# $\mathcal{L}_p$-norm IDF for Large Scale Image Search

Liang Zheng[1], Shengjin Wang[1], Ziqiong Liu[1], and Qi Tian[2]
[1]Tsinghua University, Beijing, China
[2]University of Texas at San Antonio, TX, 78249, USA
zheng-l06@mails.tsinghua.edu.cn wgsgj@tsinghua.edu.cn
liuziqiong@ocrserv.ee.tsinghua.edu.cn qitian@cs.utsa.edu

## Abstract

*The Inverse Document Frequency (IDF) is prevalently utilized in the Bag-of-Words based image search. The basic idea is to assign less weight to terms with high frequency, and vice versa. However, the estimation of visual word frequency is coarse and heuristic. Therefore, the effectiveness of the conventional IDF routine is marginal, and far from optimal. To tackle this problem, this paper introduces a novel IDF expression by the use of $\mathcal{L}_p$-norm pooling technique. Carefully designed, the proposed IDF takes into account the term frequency, document frequency, the complexity of images, as well as the codebook information. Optimizing the IDF function towards optimal balancing between TF and pIDF weights yields the so-called $\mathcal{L}_p$-norm IDF (pIDF). We show that the conventional IDF is a special case of our generalized version, and two novel IDFs,* i.e. *the average IDF and the max IDF, can also be derived from our formula. Further, by counting for the term-frequency in each image, the proposed $\mathcal{L}_p$-norm IDF helps to alleviate the visual word burstiness phenomenon.*

*Our method is evaluated through extensive experiments on three benchmark datasets (Oxford 5K, Paris 6K and Flickr 1M). We report a performance improvement of as large as 27.1% over the baseline approach. Moreover, since the $\mathcal{L}_p$-norm IDF is computed offline, no extra computation or memory cost is introduced to the system at all.*

## 1. Introduction

This paper considers the task of large scale image search for particular object. Given a query image of an object, our goal is to retrieve from a large image database all the images containing the same object in real time.

Recent years have witnessed a rapid growth of research in image search and a myriad of models have been proposed [11, 18]. Among them, the Bag-of-Words model [15] is the most popular and perhaps the most successful one. This model starts from the extraction of salient local regions from an image and representing each local patch as a high-dimensional feature vector (*e.g.* SIFT [7] or its variants [13]). Then the continuous high dimensional feature space is divided into a discrete space of *visual words*. This step is achieved by constructing a *codebook* through unsupervised clustering, e.g. k-means algorithm. To improve efficiency, approximate k-means [11] and hierarchical k-means [8] have been used. The Bag-of-Words model then treats each cluster center as a word in the codebook. In the spirit of text retrieval, the method quantizes each detected keypoint into its nearest visual word(s) and represents each image as a histogram of visual words. Finally, images are ranked using various indexing methods [15, 21] in real time.

*Image Collection*

| *Visual Words* | $I_1$ | $I_2$ | $I_3$ | $I_4$ | $I_5$ | $I_6$ |
|---|---|---|---|---|---|---|
| $z_x$ | 5 | 7 | 1 | 24 | 2 | 9 |
| $z_y$ | 3 | 1 | 10 | 7 | 4 | 2 |

Figure 1. An toy example of an image collection. Visual words $z_x$ and $z_y$ both occurs in all the six images, but with varying T-F distributions over the entire image collection. In conventional IDF, the IDF weights are equal to zero for both words. But when resorting to TF, $z_x$ and $z_y$ both have some discriminative power, the problem of which will be tackled in this paper.

To measure the importance of visual words, most of the existing approaches use the TF-IDF (Term Frequency-Inverse Document Frequency) weighting scheme. However, the conventional IDF method has two drawbacks. First, the conventional IDF functions on the image collection level. It

1063-6919/13 $26.00 © 2013 IEEE
DOI 10.1109/CVPR.2013.213

does not take a closer look into the visual word level, where multiple occurrences of a visual word are often observed. Consequently, it only makes a coarse estimation of visual word frequency. Second, as suggested in [5], IDF weighting does not address the problem of *burstiness*. Burstiness brings about a burst in false matches and compromises the image search accuracy.

In this paper, we propose a novel IDF formula, called "$\mathcal{L}_p$-norm IDF", which makes a careful estimation of visual word frequency and achieves significant improvement in performance. The key idea is that the estimated visual word frequency is the weighted sum of the TF data across the whole database. We show that the conventional IDF is a special case of our generalized version. Meanwhile, two other novel IDFs, termed average IDF and max IDF, can be derived from our method. Experimental studies on three image search datasets confirm that by integrating the term frequency into IDF using the proposed method, image search performance is improved dramatically. Furthermore, since the $\mathcal{L}_p$-norm IDF is computed offline, no extra computational cost is introduced and efficiency is ensured.

The rest of the paper is organized as follows. After a brief review of related work in Section 2, we introduce the proposed $\mathcal{L}_p$-norm IDF formula in Section 3. Visual word burstiness is illustrated in Section 4. In Section 5, we demonstrate the experimental results of our method. Finally, conclusions are drawn in Section 6.

## 2. Related Work

Built on the Bag-of-Words (BoW) model, a large body of literature has been proposed to improve performance.

One group of work mainly deals with the quantization error. For example, soft matching [10, 16] assigns each descriptor to multiple visual words, but instead increases the query time and memory overload. Hamming embedding [4] provides binary signatures to filter out false matches. [2] designs quantization method by kernel density estimation, while [1, 24] utilize binary features to improve efficiency and reduce quantization error.

Another popular topic is to encode spatial constraints into the search framework, such as the weak geometric consistency [4], and RANSAC verification [11]. The geometric context among local features can be also encoded into visual word assemblies [20, 22, 19]. By geometric constraints [23], inconsistent matches are filtered out.

The third group of work concerns about visual word weighting. For example, [5] uses IDF-like weighting formulas to tackle the burstiness problem. The combination of these methods has obtained good accuracy. X. Wang *et al*. [17] proposes to incorporate the information of both the vocabulary tree and the image spatial domain into the contextual weighting. Our work, instead, re-estimate the visual word frequency by $\mathcal{L}_p$-norm pooling in an offline manner. We optimize the parameter to achieve a good balance between TF and IDF weights.

## 3. Proposed Approach

This section gives a formal description of our proposed $\mathcal{L}_p$-norm IDF formula. An image collection possesses $N$ images, denoted as $D = \{I_i\}_{i=1}^{N}$. Each image $I_i$ has a set of keypoints $\{x_j\}_{j=1}^{d_i}$, where $d_i$ is the number of keypoints in $I_i$. Given the codebook $\{z_i\}_{k=1}^{K}$ with a vocabulary size of $K$, image $I_i$ is quantized into a vector representation $\boldsymbol{v}_i = [v_{i,1}, v_{i,2}, ..., v_{i,K}]^T$, where $v_{i,k}$ stands for the response of visual word $z_k$ in $I_i$.

### 3.1. Conventional TF-IDF

The TF part of the weighting scheme reflects the number of keypoints featured by this visual word. As a result, the TF distribution in an image is informative about textures, such as repetitive structures. On the other hand, the IDF part determines the contribution of a given visual word. The presence of a less common visual word in an image may be a better discriminator than that of a more common one. The IDF weight of a visual word $z_k$ is denoted as:

$$IDF(z_k) = \log \frac{N}{n_k} \tag{1}$$

where $N$ denotes the total number of images in the collection, and $n_k$ encodes the number of images where $z_k$ occurs.

Taking into account the TF-IDF weighting, the similarity score between two images is,

$$sim(\boldsymbol{q}, \boldsymbol{d}) = \frac{\sum_{i=1}^{K} q_i d_i IDF(i)^2}{\|\boldsymbol{q}\|\|\boldsymbol{d}\|}, \tag{2}$$

where $\|\cdot\|$ is the normalization factor. A comparison between different normalization methods is shown in Section 5.3. We use the same similarity function as in [4].

In addition, in text retrieval, a variety of weighting methods have been proposed, such as the Okapi-BM25 [12], the pivoted normalization weighting [14], etc. Typically, these methods focus on the TF part.

### 3.2. $\mathcal{L}_p$-norm IDF

The basic idea of IDF is the negative correlation between the visual word frequency $u_k$ and the IDF weight. The conventional IDF treats $u_k$ as the number of images possessing word $z_k$. Although this strategy agrees with the basic idea, the estimation of $u_k$ is coarse. In an extreme case as illustrated in Fig. 1, visual words $z_x$ and $z_y$ appear in all the images through $I_1$ to $I_6$. According to Eq. 1, both IDF($z_x$) and IDF($z_y$) are equal to zero. It indicates that $z_x$ and $z_y$ are totally worthless for image search. However, if we consider the fact that the frequency distribution of $z_x$ and $z_y$ over the entire image collection are quite different, we may realize

that these visual words indeed possess some discriminative power, which is ignored by the conventional IDF formula. Therefore, we seek to augment the IDF formula with the TF distribution, a process featured by $\mathcal{L}_p$-norm pooling.

Specifically, assume a collection of images consists of $N$ images, $n_k$ of which contain visual word $z_k$. We denote the image set containing $z_k$ as $\mathcal{P}_k = \{I \in D | z_k \in I\}$, and $|\mathcal{P}_k| = n_k$. From the quantized images $\{\boldsymbol{v}_i\}_{i=1}^N$, we seek to estimate the frequency $u_k$ of $z_k$. Conventional IDF treats $u_k = |\mathcal{P}_k| = n_k$. Our method, instead, employs the $\mathcal{L}_p$-norm pooling [3] to perform the estimation, *i.e.*

$$u_k = \sum_{I_i \in \mathcal{P}_k} w_{i,k} v_{i,k}^p, p \geq 0. \tag{3}$$

Built upon the adjusted estimation of visual word frequency $u_k$, our framework is presented as follows:

$$pIDF(z_k) = \log \frac{N}{u_k} = \log \frac{N}{\sum_{I_i \in \mathcal{P}_k} w_{i,k} v_{i,k}^p}, p \geq 0, \tag{4}$$

where $v_{i,k}$ denotes the occurrences of $z_k$ in image $I_i$. Parameter $p$ determines the extent to which the term frequency contributes to the estimated value. The coefficient $w_{i,k}$ reflects the contribution of each image containing $z_k$ to the frequency estimation. Therefore, $w_{i,k}$ should encode the following properties.

First, images vary a lot in length, *i.e.* the number of visual words it contains. Images with a greater length tend to contain more instances of $z_k$. Put it another way, it is more probable that $z_k$ appears in large images. If it is the case, we should overestimate its frequency, thus lowering its IDF score. Consequently, we should extend the $w_{i,k}$ interpretation by positively correlating it with image length $d_i$. For numerical reasons, it is appropriate to introduce the normalization by relating image length to the average value $\bar{d}$. It ensures that an image of average length has the same weight after image length normalization.

Next, we seek to incorporate codebook information into $w_{i,k}$. Given that $u_k$ is larger for a smaller codebook, another normalization should be considered. We propose to normalize $w_{i,k}$ by the average value of $v_{i,k}$, in the form of $\log(1 + \frac{1}{n_k} \sum_{I_i \in \mathcal{P}_k} v_{i,k})$.

In addition, for practical implementation, the IDF weights of visual words should be non-negative (each visual word, no matter how often it appears in bursts, should at least have some discriminative power).

Taking the aforementioned considerations into account, the $\mathcal{L}_p$-norm IDF is finally formulated in Eq. 5:

$$\begin{aligned} pIDF(z_k) &= \log(1 + \frac{N}{\sum_{I_i \in \mathcal{P}_k} w_{i,k} v_{i,k}^p}), \\ \text{where } p \geq 0, w_{i,k} &= \frac{d_i / \bar{d}}{\log(1 + \frac{1}{n_k} \sum_{I_i \in \mathcal{P}_k} v_{i,k})} \end{aligned} \tag{5}$$

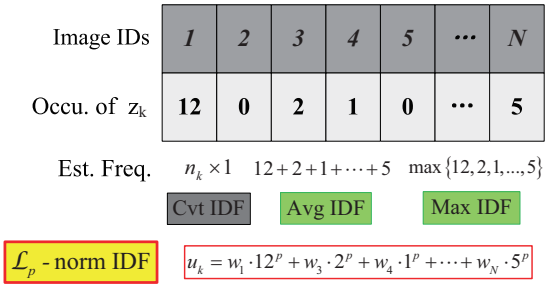



Figure 2. Illustration of four different IDF schemes. A collection consists of $N$ images indexed from 1 to $N$. The term frequencies of word $z_k$ in each image is depicted below. The formulas demonstrate how to calculate the estimated word frequency of $z_k$ for the four IDFs, e.g., the conventional IDF, average IDF, max IDF, as well as the $\mathcal{L}_p$-norm IDF introduced in this paper.

In Eq. 5, $d_i$ and $\bar{d}$ denote the number of features in image $I_i$ and the average number of features for images in the database, respectively.

### 3.3. Average IDF and Max IDF

Suppose $w_{i,k} = 1$ and $p = 0$, Eq. 4 reduces to the conventional IDF representation in Eq. 1. Therefore, the conventional IDF is a special case of the $\mathcal{L}_p$-norm IDF. Moreover, from Eq. 4, we can define two novel IDF variants, *i.e.* the average IDF and the max IDF, if we set $p = 1$ and $p = \infty$, respectively,

$$aIDF(z_k) = \log \frac{N}{u_k} = \log \frac{N}{\sum_{I_i \in \mathcal{P}_k} v_{i,k}}, \tag{6}$$

and

$$mIDF(z_k) = \log \frac{N}{u_k} = \log \frac{N}{\max_i v_{i,k}}, \tag{7}$$

where $u_k$ is approximated by the $\mathcal{L}_1$-norm and $\mathcal{L}_\infty$-norm of $v_{i,k} (i = 1, \cdots, n_k)$, corresponding to the average pooling and max pooling, respectively. An example of the four different IDFs discussed above is presented in Fig. 2.

The pooling technique we used here differs from that in feature pooling in two aspects. First, the subject of the pooling here is the bag-of-words image representation $\boldsymbol{v}_i$, instead of a subregion in the partitioned image in SPM [6]. Second, pooling used here is to aggregate the response of the whole image collection to the frequency accumulates of each visual word, while in feature pooling, the result is the response of an image to each visual word.

### 3.4. Towards Optimal $\mathcal{L}_p$-norm IDF

To determine the optimal value of $p$ in Eq. 5, we seek to minimize a cost function of visual word discriminative power. The TF-IDF weight encodes the importance of a visual word in separating one image from the others, *i.e.* the discriminative power. For $\mathcal{L}_p$-norm IDF, TF and pIDF are

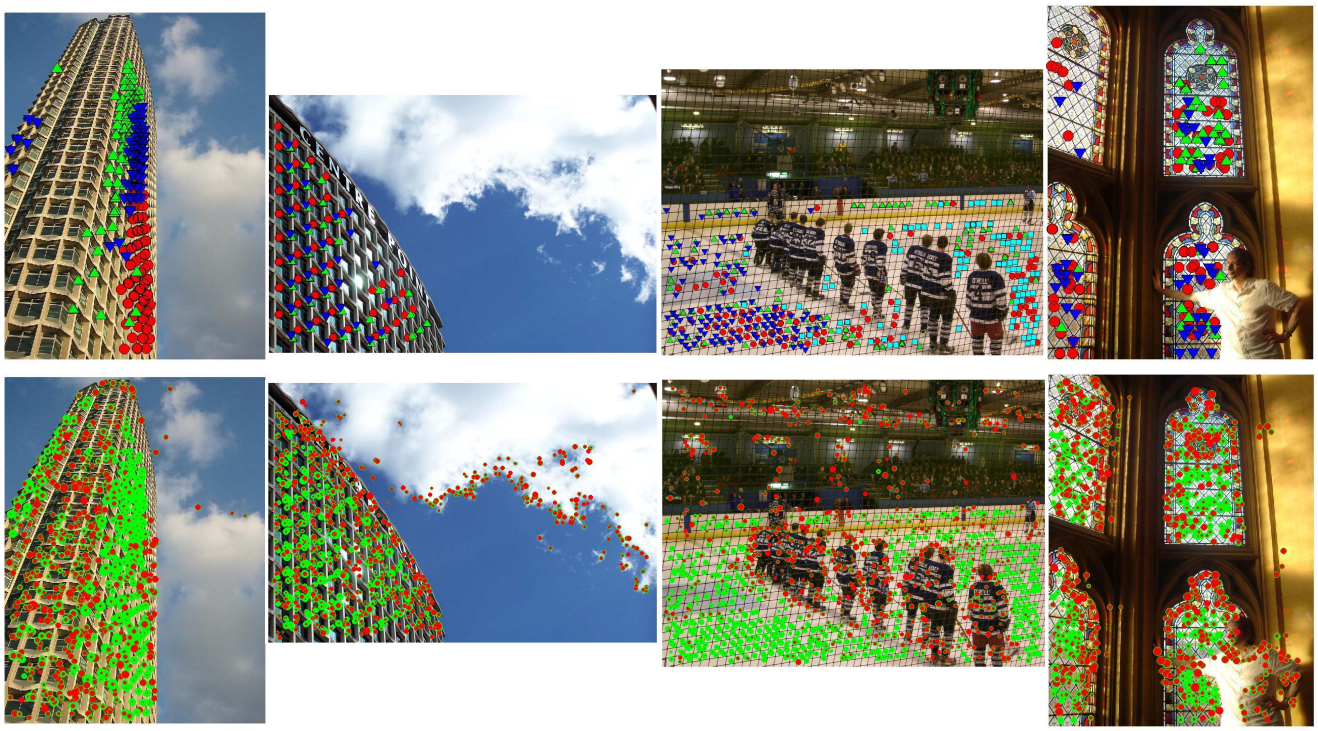


Figure 3. Visual word burstiness and the impact of $\mathcal{L}_p$-norm IDF. (**Top**): the burstiness phenomenon. The same markers represent the same visual words. Note the repetition in these images. (**Bottom**): the impact of $\mathcal{L}_p$-norm IDF. Red and green markers denote the $\mathcal{L}_p$-norm IDF and the conventional IDF, respectively. The area of the markers encodes the strengths of the weights. Note that visual words in repetitive structures (burstiness) are heavily down-weighted, while the weights of discriminative structures are preserved.

negative correlated: pIDF punishes large TF and favors small TF. In other words, the $\mathcal{L}_p$-norm IDF aims at achieving a balance between the two weighting factors. More specifically, the objective function is to minimize the discriminative power diversity among visual words, namely,

$$\arg\min_{p} \underset{k}{var}\{\frac{1}{n_k}\sum_{I_i\in\mathcal{P}_k} v_{i,k}\cdot pIDF(z_k)\} \qquad (8)$$

where the *variance* operator characterizes the diversity of discriminative power among visual words. The discriminative power of a visual word is described by its average TF-pIDF value. Eq. 8 aims to balance optimally the relationship between TF and IDF weights, thus minimizing the discriminative power diversity.

The optimization problem in Eq. 8 does not have a closed form solution for $p$. Therefore, we adopt a greedy search method to obtain the optimal value of $p$. The result is demonstrated in Section 5.3.

## 4. Visual Word Burstiness

In text retrieval, the term *positive adaption* or *burstiness* refers to the phenomenon in which words tend to appear in bursts, *i.e.* once they appear in a document, they are more likely to appear again.

In the image search community, burstiness often describe the phenomenon that repetitive structures are present (see Fig. 3). Burstiness deteriorates the system performance to a very large extent. In [5], intra-image and inter-image burstiness are discussed. Our work differs in that we leverage the term frequencies across the database, and optimize the IDF weight. Another difference lies in that [5] penalizes burstiness by computing a normalization factor on-the-fly, while our method assigns weights to visual words on the visual word level and in an offline manner. A comparison of our method and [5] is shown in Section 5.4.

To analyze the burstiness phenomenon, we plot the visual word distribution in Fig. 4. For each visual word in the codebook, we first count its maximum term frequency across the image collection and form a visual word histogram in Fig. 4(a). Then, we denote the maximum term frequency of image $I$ as $N_I$, and count the number of images that fall into different values of $N_I$, as is shown in Fig. 4(b). The statistics suggests that a majority of visual words maximally occur 2 or 3 times in an image and that most images have a maximal term frequency of 5 or 6. Therefore, the burstiness phenomenon (see the first row of Fig. 3) widely exists in the image search settings, a problem that should be tackled.

The second row of Fig. 3 depicts the impact of the pro-

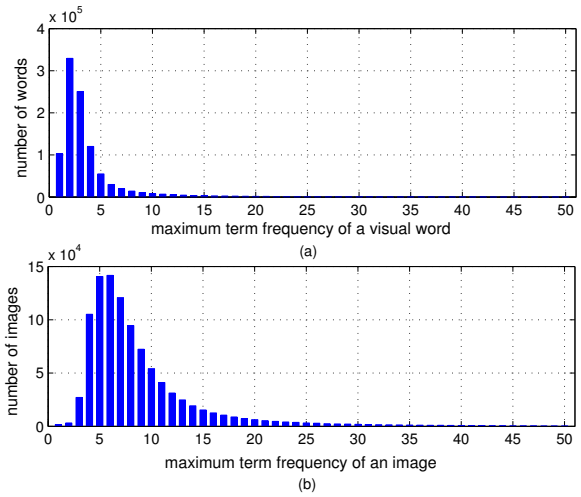



Figure 4. (a): Histogram of visual words for different values of maximum term frequency; (b): Histogram of images for different values of maximum term frequency. The data is evaluated over Flickr 1M dataset, and the codebook size is 1M.

posed $\mathcal{L}_p$-norm IDF. The green and red markers are co-located with visual words, and correspond to the conventional IDF and the $\mathcal{L}_p$-norm IDF weights, respectively. The size of the markers is proportional to the IDF value. Small red markers indicate that the visual words are heavily down-weighted, which suggests that these visual words be part of repetitive structures. On the other hand, big red markers denote slightly, if any, down-weighted visual words, which are quite discriminative structures. In Fig. 3, it is obvious that the more elaborated structures such as people and discriminative shapes are retained as before. However, the repetitive structures are heavily punished, involving man-made constructions and structured background, etc. As a result, the $\mathcal{L}_p$-norm IDF punishes visual word burstiness, while retaining the discriminative structures.

# 5. Experiments

We evaluate the performance of the proposed $\mathcal{L}_p$-norm IDF method on large scale image search task. Experiments are conducted on two datasets populated with 1M distractor images. In this section, the experimental results are summarized and analysed.

## 5.1. Baseline

We adopt the method in [11] as the baseline approach.

During preprocessing, we extract Hessian-affine regions from which the 128-D SIFT descriptors are computed. In our implementation, we only allow a one-to-one mapping between SIFT descriptors and Hessian affine regions. This modification reduces the false matches brought by multiple SIFTs per location, producing a higher baseline result.

The codebook is constructed by Approximate K-means (AKM) method, using SIFT features from Oxford 5K dataset. Quantification applies the approximate nearest neighbors (ANN) indexing structure. In the searching step, scores for each image are calculated using conventional TF-IDF in Eq. 2. Mean Average Precision (mAP) is used to measure search performance.

## 5.2. Datasets

To evaluate the effectiveness of the $\mathcal{L}_p$-norm IDF, we conducted experiments on three publicly available datasets: Oxford 5K [11], Paris 6K [9], and Flickr 1M [4].

**Oxford 5K and Paris 6K datasets**. Oxford 5K dataset was collected from Flickr and a total number of 5062 images have been obtained. This dataset has been generated as a comprehensive ground truth for 11 distinct landmarks, each containing 5 queries. In total there are 55 query images. Paris dataset was generated in couple with Oxford 5K. This dataset contains 6385 high resolution images from Flickr by queries of Paris landmarks. Again, Paris dataset is featured by 55 queries of 11 different landmarks.

**Flickr 1M dataset**. The Flickr 1M dataset are distractor images arbitrarily retrieved from Flickr. These images are added into the Oxford 5K and Paris 6K datasets to test the scalability of our approach.

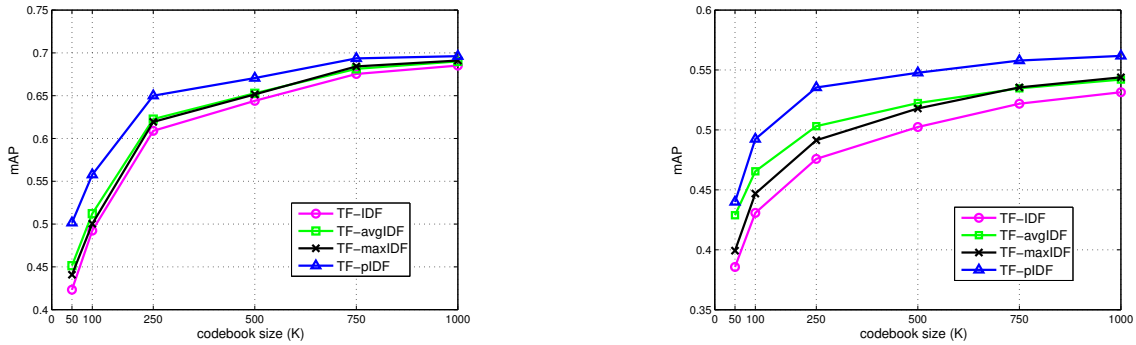
## 5.3. Parameter Analysis

For $\mathcal{L}_p$-norm IDF defined in Eq. 5, the parameter $p$ determines the extent to which burstiness is punished, so its value should be tuned carefully.

First, we search for $p$ that minimizes the cost function in Eq. 8 on Flickr 1M dataset. The result is illustrated in Fig. 5(b). Then, different values of $p$ are implemented on Oxford 5K + Flickr 1M, and Paris 6K + Flickr 1M datasets. The mAP results are presented in Fig. 5(a).

The cost function, *i.e*. Eq. 8 is minimal when $p$ is about 3.5. Meanwhile, from Fig. 5(a), the profile of the two curves are quite similar, and the mAP value is stable when $p$ takes the value of 3-4. Note that the results in Fig. 5(a) and 5(b) are to some extent consistent. Therefore, we set $p$ to 3.5 in the following experiments. It is evident from Eq. 5 that larger value of $p$ indicates amplified punishment on visual word burstiness. An optimal value helps produce satisfying performance.

Table 1 demonstrates the results of using different normalization strategies in Eq. 2. $L_1$ normalization measures the rate of descriptor matches. For a small query region in a large image, the $L_1$ normalization will probably fail. Therefore, $L_1$ normalization produces low baseline result. On the other hand, no normalization means to count the number of descriptor matches, where large images tend to produce more matches. $L_2$ normalization takes a compromise between the above two methods, and produces the highest




(a) Oxford 5K (b) Paris 6K

Figure 6. Image search performance as a function of the codebook size for different weighting schemes. Mean Average Precision (mAP) for (a) Oxford 5K and (b) Paris 6K datasets are presented.

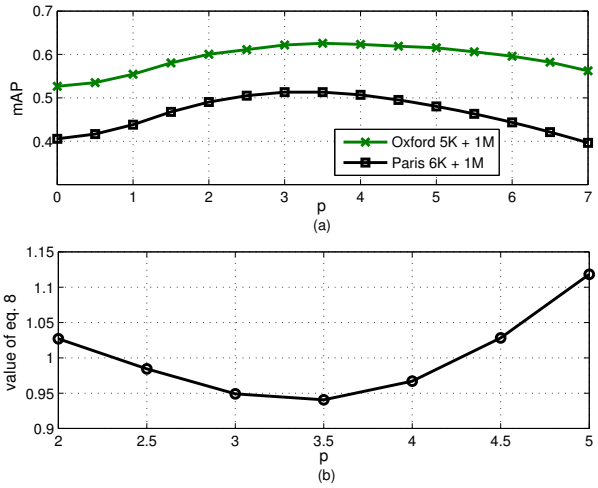



Figure 5. Selection of Parameter $p$. (a): On Oxford 5K + 1M and Paris 6K + 1M datasets, mAP is computed and plotted against different values of parameter $p$ defined in Eq. 5. (b): The value of cost function Eq. 8 when $p$ varies.

Table 1. mAP of Different Normalization Methods

| Normalization | Oxford 5K + Flickr 1M | |
|---|---|---|
| | TF-IDF | TF-pIDF |
| No | 0.432 | 0.629 |
| $L_1$ | 0.192 | 0.208 |
| $L_2$ | 0.523 | 0.626 |

baseline result. Therefore, in Table 1, although the $\mathcal{L}_p$-norm IDF achieves 45.6% improvement (from 0.432 to 0.629) in the case of no normalization, we choose the $L_2$ normalization in the following experiments.

## 5.4. Evaluation

**Comparison of Four IDFs:** We discussed four IDFs in Section 3, *i.e*. the conventional IDF, average IDF (avgIDF), max IDF (maxIDF), and the $\mathcal{L}_p$-norm IDF (pIDF). The last three are defined for the first time by our pooling method. Fig. 6 and Table 3 compares the search accuracy of the four IDFs. Results on the three benchmark datasets are reported, which leads to three major observations.

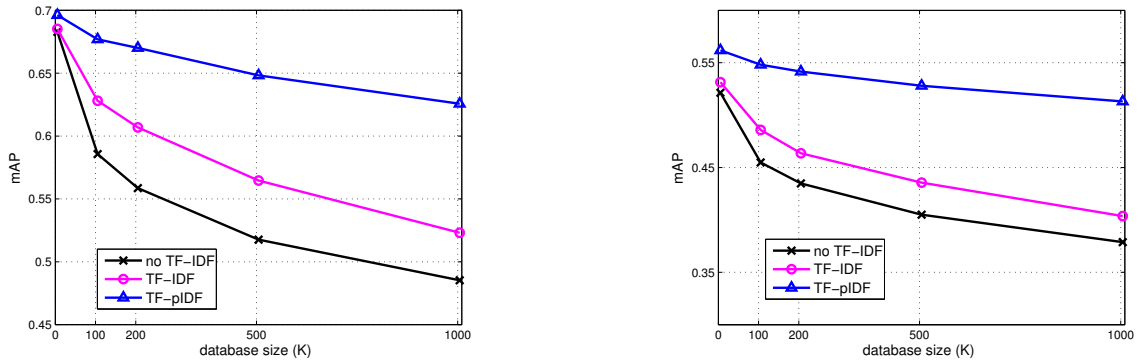
First, from Table 3, max IDF is inferior to the baseline in large scale experiment. On the two datasets, a drop of 19.1% and 26.2% is observed, respectively. Max IDF takes the maximum value of a word's TF as the estimation of its frequency. So it neglects the document frequency, while conventional IDF neglects TF. On the 1M dataset where the number of images is large, this limitation is amplified. Consequently, on small datasets, max IDF slightly outperforms the baseline, but on large dataset, situation is reversed.

Second, Fig. 6 and Table 3 shows that average IDF is shown to be slightly superior to both the conventional IDF and the max IDF. Average IDF improves accuracy by 3.3% and 5.5% on the two 1M datasets. In its nature, average IDF explicitly considers both the TF of visual words in each image and the document frequency. Therefore, on both small and large datasets, average IDF gives better performance.

Finally, it is evident in Fig. 6 and Table 3 that our proposed $\mathcal{L}_p$-norm IDF method consistently outperforms the other three IDFs. The $\mathcal{L}_p$-norm IDF estimates the word frequency using both term frequency and document frequency of each visual word. By carefully weighting the contribution of every database image, and optimizing the parameter $p$, $\mathcal{L}_p$-norm IDF gives better weights to visual words, thus elaborately making significant improvement over the baseline approach.

**Scalability:** To evaluate the scalability of the proposed method, we populated the Oxford 5K and Paris 6K datasets with various fractions of the Flickr 1M dataset. Experimental results are demonstrated in Fig. 7. Again, three major conclusions can be drawn.

First, the introduction of conventional TF-IDF helps to improve performance over the "no TF-IDF" case, but the




(a) Oxford 5K + Flickr 1M

(b) Paris 6K + Flickr 1M

Figure 7. mAP for (a) Oxford 5K and (b) Paris 6K datasets scaled with Flickr 1M dataset as distractor images. Three methods are employed, *i.e*. method without TF-IDF weighting, the TF-IDF baseline and the proposed TF-pIDF, respectively.

Table 2. Efficiency Comparison

| Methods | Average Search Time [1] (s) | | | |
|---|---|---|---|---|
| | Oxford 5K | Oxford 5K + 1M | Paris 6K | Paris 6K + 1M |
| TF-IDF | 0.026 | 0.677 | 0.041 | 0.856 |
| TF-pIDF | 0.026 | 0.678 | 0.041 | 0.856 |
| BM25 [12] | 0.052 | 0.829 | 0.070 | 1.047 |
| Jégou *et al*. [5] | 0.043 | 0.762 | 0.064 | 0.996 |

improvement is not so significant.

Second, as the database gets scaled up, mAP of the proposed methods drops more slowly. That is to say, more significant improvement is obtained on larger database. Notably, the $\mathcal{L}_p$-norm IDF outperforms the TF-IDF baseline by 19.61% and 27.1% on two 1M datasets, respectively. These results validate the scalability of the proposed method.

Third, we note that although the codebook is trained on Oxford dataset, more notable improvement is observed on Paris dataset. The codebook may be quite discriminative for Oxford dataset, but much more ambiguous [16] for Paris. Therefore, the burstiness problem is more severe on Paris dataset. Our proposed method helps to down-weight visual word in bursts and alleviate the burstiness problem, so more improvement is brought on Paris dataset. It indicates that $\mathcal{L}_p$-norm IDF based approach generalizes well to the case where the codebook is trained on irrelevant data and the improvement is much more considerable.

**Efficiency:** We run our experiment using Matlab 2010b on a 2.40-GHz CPU of a Sixteen-Core Intel Xeon server with 32 GB memory. Table 2 compares the average search time [1] of different approaches. Since the $\mathcal{L}_p$-norm IDF proposed in this paper is an offline approach, this method only marginally increases the offline training time but shed no influences on the online image search. Therefore, the $\mathcal{L}_p$-norm IDF share the same time and memory efficiency with the baseline approach. The average search time is $0.678s$ and $0.856s$ for Oxford 5K + 1M and Paris 6K + 1M datasets, respectively. Okapi-BM25 weighting is the least efficient one, because this method computes the TF-IDF weights online, resulting in more efficiency loss. As a result, compared with the baseline, the BM25 weight, and [5], our method better meets user's expectation of fast response time while enjoying much higher search accuracy.

**Comparison with other methods:** We compare the proposed $\mathcal{L}_p$-norm IDF with [5] and the Okapi-BM25 weighting [12], as in Table 3.

Okapi-BM25 mainly deals with the TF part. We select the parameter that produces the best performance of BM25. From Table 3, our method has a slightly lower mAP on small datasets, probably because the $\mathcal{L}_p$-norm IDF is optimized on the 1M dataset. However, our method clearly outperforms the Okapi-BM25 weighting on both large datasets. Notably, on Oxford 5K + 1M dataset, the BM25 weighting obtains mAP of 0.568, while our method is 0.626.

We also compare our method with [5]. We couple both the inter- and intra-image burstiness solutions in our experiment. We select the best-performance formula in [5]. From Table 3, it is evident that the proposed $\mathcal{L}_p$-norm IDF is shown to be superior on the large datasets. Therefore, the above results validate the feasibility of $\mathcal{L}_p$-norm IDF to the large scale real-world applications.

[1] Average search time does not include feature extraction and quantization. Quantization takes 0.92s on average.

Table 3. Comparison of mAP for Various Methods on Benchmark Datasets

| Dataset | TF-IDF | TF-avgIDF | TF-maxIDF | Okapi-BM25 [12] | Jégou *et al*. [5] | TF-pIDF |
|---|---|---|---|---|---|---|
| Oxford 5K | 0.685 | 0.690 | 0.691 | **0.704** | 0.695 | 0.696 |
| Oxford 5K + 1M | 0.523 | 0.540 | 0.423 | 0.568 | 0.558 | **0.626** |
| Paris 6K | 0.531 | 0.542 | 0.544 | **0.571** | 0.565 | 0.562 |
| Paris 6K + 1M | 0.404 | 0.426 | 0.298 | 0.494 | 0.491 | **0.513** |

## 6. Conclusion

This paper proposed an effective IDF weighting scheme, *i.e*. the $\mathcal{L}_p$-norm IDF. By $\mathcal{L}_p$-norm pooling, we integrate term frequency, document frequency, document length as well as the codebook information, into the final IDF representation. The $\mathcal{L}_p$-norm IDF functions on the visual word level, and can deal with the burstiness problem by down-weighting visual words in bursts. The parameter $p$ is optimized by minimizing a cost function. Extensive experiments on several benchmark datasets show that our method achieves significant improvement. Furthermore, the $\mathcal{L}_p$-norm IDF outperforms several state-of-the-art weighting approaches, and more improvement can be observed when the database size gets larger. Finally, the $\mathcal{L}_p$-norm IDF is an offline approach, so it remains the same memory usage and computation efficiency as the baseline model.

In the future, more investigation will be focused on the empirical studies of visual word frequency distribution and its discriminative power. This study re-issues the importance of visual word weighting, and various weighting strategies will be studied.

**Acknowledgements** This work was supported by the National High Technology Research and Development Program of China (863 program) under Grant No. 2012AA011004 and the National Natural Science Foundation of China under Grant No. 61071135. This work was also supported in part by ARO grant W911BF-12-1-0057, NSF IIS 1052851, Faculty Research Awards by Google, FXPAL, and NEC Laboratories of America, 2012 UTSA START-R Research Award, and NSFC 61128007 respectively.

## References

[1] X. Blix, G. Roig, and L. V. Gool. Nested sparse quantization for efficient feature coding. In *ECCV*, 2012.

[2] Y. Cai, W. Tong, L. Yang, and A. Hauptmann. Constrained keypoint quantization: towards better bag-of-words model for large-scale multimedia retrieval. In *ICMR*, 2012.

[3] J. Feng, B. Ni, Q. Tian, and S. Yan. Geometric $\mathcal{L}_p$-norm feature pooling for image classification. In *CVPR*, 2011.

[4] H. Jégou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *ECCV*, 2008.

[5] H. Jégou, M. Douze, and C. Schmid. On the burstiness of visual elements. In *CVPR*, 2009.

[6] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: spatial pyramid matching for recognizing natual scene categories. In *CVPR*, 2006.

[7] D. G. Lowe. Distinctive image features from scale invariant keypoints. *IJCV*, 2004.

[8] D. Niester and H. Stewenius. Scalable recognition with a vocabulary tree. In *CVPR*, 2006.

[9] M. Perd'och, O. Chum, and J. Matas. Efficient representation of local geometry for large scale object retrieval. In *CVPR*, 2009.

[10] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Lost in quantization: Improving particular object retrieval in large scale image databases. In *CVPR*, 2008.

[11] J. Philbin, O. Chum, M. Isard, and A. Zisserman. Object retrieval with large vocabularies and fast sptial matching. In *CVPR*, 2007.

[12] S. Robertson and S. Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. *SIGIR of Documentation*, 1994.

[13] K. Simonyan, A. Vedaldi, and A. Zisserman. Descriptor learning using convex optimisation. In *ECCV*, 2012.

[14] A. Singhal. Modern information retrieval: A brief overview. *IEEE Data Eng. Bull.*, 2001.

[15] J. Sivic and A. Zisserman. Video google: a text retrieval approach to object matching in videos. In *ICCV*, 2003.

[16] J. C. van Gemert, C. J. Veenman, and W. M. Smeulders. Visual word ambiguity. *PAMI*, 2010.

[17] X. Wang, M. Yang, T. Cour, S. Zhu, K. Yu, and T. X. Han. Contextual weighting for vocabulary tree based image retrieval. In *ICCV*, 2011.

[18] Y. Yang, F. Nie, D. Xu, J. Luo, Y. Zhuang, and Y. Pan. A multimedia retrieval framework based on semi-supervised ranking and relevance feedback. *PAMI*, 2012.

[19] S. Zhang, Q. Huang, G. Hua, S. Jiang, W. Gao, and Q. Tian. Building contextual visual vocabulary for large-scale image applications. In *ACM MM*, 2010.

[20] S. Zhang, Q. Tian, G. Hua, Q. Huang, and S. Li. Descriptive visual words and visual phrases for image applications. In *ACM MM*, 2009.

[21] X. Zhang, Z. Li, L. Zhang, W. Ma, and H.-Y. Shum. Efficient indexing for large scale visual search. In *ICCV*, 2009.

[22] L. Zheng and S. Wang. Visual phraselet: Refining spatial constraints for large scale image search. *Signal Processing Letters, IEEE*, 20(4):391–394, 2013.

[23] W. Zhou, Y. Lu, H. Li, Y. Song, and Q. Tian. Spatial coding for large scale partial-duplicate web image search. In *ACM MM*, 2010.

[24] W. Zhou, Y. Lu, H. Li, and Q. Tian. Scalar quantization for large scale image search. In *ACM MM*, 2012.