WHAT COLOR IS AN OBJECT?

Xiaofan Zhang[1], *Zengchang Qin*[1*], *Xu Liu*[1], *Tao Wan* [2*]

[1] Intelligent Computing and Machine Learning Lab
School of ASEE, Beihang University, Beijing, 100191, China
[2] Department of Biomedical Engineering
Case Western Reserve University, Cleveland, OH 44106, USA

ABSTRACT

Color perception is one of the major cognitive abilities of human being. Color information is also one of the most important features in various computer vision tasks including object recognition, tracking, scene classification and so on. In this paper, we proposed a simple and effective method for learning color composition of objects from large annotated datasets. The new proposed model is based on a region-based bag-of-colors model and saliency detection. The effectiveness of the model is empirically verified on manually labelled datasets with single or multiple tags. The significance of this research is that the color information of an object can provide useful prior knowledge to help improving the existing computer vision models in image segmentation, object recognition and tracking.

Index Terms— Bag-of-colors, color histogram, saliency detection

1. INTRODUCTION

Color is one of the most direct information of an image in human perception. Like our senses of taste and smell, visual sense of colors helps us to understand the physical world around us. While it gives us elementary survival skills of observing surrounding environment and recognizing objects, color also enriches our lives, allowing us to appreciate everything from the beauty of a rainbow, to the aesthetic pleasure of a painting. However, Picasso used to exclaim that colors in an image are only symbols and reality is to be found in luminance alone. This message is also well taken by us computer vision scientists. When trying to interpret the content of an image, we always focus is on gray-scale images and invented various gradient-based features including edges, ridges, and corners, but ignore the direct color information [1].

Two major advantages of using color vision are revealed from the previous research. First, color provides extra information which allows the distinction between various physical causes for color variations in the world, such as changes due to shadows, light source reflections, and object reactance variations. This helps to quickly identify the black object on the road as a shadow. Next to this, color is an important discriminative property of objects. We can easily tell the color of an object and find it immediately from a noisy background if it has a clear contrast in color, for example, to distinguish a red flower from a grassland. Although an object may exhibit different colors, e.g., an object *Apple* can be either red, green, yellow, and even white or silver (considering the Apple computer). It still has a limited color range, it is like we rarely see an *apple* being black or blue. Color information plays such an important role in object recognition. However, as we are aware that there is not much research to investigate how to find the color composition of a given object.

In this paper, we proposed a method to learn color composition of objects. The idea is simple and straightforward: given a collection of images and each of them contains one same object with possibly different background, it is easy to see that the most colors in common are likely to be the colors of the object. With a proper color representation model we can reduce the number of color without losing information too much. By using a saliency detection algorithm, we can significantly reduce the negative influence from irrelevant background colors. Finally, we can learn the color composition of a given object automatically.

There is a rich literature in studying color models [2, 3] and color features [4, 5, 6] . For example, Color indexing [2], which counts the color distribution in a discrete color space defined by RGB color. However, it suffers from a typical problem that the most frequent colors dominating the other colors in the final representation and that may deteriorate the performance. Based on the recent study of the bag-of-colors model [7] which is inspired by the bag-of-words model [8] and its variants [1, 9], we define a set of colors by clustering the colors sampled from the whole training dataset to generate the color codebook [7]. Such a codebook can be used for generating color histogram. Saliency detection [10, 11, 12] is a hot topic in image processing for detecting objects that attract visual attention. In previous work, quaternion discrete cosine

[*] Corresponding author emails: zcqin@buaa.edu.cn and tao.wan.wan @gmail.com. This work is partially funded by the NCET Program of MOE and the Fundamental Research Funds for the Central Universities in China.

transform (QDCT) [13] with Mean-Shift [14, 15] segmentation is used to generate a saliency map, where each pixel is weighted by *saliency intensity* which is measured by the probability of this pixel being the salient part. Each segmented region of this saliency map is updated based on a posterior probability of being in foreground and background. The saliency map provide us the location information of the object we concerned.

The paper is organized as the following. Section 2 introduces the basic idea of the bag-of-colors method and how to generate a region-based color codebook. In Section 3, we briefly introduce a saliency detection algorithm to isolate the object from background by a saliency map. Our approach for extracting colors of an object is given in Section 4. In Section 5, experimental results are analyzed to show the effectiveness of the new proposed model.

2. REGION-BASED BAG-OF-COLORS MODEL

Scale-invariant feature transform (SIFT) descriptor [16] and its variants are one of most used features in content-based image retrieval. These descriptors are often used with BOW framework [8] to produce a fixed-size vector. Wengert *et al.* [7] proposed the bag-of-colors model. They collect 10000 images from Flickr randomly and separate them into small square patches. For each patch, choose the most occurring color as the main color. If this color corresponds to less than 5 occurrences, just select an arbitrary color from the block. Then they have millions of colors and cluster the colors to get a low-dimensional color codebook. The histogram based on codebook can represent the frequency of colors appeared in the image. The color histogram with idf weighting, power-law transformation and normalization obtained excellent performance in several popular image retrieval datasets. However, it can be improved based on the following observation.

The way of finding main colors in [7] is not very precise. Based on our empirical studies on the Holiday dataset [17], about 50% patches do not have a color which occurs more than 5 times. Therefore, many so called main colors are not effectively picked.

According to the observation, we propose a region-based approach described by the following. (1) Segment the image into regions with the mean-shift algorithm in Lab color space [1]. The reason for using this space is that it is more consistent with the Euclidean space structure. Since the regions are segmented based on color information, colors within a particular region are very close. Finding the main colors from these regions instead of the square patches could be more appropriate. (2) For each region, count the frequency of all colors, chose the most occurring colors as the main colors. Number of main colors N picked from a region depends on the area of this region A.

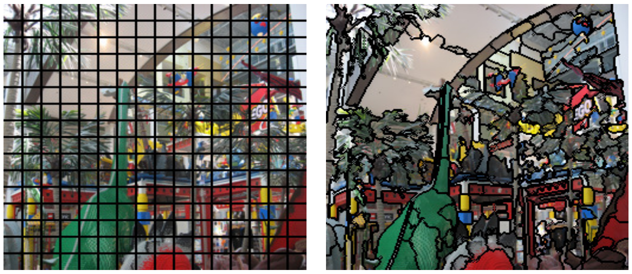
(a) Patch-based Approach (b) Region-based Approach

Fig. 1. Comparisons between the patch-based approach and region-based approach on an image.

$$N = \lceil \frac{A}{r} \rceil$$

where $\lceil \cdot \rceil$ is the ceiling function and r is usually set to three or four times larger than the minimum region area in order to add some weights to small but meaningful regions. Fig. 1 shows a comparison between patch-based and region-based approaches on a sample image. It is easy to see that colors in region-based patches are more consistent therefore can extract more accurate main colors from the image. Fig. 2 also shows the learned color codebook on the *Caltech101* [18].

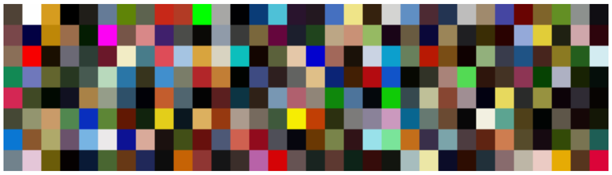

Fig. 2. Color codebook for the region-based bag-of-colors model on *Caltech101* [18].

3. SALIENCE DETECTION

Fail to locate the object from a background will deteriorate the performance significantly. For example, when we want to learn the color of a plane, since a plane always appears in the sky, colors summarized across training images may appear to be blue, which is the color of the sky. To solve this problem, we can use saliency detection to distinguish the object from background.

Here we introduce an algorithm by combining QDCT and Mean-Shift for generating the saliency map [19]. The main idea can be summarized as follows. We first use the QDCT algorithm [13] to generate a *rough saliency map* that looks like several dim light points on the black background. Segment the image into regions with mean-shift algorithm [14, 15] in Lab color space. Tune the parameters let it be a little over segmented. For each segmented region, calculate the mean

[1] http://en.wikipedia.org/wiki/Lab_color_space

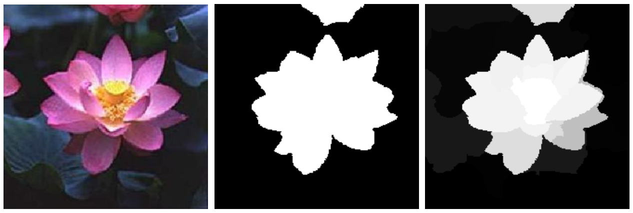

Fig. 3. Left: Original image; Middle: Binary saliency map. Right: Saliency map with gray scales showing the probabilities of pixels being the foreground.

saliency value based on the rough saliency map, and set this mean value to all the pixels within this region. We repeat the same operation region by region, then can obtain a new saliency map to separates the foreground and background by its mean. Because we over segment the image in order to obtain relatively consistent colors within each region. Some small regions, e.g., in the foreground, may be misclassified as background. It is like one part of an object is missing because of the misclassification. However, given the prior color distributions of foreground and background, we may find that the color of this small region is more likely to be in the foreground rather than the background. Here the "color" we said is the nearest color in codebook. The Bayes theorem is used to calculate the probability of a color c to be in foreground f:

$$P(f|c) = \frac{P(c|f)P(f)}{P(c|f)P(f) + P(c|b)P(b)} \tag{1}$$

where b denotes the background. $P(c|f)$ is evaluated by the area percentage of the color c in the foreground. $P(f)$ is the area percentage of foreground in the whole image. Repeat the steps that update the saliency value with posterior probability till the saliency map is stable enough. By setting a threshold, the saliency map with continuous values can be converted to a *binary saliency map* showing foreground and background without saliency intensity values. Fig. 3 shows the saliency maps of the image of *lotus*.

4. LEARNING COLORS OF OBJECT

Based on the given color representation and saliency detection result, we can learn colors of an object through the following steps.

We first calculate color histogram of every image based on the color codebook. Resize the all image into the same scale S ($S = M \times N$) and add a weighted vote to the corresponding bins of histogram based on the closest color in the codebook. The weight of the pixel (m, n) is just the saliency intensity $w_{m,n}$ from the saliency map.

$$H_{i,j} = \sum_{m=1}^{M} \sum_{n=1}^{N} w_{m,n} \delta(I_{m,n}, C_i) \tag{2}$$

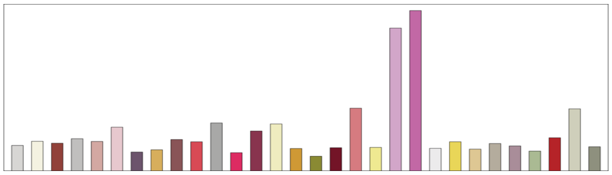

Fig. 4. Histogram of ground truth for *lotus*. Only top 30 colors are shown in the histogram.

where $\delta(\cdot)$ is an indicator function that:

$$\delta(A, B) = \begin{cases} 1 & A = B \\ 0 & A \neq B \end{cases}$$

where $H_{i,j}$ stands for ith bin of color histogram for the image j. $I_{m,n}$ is the closest color for pixel (m, n) in the color codebook and C_i denotes the ith color in the codebook.

Secondly, collect color histograms whose images contain the same object. Sum up the color histograms and renormalize to get the color distribution of this object. $D_i = \sum_{j=1}^{T} H_{i,j}$ where D_i is the probability of color C_i in the composition of the object, T denotes the total number of images which have the object in them.

5. EXPERIMENTAL STUDIES

The model is tested on both single-tagged dataset *Caltech101* [18] [2] and multiple tagged dataset *LabelMe* [20], in which each image has a few object tags. Because there is no ground truth for object colors in *Caltech101*, we manually created benchmark dataset by mouse-clicking the given object using human judgment. Each image is first segmented into a few regions. A human subject is asked to click on 5-7 representative regions of the object. The dominating color of the selected region is extracted as the main color of the object. For example, Fig. 4 shows the histogram of ground truth for the object *lotus*. The manually tagged dataset with 101 objects is available for download at the project page [3]. In *LabelMe* [20], we use 1133 fully labelled images which contains 838 categories including bird, building, painting, leaf and etc. Among these 838 kinds of objects, we choose 135 objects that appear in more than 10 images to build the ground truth. Images that contain a particular object can be retrieved and the object can be outlined by using the *LabelMe* Toolbox [20]. Ground truth color histograms are extracted based on previous outlined objects.

In our experiments, minimum area of region S_{min} in the Mean-Shift is set to 30, sample ratio r is set to 100 and the codebook size is set to 256. The generated color codebooks for *Caltech101* and *LabelMe* are shown in Fig. 2 and 5, respectively. To measure the similarity between the test

[2] `www.vision.caltech.edu/Image_Datasets/Caltech101/`
[3] `http://icml1.buaa.edu.cn/members/xfz/index.html`

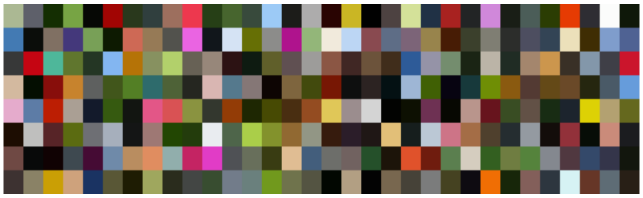

Fig. 5. Color codebook for *LabelMe*.

Table 1. JSD values between the ground truth and the learned color composition based on different models.

JSD Value (10^{-2})	Caltech101	LabelMe
Baseline	33.83±7.70	27.32±7.34
No saliency detection	17.39±7.36	11.05±6.83
Float saliency weight	8.00±5.37	10.77±6.76
Binary saliency weight	7.99±5.45	10.75±6.76

results and ground truth, we choose the average of Jensen-Shannon divergence (JSD) as the final evaluation criteria. Jensen-Shannon divergence can be regarded as an averaged Kullback-Leibler (KL) divergence:

$$JSD(P||Q) = \frac{1}{2}D_{KL}(Q||M) + \frac{1}{2}D_{KL}(P||M) \qquad (3)$$

where $D_{KL}(P||Q) = \sum_i P(i)\log\frac{P(i)}{Q(i)}$ and $M = \frac{1}{2}(P+Q)$. We set the divergence between a random distribution on colors in the codebook and ground truth as the baseline. The mean JSD values with standard deviations ($\pm$ std) on both *Caltech101* and *LabelMe* are shown in Table 1.

As we can see from the results: JSD value implies the closeness of two color compositions, smaller the JSD value is, closer two color compositions are. Fig. 6 and 7 show the color compositions given some sample objects in both *Caltech101* and *LabelMe*. It is obvious that the color compositions we learned are effective. With the experiments using two kinds of saliency maps, we can see that the differences are not significant. Unlike in Caltech101, the images in *LabelMe* usually have multiple objects with multiple tags in complex background. It is hard to use saliency detection to extract all salient objects. Fig. 8 and 9 give two examples of saliency detection. Particularly, Fig. 9 shows an image from *LabelMe*, its tags are *car, building, fence, door, window, roof* and etc.

The saliency map we obtained fails to accurately outline all these objects. However, by considering the common colors across a collection of images with the same tag, we can still effectively learn its colors. As shown in Table 1, the average difference of using and not using saliency detection in *LabelMe* is $0.1105 - 0.1077 = 0.0028$ while this difference in *Caltech101* is $0.1739 - 0.0800 = 0.0939$. That means although saliency detection in multiple tagged dataset is not as effective as in single tagged dataset, the system still can achieve similar performance in extracting the object color compositions.

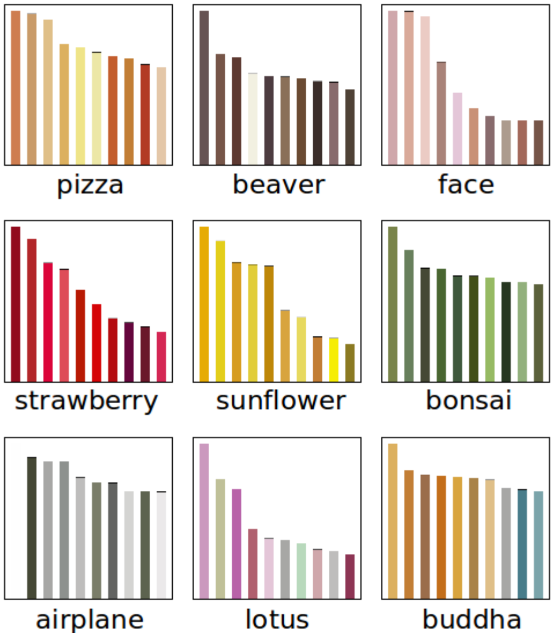

Fig. 6. Color compositions of several objects with binary saliency map learned in *Caltech101*. Only top ten colors are shown.

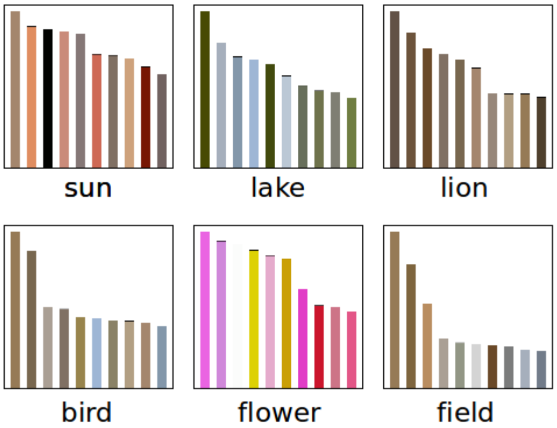

Fig. 7. Automatically learned color compositions of the objects with binary saliency maps in *LabelMe*. Only top ten colors are shown.

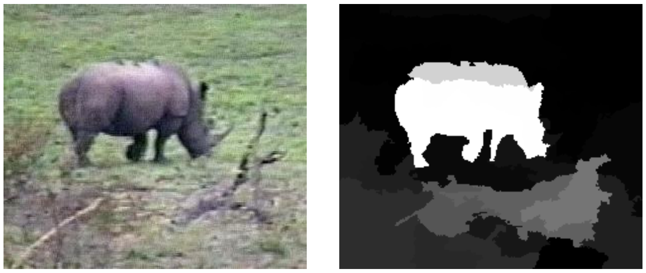

Fig. 8. A sample saliency map in *Caltech101*.

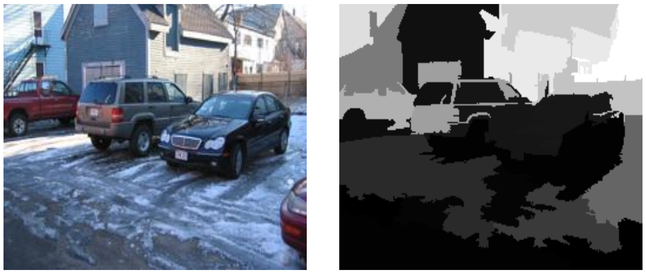

Fig. 9. A sample image with its corresponding saliency map in *LabelMe*.

6. CONCLUSION AND FUTURE WORK

In this paper, we presented a simple and effective method to learn the color composition of an given object based on tagged image dataset. This method is based on the bag-of-colors model with updated saliency maps. Empirical studies on both single-tagged dataset *Caltech101* and multiple-tagged dataset *LabelMe* show the effectiveness of the new proposed model. In future work, we need to develop an online algorithm to find the color composition of an object by querying it using an image search engine and training on the retrieved images. Another important contribution of this work is to provide useful prior color information that can be used in many other computer vision tasks including object recognition, tracking and image annotation.

7. REFERENCES

[1] X. Yuan, J. Yu, Z. Qin, and T. Wan, "A bag-of-features model with integrated sift-lbp features for content-based image retrieval," in *ICIP*, 2011.

[2] M. J. Swain and D. H. Ballard, "Color indexing," *International Journal of Computer Vision*, vol. 7:1, pp. 11–32, 1991.

[3] I. Omer and M. Werman, "Color lines: image specific color representation," in *CVPR*, 2004, pp. 946–953.

[4] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, May 2001.

[5] Alaa E. Abdel-Hakim and Aly A. Farag, "Csift: A sift descriptor with color invariant characteristics," in *CVPR*, 2006, pp. 1978–1983.

[6] S.L. Wang and A.W.C. Liew, "Information-based color feature representation for image classification," in *ICIP*, Oct. 2007, vol. 6, pp. 353–356.

[7] C. Wengert, M. Douze, and H. Jégou, "Bag-of-colors for improved image search," in *ACM International Conference on Multimedia*, 2011, pp. 1437–1440.

[8] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *ICCV*, 2003, pp. 1470–1477.

[9] W. Zhang, Z. Qin, and T. Wan, "Image scene categorization using multi-bag-of-features," in *International Conference on Machine Learning and Cybernetics*, July 2011, vol. 4, pp. 1804 –1808.

[10] B. Alexe, T. Deselaers, and V. Ferrari, "What is an object?," in *CVPR*, June 2010, pp. 73–80.

[11] T. Judd, K. Ehinger, F. Durand, and A. Torralba, "Learning to predict where humans look," in *ICCV*, Oct. 2009, pp. 2106–2113.

[12] X. Hou, J. Harel, and C. Koch, "Image signature: Highlighting sparse salient regions," *Pattern Analysis and Machine Intelligence*, vol. 34, no. 1, pp. 194–201, 2012.

[13] B. Schauerte and R. Stiefelhagen, "Predicting human gaze using quaternion dct image signature saliency and face detection," in *Workshop on the Applications of Computer Vision*, 2012, pp. 137–144.

[14] D. Comanicu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, May 2002.

[15] P. Meer and B. Georgescu, "Edge detection with embedded confidence," *Pattern Analysis and Machine Intelligence*, vol. 23, no. 12, pp. 1351–1365, Dec. 2001.

[16] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[17] H. Jégou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *ECCV*, 2008, pp. 304–317.

[18] 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," *Comput. Vis. Image Underst.*, vol. 106, no. 1, pp. 59–70, Apr 2007.

[19] X. Liu, Z. Qin, X. Zhang, and T. Wan, "Color saliency model based on mean shift segmentation," in *ICASSP*, 2013.

[20] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "Labelme: A database and web-based tool for image annotation," *International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 157–173, May 2008.