):861–876, October 2004.

[10] F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. In *ICCV*, volume 1, pages 604–610, 2005.

[11] X. Liu, L. Zhangb, M. Lib, H. Zhangb, and D. Wang. Boosting image classification with lda-based feature combination for digital photograph management. *Pattern Recognition*, 38(6):887–901, June 2005.

[12] M. Marszalek and C. Schmid. Spatial weighting for bag-of-features. In *CVPR*, pages 2118–2125, 2006.

[13] G. Pass, R. Zabih, and J. Miller. Comparing images using color coherence vectors. In *ACMMM*, 1997.

[14] A. Rocha and S. Goldenstein. PR: More than Meets the Eye. In *ICCV*, 2007.

[15] N. Serrano, A. Savakis, and J. Luo. A computationally efficient approach to indoor/outdoor scene classification. In *ICPR*, 2004.

[16] J. Sivic, B. Russell, A. Efros, A. Zisserman, and W. Freeman. Discovering objects and their location in images. In *ICCV*, pages 370–377, 2005.

[17] R. Stehling, M. Nascimento, and A. Falcão. A compact and efficient image retrieval approach based on border/interior pixel classification. In *CIKM*, pages 102–109, 2002.

[18] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[19] M. Unser. Sum and difference histograms for texture classification. *TPAMI*, 8(1):118–125, January 1986.

[20] P. Viola and M. Jones. Robust real-time face detection. *IJCV*, 57(2):137–154, February 2004.

[21] M. Weber. *Unsupervised learning of models for object recognition*. Phd thesis, Caltech, Pasadena, United States, May 2000.