FEATURE NORMALIZATION FOR PART-BASED IMAGE CLASSIFICATION

Lingxi Xie[1], Qi Tian[2], and Bo Zhang[1]

[1]State Key Laboratory of Intelligent Technology and Systems (LITS)
[1]Tsinghua National Laboratory for Information Science and Technology (TNList)
[1]Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China
[2]Department of Computer Science, University of Texas at San Antonio, Texas, USA
[1]198808xc@gmail.com,[2]qitian@cs.utsa.edu,[3]dcszb@mail.tsinghua.edu.cn

ABSTRACT

Part-based Bag-of-Features (BoF) models such as Spatial Pyramid Matching (SPM) play an important role in image classification. Before sending the feature vectors into classifiers for training and testing, it is required to normalize them in order to approximately equalize ranges of the attributes and make them have comparable effects in distance computation. Although some works have been focused on general feature normalization, we do not see any discussion on specialized normalization algorithms for part-based BoF models.

In this paper, we fill in the blank with extensive experiments and discussions. Based on solid normalization parameters (power and coefficient), we further study two straightforward part-based properties, *i.e.*, the independent assumption and the hierarchical-contribution assumption, to scale the feature super-vectors separately. Finally, we test our algorithm on challenging image sets, *i.e.*, Caltech101 and CUB-200-2011, for general and fine-grained classification, and show its efficiency, scalability and adaptability in both scenarios.

Index Terms— Part-Based Bag-of-Features Models, Image Classification, Feature Normalization, Experiments.

1. INTRODUCTION

Large-scale image classification has been a hot topic for many years. It is a challenge towards image understanding and implies a wide range of real applications. Today, one of the most popular methods of image classification is to represent images with long super-vectors, and use a generalized classifier (such as SVM [1]) for training and testing.

The Bag-of-Features (BoF) model [2] [3] is widely used for image representation. It is a statistics-based model which summarizes local features in a sparse vector. The major shortcomings of the BoF model come from the well-known gap between low-level pixels and high-level concepts [4] [5]. In recent years, new modules were proposed to bridge the semantic gap and provide more robust image representations. System with the state-of-the-art techniques [6] produces relatively discriminative representation in terms of super-vectors.

Before using the super-vectors for classification, feature normalization is considered as an important data preprocessing step to avoid attributes in greater numeric ranges dominating those in smaller numeric ranges. Despite the existing works on feature normalization applied to various classification models, there still lacks a specialized algorithm for part-based BoF models.

In this respect, we conduct an in-depth study on part-based feature normalization by proposing several novel algorithms and comparing them with baseline systems. We claim a two-fold contribution. First, we formulate power and coefficient, two normalization parameters, and show their tremendous impact on classification accuracies in experiments. Second, taking the part-based properties into consideration, we develop specialized normalization approaches for part-based BoF models. Integrating both techniques obtains an enhanced algorithm to outperform the state-of-the-art models.

The rest of this paper is organized as follows. First, we provide a brief overview of the BoF model in Section 2. Then Section 3 proposes several novel algorithms for feature normalization. After extensive experiments and discussions are given in Section 4, we draw our conclusions in Section 5.

2. THE BAG-OF-FEATURES MODEL

The Bag-of-Features model starts from a raw image $\mathbf{I}$:

$$\mathbf{I} = (a_{ij})_{W \times H} \tag{1}$$

where a_{ij} is the **pixel** at position (i, j). For better local representation, we extract a set of SIFT [7] descriptors $\mathcal{D}$:

$$\mathcal{D} = \{\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_M\} \tag{2}$$

where $\mathbf{d}_m$ denotes the **description vector** of the m-th descriptor. M is the total number of descriptors.

After descriptors have been extracted, they are quantized to be compact. For this purpose, we train a **codebook** $\mathbf{C}$, which is a $B \times D$ matrix consisting of B vectors with dimension D, each of which is called a **codeword**. The **codebook size** B, is thousands in classification tasks.

978-1-4799-2341-0/13/$31.00 ©2013 IEEE

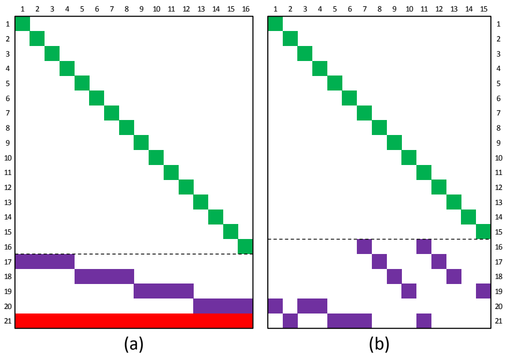

Fig. 1. Illustrations of pooling matrices. Rows are pooling bins, columns are basic regions, and shaded blocks are entries with value 1. Above the dash line are basic pooling bins (green), and below are high-level ones. (a): SPM [9], where the basic regions are $16(4 \times 4)$ grids on the image plane, and we manually set 2×2 and 1×1 grids as high-level bins. (b): HPM [10], where the basic regions are semantic body parts of birds, *e.g.*, beak, nape, left/right wing, and high-level bins are heuristically learned parts, *e.g.*, neck (nape+throat).

Next, descriptors are quantized using the codewords. This process is called **coding**, for we encode each descriptor as a sparse vector using the Locality-constrained Linear Coding (LLC) [8] algorithm. Given a codebook with B codewords, the encoded vector or **feature vector** would be a B-dimensional code $\mathbf{w}_m$, which is named the corresponding **visual word** of **descriptor** $\mathbf{d}_m$. Let $\mathcal{W}$ be the visual word set:

$$\mathcal{W} = \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_M\} \tag{3}$$

We aggregate the visual words using global max-pooling:

$$\mathbf{f} = \max_{1 \leqslant m \leqslant M} \mathbf{w}_m \tag{4}$$

and take $\mathbf{f}$ as the global **representation vector** of image $\mathbf{I}$.

However, global pooling ignores spatial information, which is very useful to image understanding. The state-of-the-art image classification systems divide images into smaller regions and construct a hierarchical structure for spatial context modeling. Successful cases include Spatial Pyramid Matching (SPM) [9] and Hierarchical Part Matching (HPM) [10]. After dividing the image (object) into U **basic regions**, we use prior knowledge [9] or heuristic learning [10] to define S spatial **pooling bins** for multi-level individual statistics. The corresponding relation between pooling bins and basic regions are represented as an $S \times U$ **pooling matrix**:

$$\mathbf{P} = (p_{su})_{S \times U} \tag{5}$$

where p_{su} is either 1 (which means s-th pooling bin contains u-th basic region) or 0 (otherwise). Standard pooling matrices for SPM [9] and HPM [10] are plotted in Figure 1.

The individually pooled **representation vectors** are finally concatenated as a **super-vector**:

$$\mathbf{F} = (\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_S) \tag{6}$$

which is the output of the part-based BoF model.

3. FEATURE NORMALIZATION

Feature normalization, or feature scaling, is a basic technique for data pre-processing. With a clear motivation to approximately equalize the range and weight of input attributes, researchers proposed various techniques to apply on different classification models, such as Support Vector Machines [11] [12] [13], Naive Bayes Classifier [14], Hidden Markov Model Estimation [15], Kernel Fisher Discriminant Analysis [16], and even image retrieval systems [17].

3.1. Normalization Power and Coefficient

One of the simplest and most widely-used formulation is the ℓ_p-normalization, which calculates the super-vector's linear projection on the ℓ_p-norm unit hyper-sphere:

$$\widetilde{\mathbf{F}} = \frac{\mathbf{F}}{\left\|\mathbf{F}\right\|_p} \tag{7}$$

where $\left\|\mathbf{F}\right\|_p$ is the ℓ_p-norm: $\left\|\mathbf{F}\right\|_p = \left(\sum F_i^p\right)^{1/p}$, and p is named the **normalization power**. In most cases, we use SVM for image classification. Since the formulation of SVM is quite sensitive to the numerical ranges of input data, it is reasonable to choose a proper normalization coefficient for robust feature super-vectors. For this, we modify (7) as:

$$\widetilde{\mathbf{F}} = w \times \frac{\mathbf{F}}{\left\|\mathbf{F}\right\|_p} \tag{8}$$

where w is the **normalization coefficient**.

We plot the classification results using (8) in Figure 2. Detailed settings are mentioned in Section 4. It is shown that proper p and w are very important for normalization. Further, we claim the ability of ℓ_1-normalization by showing its comparable performances with ℓ_2 and ℓ_∞, under a large coefficients $w = 100$ which is suitable for all powers. This refutes the opinion that ℓ_1-normalization would cause classification accuracies drop dramatically [8]. **In later experiments, we fix $w = 100$ for all normalization powers.**

3.2. Separate Normalization by Parts

Till now, we make no efforts on designing specialized normalization methods for part-based BoF models. If we assume that each part contributes independently to the entire model (**the independent assumption**), it is straightforward to keep

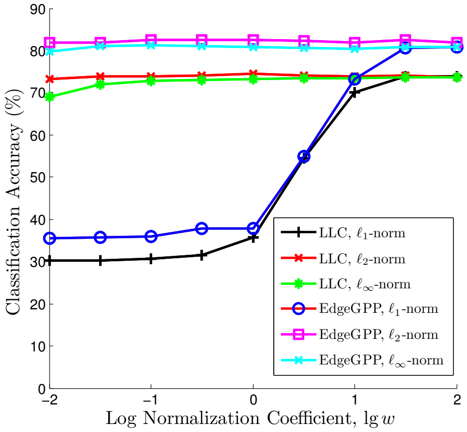

Fig. 2. LLC [8] and EdgeGPP [6] classification results on the Caltech101 Dataset using different normalization powers p and coefficients w. w is dominant under ℓ_1-normalization.

the same ℓ_p-norm individually in all the pooling bins, which leads in normalizing $\mathbf{f}_1$, $\mathbf{f}_2$, ..., $\mathbf{f}_S$ in (6) separately:

$$\widetilde{\mathbf{f}_s} = \frac{w}{S^{1/p}} \frac{\mathbf{f}_s}{\|\mathbf{f}_s\|_p}, \qquad s = 1, 2, \ldots S \tag{9}$$

and concatenating $\widetilde{\mathbf{f}_1}$, $\widetilde{\mathbf{f}_2}$, ..., $\widetilde{\mathbf{f}_S}$ again:

$$\widetilde{\mathbf{F}} = \left(\widetilde{\mathbf{f}_1}, \widetilde{\mathbf{f}_2}, \ldots, \widetilde{\mathbf{f}_S}\right) \tag{10}$$

The modified normalization coefficient $\frac{w}{S^{1/p}}$ in (9) aims to preserve $\|\mathbf{F}\|_p = w$ as it is in (8).

3.3. Hierarchical Weighting Scheme (HRC)

It is not always true that each part contributes equally. Most often, high-level pooling bins consist of more basic regions and therefore are more robust and discriminative. Denote k_s as the number of basic regions contained in s-th pooling bin, and assume that s-th bin's contribution is proportional to $k_s^{1/p}$ (**the hierarchical-contribution assumption**). Appending the fixed constraint $\|\mathbf{F}\|_p = w$ obtains the equations system:

$$\left\{ \begin{array}{rcl} w_s^p & \propto & k_s \\ \sum_{s=1}^{S} w_s^p & = & w^p \end{array} \right. \tag{11}$$

Solving (11) gives a group of new coefficients for part-wise normalization, in which we enhance the spatial weights on high-level pooling bins to emphasize global information.

4. EXPERIMENTAL RESULTS

This section shows experimental results on different part-based BoF models: SPM [9] [6] on Caltech101 [18], and HPM [10] on CUB-200-2011 [19]. To make comparison, we keep the same settings as the referred literatures:

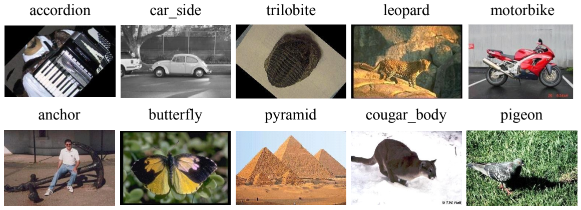

Fig. 3. Sample images from the Caltech101 dataset.

Table 1. Classification results on Caltech101 using different models and normalization techniques The normalization coefficient w is set as 100 in all cases. The standard deviation of each result is around 0.6%.

Algorithm	LLC	GPP	EdgeGPP
No normalization	73.14	76.35	80.78
Global ℓ_1	73.91	76.26	80.86
Global ℓ_2	**74.41**	**77.03**	82.45
Global ℓ_∞	73.25	76.47	80.89
Separate ℓ_1	71.99	75.20	78.05
Separate ℓ_2	73.68	75.47	81.24
Separate ℓ_∞	73.39	76.43	80.93
Separate ℓ_1 + HRC	72.71	75.88	80.40
Separate ℓ_2 + HRC	74.31	76.86	**83.19**
Separate ℓ_∞ + HRC	73.89	76.55	81.37

- **Local descriptors.** We use the VLFeat [20] library to extract SIFT [7] or or OppSIFT [21] descriptors for grayscale and color images, respectively.
- **Codebook learning.** We train a 2048-entry codebook with K-Means clustering. The number of SIFT descriptors collected for training is around 2 million.
- **Coding.** We use LLC [8] for coding, and Geometric Phrase Pooling [10] for visual phrase enhancement.
- **Classification.** A linear SVM, LibLinear [22], is used for training and testing. The penalty parameter C for slack variables is 10.
- **Accuracy evaluation.** To test our algorithm, we use random data split which is repeated 10 times and average accuracies and standard deviations are reported.

4.1. The Caltech101 Dataset

The Caltech101 Dataset [18] is a basic object collection containing 9144 images from 102 categories. Sample images are

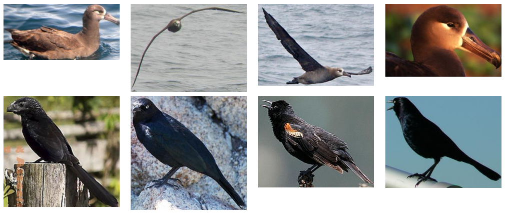

Fig. 4. Samples from the CUB-200-2011 Dataset. Upper: widely different birds from the same category (Black-footed Albatross). Lower: similar birds from different species.

Table 2. Classification results on CUB-200-2011 using different models and normalization techniques The normalization coefficient w is set as 100 in all cases. The standard deviation of each result is around 0.3%.

Algorithm	LLC	LLC-HP	GPP-HP
No normalization	25.58	27.55	30.67
Global ℓ_1	27.61	29.85	33.22
Global ℓ_2	27.06	29.30	32.96
Global ℓ_∞	25.04	27.41	30.71
Separate ℓ_1	24.58	26.94	31.84
Separate ℓ_2	**30.93**	32.75	35.98
Separate ℓ_∞	27.73	29.67	31.92
Separate ℓ_1 + HRC	–	28.12	33.85
Separate ℓ_2 + HRC	–	**32.89**	**36.48**
Separate ℓ_∞ + HRC	–	29.45	32.08

listed in Figure 3. Following [6], we use 30 images per category for training, and others for testing.

We apply 3 versions of SPM models, *i.e.*, a traditional BoF model (LLC [8]) for sparse coding, an improved model (GPP [6]) for spatial context modeling, and an ultimate version (EdgeGPP [6]) for combining heterogeneous features. Classification results are listed in Table 1.

4.2. The Caltech-UCSD Birds-200-2011 Dataset

The Caltech-UCSD Birds-200-2011 Dataset [19] contains 200 bird species and 11788 images in total. As shown in Figure 4, it is a very challenging fine-grained image collection. Following [10], we use 5 images per category to train the model, and test it on remaining images.

We apply 3 versions of HPM models learned from [10], with the difference of whether to learn a hierarchical part structure and whether to use Geometric Phrase Pooling. We name them LLC, LLC-HP (Hierarchical Part) and GPP-HP, respectively. Classification results are listed in Table 2.

4.3. Discussions

Here are some discussions based on experimental results.

- **Global ℓ_1-norm vs. ℓ_2-norm.** It is a popular opinion that ℓ_1-normalization would cause the classification accuracies drop, especially in the models using max-pooling [8] [23]. However, we show in experiments that with a large enough coefficient w, global ℓ_1-normalization gives comparable and even higher accuracies than global ℓ_2 (see Table 2). *i.e.*, hierarchical weighting beyond separate normalization.
- **Separate normalization vs. model selection.** The effect of separate normalization is highly related to model selection. SPM uses a naive spatial division (4×4 grids), while HPM works on semantic body parts (beak, head, wings, tail, etc.). Comparatively speaking, HPM is more likely to preserve part-based properties, especially the independent assumption. That is why separate normalization produces much higher accuracies in HPM, but works slightly worse in SPM.
- **Hierarchical weighting.** Hierarchical weighting works better than equal weighting in both scenarios. Here, we benefit from the hierarchical-contribution assumption, and exploit a straightforward estimation to increase the weighting on more robust spatial bins.

In general, the choice of normalization strategy has a great impact on classification accuracies. The more we exploit prior knowledge, the better normalization results we could obtain.

5. CONCLUSIONS

In this paper, we claim the importance of feature normalization in part-based Bag-of-Features models, by representing novel algorithms, extensive experiments and in-depth discussions. Based on proper normalization power and coefficient, our approach differs from previous ones in the careful consideration of part-based properties, *i.e.*, the independent assumption and the hierarchical-contribution assumption. Experimental results provide strong evidences to support our statements: (1) ℓ_1-normalization is also good for image classification, and (2) hierarchical weighting beyond separate normalization is supreme in part-based BoF models.

6. ACKNOWLEDGEMENT

This work was supported by the National Basic Research Program (973 Program) of China under Grant 2012CB316301, and Basic Research Foundation of Tsinghua National Laboratory for Information Science and Technology (TNList). This work was also supported in part to Dr. Qi Tian by ARO grant W911NF-12-1-0057, NSF IIS 1052851, Faculty Research Awards by Google, NEC Laboratories of America and FXPAL, UTSA START-R award and NSFC 61128007.

7. REFERENCES

[1] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer, 1999.

[2] Josef Sivic and Andrew Zisserman, "Video Google: A Text Retrieval Approach to Object Matching in Videos," *International Conference on Computer Vision*, 2003.

[3] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual Categorization with Bags of Keypoints," *Workshop on Statistical Learning in Computer Vision, European Conference on Computer Vision*, 2004.

[4] S. Zhang, Q. Tian, G. Hua, Q. Huang, and S. Li, "Descriptive Visual Words and Visual Phrases for Image Applications," 2009.

[5] Y. Lu, L. Zhang, J. Liu, and Q. Tian, "Constructing Lexica of High-level Concepts with Small Semantic Gap," *IEEE Transactions on Multimedia*, 2010.

[6] L. Xie, Q. Tian, and B. Zhang, "Spatial Pooling of Heterogeneous Features for Image Applications," *ACM Multimedia*, 2012.

[7] David G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, 2004.

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

[9] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce, "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories," *CVPR*, 2006.

[10] L. Xie, Q. Tian, S. Yan, and B. Zhang, "Hierarchical Part Matching for Fine-Grained Visual Categorization," *Technical Report, Department of Computer Science and Technology, Tsinghua Univerity*, 2013.

[11] A. Graf and S. Borer, "Normalization in Support Vector Machines," *Pattern Recognition*, 2001.

[12] P. Juszczak, DMJ Tax, and RPW Duin, "Feature Scaling in Support Vector Data Description," *Annual Conference of the Advanced School for Computing and Imaging*, 2002.

[13] A. Stolcke, S. Kajarekar, and L. Ferrer, "Nonparametric Feature Normalization for SVM-Based Speaker Verification," *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008.

[14] E. Youn and M.K. Jeong, "Class Dependent Feature Scaling Method Using Naive Bayes Classifier for Text Datamining," *Pattern Recognition Letters*, 2009.

[15] S. Tsakalidis, V. Doumpiotis, and W. Byrne, "Discriminative Linear Transforms for Feature Normalization and Speaker Adaptation in HMM Estimation," *IEEE Transactions on Speech and Audio Processing*, 2005.

[16] L. Bo, L. Wang, and L. Jiao, "Feature Scaling for Kernel Fisher Discriminant Analysis Using Leave-One-Out Cross Validation," *Neural Computation*, 2006.

[17] S. Aksoy and R.M. Haralick, "Feature Normalization and Likelihood-based Similarity Measures for Image Retrieval," *Pattern Recognition Letters*, 2001.

[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," *Computer Vision and Image Understanding*, 2007.

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

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

[21] K.E.A. Van De Sande, T. Gevers, and C.G.M. Snoek, "Evaluating Color Descriptors for Object and Scene Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010.

[22] R.E. Fan, K.W. Chang, C.J. Hsieh, X.R. Wang, and C.J. Lin, "LIBLINEAR: A Library for Large Linear Classification," *Journal of Machine Learning Research*, 2008.

[23] Y. Boureau, F. Bach, Y. LeCun, J. Ponce, et al., "Learning Mid-Level Features for Recognition," *Computer Vision and Pattern Recognition*, 2010.