IMAGE CLASSIFICATION WITH MAX-SIFT DESCRIPTORS

Lingxi Xie[1], Qi Tian[2], Jingdong Wang[3], and Bo Zhang[4]

[1,4]LITS, TNLIST, Dept. of Computer Sci&Tech, Tsinghua University, Beijing 100084, China
[2]Department of Computer Science, University of Texas at San Antonio, TX 78249, USA
[3]Microsoft Research, Beijing 100080, China
[1]198808xc@gmail.com,[2]qitian@cs.utsa.edu,
[3]jingdw@microsoft.com,[4]dcszb@mail.tsinghua.edu.cn

ABSTRACT

In the conventional Bag-of-Features (BoF) model for image classification, handcrafted descriptors such as SIFT are used for local patch description. Since SIFT is not flipping invariant, left-right flipping operation on images might harm the classification accuracy. To deal with, some algorithms augmented the training and testing datasets with flipped image copies. These models produce better classification results, but with the price of increasing time/memory consumptions.

In this paper, we present a simple solution that uses Max-SIFT descriptors for image classification. Max-SIFT is a flipping invariant descriptor which is obtained from the maximum of a SIFT descriptor and its flipped copy. With Max-SIFT, more robust classification models could be trained without dataset augmentation. Experimental results reveal the consistent accuracy gain of Max-SIFT over SIFT. The much cheaper computational cost also makes it capable of being applied onto large-scale classification tasks.

Index Terms— Image Classification, BoF Model, Max-SIFT, Flipping Invariance, Experiments

1. INTRODUCTION

Image classification is a fundamental problem in the community of computer vision. It is a basic task towards image understanding, and implies a wide range of real-world applications, including object recognition, scene understanding, object labeling, image tagging, *etc*. Recent years, fine-grained and large-scale classification tasks bring a lot of new challenges into this traditional research field.

One of the most popular approaches for image classification is the Bag-of-Features (BoF) model [1]. It is a statistics based model, in which local features are extracted, encoded and summarized into a global image representation. As a scale and rotation invariant feature transform, the SIFT descriptor [2] is widely adopted. However, since SIFT is not flipping invariant, its ability of matching flipped objects are not satisfied. To deal with, some researchers propose to consider the flipping operation by adding a flipped copy for each original image, and evaluate the classification model with the augmented datasets [3][4]. Although this method improves the classification accuracy, the doubled computational costs limit it from being applied onto large-scale tasks.

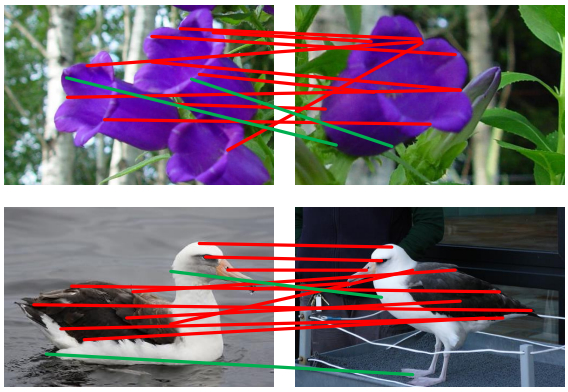

Fig. 1. Feature matching with SIFT [2] (green) and Max-SIFT (red) descriptors. For each image pair, we show 2 most significant matches by SIFT, and 10 by Max-SIFT. Unlike SIFT which often finds incorrect matches, Max-SIFT works very well to match the flipped objects.

We study this problem from observing the inner structure of SIFT as well as how it is changed by a flipping operation. Then, we compute the **maximum** of SIFT and its flipped copy to cancel out the flipping operation. As a consequence, we obtain **Max-SIFT**, a kind of **flipping invariant** descriptors. Examples of feature matching with SIFT and Max-SIFT descriptors are illustrated in Figure 1, in which Max-SIFT is verified to find much more feature matches between flipped object pairs. When Max-SIFT is adopted with the BoF model, we can guarantee to generate exactly the same representation on an image and its flipped copy. Experimental results reveal that the accuracy using Max-SIFT is consistently better than using SIFT, and also comparable with dataset augmentation methods which are much more computationally expensive.

The remainder of this paper is organized as follows. Section 2 briefly introduces some related works. The Max-SIFT descriptor and its application on image classification are illustrated in Section 3. After experimental results are shown in Section 4, we conclude our work in Section 5.

2. RELATED WORKS

The Bag-of-Features (BoF) model [1] is one of the most popular approaches for image classification. It is a statistics based model, which extracts local features, encodes them and summarizes them into a global image representation.

The BoF model starts from extracting local descriptors. Due to the limited descriptive power of raw pixels, handcrafted image descriptors such as SIFT [2][5] are widely adopted. These descriptors could be automatically detected using operators such as DoG [2] and MSER [6], or dense sampling [7] which is verified much better for classification.

Next, a visual vocabulary or codebook is trained using the descriptors collected from the whole dataset. The codebook could be computed iteratively with K-Means or Gaussian Mixture Model (GMM), in which the latter preserves more geometric information of the feature space. The descriptors are then projected onto the codebook as a compact feature representation. Popular feature encoding methods include hard quantization, LLC encoding [8], FV encoding [9], *etc*. Extracting visual phrases also helps feature encoding [10].

As a final stage of the BoF model, the quantized feature vectors are aggregated as a final vector for image representation. Sum pooling and max-pooling are different choices for feature summarization, and different pooling bins [11][12][13] are constructed for richer context modeling. The image representation vectors are then normalized [14] and fed into generic machine learning algorithms such as SVM. The BoF model could also be adopted with indexing structures for image retrieval [15][16][17][18][19].

Despite the simplicity, efficiency and scalability of the BoF model, it still suffers from several disadvantages. One of them comes from the use of SIFT descriptors, which is not flipping invariant: the SIFT descriptors extracted on the corresponding position of a flipped image pair might be totally different. As a consequence, an image before and after flipping operation might produce totally different representation vectors. To cope with, some researchers propose to augment the image datasets by adding a flipped copy for each original image, and evaluate the classification model on the enlarged training and testing sets [3][4]. In [20], it is even suggested to augment the datasets with a larger set of image transformations. Although these complex training processes are verified to improve the classification accuracy, the expensive computational costs in both time and memory limit their scalability, and make it difficult to apply these methods onto large-scale image classification tasks.

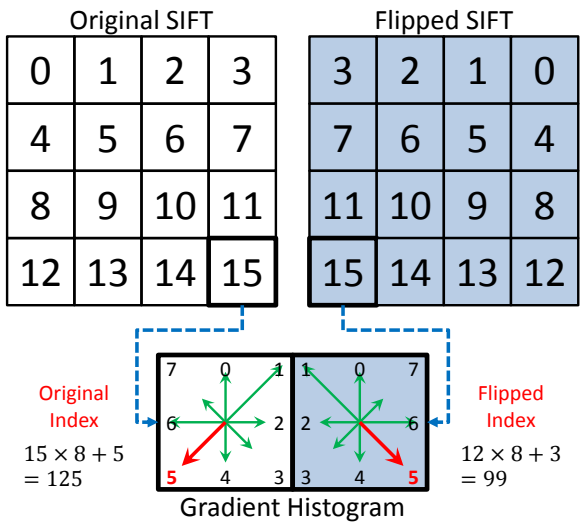

Fig. 2. The impact of flipping operations on SIFT descriptors. The grids with light blue background indicate those bins in which the order of gradient values is reversed.

3. MAX-SIFT FOR IMAGE CLASSIFICATION

This section illustrates the Max-SIFT descriptor and its application on image classification. Our method is inspired by the observation that how a SIFT descriptor is changed by the flipping operation, based on which we propose a straightforward solution to cancel up the operation, producing flipping invariant descriptors for image representation.

3.1. The Max-SIFT Descriptor

The inner structure of a SIFT descriptor is illustrated in the left part of Figure 2. A local patch is partitioned into 4×4 spatial bins, and in each grid an 8-dimensional gradient histogram is computed. The 16 gradient vectors are thereafter concatenated as the final 128-dimensional SIFT descriptor. The number in each bin indicates its order, and the 8 dimensions of the gradient vector is collected in a clockwise manner. When an image is left-right flipped, all the patches on it are left-right flipped as well. In the flipped patch, shown in the right part of Figure 2, both the order of 16 bins and the order of collecting the gradient vectors are changed, although the absolute values of gradient in each corresponding direction do not change. Taking the lower-right bin (#15, emphasized in Figure 2) in the original SIFT descriptor as the example. When the image is left-right flipped, this bin is moved to the lower-left position (#12), and the order of gradients in the bin changes from $(0, 1, 2, 3, 4, 5, 6, 7)$ to $(0, 7, 6, 5, 4, 3, 2, 1)$.

In formal, let us denote the original SIFT descriptor as $\mathbf{d} = (d_0, d_1, \ldots, d_{127})$, where we have $d_{i\times 8+j} = a_{i,j}$ indicating the j-th gradient value in the i-th spatial bin, for all

$i = 0, 1, \ldots, 15$ and $j = 0, 1, \ldots, 7$. With the illustration in Figure 2, we could map each index (0 to 127) of the original SIFT descriptor to another index of the flipped SIFT descriptor. Taking d_{125} ($a_{15,5}$, the red arrow in Figure 2) as the example. The same gradient value would appear at d_{99} ($a_{12,3}$) when the image (descriptor) is left-right flipped. We denote the mapping as $f^{\mathrm{F}}(125) = 99$. Due to the symmetry of the flipping operation, one can easily observe that for all $k = 0, 1, \ldots, 127$, we have $f^{\mathrm{F}}\big(f^{\mathrm{F}}(k)\big) = k$, which implies that flipping an image twice obtains the original image unchanged. Since the function $f^{\mathrm{F}}(\cdot)$ is a constant index permutation, we can compute the flipped copy of a SIFT descriptor very quickly: $\mathbf{d}^{\mathrm{F}} = f^{\mathrm{F}}(\mathbf{d}) = \big(d_{f^{\mathrm{F}}(0)}, d_{f^{\mathrm{F}}(1)}, \ldots, d_{f^{\mathrm{F}}(127)}\big)$.

With the original and flipped versions of a SIFT descriptor, we can cancel out the flipping operation by selecting the **maximum** of them, denoted as $\mathbf{d}^{\mathrm{MAX}} = \max\big\{\mathbf{d}, \mathbf{d}^{\mathrm{F}}\big\}$. Here, $\mathbf{d}$ and $\mathbf{d}^{\mathrm{F}}$ are compared with the alphabetical order from 0-th to 127-th dimension. $\mathbf{d}^{\mathrm{MAX}}$ is named the Max-SIFT descriptor generated from the original descriptor $\mathbf{d}$. Since $\big(\mathbf{d}^{\mathrm{F}}\big)^{\mathrm{F}} = \mathbf{d}$, one can easily find that the Max-SIFT descriptors generated from $\mathbf{d}$ and $\mathbf{d}^{\mathrm{F}}$ are exactly the same. Therefore, Max-SIFT is a kind of **flipping invariant** descriptor with the descriptive power of original SIFT preserved.

3.2. Application on Image Classification

Consider an image $\mathbf{I}$, and a set of SIFT descriptors extracted from the image: $\mathcal{D} = \{\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_M\}$. When the image is left-right flipped, the set of SIFT descriptors extracted becomes: $\mathcal{D}^{\mathrm{F}} = \big\{\mathbf{d}_1^{\mathrm{F}}, \mathbf{d}_2^{\mathrm{F}}, \ldots, \mathbf{d}_M^{\mathrm{F}}\big\}$. Since SIFT is **not** flipping invariant, the feature vectors encoded from $\mathcal{D}$ and $\mathcal{D}^{\mathrm{F}}$ might be very different, resulting in distinct feature representations on the same but left-right flipped images. The above problem could be solved with the Max-SIFT descriptor which is flipping invariant: the sets of Max-SIFT descriptors generated from original and flipped images are just the same.

When Max-SIFT descriptors are extracted on an image and its flipped copy, a same descriptor might appear on different locations, *e.g.*, a descriptor at the upper-left corner of the original image could also be found at the upper-right corner of the flipped image. When spatial pooling techniques such as SPM [11] are performed, it might cause the incorrect spatial bin correspondence of the feature vector. To deal with, we use a small trick which counts the number of flipped SIFT descriptors, *i.e.*, the descriptors that $\mathbf{d}^{\mathrm{MAX}} = \mathbf{d}^{\mathrm{F}}$. If the number is larger than half of the total number of descriptors M, we left-right flip the whole image by replacing the x coordinate of each descriptor with $W - x$, where W is the image width. This is equivalent to align the images automatically according to their directions.

In conclusion, with the Max-SIFT descriptor, we can guarantee to generate exactly the same representation vector for an image and its flipped copy. This helps us to train robust classification models, as shown in below experiments.

4. EXPERIMENTS

In this section, we perform extensive experiments to show the benefit and efficiency of using Max-SIFT descriptors for image classification.

4.1. Datasets and Settings

We evaluate our method on six publicly available image classification datasets, two for scene classification and other four fine-grained object recognition.

For scene classification, we use the **LandUse-21** dataset [21] (21 land-use scenes with 100 images for each class) and the MIT **Indoor-67** dataset [22] (67 indoor scenes and 15620 images). 80 images per category are randomly selected for training. For fine-grained object recognition, we use the Oxford **Pet-37** dataset [23] (37 cat or dog breeds and 7349 images), the **Aircraft-100** dataset [24] (100 aircraft models and 100 images for each model), the Oxford **Flower-102** dataset [25] (8189 flower images from 102 categories) and the Caltech-UCSD **Bird-200** dataset [26] (11788 bird images of 200 different species). For the Aircraft-100 and Bird-200 datasets, a bounding box is provided on each image. The numbers of training images per category for the above four datasets are 100, 20, 66 and 30, respectively.

Basic experimental settings follow the recent proposed BoF model with Fisher vector encoding [9]. Images are scaled, with the aspect ratios preserved, so that the larger axis is 300 pixels. When a bounding box is available (often for the fine-grained datasets), we only use the region within the box. We use the VLFeat [27] library to extract dense RootSIFT [28] descriptors. The spatial stride and window size of dense sampling are 6 and 12, respectively. The dimension of SIFT or Max-SIFT descriptors are reduced to 64 using Principal Component Analysis (PCA). We then cluster the descriptors with Gaussian Mixture Model (GMM) of 32 components, and use the improved Fisher vector (IFV) for compact feature encoding. We use sum-pooling with a $\{1 \times 1, 1 \times 3\}$ spatial pyramid. The final vectors are square-root normalized followed by ℓ_2 normalized, and then fed into LibLINEAR [29], a scalable SVM implementation. The average accuracy over all the categories are reported. We repeat the random selection 10 times and report the averaged results.

To compare our model with the state-of-the-art classification performance, stronger features are extracted by resizing the images into 600 pixels in the larger axis, using 10 and 16 for SIFT spatial stride and window size, and using 256 GMM components. Another vector of the same length but generated from the LCS descriptors is concatenated to describe the color features.

4.2. The Comparison of Different Models

First, we report classification performance using SIFT and Max-SIFT descriptors, respectively. For comparison, we also

Dataset	**ORG**	**MAX**	**FLP**
LandUse-21	88.86	89.57	**89.79**
Indoor-67	41.17	**43.20**	42.88
Pet-37	37.92	41.78	**42.01**
Aircraft-100	53.13	**57.72**	57.19
Flower-102	53.68	**58.12**	58.01
Bird-200	25.77	31.59	**31.83**

Table 1. Classification accuracy (%) of different models: using original SIFT descriptor, Max-SIFT descriptor, or using original SIFT with left-right flipping image augmentation. Here, **ORG** and **MAX** denote using original SIFT and Max-SIFT descriptors, while **FLP** represents using SIFT and training and testing with both original and flipped images.

Dataset	**ORG**	**MAX**	Compared with
LandUse-21	93.63	**94.13**	92.8 ([30], 2014)
Indoor-67	61.87	**63.60**	63.4 ([30], 2014)
Pet-37	59.24	**62.39**	56.8 ([31], 2014)
Aircraft-100	70.12	**72.88**	48.7 ([24], 2013)
Flower-102	83.03	**85.45**	84.6 ([31], 2014)
Bird-200	46.61	**49.41**	33.3 ([31], 2014)

Table 2. Comparison of our accuracy (%) with recently published papers. We report both **ORG** and **MAX** results.

report the use of SIFT and Max-SIFT with dataset augmentation. By augmentation we mean to generate a flipped copy for each training/testing image, use the enlarged set to train the model, test each image with original and flipped samples, and predict the label with a soft-max function [20].

Results are summarized in Figure 1. One can see that Max-SIFT (**MAX**) produces consistent accuracy gain beyond original SIFT (**ORG**). Although the accuracy (**MAX**) is sometimes a little bit lower than that using dataset augmentation (**FLP**), cheap computational costs allow us use more powerful features with Max-SIFT, *e.g.*, a larger codebook.

It is also interesting to note that Max-SIFT produces significant improvement on the fine-grained object recognition tasks. For example, the absolute accuracy gain on the Pet-37, Aircraft-100 and Flower-102 datasets is about 4%, and the number is nearly 6% on the Bird-200 dataset (22.58% relative improvement). The reason lies in the significant asymmetry of fine-grained objects (*e.g.*, pets, aircrafts, flowers and birds), which implies that an object and its flipped copy might produce totally different representation vectors using SIFT descriptors. In such cases, machine learning algorithms have to consider two distinct prototypes for each object, which significantly reduces the number of training samples per prototype and increases the chance of over-fitting.

To verify that our algorithm could produce competitive classification performance, we use the strong feature settings and compare the results with some recently published papers on these datasets. We report the results with SIFT and Max-SIFT descriptors, *i.e.*, the **ORG** and **MAX** models. For the fine-grained datasets, we do not compare our method with those using complicated part detection, although these methods are verified to improve the classification accuracy significantly but they go out of the goal of this paper. Results are shown in Table 2. One can observe that our algorithm achieves the state-of-the-art accuracy on all the six datasets.

4.3. Computational Costs

Finally, we report the time/memory cost of our algorithm. Since the only difference between SIFT and Max-SIFT descriptors is the permutation and maximum operation, the extra time cost of Max-SIFT is merely about 1% of original SIFT computation. Moreover, Max-SIFT does not require any additional memory consumptions since it is just a permutation of original SIFT. However, if the datasets are augmented with left-right flipping operation, one needs to store two instances for each image, descriptor set and feature vector along the BoF flowchart, resulting in almost doubled time and memory consumptions in both training and testing processes. Therefore, our algorithm is much more scalable and applicable onto large-scale image classification tasks.

5. CONCLUSIONS

In this paper, we propose Max-SIFT, a kind of flipping invariant descriptors for image classification. Based on the SIFT descriptor, we achieve the flipping invariance by observing the impact of flipping operations on SIFT, and then cancel out the flipping operation by performing maximum on original and flipped descriptors. Experiments reveal that our algorithm achieves comparable classification accuracy with those performing dataset augmentation, meanwhile it is much more scalable since lower time and memory consumptions are required. In the future, we can further apply the Max-SIFT descriptor onto large-scale image classification tasks.

6. ACKNOWLEDGEMENTS

This work was supported by the National Basic Research Program (973 Program) of China (Grant Nos. 2013CB329403, 2012CB316301, and 2014CB347600), the National Natural Science Foundation of China (Grant Nos. 61332007, 61273023 and 61429201), and the Tsinghua University Initiative Scientific Research Program (Grant No. 20121088071). This work was also supported in part to Dr. Tian by ARO grant W911NF-12-1-0057, Faculty Research Awards by NEC Laboratories of America.

7. REFERENCES

[1] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual Categorization with Bags of Keypoints," *Workshop of SL in CV, ECCV*, 2004.

[2] D.G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *IJCV*, 2004.

[3] K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman, "The Devil is in the Details: An Evaluation of Recent Feature Encoding Methods," *BMVC*, 2011.

[4] Y. Chai, V. Lempitsky, and A. Zisserman, "Symbiotic Segmentation and Part Localization for Fine-Grained Categorization," *ICCV*, 2013.

[5] L. Xie, Q. Tian, and B. Zhang, "Max-SIFT: Flipping Invariant Descriptors for Web Logo Search," *ICIP*, 2014.

[6] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust Wide-Baseline Stereo from Maximally Stable Extremal Regions," *Image and Vision Computing*, 2004.

[7] A. Bosch, A. Zisserman, and X. Munoz, "Scene Classification via pLSA," *ICCV*, 2006.

[8] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-Constrained Linear Coding for Image Classification," *CVPR*, 2010.

[9] J. Sanchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image Classification with the Fisher Vector: Theory and Practice," *IJCV*, 2013.

[10] L. Xie, Q. Tian, M. Wang, and B. Zhang, "Spatial Pooling of Heterogeneous Features for Image Classification," *TIP*, 2014.

[11] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories," *CVPR*, 2006.

[12] L. Xie, Q. Tian, R. Hong, S. Yan, and B. Zhang, "Hierarchical Part Matching for Fine-Grained Visual Categorization," *ICCV*, 2013.

[13] L. Xie, J. Wang, B. Guo, B. Zhang, and Q. Tian, "Orientational Pyramid Matching for Recognizing Indoor Scenes," *CVPR*, 2014.

[14] L. Xie, Q. Tian, and B. Zhang, "Feature Normalization for Part-based Image Classification," *ICIP*, 2013.

[15] Q. Tian, N. Sebe, M.S. Lew, E. Loupias, and T.S. Huang, "Content-based Image Retrieval using Wavelet-based Salient Points," *Photonics West-Electronic Imaging*, 2001.

[16] Q. Tian, J. Yu, Q. Xue, and N. Sebe, "A New Analysis of the Value of Unlabeled Data in Semi-supervised Learning for Image Retrieval," *ICME*, 2004.

[17] S. Zhang, Q. Huang, S. Jiang, W. Gao, and Q. Tian, "Affective Visualization and Retrieval for Music Video," *TMM*, 2010.

[18] W. Zhou, H. Li, Y. Lu, and Q. Tian, "Principal Visual Word Discovery for Automatic License Plate Detection," *TIP*, 2012.

[19] L. Xie, Q. Tian, W. Zhou, and B. Zhang, "Fast and Accurate Near-Duplicate Image Search with Affinity Propagation on the ImageWeb," *CVIU*, 2014.

[20] M. Paulin, J. Revaud, Z. Harchaoui, F. Perronnin, and C. Schmid, "Transformation Pursuit for Image Classification," *CVPR*, 2014.

[21] Y. Yang and S. Newsam, "Bag-of-Visual-Words and Spatial Extensions for Land-Use Classification," *International Conference on AGIS*, 2010.

[22] A. Quattoni and A. Torralba, "Recognizing Indoor Scenes," *CVPR*, 2009.

[23] O.M. Parkhi, A. Vedaldi, A. Zisserman, and C.V. Jawahar, "Cats and Dogs," *CVPR*, 2012.

[24] S. Maji, E. Rahtu, J. Kannala, M. Blaschko, and A. Vedaldi, "Fine-Grained Visual Classification of Aircraft," *Technical Report*, 2013.

[25] M.E. Nilsback and A. Zisserman, "Automated Flower Classification over a Large Number of Classes," *Indian Conference on CVGIP*, 2008.

[26] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD Birds-200-2011 Dataset," *Technical Report*, 2011.

[27] A. Vedaldi and B. Fulkerson, "VLFeat: An Open and Portable Library of Computer Vision Algorithms," *ACM Multimedia*, 2010.

[28] R. Arandjelovic and A. Zisserman, "Three Things Everyone Should Know to Improve Object Retrieval," *CVPR*, 2012.

[29] R.E. Fan, K.W. Chang, C.J. Hsieh, X.R. Wang, and C.J. Lin, "LIBLINEAR: A Library for Large Linear Classification," *JMLR*, 2008.

[30] T. Kobayashi, "Dirichlet-based Histogram Feature Transform for Image Classification," *CVPR*, 2014.

[31] N. Murray and F. Perronnin, "Generalized Max Pooling," *CVPR*, 2014.