COLOR DESCRIPTION OF LOW RESOLUTION IMAGES USING FAST BITWISE QUANTIZATION AND BORDER-INTERIOR CLASSIFICATION

Moacir Ponti, Camila T. Picon

ICMC — Universidade de São Paulo — Campus de São Carlos
13566-590 São Carlos, SP, Brazil

ABSTRACT

Image classification often require preprocessing and feature extraction steps that are directly related to the accuracy and speed of the whole task. In this paper we investigate color features extracted from low resolution images, assessing the influence of the resolution settings on the final classification accuracy. We propose a border-interior classification extractor with a logarithmic distance function in order to maintain the discrimination capability in different resolutions. Our study shows that the overall computational effort can be reduced in 98%. Besides, a fast bitwise quantization is performed for its efficiency on converting RGB images to one channel images. The contributions can benefit many applications, when dealing with a large number of images or in scenarios with limited network bandwidth and concerns with power consumption.

Index Terms— Feature extraction, image classification.

1. INTRODUCTION

Image classification and retrieval tasks can be computationally demanding on applications with large image databases or images of high resolution. Since the image acquisition with high spatial sampling is currently a standard, it is important to analyze alternatives to process and classify such large images.

Swain and Ballard showed that it was possible to use different resolutions of images for color descriptor [1]. Later, with new applications including more classes and instances, other authors recommended the highest spatial resolution available for histogram-based descriptors [2], and also for face detection applications [3], performing downsampling only if the original resolution is too high to process. On the other hand Brunelli and Mich [4] claimed that reducing the resolution or quality of the images does not degrade severely the retrieval performance of histogram-based systems. Nevertheless, there are no studies that tries to answer this questing by analyzing the problem in a objective way.

In this paper we evaluate the classification results when using lower resolution versions of images, in order to understand how much it is possible to reduce the image size without hampering the accuracy. Color description is used because it retains shape, location, and texture information, especially for images without distinctive objects and scenes [5]. A fast color quantization method is also propose.

An example of an application that can benefit for our work is the Produce dataset. It was acquired using RGB color images with 1024×786 pixels [6]. In the original work, the authors achieved good classification performance using a pipeline of operations: i) downsampling by half of the original size using linear interpolation, ii) background subtraction based on k-Means algorithm, iii) extraction of five different feature vectors and iv) feature/classifier fusion. We believe the whole system, as well as other applications, can benefit from the use of faster algorithms and lower resolution images.

Contributions Recent studies showed that the choice of the method used to convert color images to grayscale have significant impact on image recognition [7, 8], and that simplification of images often has good impact in image segmentation [9]. Our contribution is to propose both a reduction in resolution, by undersampling the image, and a quantization while mantaining the discrimination capability of the color descriptors. This could reduce the cost of feature extraction on content based image retrieval and classification tasks. As far as we know there is no previous study using the same methodology to ask such questions.

2. COLOR DESCRIPTORS

- BIC (Border-interior classification): uses a re-quantized image (often with 64 colors) and generates a representation of the image color distribution by computing two histograms: one for the pixels classified as border and another one for those classified as interior. A pixel is classified as border if at least one of its neighbors has a different quantized color, and classified as interior otherwise [10].

- GCH (Global Color Histogram): its feature vector is composed by ordered values, one for each distinct color, representing the probability of a given pixel have that specific color, computed as a normalized histogram of color frequencies on the whole image. A reduced number of color is often used [1].

978-1-4673-6997-8/15/$31.00 ©2015 IEEE

Fig. 1. Samples of the 15 classes of Produce dataset

Fig. 2. Samples of the 10 classes of COREL dataset

- LCH (Local Color Histogram): similar to the GCH, but it divides the image in fixed rectangular regions, computing the normalized histogram of each region [11].

3. BASE CLASSIFIERS

SVM (Support Vector Machines): a classifier developed using the statistical learning theory [12]. It maps the vector space to a higher dimensional space, uses kernel functions to compute the data distribution, and uses optimization techniques to find the support vectors. It is complex, but works well for different tasks, by tuning parameters such as the kernel function.

OPF (Optimum-Path Forest): in this classifier the samples are interpreted as vertices of a graph. The training step connects the samples from the same class in order to produce trees, using a specified distance space adjacency relation. The set of trees is the optimum path forest (OPF). A new sample is classified by connecting it to the tree that offers the optimum cost path to its root. This classifier was proposed in 2009 [13] and showed good performance on different applications. It handles multi-class problems natively and has no parameters to adjust. In this paper we used an ensemble version of OPF for a better performance [14].

4. METHOD AND EXPERIMENTS

4.1. Image databases

Produce (also refered as *Tropical Fruits and Vegetables* dataset): contains RGB images with a controlled background but changes in illumination, pose, number of objects and scale as can be seen in Figure 1. Some images also present partial occlusion of the objects, blur and shadow [6] with:

- 2633 images with 1024×768 pixels resolution.
- 15 unbalanced classes: Agata Potato, Asterix Potato, Pear, Cashew, Peach, Red Apple, Green Apple, Melon, Kiwi, Nectarine, Onion, Orange, Plum, Lime and Watermelon. From 75 to 264 objects per class.

COREL-1000: RGB images (samples shown in Figure 2) compiled by Wang et al. [5] with:

- 1000 images with 384×256 pixels resolution.
- 10 classes: Africa, Beach, Building, Bus, Dinosaur, Flower, Elephant, Horse, Mountain and Food, with 100 objects per class.

4.2. Logarithmic distance

When computing the difference between two colors frequencies with very different values, the result given by the euclidean distance is a large number that is summed over smaller differences. Thus, images with the same background color but different foreground can be assigned to the same class. The logarithmic distance (dLog) between two histograms q and d, tries to reduce this problem[10]:

$$dLog(q, d) = \sum_{i=0}^{i=M} \| f(q[i]) - f(d[i]) \|, \tag{1}$$

where

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

In order to use this function, the feature vectors are normalized to values between 0 and 255.

4.3. Fast downsampling and quantization

In order to study the impact of different image resolutions on the performance of the classification, the images were downsampled to 354×256 as a starting point and used in versions called 100% (354×256), 75% (266×192), 50% (177×128) and 25% (89×64) of the starting size. An example of such reduction is shown in Figure 3. The original studies used images with 354×256 for COREL [5] and a downsampling to 640×480 for Produce [6].

The OpenCV library [15] and `C` were used to perform the image resizing, quantization, feature extraction and dLog. Fast preprocessing algorithms were implemented as follows:

- The images were **downsampled** without antialiasing, since it was the fastest option available.

(a) (b) (c)

Fig. 3. Nectarine image low resolution samples: a) 100% b) 50% c) 25%

- A **fast re-quantization** method was developed to obtain 64 color images using the two most significant bits of each color channel. Each 24-bit pixel, 8 bits/channel in the format: $(B_1B_2B_3B_4B_5B_6B_7B_8)$, $(G_1G_2G_3G_4G_5G_6G_7G_8)$, $(R_1R_2R_3R_4R_5R_6R_7R_8)$, was changed into a single 6-bit channel, i.e.: $[0\ 0\ \mathbf{B_1B_2G_1G_2R_1R_2}]$. This procedure is linear on the number of pixels, and fast to compute since it uses only bitwise operations.

4.4. Experimental settings

- **BIC**: in order to simplify the border/interior classification, a neighborhood of 4 pixels was used. All pixels on the outer edge of the image were considered border pixels. The final histogram was calculated by the concatenation of the border and interior histograms.
- **LCH**: the image was divided into four rows and four columns, a setting often used in the literature [16].
- **GCH** and **LCH**: feature vector values were normalized to double precision numbers in a [0-1] interval so that the sum of the elements equals unity.

Four different hold-out configurations were used. From the available samples, 4 experiments were carried out using different number of object per class as training set:

- Produce: 64, 48, 32 and 16 samples per class, similar to the experiments performed in [6].
- COREL: 40, 30, 20 and 10 samples per class.

Each experiment was repeated 10 times using a repeated sampling method in order to compute average and standard deviation values. A balanced accuracy was used since the Produce dataset has unbalanced classes.

The classifiers were trained using the L^2-norm distance and also the logarithmic distance as described in Eq.1. The SVM parameters were adjusted by grid search with an evaluation set (10% of the training set) for each descriptor/settings. The best parameters are shown in Table 1. The libSVM default values were used on all other parameters and the distance function used was also the default.

Table 1. Parameters for the SVM classifier

	Kernel	Cost	γ
GCH	Radial base	200	1/5
LCH	Sigmoidal	300	1/20
BIC	Polinomial	300	1/120

Table 2. Results for COREL-1000 with 40 samples per class

Size	LCH	GCH	BIC	BIC-dLog
		OPF Ensemble		
100%	92.4%±0.7	93.9%±0.8	98.2%±0.4	98.9%±0.2
75%	92.8%±0.8	94.2%±0.7	98.5%±0.3	99.2%±0.3
50%	93.2%±0.9	94.0%±1.2	98.7%±0.3	98.9%±0.6
25%	92.2%±1.1	94.1%±1.8	98.6%±0.3	99.1%±0.7
		SVM		
Size	LCH	GCH	BIC	BIC-dLog
100%	90.0%±0.9	95.1%±0.3	97.3%±0.6	97.6%±0.6
75%	91.8%±1.7	94.3%±0.7	97.5%±1.5	98.5%±1.0
50%	91.5%±2.1	94.6%±1.2	97.0%±0.9	98.8%±2.1
25%	90.3%±2.4	94.3%±0.9	97.2%±1.1	98.3%±1.3

5. RESULTS AND DISCUSSION

The results are shown in Table 2 for COREL with 40 samples/class and Table 3 for the Produce dataset with 64 samples/class. The OPF classifier showed accuracies similar to the SVM classifier most experiments. For this reason, from this point, only OPF classifier results will be shown.

In order to see the effect of using different training samples per class, results with GCH and BIC are shown in Figure 4 for COREL dataset. The BIC-dLog method showed more robust results with respect to the number of training samples. The logarithmic distance used with GCH degraded the results. The LCH results were not displayed since the results were similar to the GCH method.

Table 3. Results for Produce with 64 samples per class

Size	LCH	GCH	BIC	BIC-dLog
		OPF		
100%	91.5%±0.4	91.1%±0.3	97.1%±0.9	98.1%±0.7
75%	90.9%±0.5	91.8%±0.6	96.2%±0.8	97.9%±0.5
50%	90.6%±1.3	91.3%±1.0	96.9%±0.8	98.6%±0.9
25%	89.8%±1.2	91.8%±1.2	96.0%±1.4	98.2%±1.0
		SVM		
Size	LCH	GCH	BIC	BIC-dLog
100%	85.0%±1.2	85.1%±1.3	95.3%±1.0	97.9%±0.5
75%	83.8%±1.7	84.3%±0.6	95.2%±0.5	97.5%±2.3
50%	83.5%±2.3	84.6%±1.4	95.0%±0.9	97.8%±0.9
25%	82.5%±1.5	84.3%±1.8	95.2%±1.2	97.7%±1.4

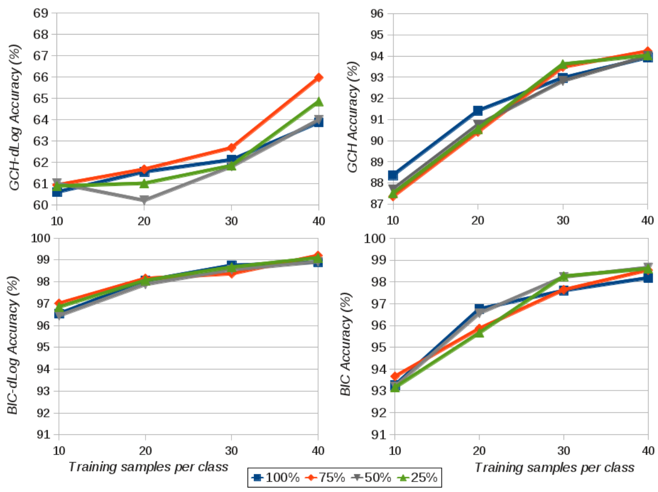

Fig. 4. COREL results of GCH descriptor (first row) and BIC descriptor (second row) with different training samples

In order to show better the effect of using lower resolution images on the classification accuracy, two graphs with the results for the both COREL and Produce dataset are shown in Figure 5, where is possible to see that the decrease on the resolution can increase the standard deviation, but with a stable classification accuracy, even for the 25% setting. In contrast to the previous recommendations for using the highest spatial resolution available [2], our findings shows that it is possible to significantly reduce the spatial resolution, speeding-up the process without hampering the classification performance.

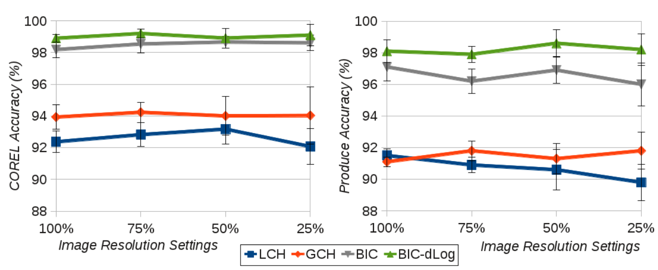

Fig. 5. Results changing resolutions for COREL (left) and Produce (right)

The produce classification task, in the previous study, required background removal and extraction of color and texture features. It was possible to show that an image with $89 \times 64 = 5,696$ pixels is sufficient to obtain an efficient color descriptor. When compared to the images previously used, that used images with $640 \times 480 = 307,200$ pixels, this represents a reduction of $\approx 98\%$ on the computational effort. Figure 6 shows two trees for the visualization of Produce image collection instances [17], showing that instances will lie in similar branches after resolution reduction.

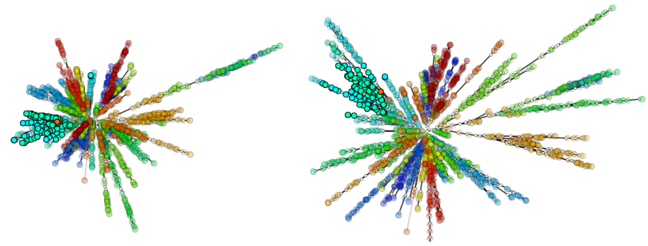

Fig. 6. Visualization of Produce image collection using similarity trees: left 100% resolution, right, 50% resolution. Each node is a feature vector and each color represents a class assigned by the classifier. Selected samples from one class are highlighted, showing that after processing a similar distribution of nodes is observed.

6. CONCLUSION

Experiments were performed to investigate the accuracy using low resolution images and bitwise quantization. These procedures can speed-up the preprocessing and feature extraction steps. With respect to the resolutions, we observed similar results with different classifiers and feature extractors.

According to the experimental evidences, the BIC descriptor and the use of a logarithmic function are specially suited to the use of color descriptors with low resolution images. The bitwise quantization method also worked well for description. Besides, the OPF classifier showed performance compared to the SVM without the need of grid search. The advantages of this set of methods are: i) OPF does not parameter search, ii) BIC discriminative properties are better preserved in different resolutions and training set sizes, iii) bitwise algorithms are faster to compute, iv) the logarithmic distance with BIC prevent confusion when spatial resolution is lost and less training samples are used.

The results indicate that it is possible to use color descriptors extracted from low resolution images. Also, for classification purposes, there is no need for interpolation or a quantization method that preserves visual quality. Our study gives an important guideline for using color descriptors with maximum efficiency, benefiting many applications, specially those with a large number of images to be processed. Such guidelines are, for example, useful for applications with limited network bandwidth and concerns with power consumption. Future studies can investigate also texture and shape descriptors in this context.

7. ACKNOWLEDGMENT

We would like to thank FAPESP (grants # 10/19159-1 and 11/22749-8) and CNPq (grant # 482760/2012-5).

8. REFERENCES

[1] M. J. Swain and D. H. Ballard, "Color indexing," *Int. J. Computer Vision*, vol. 7, no. 11, pp. 11–32, 1991.

[2] O. Chapelle, P. Haffner, and V. N. Vapnik, "Support vector machines for histogram-based image classification," *IEEE Trans. Neural Networks*, vol. 10, no. 5, pp. 160, 1999.

[3] J. Wright, A.Y. Yang, A. Ganesh, Sastry S.S., and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[4] R. Brunelli and O. Mich, "On the use of histograms for image retrieval," in *IEEE Int. Conf. Multim. Comp. Syst.*, 1999, vol. 2, pp. 143–147.

[5] J.Z. Wang, J. Li, and G. Wiederhold, "Simplicity: Semantics-sensitive integrated matching for picture libraries," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 9, pp. 947–963, 2001.

[6] A. Rocha, D. C. Hauagge, J. Wainer, and S. Goldenstein, "Automatic fruit and vegetable classification from images," *Computers and Electronics in Agriculture*, vol. 70, no. 1, pp. 96–104, 2010.

[7] C. Kanan and G.W. Cottrell, "Color-to-grayscale: does the method matter in image recognition?," *PLOS One*, vol. 7, no. 1, pp. e29740, 2012.

[8] M. Ponti and L. Escobar, "Compact color features with bitwise quantization and reduced resolution for mobile processing," in *IEEE Global Conference on Signal and Information Processing*, Austin/TX, 2013, pp. 751–754.

[9] M. Ponti, "Segmentation of low-cost remote sensing images combining vegetation indices and mean shift," *IEEE Geoscience and Remote Sensing Letters*, vol. 10, no. 1, pp. 67–70, 2013.

[10] R. O. Stehling, M. A. Nascimento, and A. X. Falcao, "A compact and efficient image retrieval approach based on border/interior pixel classification," in *CIKM'02 - 11th ACM Int Conf Information Knowledge Management*, 2002, pp. 102–109.

[11] H. Lu, B.-C. Ooi, and K.-L. Tan, "Efficient image retrieval by color contents," in *Applications of Databases*, 1994, vol. 819, pp. 95–108.

[12] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[13] J.P. Papa, A. X. Falcao, and C. T. N. Suzuki, "Supervised pattern classification based on optimum-path forest," *Int. J. Imaging Systems and Technology*, vol. 19, no. 2, pp. 120–131, 2009.

[14] M. Ponti-Jr. and I. Rossi, "Ensembles of optimum-path forest classifiers using input data manipulation and undersampling.," in *Multiple Classifier Systems*, 2013, vol. 7872 of *LNCS*, pp. 236–246.

[15] Intel, *OpenCV: open source computer vision library*, Intel Corporation, `http://opencv.willowgarage.com/wiki/`, 2.3 edition, 2012, BSD license.

[16] O.A.B. Penatti, E. Valle, and R.S. Torres, "Comparative study of global color and texture descriptors for web image retrieval," *Journal of Visual Communication and Image Representation*, vol. 23, no. 2, pp. 359–380, 2012.

[17] J. G. S. Paiva, L. Florian-Cruz, H. Pedrini, G. P. Telles, and R. Minghim, "Improved similarity trees and their application to visual data classification," *IEEE Trans. Visualization and Computer Graphics*, vol. 17, pp. 2459–2468, 2011.