Journal of Multimedia

ISSN 1796-2048 Volume 10, Number 1, January 2015

Contents

Contents	
REGULAR PAPERS	
From Text to Images: Weighting Schemes for Image Retrieval Pierre Tirilly, Vincent Claveau, and Patrick Gros	1
The Study of PID AQM Algorithm for FPGA Implementation Xiaoping Yang, Xiaojing Wang, Te Di, and Chao Niu	22
Chaos Character of Test Data Processing in Complex Wireless Sensor Network Changjian Deng	28

From Text to Images: Weighting Schemes for Image Retrieval

Pierre Tirilly^a, Vincent Claveau^b, Patrick Gros^c

^a LIFL / Université Lille 1, Cité Scientifique, 59655 Villeneuve d'Ascq, France
Email: pierre.tirilly@lifl.fr

^b CNRS / IRISA, Campus de Beaulieu, 35042 Rennes, France
Email: vincent.claveau@irisa.fr

^c INRIA Rennes-Bretagne Atlantique, Campus de Beaulieu, 35042 Rennes, France
Email: patrick.gros@inria.fr

Abstract—Bags of visual words are the most studied image description technique in the last years. This representation of images raised new possibilities as well as new research issues. In particular, it is important to automatically determine which visual words are the most relevant to describe the images, and which ones should be ignored. This issue is a classical problem of textual information retrieval, usually addressed by the use of weighting schemes. In this paper, the most common weighting schemes from text retrieval are applied to the case of visual word-based retrieval. New weighting schemes are also proposed, and several Minkowski-like distances are tested. The experiments are performed on four different datasets that correspond to two different retrieval tasks; it allows us to bring to light some properties of visual words and weighting schemes. This study results in several findings. It first shows that the optimal setting for distances and weighting schemes depends on the nature of the visual content of the images considered. Especially, raw frequency can be the most effective weight when dealing with complex datasets; it questions the habit to systematically use the tf.idf weighting scheme. It also shows that weighting schemes and Minkowski distances have similar effect and should be used together in a consistent way. Based on these findings, general guidelines for the choice of distances and weighting schemes are proposed.

Index Terms—Image retrieval, visual words, weighting schemes, divergence from randomness, vector space model, Minkowski distances, fractional distances.

I. INTRODUCTION

MAGE retrieval has been an active research field for many years now, and the recent and dramatic growth of image collections, due to the popularization of image acquisition devices such as camera cellphones, raises the need for effective retrieval systems able to handle large quantities of data. In the last years, the use of quantized local descriptors as an image description has received a huge interest, and most of modern image retrieval systems rely on this technique, referred to as *bag of visual words* or *bag of features*. Such systems have shown excellent capabilities in various tasks such as particular object or

Manuscript received August 23, 2013; revised March 19, 2014; revised September 29, 2014; accepted September 30, 2014.

scene retrieval [2], [3], image categorization [4] and image annotation [5], but also raised new research issues. One of these is: how to automatically determine which visual words best describe the images? This problem has been widely studied in the field of text retrieval and, since bag of visual words are inspired from the bag of words used to describe textual documents, many solutions exist that can be adapted to images. One of these is the use of weighting schemes: they allow to control the importance of words in the description of documents, depending on how representative of document contents they are. The most popular of these weighting schemes, called tf.idf, has been part of visual word-based systems since the seminal work of [2]; however there is no comprehensive study about the relative performance of the available weighting schemes. This is the issue addressed in this paper: the statistical properties of visual words are studied from a text retrieval perspective, and many of the weighting schemes available in the text retrieval literature, as well as some new ones, are applied in the context of image retrieval, providing experimental results for 84 different weighting configurations. Different Minkowski metrics are used to compare document vectors, showing the relationship between weights and distances. This study is performed on four datasets covering two distinct retrieval tasks to provide a global overview of the problem.

This paper is an extended version of [1]. It provides a more complete background analysis with empirical evidence of the claims that fund this work (Section III). Additional experimental results are provided (Section VII), together with more detailed analyses (Sections VIII-A, VIII-B and VIII-C) and discussions (Sections IX-C and IX-D) of the overall results. Finally, it provides some new guidelines for the choice of appropriate weighting schemes (Section X).

First, the bag of visual words approach is presented (Section II). Some elements about the relation between visual word-based image retrieval and text retrieval are provided, and some fundamental differences that must be taken into account when working with weighting schemes are detailed (Section III). Then, the datasets used in this study (Section IV) are presented, followed by the

distances (Section V) and the weighting schemes tested in the experiments (Section VI). Some new, image-specific, weighting schemes are also proposed (Section VI-C). Finally, the experimental settings and the results of the experiments are detailed (Section VII), analyzed (Section VIII) and discussed (Section IX). Based on this discussion, Section X presents some general guidelines to choose appropriate weighting schemes. Related work from the literature is reviewed in Section XI.

II. THE BAG OF VISUAL WORDS MODEL

A. Definition

The visual word-based representation of image relies on a finite number of image elements, named visual words, that are combined to obtain a global representation of images. The construction of the bag of visual words of an image includes two steps (see Fig.1):

- the construction of a visual vocabulary, i.e. of the set of possible visual words describing an image. This vocabulary is unique for the whole image dataset, so images are all described using the same elements. The vocabulary is therefore computed once, although some authors combine several vocabularies to represent images [6].
- the construction of the bag of visual words representing an image, by counting the number of occurrences of each visual word in the image to construct a vector of visual word counts.

Construction of the visual vocabulary: First, local descriptors (e.g. SIFT descriptors [7]) are extracted from a set of images and grouped using a clustering algorithm (e.g. k-means). Each of the resulting clusters corresponds to one visual word. The number of clusters, i.e. the vocabulary size, is set manually since most clustering algorithms cannot determine an optimal cluster number. There is no rule to pick the vocabulary size, although some claim that the larger the vocabulary, the better [8]. Typically, the vocabulary size stands between a few thousands and one million words. Although it is computed only once, this step raises some efficiency issues because of the large amount of input descriptors required, their high dimensionality and the high number of clusters used.

Construction of the visual word vector: Given an image and a vocabulary, a visual word vector is built to describe this image. Local descriptors are extracted from the image and matched with the centroids of the vocabulary to find which clusters they fall in. Each descriptor counts for one occurrence of the corresponding visual word, building an histogram of visual word occurrences. Generally, the number of image descriptors is smaller than the vocabulary size, resulting in a sparse vector.

B. Advantages of the Bag of Visual Words Model

1) Relation to text retrieval: The visual word vectors are similar to the vectors used in the vector space model (VSM) of information retrieval: sparse vectors of element occurrences. Many techniques from text retrieval can

therefore be applied to images, e.g. pLSA [9], inverted index [2] or query expansion [10].

- 2) Efficiency and effectiveness: Visual word vectors, especially when coupled to index structures such as an inverted index, can provide search results quickly. They are more efficient than previous approaches relying on local descriptors (e.g. [11]) while remaining effective, thanks to the expressiveness of local descriptors and histogram sparsity.
- 3) Generalization capabilities: It is often assumed that quantizing local descriptors allows to group visually similar features that would not have been matched by regular matching algorithms, making them more robust to some appearance variations.

C. Major Issues

- 1) Independence of the visual words: Visual word occurrences are usually considered independent, which is not always realistic (see Section III-B). Some authors have proposed to group occurrences to overcome this limitation (e.g. [12], [13]).
- 2) Relative importance of the visual words: While some visual words are very descriptive of the image content, some of them are of poor interest (e.g. visual words that occur in the background). This is the main issue addressed in this paper. More details are available throughout the paper, including a literature review (Section XI).
- 3) Construction of an optimal vocabulary: The vocabulary construction is an essential step, but complex clustering methods cannot be used due efficiency limitations. Several authors propose methods to get better vocabularies [14] or quicken the clustering step [15]. However, there is no consensus on how to build an optimal vocabulary.
- 4) Loss of precision: The quantization process allows to get faster systems, but this approximate descriptor matching can be prejudicial to some applications. This has been studied in the literature (e.g. [3], [16]). It can seem opposed to the supposed generalization capabilities of this approach, but it applies to different tasks (near duplicate retrieval vs. image categorization).

III. On the Relation Between Visual Words and Textual Words

As stated in Section II-B, the bag of visual words model is very close to the standard VSM of text retrieval. This similarity allows us to apply text retrieval techniques to images, as we do in this paper. To do so, some differences existing between textual and visual words must be stated in order to better understand the results obtained. This is especially true when working on weighting schemes, which rely on the distribution of visual words within and over documents.

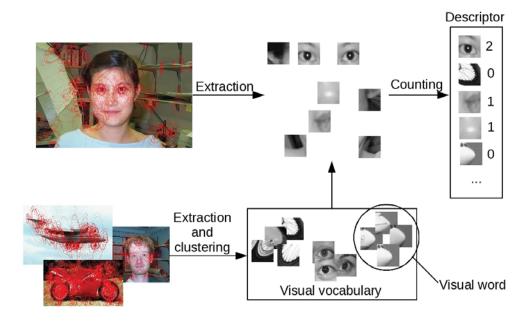


Figure 1. Construction of the visual word histogram of an image.

A. Nature of the Vocabulary

The first notable difference is the way the vocabulary is obtained. In the case of textual corpora, the vocabulary is provided implicitly by the data: it contains all the words occurring in the dataset, and can be further optimized using natural language processing tools (e.g. stemmers). In contrast, a visual vocabulary is built explicitly during a process involving many parameters; it is possible to build many different vocabularies from a single dataset.

B. Meaning of the Visual Words

Textual and visual words have different meanings. On the one hand, a textual word conveys its own meaning (for instance house, computer...), making text retrieval systems able to work under the word independence assumption, which states that words can be considered as occurring independently from each other. Although this is not always true (e.g. White House has meaning that differs from White and House taken separately), most systems still yield good results under this assumption. On the other hand, visual words are not meaningful on their own, and rather make sense when considered as a whole, as shown in Fig.1 where the face can only be identified by considering its subparts. This insight can be confirmed from simple statistics on visual word distributions. Considering a small dataset containing 5,415 images divided into 6 object categories, the number of categories in which each visual word occurs is calculated. The results are reported in Table I(a). Clearly, most of the visual words of the 6,556-word-vocabulary can occur in any category. Hence, they do not correspond to specific parts of a single object. The same statistics on a text dataset with comparable size (Reuters R8 dataset [17], 8 categories, 7,674 documents) reveal that most words occur in only one category (Table I(b)). This shows that visual

vocabularies contain much more *polysemous* words (i.e. words with several meanings) than textual vocabularies.

TABLE I.

DISTRIBUTION OF WORDS WITH RESPECT TO THE NUMBER OF
CATEGORIES THEY OCCUR IN (A) IN AN IMAGE DATASET
CONTAINING SIX IMAGE CATEGORIES (B) IN A TEXT DATASET
CONTAINING EIGHT CATEGORIES.

Number of categories	Number of		
where the visual	al visual words		
word occurs			
1	4		
2	3		
3	11		
4	15		
5	81		
6	6442		
(a)			

Number of categories	Number of	
where the textual textual word		
word occurs		
1	14043	
2	4130	
3	1629	
4	1050	
5	782	
6	653	
7	551	
8	747	
(b)		

C. Length of the Documents

In text retrieval, the length of a document corresponds to the number of word occurrences in it. Generally, the more words a documents contains, the more information it contains (either more topics or more precisions about a given topic). Moreover, the longer the document, the the larger its vocabulary, and the more each word occurs.

These facts motivated much work about document normalization. In images, the length of a document obeys to different laws, which are more difficult to define formally. At least three parameters can affect it:

- the size (or resolution) of the image: the larger the image (or the higher its resolution), the higher the number of regions provided by interest point detectors. As an illustration, Fig.2 shows the number of regions detected using a Hessian-affine detector on an image scaled at 3 different sizes;
- the visual content of the image: since interest point detectors react to the heterogeneity (or homogeneity) of the image to determine where the most interesting regions are, the nature of the image has a strong influence on the number of regions detected. Fig.3 illustrates this, by indicating the number of regions detected by a Hessian-affine detector on four images with different visual contents. On the first one, very few regions are detected, due to the homogeneity of the content. On the second one, few regions are detected because of the obscurity. On the last two, many regions are detected, due to the complexity of the objects and/or backgrounds;
- the detector parameters: the detector itself influences
 the number of regions detected, since it is generally
 possible to tune its sensitivity; however, it does not
 constitute a major issue, as the same detector and
 parameters are used for every image of the dataset.

D. Visual Word Frequencies Within a Document

In text documents, word frequencies (i.e. number of occurrences) are usually interpreted as follows: the more frequent a word, the more specific to the document it is. Frequency reflects the importance of that word in this document. As for document length, visual word frequency can be due to several factors.

- The intra-document frequency of a word will be higher if the object containing this visual word occupies a bigger proportion of the image. Also, if the object occurs several times, visual word frequencies will also increase. This is somewhat similar to text documents, but two remarks should be made. First, a bigger object will show more details and tend to contain visual words that would have not appeared at a smaller scale. Also, the fact that an object takes up a large area on the image, or appears more than once, does not necessarily mean that this object is important to describe the picture, as shown on the third picture of Fig.3 where repeated objects belong to the background, which is often of little interest.
- Intra-document frequency is also related to the properties of interest point detectors. It should especially be noted that many detectors tend to raise the same region several times, at the same or multiple scales. Fig.4 illustrates this, where the same image region is detected several times, increasing the frequency of the corresponding visual word.



Figure 4. Example of a region detected several times on a motorbike image.

- A visual word can appear in several objects of an image, making it difficult to handle correctly the frequencies due to each object. Occlusions also modify frequencies.
- Intra-document frequencies of visual words depend on the size of the vocabulary: the larger the vocabulary, the lower the average frequencies of the visual words; however, this property becomes negligible once the vocabulary size is set, as it applies to the whole collection.

E. Visual Word Frequencies Within a Collection

The frequencies of words within text collections are typically interpreted as follows: the more documents a word occurs in, the less specific it is to these documents. The document frequency of a word (i.e. the number of documents in which it occurs) is a direct measure of its specificity [18]. This assumption, which motivates the well-known idf weighting scheme, has been shown to be theoretically valid [19]. In the case of visual words, the same approach is adopted, based on this assumption: visual words that occur in many documents mostly represent the background of the images, so they do not describe objects. It has shown its effectiveness when retrieving specific objects in videos [2], where scene backgrounds tend to appear more frequently over the movie than specific objects. Although the idf weight is regularly used in the literature, no proof exists that its underlying assumption holds in the general case. As mentioned in Section III-B, visual words taken separately are usually not specific to a single object. It is possible that a visual word occurs both in objects and backgrounds, making its statistics over the collection difficult to interpret.

F. Queries

Image queries are different from text queries. Indeed, text queries are usually short (typically, tens of words in TREC¹ tracks) or very short (about two words on average for web queries). By contrast, image queries are whole images, or cropped images, which contain hundreds to thousands of words, which implies that:

• inverted files are much less efficient, as much more documents will be retrieved for a given query. Some

¹Text REtrieval Conference: http://trec.nist.gov.

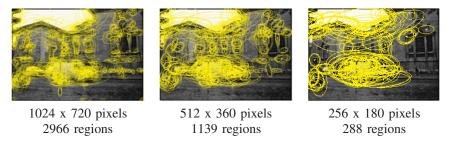


Figure 2. Number of interest regions detected according to the size of the image (Hessian-affine detector).

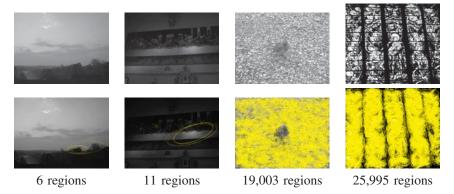


Figure 3. Detection of various numbers of interest regions according to the image content (Hessian-affine detector).

authors overcome this limitation by modifying the inverted file structure [20], while others use only small regions to query the system [2];

 the query has to be normalized like any document, which is rarely the case in text retrieval. In particular, probabilistic retrieval models do not include such normalization (see Section VI-B).

IV. DATASETS USED IN THIS STUDY

Datasets are a major issue in image retrieval, as particular datasets can include biases or are not as complex as in real-world retrieval scenarios. The content of image datasets often determines implicitly the nature of the retrieval task considered, depending of the properties picked as defining visual relevance. Among common image retrieval datasets, two tasks prevail:

- particular scene (or near-duplicate) retrieval: relevant images are images from a given scene or object, with variations in scale, orientation, illumination, occlusions, but not in the aspect (shape, color and texture) of the scenes or objects. This task is illustrated in Fig.5;
- category retrieval: relevant images represent objects from a given category (airplanes, watches...). In addition to the aforementioned transforms, changes in object shapes, colors or textures occur. This is illustrated in Fig.6. This task is more difficult to handle than the first one, considering the intra-class variations of the objects.

In this paper, both tasks are considered, and two datasets used for each.

A. Datasets for Particular Scene Retrieval

We selected two datasets that have been intensively used in the literature for this task:

- Nistér dataset [15]: it contains 10,200 images divided into groups of four relevant images. The results are usually evaluated using a specific evaluation score which is equivalent to a precision after 4 documents (including the query among the results). Since the ground truth is provided for all documents, any of them can serve as a query;
- Oxford dataset [10]: it contains, among others, images of Oxford buildings. It contains 5,063 images and provides 55 queries with their associated ground truth. The relevant documents associated to each query are classified into *good*, *OK* and *junk* categories, depending of the proportion of the building present on the image. In our experiments, all categories are considered as relevant whatever its relevance category to ensure the consistency of the evaluation protocol. For the same reason, full images are taken as queries, instead of the cropped queries provided in the ground truth.

B. Datasets for Object Category Retrieval

We selected two datasets that are generally devoted to image categorization rather than image retrieval, because they are usually used to evaluate visual word-based systems on this task (e.g. [12]).

• Caltech6: it is a subset of the classical Caltech datasets. It contains 5,415 images split into 6 categories: airplanes, backgrounds, faces, cars, motorbikes and guitars. Each class contains 450 to 1,155 images.

 Caltech101 [21]: it contains 8,196 images split into 101 classes. Each category contains 31 to more than 800 images. This dataset is the most difficult to deal with, as it presents strong intra-class variations and some visually similar categories.

C. On the Properties of the Datasets Used

Although these datasets are not real-world datasets, their properties are helpful to provide comprehensive conclusions about the evaluated systems. Especially, some of the issues mentioned in Section III can be avoided, which allows us to focus on the distributions of visual words with respect to the nature of the image content only. Moreover, these datasets are widely used in the literature, which makes the comparison with other studies possible.

- 1) Size of the datasets: The datasets are of reduced size, especially compared to some large-scale datasets recently used (up to 10 million images [22]). However, the whole ground truth is known on these datasets, making the evaluation more reliable. Moreover, their limits and properties allows them to highlight some facts about our systems that would have not been obvious in more diverse datasets.
- 2) Size of the images and objects: The images within each dataset are of similar size and generally present single objects at a similar scale. It limits the effects that image size, object scale...have on visual word statistics that were mentioned in Section III, so the analysis of the results can focus on visual word statistics related to scene or object appearance only.

V. DISTANCES

Minkowski distances are usually used to compute the similarity between visual word vectors. The general equation of a Minkowski distance is provided by (1). The parameter k of these distances allows one to define different distances with different properties. In particular, k does not have to be an integer value. Distances so that k < 1, called fractional distances (although they are actually semimetrics), have interesting properties in high dimensional spaces [23], and are effective in image retrieval tasks [24]. They have two remarkable properties:

- they reduce the impact of the curse of dimensionality;
- they can handle, to some extent, the presence of noise in the data; however, Aggarwal et al. [23] showed that when the data is too noisy, they tend to be equivalent, whatever the value of k.

In practice, L_1 and L_2 are the most used in visual word-based image retrieval, and especially L_1 since it was reported as providing better performance [15], although this was observed on a single dataset and not further investigated. To our knowledge, fractional distances have never been tested in this context.

$$d_{L_k}(x,y) = \left(\sum_{i=1}^n |x_i - y_i|^k\right)^{\frac{1}{k}}, \ \forall x, y \in \mathbb{R}^n \quad (1)$$

VI. WEIGHTING SCHEMES

Weighting schemes are very popular and powerful tools of text retrieval. Their goal is to consider not only the presence or absence of words in the documents, as done in former information retrieval models such as boolean models [25], but also their importance with respect to the content of the document. The basic idea is to assign higher weights to the words that best describe documents. Such weights provide finer descriptors, so that retrieval performance can be improved.

In this section, the weighting schemes that are tested in the experiments are presented. First, some weighting schemes traditionally used in the context of the vector space model (VSM) are introduced, then weighting schemes adapted from probabilistic models, along with a framework to adapt them to the VSM. These weighting schemes follow the traditional assumptions about the distribution of words in texts, mentioned in Section III. Finally, some new weighting schemes are proposed, based on some of the specific properties of visual words that were provided in Sections III-B and III-D. Table II presents the notations used throughout the paper, and Tables III and IV list the weighting schemes considered. In the following, the word term, common in the information retrieval literature, stands for any kind of word (textual or visual).

TABLE II.
NOTATIONS USED THROUGHOUT THE PAPER.

t_i	<i>i</i> -th term of the vocabulary
d_{j}	<i>j</i> -th document of the collection
w_{ij}	Weight of term t_i in document d_j
N	Total number of documents in the collection
df_i	Number of documents containing visual word t_i
tf_{ij}	Number of occurrences (or frequency) of t_i in doc-
-	ument d_j
$\overline{\mathrm{tf}_i}$	Average frequency of t_i in the documents in which
	it occurs
CF_i	Number of occurrences of t_i in the whole dataset
CF_*	Total number of visual word occurrences in the
	whole dataset
dl_j	Length (i.e. number of term occurrences) of docu-
	ment d_i
$\mathrm{dl}_{\mathrm{avg}}$	Average length of the dataset's documents

A. Weighting Schemes from the VSM

The weights w_{ij} used in the VSM are generally based on empirical observations rather than complex models, but this is generally sufficient to provide significant improvements of the retrieval performance. The three most common assumptions are:

- the more frequent a term in a document, the best it describes this document. It is based on Luhn's first observations about the statistical description of documents [26];
- the less documents a term appears in, the more specific it is to these documents. This observation is due to [18];
- the longer a document, the higher the frequencies of its terms. Hence the need for normalization techniques.

TABLE III. Local (Top of the Table) and Global (Bottom) Weights Considered in this Paper. t_i and d_j Denote a Visual Word and a Document, Respectively.

Identifier	Description	Equation			
$l_1(t_i,d_j)$	Frequency tf	tf_{ij}			
$l_2(t_i, d_j)$	Frequency logarithm	$ \begin{array}{ccc} 0 & \text{if } \text{tf}_{ij} = 0 \\ 1 + \log(\text{tf}_{ij}) & \text{otherwise} \end{array} $			
$l_3(t_i,d_j)$	Normalized augmented frequency	$a + (1 - a) \cdot \frac{\underset{t_{k} \in d_{j}}{\text{tf}_{ij}}}{\underset{t_{k} \in d_{j}}{\text{otherwise}}} \text{otherwise}$			
$l_4(t_i, d_j)$	Binary weight	$ \begin{array}{ccc} 1 & \text{if } \text{tf}_{ij} > 0 \\ 0 & \text{otherwise} \end{array} $			
$l_5(t_i,d_j)$	DFR-like normalization	$\operatorname{tf}_{ij} \cdot \frac{\operatorname{dl_{avg}}}{\operatorname{dl}_{j}}$			
$l_6(t_i, d_j)$	Squared tf	tf_{ij}^2			
$l_7(t_i, d_j)$	BM25 tf	$\frac{\operatorname{tf}_{ij}.(\vec{k}_1 + 1)}{K + \operatorname{tf}_{ij}}$			
$g_0(t_i)$	No weighting	1			
$g_1(t_i)$	Inverse document frequency (idf)	$\log\left(rac{N}{\mathrm{df}_i} ight)$			
$g_2(t_i)$	Probabilistic idf	$\log\left(\frac{(\hat{N}-d\hat{\mathbf{f}}_i)}{d\mathbf{f}_i}\right)$			
$g_3(t_i)$	Squared idf	$\left[\log\left(rac{N}{{ m d} f_i} ight) ight]^2$			
$g_4(t_i)$	Average tf.idf	$\overline{\operatorname{tf}_i} \cdot \log \left(\frac{N}{\operatorname{df}_i} \right) = \frac{\operatorname{CF}_i}{\operatorname{df}_i} \cdot \log \left(\frac{N}{\operatorname{df}_i} \right)$			
$g_5(t_i)$	Squared average tf.idf	$[\overline{\operatorname{tf}_i}\log(\frac{N}{df_i})]^2$			

TABLE IV. Weight Elements Derived from the Divergence from Randomness (DFR) Models (from Top to Bottom: Randomness Models, Information Gain Models, Normalization Models). t_i and d_j Denote a Visual Word and a Document, Respectively.

Identifier	Model	Equation
P	Bernoulli	$tf_{ij} \cdot \log_2\left(\frac{tf_{ij}}{\lambda}\right) + \left(\lambda + \frac{1}{12 \cdot tf_{ij}} - tf_{ij}\right) \cdot \log_2(e) $ $+0.5 \log_2(2\pi \cdot tf_{ij})$ $where \lambda = \frac{CF_i}{N}$ $CF_i \cdot D(\phi_{ij}, p) + 0.5 \log_2(2\pi \cdot tf_{ij}(1 - \phi_{ij}))$
D	Bernoulli	where $\phi_{ij} = \frac{\operatorname{tf}_{ij}}{\operatorname{CF}_i}$, $p = \frac{1}{N}$ and $D(\phi_{ij}, p) = \phi_{ij} \cdot \log_2\left(\frac{\phi_{ij}}{p}\right) + (1 - \phi_{ij}) \cdot \log_2\left(\frac{1 - \phi_{ij}}{1 - p}\right)$
G	Bose-Einstein	$-\log_2\left(\frac{1}{1+\lambda}\right) - \operatorname{tf}_{ij} \cdot \log_2\left(\frac{\lambda}{1+\lambda}\right)$ where $\lambda = \frac{\operatorname{CF}_i}{N}$ $-\log_2(N-1) - \log_2(e)$
Be	Bose-Einstein	$-\log_2(N-1) - \log_2(e)$ $+f(N + \operatorname{CF}_i - 1, N + \operatorname{CF}_i - \operatorname{tf}_{ij} - 2)$ $-f(\operatorname{CF}_i, \operatorname{CF}_i - \operatorname{tf}_{ij})$ where $f(n,m) = (m+0.5) \cdot \log_2\left(\frac{n}{m}\right) + (n-m) \cdot \log_2(n)$
In	Inverse document frequency	$tf_{ij} \cdot \log_2\left(\frac{N+1}{\mathrm{d}f_{ij}+0.5}\right)$
In_e	Inverse document frequency	$\operatorname{tf}_{ij} \cdot \log_2 \left(\frac{N+1}{n_e + 0.5} \right)$ where $n_e = N \cdot \left(1 - \left(\frac{N-1}{N} \right)^{\operatorname{CF}_i} \right)$
HG	Hypergeometric	$\begin{aligned} &\log((\mathrm{tf}_{ij})!) + \log((\mathrm{CF}_i - \mathrm{tf}_{ij})!) + \log((\mathrm{dl}_j - \mathrm{tf}_{ij})!) \\ &+ \log((\mathrm{CF}^* - \mathrm{CF}_i - \mathrm{dl}_j + \mathrm{tf}_{ij})!) + \log(\mathrm{CF}^*!) - \log(\mathrm{CF}_i!) \\ &- \log((\mathrm{CF}^* - \mathrm{CF}_i)!) - \log(\mathrm{dl}_j!) - \log((\mathrm{CF}^* - \mathrm{dl}_j)!) \end{aligned}$
L	Laplacian	$\frac{1}{\operatorname{tf}_{ij}+1} \\ \operatorname{CF}_i+1$
В	Bernoulli	$\overline{\mathrm{df}_i \cdot (\mathrm{tf}_{ij} + 1)}$
Н0		tf_{ij}
H1		$rac{ ext{tf}_{ij}}{ ext{tf}_{ij} \cdot rac{ ext{dl}_{ ext{avg}}}{ ext{dl}_{j}}}$
H2		$\operatorname{tf}_{ij} \cdot \log_2(1 + \frac{\operatorname{dlavg}}{\operatorname{dl}_j})$









Figure 5. Images of a particular scene (Nistér dataset).









Figure 6. Images from an object category (Caltech101 dataset).

These observations lead to the following formulation of weights:

$$w_{ij} = l_{ij}.g_i.n_j \tag{2}$$

where l_{ij} represents the local importance of t_i in document d_j (based on Luhn's observation), g_i represents a global weight reflecting the specificity of t_i in the collection (based on Spärck-Jones' observation) and n_j is a normalization factor depending on document d_j . The remaining of this section describes the most common alternatives used to compute these components. Their mathematical formulation are provided in Table III.

1) Local Weights:

- a) Frequency: This is the raw term frequency tf_{ij} , i.e. the number of occurrences of visual word t_i in document d_i .
- b) Frequency logarithm: It limits the highest word frequencies. Indeed, if a document contains a high number of occurrences of a given term, it will tend to be returned for any query containing this term, making the contribution of other terms negligible. This weight overcomes this issue [27].
- c) Normalized augmented frequency: It is another means to limit the effect of high-frequency terms. Its contains two parts: one corresponds to the presence of the term only, the other corresponds to its frequency. Tuning the value of a provides more importance to one part or the other [28]. In the experiments, a=0.5 is chosen, so that both parts have equal importance.
- d) Binary weighting: It takes into account the mere presence of the term, discarding any frequency information.
- *e)* Squared frequency: It emphasizes the importance of frequent visual terms.
 - 2) Global Weights:
- a) Inverse document frequency: Inverse document frequency (idf) is born from the observation that words

occurring in few documents are specific to the content of these documents, whereas words that occur in many documents are more general, therefore less descriptive. It is inversely proportional to the document frequency $\mathrm{d} f_i$ (number of documents where t_i occurs). This weight is widely used in the visual word literature [2].

- b) Squared idf: Squared idf increases the weight of rare (i.e. specific) words, which can improve the precision of retrieval systems [29].
- *3) Normalization Factor:* The role of the normalization factor is twofold:
 - it provides weights that are relative to the document length. This effect is also provided by some local weights, e.g. normalized augmented frequency;
 - it ensures that all document vectors have the same norm, so that query-document distances can be compared consistently. The norm used depends on the distance. In the case of Minkowski distances, the normalization factor associated to distance L_k is:

$$n_j = \frac{1}{||d_j||_k} = \frac{1}{\left(\sum_{i=1}^m w_{ij}^k\right)^{\frac{1}{k}}} \tag{3}$$

where $d_j = (w_{1j}, w_{2j}, \dots, w_{mj})$ is the vector of term weights.

B. Weighting Schemes from Probabilistic Models

Introduced by Robertson, probabilistic models rely on his probability ranking principle (PRP) [30]: If retrieved documents are ordered by decreasing probability of relevance on the data available, then the system's effectiveness is the best to be gotten for the data. This intuitive principle is the basis for any probabilistic retrieval system, no matter what kind of information (textual, visual...) it deals with; however, it does not tell how to compute relevance probabilities. Several models of relevance have

been proposed and validated on text data. To our knowledge, none of them has been applied to visual words. This paper focuses on two popular probabilistic models. Before presenting them, we explain how they can be fit into the VSM, making them more flexible.

1) From Probabilistic Models to the VSM: The probabilistic models presented here can be seen as specific cases of the VSM. Indeed, the relevance probability of a document to a query depends only on the match of their terms under the independence assumption, raising this general probabilistic matching (PM) measure:

$$PM(d_j, q) = \sum_{t_i \in q} q_i . w_{ij}$$
(4)

where $q_i=1$ if term t_i is present in query q, and $q_i=0$ otherwise. This is equivalent to computing a scalar product between two vectors, with the following properties:

- the query vector contains only some information of term presence and absence. This is suited to text retrieval since the queries are usually short and do not contain repeated words. However, as stated in Section III-F, it does not apply to image retrieval, where queries are longer and terms can be repeated. Moreover, the query vector is not normalized, which is not suited to image queries neither;
- the document vector contains the probabilities of occurrence of terms in the document, which can be interpreted as a term weight in the VSM.

This scalar product is itself the main component of the most popular similarity measure of text retrieval, the cosine similarity², which is basically a normalized scalar product. It must be noted here that the probabilistic term weights are implicitly normalized, since probabilities always sum to 1. By considering the term probabilities as weights, these probabilistic models can be used within the VSM in a more flexible way, as:

- query terms can be weighted in the same way as document terms;
- any distance can be used, provided the vectors are properly re-normalized (see Section VI-A.3).
- 2) Weighting Schemes from Best Match Models: The most popular probabilistic model is the best match (BM) model proposed by Robertson, and especially the BM25 version, that yields excellent results during TREC evaluations [31] and is considered as a reference system. The complete BM25 formula includes some relevance information, which can be obtained, for instance, by (pseudo-)relevance feedback; however, as (pseudo-)relevance feedback is beyond the scope of this paper, only a simplified version of BM25, where relevance information is dis-

carded, is considered:

BM25
$$(d_j, q) = \sum_{t_i \in q} \left[\frac{\operatorname{tf}_{ij} * (k_1 + 1)}{K + \operatorname{tf}_{ij}} \cdot \operatorname{tf}_{iq} \cdot \log \left(\frac{(N - \operatorname{df}_i)}{\operatorname{df}_i} \right) \right]$$
(5)

where $K = k_1 \left((1-b) + b \cdot \frac{\mathrm{dl}_j}{\mathrm{dl}_{\mathrm{avg}}} \right)$ and k_1 and b are constant values. In the experiments, the following default values are used: $k_1 = 1.2$ and b = 0.75. From this formulation, the following weighting scheme can be extracted:

$$w_{ij} = \frac{\operatorname{tf}_{ij} \cdot (k_1 + 1)}{K + \operatorname{tf}_{ij}} \cdot \log \left(\frac{(N - \operatorname{df}_i)}{\operatorname{df}_i} \right) \tag{6}$$

It is the combination of a local weight and a global weight (detailed in Table III). The local weight is derived from the 2-Poisson model proposed by [32] to represent the distribution of terms within a document [33]. The second part, derived from the PRP, is known to provide optimal global weights when relevance data is provided [19]. It is close to the idf weight, and sometimes named *probabilistic idf*.

- 3) Weighting Schemes from Divergence from Randomness Models: Introduced by Amati and van Rijsbergen [34], divergence from randomness (DFR) models rely on the following assumptions:
 - the distribution of terms in a collection is a random process;
 - when a term is specific to a given document, its frequency is different from its frequencies in other documents, so that specificity can be measured as the divergence of term frequency from the initial randomness model.

The probabilities associated to terms are computed by combining two information sources, Inf_1 and Inf_2 , which are derived from two probabilities, Prob_1 and Prob_2 , following this expression:

$$w_{i,j} = \inf_{1}(t_i, d_j).\inf_{2}(t_i, d_j)$$

$$= -\log_{2}(\operatorname{Prob}_{1}(t_i, d_j)).(1 - \operatorname{Prob}_{2}(t_i, d_j))$$

$$= -\log_{2}(\operatorname{Prob}_{1}(t_i, d_j)^{1 - \operatorname{Prob}_{2}(t_i, d_j)})$$
(7)

Prob₁ is the probability that a term has a given frequency in any document, based on collection information and the randomness model. It is therefore similar to a global weight. The associated information source Inf₁ is called the *informative content* of the term. Prob₂ corresponds to the risk to pick the term as a good document descriptor. It depends on the current document only, similarly to a local weight. The associated information source Inf₂ is called *information gain*, which designates the additional information brought by this term to the current document as compared to any other document. Amati and van Rijsbergen proposed three different randomness models and, since these models cannot be computed exactly, two approximations to compute each of them. Here, we review these models and provide their approximations in Table IV. More details about them can be found in [34].

²The cosine similarity is itself equivalent to distance L_2 when computed on normalized vectors: $d_{L_2}(x,y) = 2 \cdot (1 - \cos(x,y))$.

a) Bernoulli's model: This model assumes that the frequency of term t_i is the result of a Bernoulli process, i.e. the result of successive independent random draws of occurrences of term t_i among its occurrences in the whole dataset. This model raises the following equation for Prob_1 :

$$\operatorname{Prob}_{1}(t_{i}, d_{j}) = \begin{bmatrix} \left(\operatorname{CF}_{i} \\ \operatorname{tf}_{ij} \right) \left(\frac{1}{N} \right)^{\operatorname{tf}_{ij}} \left(\frac{N-1}{N} \right)^{\operatorname{CF}_{i} - \operatorname{tf}_{ij}} \end{bmatrix}$$
(8)

b) Bose-Einstein model: Inspired from random models used in physics, its formulation for $Prob_1$ is:

$$\operatorname{Prob}_{1}(t_{i}, d_{j}) = \frac{(\operatorname{CF}_{i} - \operatorname{tf}_{ij} + 1) \cdot \ldots \cdot \operatorname{CF}_{i} \cdot (N - 1)}{(N + \operatorname{CF}_{i} - \operatorname{tf}_{ij} - 1) \cdot \ldots \cdot (N + \operatorname{CF}_{i} - 1)}$$

c) Inverse document frequency model: It is inspired from the aforementioned principle of idf. Its formulation for Prob_1 is:

$$\operatorname{Prob}_{1}(t_{i}, d_{j}) = \left(\frac{\operatorname{df}_{i} + 0.5}{N+1}\right)^{\operatorname{tf}_{ij}} \tag{10}$$

- d) Information gain models: Together with these randomness models, Amati and van Rijsbergen proposed two models of information gain. We review them here and provide their equations in Table IV (see [34] for details). The first one, called Laplacian model, is based on Laplace's succession law, which provides the probability that a given event (here, a term in a document) that already occurred occurs once more. The other, Bernoulli's model, calculates the information gain as the ratio between the probability to have a given frequency tf_{ij} for term t_i and the probability to have the same frequency plus one. These probabilities are assumed to be Bernoulli processes, hence the name of the model.
- e) Normalization models: Each of these information gain models can be combined with any of the randomness models presented before. Since these models do not take the document length into account, tf_{ij} has to be normalized first. The normalization functions proposed in [34] are provided in Table IV. They are based on the average length of all documents. Since this value is not used in existing local weights, DFR normalization H1 is added to the list of local weights (l_5 in Table III) to evaluate its relevance independently from DFR models.

C. New Weighting Schemes

In this section, two additional weighting schemes that seem suited to images are proposed, based on the statements of Section III-B and III-D. The first one is a global weight based on empirical observations. The second one is a DFR randomness model.

- 1) Average Frequency and idf: This global weight relies on two distinct values:
 - the inverse document frequency idf: it is used as a basis, following the assumption presented in Section III-E;

local mean frequency: objects often contain repeated parts (for instance, eyes in faces or wheels in motorbikes), so the visual words corresponding to such parts are more informative than visual words occurring alone. We propose to take this into account by weighting the term using its average intra-document frequency tf. This average frequency is computed only on the documents where the visual word occurs, because considering the whole collection would penalize the terms that occur only in few images.

The proposed weight combines these two elements into a new global weight (g_4 in Table III). A squared version of this weight is also proposed (g_5 in Table III) to emphasize these terms even more.

2) Hypergeometric Randomness Model: We propose to add a hypergeometric randomness model to DFR models. Contrary to the Bernoulli model which considers the terms occurring in a document as successive random draws of term occurrences among all the term occurrences of the collection, the hypergeometric model considers a single draw of several terms at once. The Bernoulli model accumulates terms one by one in a document, which fits well the common hypothesis used in text retrieval, where terms are added as the document grows. Since, in image retrieval, as shown in Section III-B, sets of terms are more representative of the image content than single terms, a hypergeometric model seems more suited, as it considers terms taken together. From the definition of hypergeometric distributions, the hypergeometric model of Prob₁ is:

$$\operatorname{Prob}_{1}(t_{i}, d_{j}) = \frac{\left(\begin{array}{c} \operatorname{CF}_{i} \\ \operatorname{tf}_{ij} \end{array}\right) \left(\begin{array}{c} \operatorname{CF}^{*} - \operatorname{CF}_{i} \\ \operatorname{dl}_{j} - \operatorname{tf}_{ij} \end{array}\right)}{\left(\begin{array}{c} \operatorname{CF}^{*} \\ \operatorname{dl}_{j} \end{array}\right)}$$
(11)

To estimate factorials, Ramanujan's approximation [35], which provides more precise values than Stirling's approximation used by Amati and van Rijsbergen, is used:

$$\log(k!) \approx k \cdot \log(k) - k + \frac{k(1 + 4k(1 + 2k))}{6} + \frac{\log(\pi)}{2}$$
(12)

The final approximation for this model is given in Table IV.

VII. EXPERIMENTS

This section details the experiments performed using the distances and weighting schemes previously presented. First, the general experimental settings are presented. Then some results obtained with different distances (Section VII-B) are provided, followed by results obtained using different combinations of weighting schemes and distances (Section VII-C). More detailed results are available as an internal publication [36].

A. Experimental Settings

1) Visual Vocabulary:

Interest region detector: The Hessian-affine region detector [37] is used. This region detector provides good performance [38] and is widely used in the visual word-related literature (e.g. [3], [10].

Interest region descriptor: SIFT descriptors [7] are used, as they provide excellent results for many applications [39] and are the most popular in the literature related to visual word-based systems (e.g. [2]–[4], [10]).

Clustering algorithm: For efficiency reasons, we use a hierarchical k-means algorithm [15]. Although it does not provide optimal clusters, it is still a good approximation of more costly solutions such as the classical k-means. Moreover, as stated before, no clustering algorithm is able to provide optimal performance in such high dimensional spaces (128 dimensions for SIFT descriptors).

Vocabulary size: Table V presents, for each dataset, the size of the vocabulary. These sizes have been chosen by validation on some randomly chosen queries, different from the queries used in the experiments (except on the Oxford dataset where the number of available queries is limited; in this case, the largest vocabulary that the implementation could handle is used, following the observations of Philbin et al. [8] on this dataset).

TABLE V. VOCABULARY SIZE FOR EACH DATASET.

Dataset	Vocabulary size
Caltech6	6 556
Caltech101	61 687
Nistér	19 545
Oxford	117 151

2) Queries: All queries are whole images, and their terms are weighted as the database images' ones are. They have been randomly picked from each database, except for the Oxford dataset in which they are provided as part of the ground truth. Table VI shows the number of queries used for each dataset. These quantities are sufficient to provide reliable results [40].

TABLE VI. QUERY NUMBER FOR EACH DATASET.

Dataset	Query number
Caltech6	200
Caltech101	200
Nistér	300
Oxford	55

3) Evaluation: The performance of the tested systems is evaluated using precision and mean average precision (MAP). Precision is computed after different numbers of documents (document cut-off values, DCV), depending of the average number of relevant documents existing in the dataset considered. In particular, a DCV of 4 documents is sufficient for the Nistér dataset. As a sequential search algorithm is used, recall is not necessary: it varies in the same way as precision. The statistical significance of the

results is tested using Wilcoxon's test (p-value = 0.1). The results of these tests are not reported here due to limited space, but are available in [36]; non-significant results are indicated in the comments of the experiments.

B. Experiments on Distances

Minkowski distances are tested for different values of k between 0.01 and 3. The vectors are weighted using tf. idf, as it is done in most publications, and normalized according to the distance.

Fig. 7 shows the performance of the system w.r.t. k. One can make the following observations:

- on Caltech6, the overall results are improved when k < 1, and optimal when k = 0.75. The worst results are obtained with high values of k.
- on Nistér, the results are similar to those obtained on Caltech6. It must be noted that the precision results are less significant here, as there are only 4 relevant documents per query;
- on Caltech101, the behavior is globally similar to what happened with the previous datasets, but the differences of performance are less prominent;
- on Oxford, the results are opposed to the previous ones: best with high values of k (k = 2), worse with low values of k.

These results are discussed in Section VIII-A; the specific case of Oxford is discussed in Section IX-B.

C. Experiments on Weighting Schemes

- 1) Weighting Schemes Tested: In these experiments, the weighting schemes are divided into two categories:
 - weighting schemes obtained by the combination (product) of a local weight and a global weight (plus a proper normalization depending on the distance) from Table III. In the results, combinations of a local weight and a global weight are noted $l_x g_y$, where l_x is the local weight and g_y the global weight;
 - weighting schemes derived from DFR models (Table IV). They are tested separately as their components make sense only taken together. The vectors are re-normalized according to the distance. In the results, these weights are noted XYZ, where X is the randomness model, Y the information gain model and Z the normalization model.

Every weighting scheme is tested using distances L_1 and L_2 . Although they do not provide optimal results (as shown before), they are the most common distances in the literature and reflect the differences observed on the datasets. Some experiments have been performed using the original formulation of the DFR models to show the importance of query normalization. Not presented here due to limited space, they are available in [36].

2) Results: The results presented here are computed as the performance gain obtained with respect to the weighting scheme l_1g_0 (mere count of term occurrences). The absolute performance of this baseline are provided in Table VII. Detailed results are available in [36].

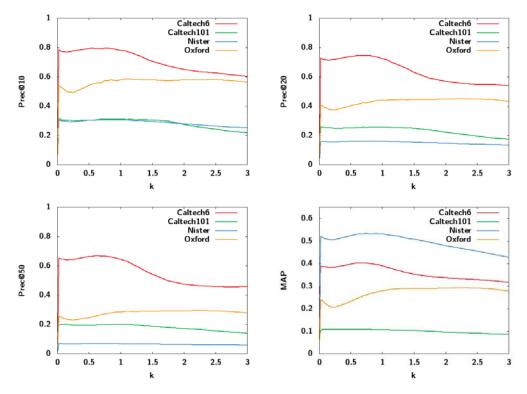


Figure 7. Effect of parameter k on the performance of L_k distances.

TABLE VII.

Results Obtained Using the Baseline Weighting Scheme l_1g_0 , with Distances L_1 (Top) and L_2 (Bottom). Precision is Computed After 4 Documents for Dataset Nistér and

AFTER	10	DOCUMENTS	FOR	THE	OTHERS.
ANI ILK	10	DOCUMENTS	IOI	IIIL	OTHERS.

Dataset	Precision	MAP
Caltech6	0.7685	0.3827
Caltech101	0.3146	0.1071
Nistér	0.6970	0.5229
Oxford	0.5600	0.2698
Dataset	Precision	MAP
Caltech6	0.6208	0.3287
Caltech101	0.2727	0.0985
Nistér	0.5985	0.4611
Oxford	0.5655	0.2730

Combined weighting schemes: Figures 8 and 9 present the results obtained using combined local and global weights. The best local weight depends on the dataset considered. When L_1 is used to compare the vectors:

- on Caltech6, l₃ and l₇ are the best local weights, followed by l₄;
- on Caltech101, all weights are equivalent (not statistically significant differences), except l₆ which worsens the results;
- on Oxford, weight l_6 yields the best results, followed by l_1 and l_5 , the two latter providing similar results. All the others worsen the results.

When using L_2 , the results' trend is similar –no weighting scheme prevails on all datasets– but the absolute performance is always worse than the performance obtained

with L_1 , except for Oxford, as in Section VII-B. These results are not further detailed but will be discussed in Section IX.

As for the local weights, no global weight outperforms all the others: it depends on the dataset considered. When using L_1 :

- on Caltech6, our new weights g₄ and g₅ provide the best results;
- on Nistér, g_3 is the best, but g_1 and g_2 yield good results as well:
- on Caltech101, the results are similar for all weights, and using no global weight at all is often the best choice;
- on Oxford, g_3 and g_5 provide the best results.

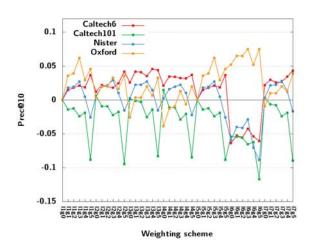
The results using L_2 are not detailed here since the performance is generally worse than what was obtained using L_1 . Globally, g_3 often yields good results; g_4 and g_5 also provide interesting results on two datasets. The results of local weights are discussed in Section VIII-B, and those of global weights in Section VIII-C. The case of Oxford is discussed in Section IX-B.

DFR weighting schemes: Figures 10 and 11 present the results obtained using DFR-based weighting schemes. The results obtained are close from one to another, and the differences are often not statistically significant. As for the local/global schemes, none of them outperforms all the others, although a small group is clearly more effective than the others: when using distance L_1 , the approximation D of the Bernoulli model performs well in many cases, as well as the inverse document frequency-based models. Model DLH1 yields the best results on the Nistér dataset, but information gain model B per-

forms better on the others. None outperforms the best local/global schemes. Here again, L_2 provides scores that are closer from one to another, and behaves globally worse than L_1 . Some aspects of these results are discussed in Section VIII-D.

Overall results: Table VIII shows the best improvement obtained on each dataset (using distance L_1). Two remarks can be done:

- the maximal gain is generally not very important, and not always statistically significant. The only gain that excess 10% is obtained on the Oxford dataset. The gain is low on the most difficult dataset, Caltech101;
- the maximal gain is obtained by a different weighting scheme for each dataset and each performance measure; none of these weighting schemes provides optimal results on any type of data.



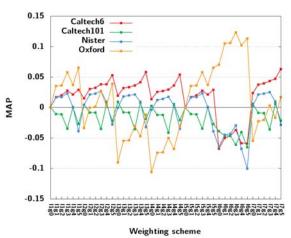
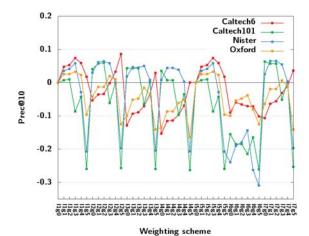


Figure 8. Performance gain using the weighting schemes from Table III and distance L_1 , with respect to the baseline weighting scheme l_1g_0 (Precision@4 for the Nistér dataset). The lines are added to facilitate visualization.

VIII. ANALYSIS OF THE RESULTS

A. On Parameter k of Minkowski Distances

On Caltech6 and Nistér datasets, the observed effect of parameter k is consistent with what was observed by [23] and, especially, [24]: low values of k improve



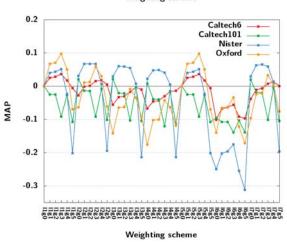


Figure 9. Performance gain using the weighting schemes from Table III and distance L_2 , with respect to the baseline weighting scheme l_1g_0 (Precision@4 for the Nistér dataset). The lines are added to facilitate visualization.

the retrieval results, but below a certain value, the performance decreases dramatically. However, this behavior is not well understood. The role of k can be seen as follows: it confers more or less importance to each of the local distances³. Using low values of k reduces the highest local distances (see Fig.12), and thus provides more importance to lower local distances in the calculation of the final global distance value. It explains the fact that fractional distances generally yield good performance in high-dimensional spaces: they allow more dimensions to "express" themselves in the overall distance calculation. More practically, the results here show that a compromise must be found between high and low local distances.

On the Caltech101 dataset, distances so that k < 1 also improve the results, but the performance gain is much more limited than it was on Caltech6 and Nistér datasets. The first difference between this dataset and the two preceding ones is the size of the vocabulary. A much larger vocabulary is used for Caltech101, resulting in much sparser vectors. However, Howarth and Rüger

³A local distance is a distance computed on only one dimension, *i.e.* the absolute value of the difference between the values of the vectors for this dimension.

TABLE VIII. Maximum Performance Gain Observed with Respect to the Baseline Weighting Scheme l_1g_0 , Using Distance L_1 .

	p@4	p@5	p@10	p@20	p@50	p@100	MAP
Caltech6	N/A	N/A	+4.3%	+6.3%	+8.0%	+8.5%	+6.3%
Caltech101	N/A	N/A	+1.5%	+1.1%	+0.9%	+0.7%	+0.9%
Nistér	+3.3%	N/A	N/A	N/A	N/A	N/A	+2.7%
Oxford	N/A	+3.0%	+7.5%	+9.3%	+10.0%	N/A	+12.3%

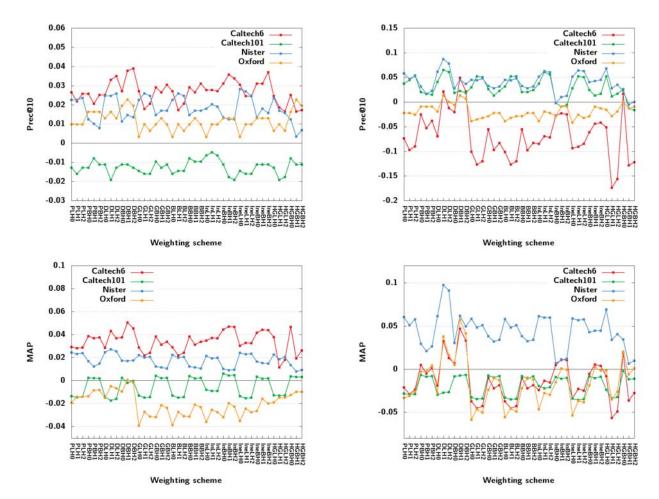


Figure 10. Performance gain using DFR weighting schemes and distance L_1 , with respect to the baseline weighting scheme l_1g_0 (Precision@4 for the Nistér dataset). The lines are added to facilitate visualization.

Figure 11. Performance gain using DFR weighting schemes and distance L_2 , with respect to the baseline weighting scheme l_1g_0 (Precision@4 for the Nistér dataset). The lines are added to facilitate visualization.

showed that fractional distances are more suited to sparse vectors than dense ones [24]. The second difference is the quality of the vocabulary: the strong intra-class variations of images, and the small inter-class variations of some classes make it much more difficult to obtain consistent visual words, in the sense of visual words corresponding to specific parts of objects in a given class. The probability that a given SIFT descriptor falls in the wrong cluster (i.e. visual word) is high⁴. The resulting visual word vectors are then very noisy. The presence of noise explains the fact that all the distances tend to be equivalent, since [23] already highlighted this behavior in their experiments. It

shows that, contrary to what is often admitted, the use of a descriptor quantization step does not provide more generalization skills to the system, but rather adds some noise.

B. Local Weights

As shown in Section VII-C, the choice of the optimal local weights strongly depends on the dataset considered. The local weights can be classified into two major categories:

- weights that limit the high frequencies: l₂, l₃, l₄ and l₇, l₄ being an extreme case where all frequency information is discarded;
- weights that emphasize high frequencies: l_6 .

⁴The fact that the vocabulary is large increases this trend, however we used the vocabulary size that provided the best results on some validation queries.

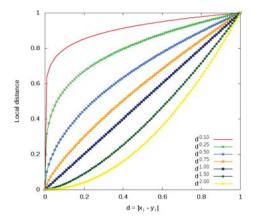


Figure 12. Importance of the local distances with respect to the value of k.

The others $(l_1 \text{ and } l_5)$ are somewhat equivalent in the sense that the relative importance of the different terms within the document remain the same. The local weights from the first category are the most effective on Caltech6 and Nistér datasets, but not on the Caltech101 dataset where no weight significantly improves the performance. The distribution of words in the documents can help explaining that. Fig.13 shows the average frequency of words in the documents in which they occur, and the standard deviation of this frequency (the documents where they do not occur are of poor interest to study local weights). Table IX also shows, for each dataset, the number of hapax, i.e. of words occurring only once in the whole collection. For Caltech6 and Nistér datasets, the average frequencies of visual words stand between 1 and 2 for most of the vocabulary, and increase dramatically for a few visual words. Moreover, several words have an average frequency of 1, without being hapax. This distributions explain the effectiveness of the weights reducing the high frequencies, since they prevent the most frequent visual words to represent most of the overall distance between the documents to the expense of less frequent visual words. It shows the need to find a compromise between taking into account the frequencies of words or their presence only, because high frequency words can hide the presence or absence of less frequent words in the final distance value. This behavior is similar to what was observed with distances; this is further discussed in Section IX-A. On Caltech101, most of the average frequencies stand between 1 and 1.2. Moreover, many words have an average frequency of one, only one third of them being hapax. This fact can be explained by the large size of the vocabulary as well as by the presence of noise due to the quantization step, as we mentioned already. Such frequencies can explain the lack of effect of most of the local weights, since they tend to be equivalent when local frequencies are equal to 1 (especially, l_1 , l_2 , l_4 and l_6 become identical). It also explains why l_6 worsens the results dramatically: it emphasize the rare visual words whose frequency is 2 or more, at the expense of most of the vocabulary. Finally, the distributions on Oxford are similar to the ones observed on Caltech101, with higher standard deviations, with different results; this special case is discussed in Section IX-B.

TABLE IX. Comparison Between Hapax and Visual Words such as $\overline{\mathrm{tf}_i} = 1.$

Dataset	Hapax	$\overline{\mathrm{tf}_i} = 1$
Caltech6	0	17
Caltech101	3021	9790
Nistér	56	358
Oxford	1434	4206

C. Global Weights

One interesting result about global weights is obtained on Caltech101: the best setting on this dataset is not to use any global weight. It shows that it is hard to provide valid assumptions about the distribution of visual words and to define effective weights when the complexity of the dataset increases. It also questions the tendency of many authors to use idf systematically, since our results show that it is not a good choice when dealing with complex data. On Caltech6 and Nistér, weights g_4 and g_5 help improving the results, in combination with local weights that decrease local frequencies. These weights allow to emphasize words that tend to be repeated locally over the dataset, as opposed to l_6 which emphasize any words that are locally frequent. In datasets where objects (e.g. motorbikes or faces in Caltech6) contain both repeated and non-repeated visual words, such weights allow to reach a better compromise between word presence and word frequency.

D. DFR Weighting Schemes

Globally, the complexity of the DFR models make the results that they yield difficult to interpret. However, about the hypergeometric model we propose, one can make the following remarks:

- contrary to the other DFR weights, the hypergeometric model-based weights perform better with a H0 normalization, i.e. without normalizing frequencies. It can be explained by the fact that this model already takes into account the length of the documents, making any additional normalization superfluous;
- although it is more appealing in theory, this model yields results that are similar to the ones obtained using a Bernoulli model (the difference between them are not statistically significant). It can be explained by the fact that hypergeometric distributions tend to be equivalent when the number of elements to draw (here, the document length) is negligible compared to the overall number of elements (here, CF*, the overall number of visual word occurrences in the collection).

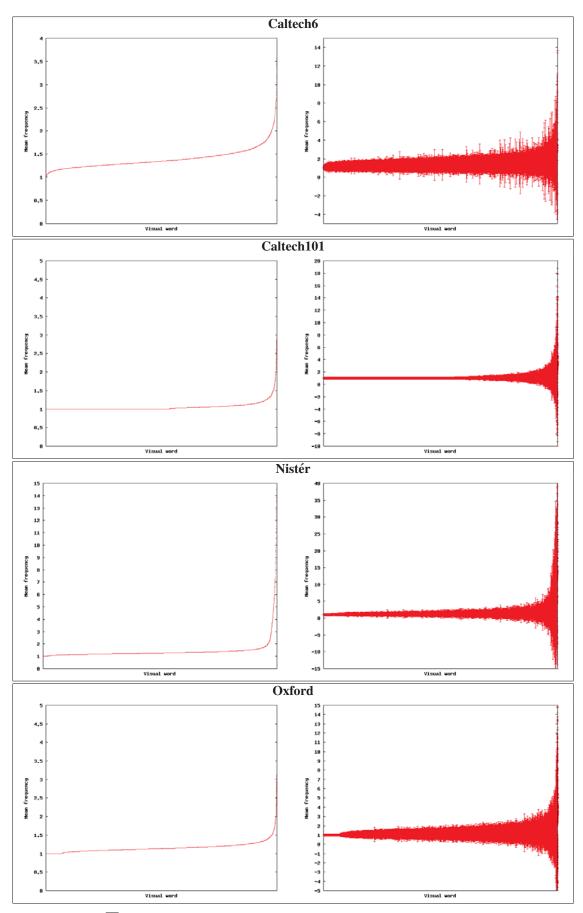


Figure 13. Mean frequencies $\overline{tf_i}$ of the visual words (left) and mean frequencies and their standard deviation (right). These values are computed on the documents where each word appears only.

IX. DISCUSSION

A. On the Relation Between Distances and Weighting Schemes

As mentioned in Section VIII-B, results suggest a relationship between some types of weighting schemes and Minkowski distances: the weighting schemes limiting high frequencies of visual words tend to yield similar effects as distances L_k with a low value of k, whereas the weighting schemes that emphasize high term frequencies provide results similar to those provided by the use of high values for k. This can be explained at the local distance scale: a low k value decreases local distances, as does the reduction of the amplitude of high frequencies, and inversely. It is therefore possible to approximate a fractional distance by using any regular distance and a suited weighting scheme. This property allows one to overcome the fact that fractional distances do not respect the triangle inequality [23]. This trick allows to combine the good properties of fractional distances with indexing methods relying on the triangle inequality to quicken nearest-neighbor searches [41]. This property also shows that both distance and weights have to be taken into account when looking for the appropriate compromise between term presence and term frequency. For instance, on the Oxford dataset, we see that l_6 is the best local weight when using distance L_1 , but not with distance L_2 , since the effects of the distance and the local weight are cumulated.

B. Influence of the Visual Nature of the Images

Results show that the visual content of the images has a strong impact on the choice of an optimal weighting scheme. On Oxford, the best results are obtained using weight l_6 (or, equivalently, a distance L_k with a high k). This happens because the query objects are buildings, which contain mostly repeated parts (e.g. windows), as shown in Fig.14. Emphasizing locally frequent words gives importance to parts of the object of interest over parts of the background, and to buildings over objects in distractor images that do not contain repeated parts. On Caltech6 and Nistér, weights g_4 and g_5 emphasize words that tend to be locally repeated over the datasets, instead of emphasizing all words that are frequent in the current image. This is effective considering the objects of interest from these datasets (faces, motorbikes...) that contain both repeated (eyes, wheels...) and non-repeated parts.

Therefore, the nature of the images has a strong impact on the best settings to use. Although no optimal setting exists, it is possible to adjust the weighting strategy to the nature of the data considered. Some guidelines to do so are provided in Section X.

C. Scalability of Weighting Schemes

The results obtained on Caltech101 questions the scalability of most weighting schemes: are weighting schemes systematically less effective when the number of object

classes increases? The results on the Nistér dataset show that this is not the case. Indeed, although this dataset contains 2,550 classes⁵, some weighting schemes have shown to be effective. The difference lies in the nature of the classes: since Nistér is dedicated to near-duplicate retrieval, the classes have a low intra-class variance and a high inter-class variance; Caltech101, which contain generic classes of objects, has opposite properties. Therefore, weighting schemes can be detrimental in the case of complex datasets, where visual word ambiguity is high, but still perform well at the scale of large numbers of classes, as long as ambiguities between classes are limited.

D. Looking Back on the Relationship to Textual Words

In Section III, a conceptual analysis of the differences between visual words and textual words was proposed. The experimental results reported here validate some of these differences:

- intra-document frequency (see Section III-D) of visual words differs from the one of textual words. Indeed, intra-document frequency of visual words is related to the visual nature of the objects, but not necessarily to their overall representativity in the image. This makes the choice of local weights less straightforward as it is for text data, as shown by the results obtained on Oxford as compared to other datasets: the choice to emphasize or not locally frequent visual words will depend on the visual nature of the query objects;
- collection frequency (see Section III-E) of visual words cannot be interpreted in the same way as collection frequency of textual words;
- query normalization is essential in image retrieval because queries are images containing a high number of word occurrences, as opposed to text queries which are usually short. A series of experiments (not reported here but available in [36]) showed that not normalizing image query vectors is always detrimental to the performance of the retrieval system.

These differences are mostly due to the semantic nature of visual words as compared to textual words (as introduced in Section III-B). The major consequence in terms of weighting strategies is that images require to adapt the weighting schemes to the visual content of the objects to be queried and the visual variability of the collection, whereas text retrieval can rely on nearly universal weighting approaches. Some authors [42] proposed to build visual vocabularies which provide similar statistics as textual vocabularies; however, this may not be desirable since, as shown in our experiments, the difference between text and image data can be leveraged to improve the retrieval results, by choosing weighting schemes according to the visual content of the datasets.

⁵By considering a query and the related images as a single class.



Figure 14. Query images from the Oxford dataset. The query buildings contain many repeated visual elements (e.g. windows, architectural features) that are very specific to the query objects and have to be emphasized to best describe the images' content.

X. Some Guidelines to Choose Weighting Schemes

From the analyses and discussions provided in Sections VIII and IX, we can draw the following major conclusions about weighting scheme design:

- local weighting schemes help to give locally more
 or less importance to visual words that are repeated
 in documents; it is possible to design global weights
 that have a similar effect, only globally;
- tuning parameter k of Minkowski distance is equivalent to using a local weighting scheme to emphasize more or less highly frequent visual words;
- the fact that locally frequent visual words are good descriptors of query objects depends on the visual nature of these objects (e.g. when they are buildings as in Fig.14);
- idf is not effective when the dataset considered is too complex (high number of classes with high intraclass variance and low inter-class variance see for instance Fig.6).

Based on this, the following general formulation of a weighted distance can be used in any system:

$$d_{\alpha,\beta,\gamma}(x,y) = \sum_{i=1}^{n} \operatorname{idf}_{i}^{\gamma} \overline{\operatorname{tf}_{i}}^{\beta} |x_{i}^{\alpha} - y_{i}^{\alpha}|$$
 (13)

This formulation uses the L_1 distance as a basis, as it provides a "neutral" distance. Emphasizing or not highly frequent words is left to the local and global weights. The parameters α , β and γ of this general weighted distance are described below.

Parameter α has the effect of a local weight: setting $\alpha < 1$ will reduce the emphasis of locally frequent visual words, while setting $\alpha > 1$ will increase their importance. In general, α should be low, to emphasize word presence over word frequency, as shown by the results obtained on the Caltech6, Caltech101 and Nistér datasets. When locally repeated visual words are important to describe query objects, α should be set to a high value. This is the case for building image datasets for instance, but also datasets containing highly textured objects in general. An

analysis of the average frequency of visual words in the images as the one provided in Fig.13 can help to adjust the value of α in the absence of specific ground truth data. Other dataset properties that could be taken into account to refine this estimation are:

- the relevance of frequent patterns to describe query or relevant images as compared to distractors (images that are not expected to be queries or relevant results), e.g. when searching for buildings (such as the ones in Fig.14) in general-purpose image collections;
- the relevance of frequent patterns to describe query or relevant objects as compared to their background in a single image. The third column of Fig.3 provides an example where frequent patterns are specific to the background rather than the object of interest.

Computing statistics about these facts would require relevance or segmentation ground truth data.

Parameter β can emphasize visual words that have a high average frequency in the images in which they occur. As a global parameter (related to weights g_4 and g_5 in our experiments), it differs from α as it does not emphasize all the word frequencies in a single image but only the frequencies of some specific words which have higher local frequencies over the collection. Whereas high α values are suited to images that contain mainly repeated parts (such as building images), β should be high when the objects of interest contain both repeated and non-repeated words (e.g. a face, as in Fig.1).

Parameter γ controls the idf part of the weighting scheme, i.e. whether specific (or rare) visual words should be emphasized. As shown in our experiments, γ should be set to a high value in datasets where the complexity of images (in terms of intra- and inter-class variance) is limited, such as Caltech6 and Nistér. When the datasets get more complex visually and descriptors cannot be assigned to visual words reliably (as in Caltech101), lower values of γ should be picked.

XI. RELATED WORK

A. Visual Words and Weighting Schemes

Although many authors include the tf.idf weighting scheme to their visual word-based systems, few papers exist that really analyze the effectiveness of visual words and try to adapt weighting schemes from text retrieval systems to image retrieval systems. To our knowledge, only Jiang et al compared the classical tf.idf to another existing scheme, the binary weighting, in the context of image classification [43] and image retrieval [44]. They used these two weighting schemes as a baseline to evaluate their own weighting scheme, which relies on a fuzzy assignment of the initial descriptors to their two nearest centroids instead of just one, as it is usually done. In the same spirit, Gemert et al. proposed to weight visual word occurrences based on the distances from the descriptors to the vocabulary centroids [45]. None of them considered weighting schemes based on the statistical distribution of the visual words only, nor did they provide any comprehensive study of such weighting schemes as it is done in this paper. Ref. [8] proposed a work similar to the one of Jiang et al., also based on the properties of the clustering rather than statistical properties. More recently, [46] proposed a weighting scheme based on a previous segmentation of the image based on color, to integrate geometrical information to their system. Here again, the statistical properties of visual words in the image and in the collection is not used. To our knowledge, [47] are the only ones who studied such properties. They studied the burstiness of visual words, which is the fact that visual words are more likely to occur in an image if this image already contain numerous occurrences of them. This phenomenon is also well-known in text retrieval. They correct their similarity scores by decreasing the score of visual words that have a high number of matches, and show that it increases the performance of their system. Their work differs from the present paper as their correction does not rely on pure statistics of term occurrences but rather uses on the distance of the descriptors to the cluster center.

B. Visual Words and Distances

Determining the optimal distance for visual word-based systems is essential but only received little attention in the literature. As mentioned in Section V, [15] compared L_1 and L_2 distances on one dataset only, and preferred L_1 , but did not provide any explanation about these results. Some other authors proposed custom distances for visual word vectors comparison, based on some specific properties. Ref. [3] proposed a distance which relies on the assignation step of the local descriptors to the centroids, to provide a finer similarity measure. Ref. [16] also proposed a new distance based on this kind of properties. Finally, [20] proposed another distance, which relies on the properties of the nearest-neighbor search in a more general way. All these distances where generally validated on only one dataset.

XII. CONCLUSION

In this paper, we proposed a study about distances and weighting schemes in the context of visual words-based image retrieval systems. We tested several parametrization of the Minkowski distance and 84 weighting schemes configurations adapted from the text retrieval literature, and proposed two new weighting schemes based on specific properties of visual words. The experiments were performed on four datasets, corresponding to the two most popular image retrieval tasks: particular object/scene retrieval and categorized image retrieval. These datasets present some interesting features that allow to bring to light some properties of the visual words. Results show that there is no universally optimal weighting schemes, but rather that weighting schemes should be chosen according to some visual properties of the dataset considered. To do so, a general weighting framework and some associated guidelines were proposed. In addition to these conclusions, our study revealed other interesting facts:

- Minkowski distances and weighting schemes are linked, in the sense that they provide the same kind of effect on the distance calculation, so that weighting schemes can approximate fractional distances;
- contrary to what is often assumed, the quantization of descriptors does not provide more generalization properties to the system, but rather adds some noise which strongly limits the performance of the system, not only when dealing with particular object retrieval (as it was implicitly shown by some previous papers such as [3], [16]), but also when considering categorized image retrieval. This result also explains the success of some approaches in supervised image classification that take into account the accuracy of descriptor quantization [6];
- using no weighting scheme at all can be the best choice when dealing with complex datasets.

Finally, this work also questions some experimental habits of the community:

- the tendency to systematically use a tf . idf weighting scheme without considering other choices;
- the tendency to test the systems on one dataset only, whereas datasets always contain biases, and can therefore yield results that cannot be generalized to others datasets.

One task that can be envisaged following these results is to try to make the systems able to adapt themselves to the nature of the query. This issue, that has not been addressed directly so far, but is implicitly present is some papers such as [48], is not limited to visual word-based systems, but to any retrieval system, whatever the kind of descriptors it uses. Indeed, it is clear that some descriptors are clearly suited to describe some kind of objects but not the others, e.g. color is suited to describe natural objects (such as trees), but not human-made objects (such as cars). Another perspective is to try to better understand the model underlying the distribution of visual word in images, to define more accurate weighting schemes.

In text retrieval, some authors have formally defined the desirable properties of IR systems according to the objectives of their task [49]. The same work could be done in image retrieval, according to the different image retrieval tasks, to help building models of interest point distribution and/or better interest point detectors, whose properties would be suited to the image retrieval task considered.

REFERENCES

- [1] P. Tirilly, V. Claveau, and P. Gros, "Distances and weighting schemes for bag of visual words image retrieval," in *Proceedings of the International Conference on Multimedia Information Retrieval (MIR)*, Philadelphia, Pennsylvania, USA, 2010, pp. 323–332.
- [2] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proceedings of* the International Conference of Computer Vision (ICCV), vol. 2, Nice, France, October 2003, pp. 1470–1477.
- [3] H. Jégou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *Proceedings of the European Conference on Computer Vision (ECCV)*, vol. 1, Marseille, France, October 2008, pp. 304–317.
- [4] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proceedings of the European Conference on Computer Vision (ECCV), workshop on statistical learning and computer vision*, Prague, Czech Republic, May 2004, pp. 1–22.
- [5] S. Nowak and M. J. Huiskes, "New strategies for image annotation: Overview of the photo annotation task at imageclef 2010," in *Proceedings of the Cross-Language* Evaluation Forum (CLEF), Padua, Italy, September 2010.
- [6] F. Perronnin and C. R. Dance, "Fisher kernels on visual vocabularies for image categorization," in *Proceedings of* the international conference on Computer Vision And Pattern Recognition (CVPR), Minneapolis, MN, USA, June 2007, pp. 1–8.
- [7] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the International Conference* of Computer Vision (ICCV), vol. 2, Kerkyra, Corfou, Greece, August 1999, pp. 1150–1157.
- [8] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proceedings of the international conference* on Computer Vision And Pattern Recognition (CVPR), Minneapolis, MN, USA, June 2007, pp. 1–8.
- [9] A. Bosch, A. Zisserman, and X. Muñoz, "Scene classification via pLSA," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Graz, Austria, May 2006, pp. 517–530.
- [10] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman, "Total recall: automatic query expansion with a generative feature model for object retrieval," in *Proceedings of the International Conference of Computer Vision (ICCV)*, Rio de Janeiro, Brazil, October 2007, pp. 1–8.
- [11] C. Schmid and R. Mohr, "Local grayvalue invariants for image retrieval," *IEEE Transactions on Pattern Analysis* and Machine Intelligence, vol. 19, no. 5, pp. 530–535, May 1997.
- [12] Q.-F. Zheng, W.-Q. Wang, and W. Gao, "Effective and efficient object-based image retrieval using visual phrases," in *Proceedings of ACM Multimedia*, Santa Barbara, CA, USA, October 2006, pp. 77–80.
- [13] P. Tirilly, V. Claveau, and P. Gros, "Language modeling for bag-of-visual words image categorization," in *Proceedings* of the Conference on Image and Video Retrieval (CIVR). Niagara Falls, Canada: ACM, July 2008, pp. 249–258.

- [14] D. Larlus and F. Jurie, "Latent mixture vocabularies for object categorization," *Image and Vision Computing*, vol. 27, no. 5, pp. 523–534, April 2009.
- [15] D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," in *Proceedings of the international* conference on Computer Vision And Pattern Recognition (CVPR), vol. 2, June 2006, pp. 2161–2168.
- [16] Y.-G. Jiang and C.-W. Ngo, "Visual word proximity and linguistics for semantic video indexing and near-duplicate retrieval," *Computer Vision and Image Understanding*, vol. 113, no. 3, pp. 405–414, March 2009.
- [17] F. Debole and F. Sebastiani, "An analysis of the relative hardness of reuters-21578 subsets," *Journal of the American Society for Information Science and Technology*, vol. 56, pp. 584–596, April 2005.
- [18] K. Spärck Jones, "A statistical interpretation of term specificity and its application in retrieval," *Journal of Documentation*, vol. 28, no. 1, pp. 132–142, 1972.
- [19] K. Spärck Jones, S. G. Walker, and S. E. Robertson, "A probabilistic model of information retrieval: Development and comparative experiments (part 1)," *Information Pro*cessing and Management, vol. 36, no. 6, pp. 779–808, November 2000.
- [20] H. Jégou, H. Harzallah, and C. Schmid, "A contextual dissimilarity measure for accurate and efficient image search," in *Proceedings of the international conference* on Computer Vision And Pattern Recognition (CVPR), Minneapolis, MN, USA, June 2007, pp. 1–8.
- [21] 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*, vol. 106, no. 1, pp. 59–70, April 2007.
- [22] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Proceedings of the international conference on Computer Vision And Pattern Recognition (CVPR)*, San Francisco, CA, USA, June 2010, pp. 3304–3311.
- [23] C. C. Aggarwal, A. Hinneburg, and D. A. Keim, "On the surprising behaviour of distance metrics in high dimensional spaces," in *Proceedings of the 8th International Conference on Database Theory*, ser. Lecture Notes in Computer Science, London, United Kingdom, 2001, pp. 420–434.
- [24] P. Howarth and S. Rüger, "Fractional distance measures for content-based image retrieval," in *Proceedings of the European Conference on Information Retrieval (ECIR)*, Santiago de Compostela, Spain, March 2005, pp. 447–456.
- [25] C. J. Van Rijsbergen, Information retrieval. Butterworth, 1977.
- [26] H. P. Luhn, "The automatic creation of literature abstracts," *IBM Journal on Research and Development*, vol. 2, no. 2, pp. 159–165, 1958.
- [27] C. Buckley, G. Salton, and J. Allan, "Automatic retrieval with locality information using SMART," in *Proceedings* of the first Text Retrieval Conference, Gaithersburg, USA, 1992, pp. 59–72.
- [28] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing and Management*, vol. 24, no. 5, pp. 513–523, 1988.
- [29] V. Claveau, F. Moreau, and P. Sébillot, "Description des textes," in L'indexation multimédia: description et recherche automatiques, P. Gros, Ed. Hermès-Lavoisier, 2007, pp. 163–190.
- [30] S. Robertson, "The probability ranking principle in information retrieval," *Journal of documentation*, vol. 33, pp. 294 304, 1977.
- [31] S. E. Robertson, S. G. Walker, M. M. Beaulieu, M. Gatford, and A. Payne, "Okapi at TREC 4," in *Proceedings of*

- the Text Retrieval Conference (TREC), Gaithersburg, MD, USA, November 1996, pp. 73–96.
- [32] S. P. Harter, "A probabilistic approach to automatic keyword indexing," *Journal of the american society for information science*, vol. 26, no. 4, pp. 197–206, July-August 1975.
- [33] K. Spärck Jones, S. G. Walker, and S. E. Robertson, "A probabilistic model of information retrieval: Development and comparative experiments (part 2)," *Information Processing and Management*, vol. 36, no. 6, pp. 809–840, November 2000.
- [34] G. Amati and C. J. Van Rijsbergen, "Probabilistic models of information retrieval based on measuring the divergence from randomness," *ACM Transactions on Information Systems*, vol. 20, no. 4, pp. 357–389, October 2002.
- [35] S. Ramanujan, *The lost notebook and other unpublished papers*. Springer-Verlag, 1988.
- [36] P. Tirilly, V. Claveau, and P. Gros, "A review of weighting schemes for bag of visual words image retrieval," IRISA -INRIA Rennes Bretagne-Atlantique, Rennes, France, Tech. Rep. 1927, April 2009.
- [37] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *International Journal of Computer Vision*, vol. 60, no. 1, pp. 63–86, October 2004.
- [38] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A comparison of affine region detectors," *International Journal of Computer Vision*, vol. 65, no. 1-2, pp. 43–72, 2005.
- [39] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, October 2005.
- [40] C. Buckley and E. M. Voorhees, "Evaluating evaluation measure stability," in *Proceedings of the ACM SIGIR* conference, Athens, Greece, July 2000, pp. 33–40.
- [41] J. Barros, J. French, W. Martin, P. Kelly, and M. Cannon, "Using the triangle inequality to reduce the number of comparisons required for similarity-based retrieval," in Proceedings of SPIE/IS&T Conference on Storage and Retrieval for Image and Video Databases IV, San Jose, CA, USA, February 1996, pp. 392–403.
- [42] J. Martinet, "From text vocabularies to visual vocabularies: what basis?" in *Proceedings of the international conference on computer Vision Theory and Applications*, Lisbon, Portugal, January 2014.
- [43] Y.-G. Jiang, C.-W. Ngo, and J. Yang, "Towards optimal bag-of-features for object categorization and semantic video retrieval," in *Proceedings of the Conference on Image and Video Retrieval (CIVR)*, Amsterdam, The Netherlands, July 2007, pp. 494–501.
- [44] J. Yang, Y.-G. Jiang, A. G. Hauptmann, and C.-W. Ngo, "Evaluating bag-of-visual-words representations in scene classification," in *Proceedings of ACM Multimedia, Work-shop on Multimedia Information Retrieval (MIR)*, Augsburg, Germany, July 2007, pp. 197–206.
- [45] J. van Gemert, C. Veenman, A. Smeulders, and G. J.M., "Visual word ambiguity," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 7, pp. 1271–1283, 2010.
- [46] X. Chen, X. Hu, and X. Shen, "Spatial weighting for bagof-visual-words and its application in content-based image retrieval," in *Proceedings of the Pacific-Asia Conference* on Knowledge Discovery and Data Mining, Bangkok, Thailand, April 2009, pp. 867–874.
- [47] H. Jégou, M. Douze, and C. Schmid, "On the burstiness of visual elements," in *Proceedings of the international conference on Computer Vision and Pattern Recognition (CVPR)*, Miami, FL, USA, June 2009, pp. 1169–1176.

- [48] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid, "Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation," in *Proceedings of the International Conference of Computer Vision (ICCV)*, Kyoto, Japan, September-October 2009, pp. 309–316.
- [49] H. Fang, T. Tao, and C. Zhai, "A formal study of information retrieval heuristics," in *Proceedings of the ACM SIGIR conference*, Sheffield, United Kingdom, July 2004, pp. 49–56.

The Study of PID AQM Algorithm for FPGA Implementation

Xiaoping Yang

State Key Laboratory of Automobile Simulation and Control, Jilin University, Changchun Jilin 130025, China College of Communication Engineering, Jilin University, Changchun Jilin 130012, China Email: yxp@jlu.edu.cn

Xiaojing Wang, Te Di, and Chao Niu College of Communication Engineering, Jilin University, Changchun Jilin 130012, China E-mail: {xiaojing13,dite14,niuchao14}@jlu.edu.cn

Abstract—This paper proposes a kind of project which is about FPGA (the Field Programmable Gate Array, FPGA) hardware implementation scheme for PID AQM algorithm. According to the analyses of the relatively mature discrete PID algorithm, we realized PID algorithm by using the combination of LPM (Library of Parameter Modules) macro module and Verilog code in FPGA. In this article, we write the program of serial communication to achieve data communication of FPGA with the router (Ipcop), and we use the interface to connect PID algorithm and FPGA serial communication module, the purpose of which is to achieve the control of PID algorithm to control network communication data flow. This article uses FPGA to reduce the consumption of router by achieving PID AQM algorithm, which can improve the algorithm's speed of execution and the real time performance.

Index Terms—PID AQM; Active Queue Management; FPGA

I. INTRODUCTION

AQM (Active Queue Management, AQM) algorithm is an important cluster of algorithms to control network congestion in the router nodes. AQM algorithms through the intermediate node of the network randomly discard IP packets, expecting the smaller queuing delay and larger data throughput. So far, researchers have proposed many AQM algorithms [1] [2], such as RED, PID, BLUE, AQM algorithm based on neural prediction model control, etc. In simulation environment, AQM algorithms with good control effect but with unsatisfactory effect in practice application, mainly because of computational complexity of such algorithms. Adding AQM algorithms to the router directly will increase the computation in control and over consumption of router resources, and make the real-time of an AQM algorithm worse. For the above defects of AQM algorithm in actual deployment, we proposed two aspects to improve the control effect of AQM algorithm. On the one hand, we try to improve the

The work is supported by the Key Scientific and Technological Project of Jilin Province (No. 20140204087GX), the National Nature Science Foundation of China (No. 6140020296) and Jilin Provincial Science Foundation of China (No. 20140520063JH).

computing speed of AQM algorithm directly; on the other hand, to add AQM algorithm to an external device.

For several AQM algorithms have problem of long response time, PID AQM algorithm was proposed [3]. And they use NS2 to simulation it under the condition of different load, different time delay. Simulation results indicate the PID algorithm is stable, robust, fast response. So far, many researchers have do their best to improve PID algorithm [4]. In order to improve the effect of algorithm in practical application, we focus to the hardware implementation scheme [5]. The emergence of the FPGA [6] and its powerful function provide the new train of thought for us in congestion control. FPGA has the capabilities of fast operation, rich hardware resources, working in parallel manner, and processing data with high efficiency, fast processing speed, strong program ability, good real-time, etc. Many researchers have used FPGA for AQM algorithm implementation [7]. And we have implemented RED algorithm in FPGA, which laid a solid foundation for this study [8].

This paper proposes a kind of project which is about FPGA hardware implementation scheme for PID AQM algorithm. Specific steps as shown below. Firstly, we analyze the mathematical formula of discrete PID algorithm. Secondly, we use the combination of the LPM macro module and Verilog code accurately to complete PID circuit scheme. Thirdly, we use the software method to realize the serial communication of FPGA side to realize data interaction between FPGA and Ipcop. Lastly, through the analyzing of simulated waveform, we verify the correctness of the PID algorithm. At the end of this article, we compared real-time values calculated by PID algorithm with the actual values by using the C programming language and Logic Analyzer to verify the functions of serial communications.

II. PID AQM ALGORITHM AND PID AQM FOR FPGA IMPLEMENTATION

AQM method of PI on the basis of the linear control theory controller is proposed by C.Holloton. The theoretical basis for the AQM research is the control theory introduced into the AQM system. PID algorithm

was introduced into the AQM mechanism, on the basis of the PI algorithm, shorter the time of queue length reach stable scheduling. Discrete PID expression as shown below,

$$p(k) = p(k-1) + ae(k) + be(k-1) + ce(k-2)$$
 (1)

When the PID controller is applied to the field of congestion control, q(k), p(k), and e(k) (e(k)=q(k)-qref), respectively denote the instantaneous queue length, drop probability, deviation, and qref represents expected queue length. The principle diagram of PID algorithm is shown in Fig. 1. Obviously, the controller consists of one subtraction module, one add module, four delay modules (D in Fig. 1), and three multiplier modules.

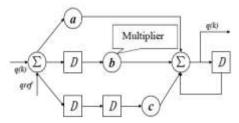


Figure 1. Discrete PID principle diagram

This design uses FPGA chip of Altera CycloneII EP2C35F484C6 series. The next work is to realize modules above.

A. The design of the Subtraction Module

Fist we write Verilog code of subtraction module, The main Verilog code of subtraction are shown in Fig. 2, and then compiler it to generate symbol file to realize the subtraction module. The purpose is to generate visual circuit diagram.

```
module qinout (tx_clk,qin,out)
input tx_clk;
input [31:0] qin;
output [31:0] out;
reg [31:0]out;
always@(negedge tx_clk)
Begin
qout [31:0]= qin[31:0];
out=qout-200;
endmodule
```

Figure 2. The main Verilog code of subtraction modules

B. The design of the Add Module

Four-terminals add module, as shown in Fig. 3, was established by using the LPM macro module of Quartus II software. And then we get symbol files of add module.

C. The Design of Delay Module

Delay module was established by using the Verilog non-blocking assignment statements. By outputting the result e(k), which was calculated by delay module, in the next one or two clock can we obtain e(k-1) or e(k-2) delay signal. The main Verilog code of delay module are shown

in Fig. 4. Then compiler to generate symbol file to realize the delay module.

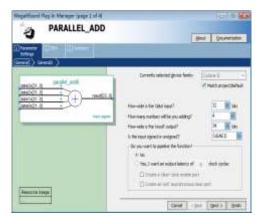


Figure 3. The LPM module of add module

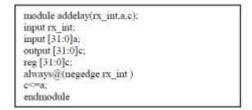


Figure 4. The Verilog code of delay modules

D. The Design of Multiplication Module

Multiplier module realized by calling the LPM macro module, as shown in Fig. 5, of which - 2226 is coefficient of *b*, and then we get symbol files of multiplier module.

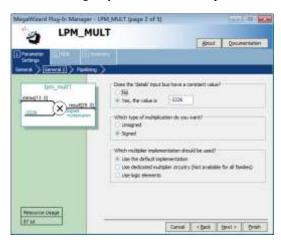


Figure 5. The LPM module of multiplier module

The next work is to link the symbol file of modules above together according to the diagram of the PID algorithm, as shown in Fig 6, we can obtain engineering principle diagram of the PID algorithm.

The next step is to build communication between the FPGA and the router in the FPGA hardware side.

III. THE IMPLEMENTATION SCHEME OF SERIAL COMMUNICATION FUNCTION IN FPGA SIDE

The Function of the FPGA-side is responsible for receiving data of q (instantaneous queue, q) send by router, and sending the data of p (packet loss rate, p)

calculated by PID algorithms to router. Serial communication uses Nios II soft-launched core processor [9]. First of all, set up a SOPC system on FPGA. FPGA can realize serial communication when the SOPC hardware platform and software drive program are completed.

A. The Hardware Development of SOPC System

This design uses the strongest type of Nios II/f to realize CPU module, as shown in Fig. 7. The CPU module is the heart of the system.

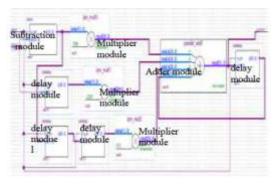


Figure 6. Engineering principle diagram of the PID algorithm



Figure 7. CPU module

Due to the limited capacity of the internal RAM on FPGA, SDRAM memory stores need to be established to storage larger program. The method of establishing a SDRAM controller is shown in Fig. 8. The SDRAM module provides enough storage space for the SOPC system.



Figure 8. SDRAM controller

PI/O modules provide the data interaction with the outside world for the SOPC system. We create three PI/O (Input Parallel Output, PI/O) modules, the two of which are 32-bit PI/O, the one of which is 1-bit PI/O. 32-bit PI/O modules established is shown in Fig. 9.



Figure 9. PI/O module

B. The Driver Software of SOPC System

According to the steps above of building hardware system, the next work is to undertake the corresponding software to drive the hardware system. For example, computer cannot work without operating system software, operating system software, etc.

UART and PI/O are peripherals in the Nios II system and be regarded as external architecture to complete the relevant functional operation. The main code of uart.c file are shown in Fig. 10.

```
*send data
int uart_send_byte(unsigned char data)
   UART->TXDATA.BITS.TRANSMIT_DATA=
       While(!UART->STATUS.BITS.TRDY);
  return 0:3
 viod wart_send_string(unsigned int len.ubsigned
char *str)
 ( While(len-)
       {uart send byte(*str++) }}
*receive data
static void uart_ISR(void)
    while(!(UART->STATUS.BITS.RRDY));
     uart.receive_buffer[uart.receive_count++]:
      UART->RXDATA.BITS.RECEIVE_DATA:
       if(4==nart,receive_count)
          (uart_receive count=0;
               uart.receive_flag=1;}}
```

Figure 10. The main code of Uart.c file

```
#ifindef UART H_
#define UART H_
#include"../inc/sopc.h"
#define BUFFER SIZE 4
typedef struct{ unsigned int receive_flag:
    unsigned int receive_count;
    unsigned int receive_count;
    unsigned receive buffer[BUFFER_SIZE]:
    int(*send_byte)(unsigned char data);
    void(*send_byte)(unsigned char data);
    void(*send_byte)(unsigned int len,
    unsigned char *sfr);
    int(*init)(void);
    unsigned int(*baudrate)(unsigned int baudrate);
    )UART_T;
    extern UART_T uart;
#endif/*UART_H_*/
```

Figure 11. The main code of Uart.h file

The header files of SOPC system and header files of serial have different functions. The header file of SOPC is automatically generated by the system and defines related peripherals of hardware structure. The header file of serial communication defines functions related to the operation of the serial. The main code of Uart. h file are shown in Fig. 11.

The main. c files contain information about the external hardware operation. The main code of main.c file as shown in Fig. 12. Specific information as follows,

Receive data q;

Send data q to the PID algorithm hardware circuit; Drive PID algorithm Hardware circuit;

Accept the outcome data p of the operation, etc.

```
int main(void)
[unsigned char buffer[4];
uart init();
unsigned long int q.k.z;
while(1){ if(uart receive flag)
(memcpy(&q.uart receive buffer,4)
LED->DATA=0;
p=INDATA=0;
p=INDATA=0 bafg=0;
uart receive flag=0;
memcpy(buffer,&p,4);
uart send string(sizeof(buffer),buffer);
}
return 0;
}
```

Figure 12. The main code of main.c file

Nios II is soft-core processor, and how to implement the interface between the Nios II and PID algorithm hardware circuit is a key issue. As shown in Fig. 12, this design uses a PI/O and adds assignment statements to the PI/O port to drive level jump to generate falling edge that can be used to drive PID algorithm hardware circuit.

The next step is to build communication between the FPGA and PID algorithm.

IV. FPGA SERIAL JOINT WITH PID ALGORITHM

After receiving q that sent by the router, FPGA sent it to the PID algorithm modules to compute to get the drop probability p. The q values are assigned to a defined 32-bit PI/O module through software programming. The link way is shown in Fig. 13.

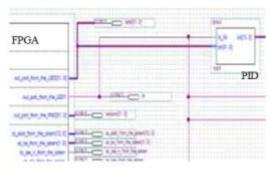


Figure 13. The receive port module with PID algorithm

The p (drop probability,p) values calculated by the PID algorithm module will be sent back to the serial communication module. And in the next step p will be sent to the router. The p values are assigned to a defined 32-bit PI/O module through software programming. The link way is shown in Fig. 14.

By the design ideas of this article, as long as the router uses serial communication for data exchange with the FPGA side , and only the PID algorithm is embedded into the router can we achieve effective network congestion control.

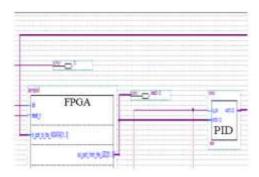


Figure 14. The send module with PID algorithm

V. FUNCTIONAL SIMULATION ANALYSIS

This design uses the Quartus II software for simulation analysis. Related simulation parameters [3] as follows: CLK is 50 MHZ, and in the PID algorithm formula, a = 0.0000139; b = 0.00002226; c = 0.00000873; the value of qref = 200 packets.

Fig. 15 records the simulation data in period of 220ns to 460ns. In Fig. 15, *CLK* represents the system clock signals, rising along effectively, and its frequency is 50 MHZ; *q* is the instantaneous queue length of input signal; *e* is the deviation; *ae* represents *ae* (*k*) which equals 1390; *delay1* represents deviation *e* that delays one clock cycle and *delay1* equals e(*k*-1); *be_1* represents *be*(*k*-1) and it equals -2226; *delay2* represents deviation *e* that delays two clock cycles and *delay2* represents e(*k*-2); *ce_2* is *ce*(*k*-2) and it equals 873; *p* is output signal of discarding probability.

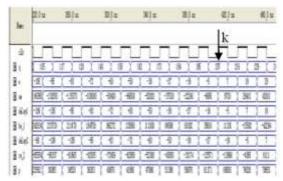


Figure 15. PID algorithm simulation

Check the accuracy of delay module. On arrival of rising along signal at the k moment, signal q equals 207. Because e(k) = q(k)-200, so e(k) equals 7. Similarly, at k-1 moment, q(k-1) = 195, e(k-1) = -5; At k-2 moment, q(k-2) = 184, e(k-2) = -16. In Fig. 15, delay1(k) equals -5 which represents the deviation e(k) outputs one clock cycle later, Similarly, delay2(k) equals -16. Therefore, the delay module realized delay function.

Check the accuracy of multiplier module. When arrival of rising along signal at the k moment, according to the PID algorithm formula ae(k) = 9730, be(k-1) = 11130,

ce(k-2) = -13968. In Fig. 15, we can see that functional simulation results are correct, so the design of multiplier module is correct.

Check the accuracy of discard probability p. When arrival of rise along q(k) is 207 at the k time, according to the formula(1), p(k) = 68063. In Fig. 15, at k moment simulation results p(k) = 68063, so discard probability is correct in this design.

The related parameters for FPGA to realize the PID AQM algorithm were set the same as the function simulation experiment. In order to verify the operation of the drop probability p values of the PID algorithm in the FPGA that correspond to the p values are measured in the real network environment. The p values and q values [10] of PID queue management algorithms in the real router as shown in Table I. Logic analyzer can observe p values in real-time. Four moment values are shown in Fig. 16. and then compares p values with the data in Table I. Because the coefficient a, b, c are expand 10^8 times, so Logic analyzer sample get p values differ with the real p values in Table I also 10^8 times.

TABLE I. IN THE ACTUAL OUTER Q AND P VALUE TABLE OF PID ALGORITHM

q		128	207	250	250
p	0.0025948	0.0005138	0.0006806	0.0009122	0.0008
	0	8	3	7	5220
q	102	91	95	105	117
р	0.0002174	0.0001243	0.0002356	0.0003008	0.0003
	5	2	2	5	4520

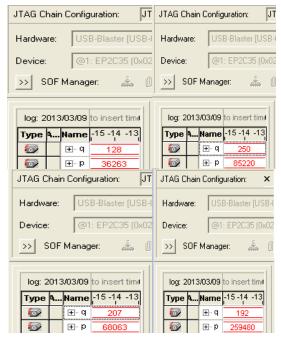


Figure 16. P value observed by the Logic analyzer

To see more data, we can use C language: "printf (" \ n q = % d", "p = % d \ n", q, p);" real-time outputs discard probability p values and the corresponding q values, as shown in Fig. 17. The router receive the p values calculated by the FPGA, Since the coefficient a, b, c expand 10^8 times, compared them with the data in Table

I, so divide the same 10^8 times can we obtain real p values as in Table I

Figure 17. Using C language to observe the FPGA end result of PID algorithm p

This design follow the FPGA top-down design process strictly. Through logical analysis, we get the whole project hardware resource consumption as shown in Fig. 18. We can see from Fig. 18, This design consume a few number of hardware resource of FPGA, So FPGA can completely meet the design. Compared Fig. 18 with Ref [8], PID AQM algorithm occupy less FPGA hardware resources than RED algorithm.

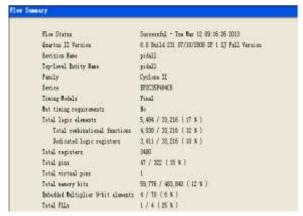


Figure 18. The hardware resource consumption of FPGA

VI. CONCLUSION

This paper proposes a kind of project which is about FPGA hardware implementation scheme for PID AQM algorithm, which improves the efficiency and accuracy of AQM and provides a prerequisite for the PID AQM algorithm fully implemented and accurate operation. This paper solved the problem of FPGA side module that drives PID algorithm module and serial communication, Laying the foundation for PID AQM algorithm implementation.

ACKNOWLEDGMENT

The authors are grateful to the anonymous referees for their valuable comments and suggestions to improve the presentation of this paper.

REFERENCES

- [1] X. Chang and J. K. Muppala. "The effects of AQM on the performance of assured forwarding service," *computer Communications*, vol. 29. pp. 2294-2307, 2006.
- [2] X. G. Li. "Algorithm Research of AQM Based on Congestion Control," Computer Technology and Development, vol. 17. pp. 199-202, 2007.
- [3] J. H. Chen and S. J. Sun. "Intelligent PID congestion control algorithm," *Journal of JiLin University* (*Information Science Edition*), vol. 22. pp. 373-377, 2004.
- [4] X. P. Yang, C. F. Song, J. W. Sun and X. J. Wang. "Simulation and Implementation of Adaptive Fuzzy PID," *Journal of Network*, vol. 9. pp. 2575-2581, 2014.
- [5] Y. L. Kang, X. Z. Lin and J. G. Li. "PID Control System Design Based on AVR Microprocessor," *journal of HeNan Normal University*, vol. 38. pp. 180-182, 2010.
- [6] H. G. Yang, J. B. Sun and W. Wang. "An Overview to FPGA Device Design Technology," *Journal of Electronics & Information Technology*, vol. 32. pp. 714-727, 2010,.
- [7] Marami Bahram and Haeri Mohammad. "Implementation of MPC as an AQM controller," *Computer Communications*, vol. 3. pp. 227-239, 2 009.
- [8] X. P. Yang, N. Zheng, H. Chen and Y. J. Wang. "FPGA hardware implementation scheme for AQM algorithm," *Journal of jilin university*, vol. 43. pp. 472-479, 2013.
- [9] K. Sun. "Nios2 system development design and the application example." *BeiJing: BeiJing University of aeronautics and astronautics press*, 2007.
- [10] H. Guan, X. P. Yang and H. Chen. "Implementation of software router embedded PID AQM algorithm in the

communication test network," *Computer, Mechatronics, Control and Electronic Engineering (CMCE)*, pp. 254-257, 2010.



Xiaoping Yang received the B.E. and M.E. degree in Detection Technology and Instrument and Communication and Electronic System from Jilin University of Technology, Changchun, China, in 1985 and 1988, respectively, and Ph.D. degree in Control Theory and Control Engineering, Jilin University, Changchun, China in 2007. She is a

professor at College of Communication Engineering, Jilin University, China. She has been committed to teaching and research including communication, signal and information processing for years. Her current research interest includes optimization control of data transmission and congestion control in communication network.

Xiaojing Wang is currently a postgraduate and working towards her M.E. degree in Jilin University, China. Her current research interest includes active queue management algorithm and simulation implementation of AQM algorithm.

Te Di is currently a postgraduate and working towards his M.E. degree in Jilin University, China. His current research interest includes active queue management algorithm and simulation implementation of AQM algorithm.

Chaos Character of Test Data Processing in Complex Wireless Sensor Network

Changjian Deng

(1 Department of Control Engineering, Chengdu University of Information Technology, Chengdu, China. 2 School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu, China.)

Abstract—when a large scale WSN is deployed in equipment condition monitoring system, some parameters of network have complex properties. If parameters were related to data processing in system diagnosis, these characteristics need be analyzed. The paper using Logistic map discusses the chaos character of the delay time in complex WSN. And presents the caculation precise of delay time related data is a function of complexity degree of WSN. And meanwhile the method of controlling complexity of network is discussed. Simulation shows the availability, practicality of analysis in the paper.

Index Terms—chaos, complexity degree, error bound, temperature varying rate, Logistic map

I. INTRODUCTION

An important property of networked systems (for example, a large scale wireless equipment condition monitoring system) is their data accuracy and reliability, but they may go bad by disorder data, missing data, different delay time and so on.

This phenomenon is very complex, it is reasonable to propose that the chaos may occur when network is complex enough. For example, some parameter in the complex network have two important property of a chaotic dynamical system, that is (a) everywhere locally unstable (positive Lyapunov exponent) and (b) globally mixing (positive entropy). And it is well known, that chaos often occurs if the Lyapunov time is significantly shorter than the observational time(for example, the time by which a phase space region initially comparable in size to the observational accuracy extends across the entire accessible phase space). A similar thing is in a complex network, one can never predict what delay time a node (transmitting data) may have. Paper studies the chaos in data processing in WSN.

Important features of chaotic signals in WSN, such as: pseudo-randomness, sensitivity to initial conditions and parameters of the system, are studied in paper [1], [2], [3], [4].

Some researchers study using chaos in physical layer, data link layer, network layer, and so on. For example, Nakano studies chaos synchronization-based data transmission scheme in multiple sink wireless sensor networks [5]. Qian Fang proposes a secure cluster routing protocol based on chaotic encryption as well as a conventional symmetric encryption scheme [6]. Berber [7] uses chaotic sequences in direct sequence spread spectrum systems in physical layer in wireless sensor

networks. But as the motivation of wireless monitoring system, how the chaos characteristic influence the test data processing has not been studied enough by now.

Meanwhile some researchers study robustness performance of chaotic test data in complex network. For example, Saito investigates a coupled map network using multicanonical Monte Carlo method [8]; Kauffman [9] introduced the Boolean network model and proposed the hypothesis some systems prefer the edge of chaos because it allows systems to have complex behaviors. The related arithmetic includes Markov chain Monte Carlo method, power series or polynomial chaos expansions, Stalled predictive feedback control (PFC), and so on [10] [11]. But the related arithmetic of chaotic test data processing is very less.

Paper focuses on chaos character of the test data processing in complex wireless sensor networks. Contributions of paper include using chaos character to analyze some network parameter, present the concept of complexity degree to build the relation between the complexes of network with the chaos character. And propose the error bound produced by complexity degree of network is proposed.

The paper is organized as follows: section II gives a brief review of chaos map in WSN. Section III presents the chaos occur in a complex WSN network. Section IV presents the theory foundation for it. In Section V, the error bound of data processing is presented. The simulation results are presented in section VI.

II. CHAOS MAP AND ITS APPLICATIONS IN WSN

A. PWLCM map

Mansour [12] uses the PWLCM (Piece Wise Linear Chaotic Map) function in security communication of WSN. PWLCM uses very few multiplications/divisions operations and several additions/comparisons for each digital chaotic iteration.

PWLCM has following dynamical properties:

- (i) Uniform and invariant density,
- (ii) Exponentially decayed correlation function
- (iii) Simple hardware and software realization and implementation.

PWLCM is a map composed of multiple linear segments and it is given by equation (1).

$$x(n) = F[x(n-1)]$$

$$= \begin{cases} x(n-1) \times \frac{1}{p}; & \text{if } (0 \le x(n-1) < p) \\ [x(n-1) - p] \times \frac{1}{0.5 - p}; & \text{if } (p \le x(n-1) < 0.5) \\ F[1 - x(n-1)]; & \text{if } (0.5 \le x(n-1) < 1) \end{cases}$$
(1)

Such as $x(i) \in [0,1]$ and $p \in [0,0.5]$.

B. a chaos-based data gathering scheme

Nakano [13][14] uses a chaos-based data gathering scheme. Its chaos map is descried as below: a WSN is consisting of M wireless sensor nodes. Each wireless sensor node $S_i (i=1,...,M)$ has a timer which controls transmit and receive sensor timing information. The timer in S_i is characterized by an oscillator having two internal states X_i and Y_i , a non-negative integer distance level l_i , and an offset time O_i . Basic dynamics of the timer in S_i is described by the following equation:

$$\frac{d}{dt} \begin{bmatrix} x_i(t) \\ y_i(t) \end{bmatrix} = \begin{bmatrix} \Delta_i & \omega_i \\ -\omega_i & \Delta_i \end{bmatrix} \begin{bmatrix} x_i(t) \\ y_i(t) \end{bmatrix}$$
 (2)

For x(i) < 1, and $x'_{N(i)}(t) < 1$

$$\begin{bmatrix} x_i(t^+) \\ y_i(t^+) \end{bmatrix} = \begin{bmatrix} q_i \\ y_i(t) - p_i(x_i(t) - q_i) \end{bmatrix}, \text{ if } x_i(t) = 1$$

$$\begin{bmatrix} x_i(t^+) \\ y_i(t^+) \end{bmatrix} = \begin{bmatrix} q_i \\ y_i(t) - p_i(x_i(t) - a_i) \end{bmatrix}, \text{ if } x'_{N(i)}(t) = 1$$

$$N(i) = \left\{ j \mid l_i < l_i, \left| S_j - S_i \right| < radio_range \right\}$$
 (5)

Where Δ_i , ω_i , p_i , q_i , a_i denote a damping, a self-running angular frequency, a slope, a base state for self-firing, and a base state for compulsory-firing.

C. Lorenz system

Paper [15] use the deterministic Lorenz system (6) studies the phenomenon that chaos is contained in data of WSN.

$$\dot{x} = s(y - x)
\dot{y} = Rx - y - xz
\dot{z} = -bz + xy$$
(6)

It chooses parameters (s,R,b) = (10,240,8/3), where the system has a stable periodic orbit γ . The return map σ to the cross-section Σ defined by z=239 has a fixed point at its intersection with γ , located at (x,y,z)=(29.7754,69.2266,239).

The stochastic Lorenz system is the SDE defined by

$$dx = s(y - x)dt + gdW$$

$$dy = (Rx - y - xz)dt + gdW$$

$$dz = (-bz + xy)dt + gdW$$
(7)

Where g is a constant that determines the amplitude of the Brownian term dW.

D. Logistic chaos

Logistic map (9) in WSN is mostly in researches [16] [17].

$$x_{n+1} = \mu x_n (1 - x_n), x \in (0,1), \mu \in [0,4]$$
 (8)

When $\mu \ge 3.5699$, the sequence generated by this map will be chaotic.

Anther Logistic map is

$$x_{n+1} = 1 - \mu x_n^2, x \in (-1,1), \mu \in [0,2]$$
 (9)

In order to enlarge the data range of xi, the extend Logistic map is like (10).

$$x_{n+1} = \mu x_n (N - x_n / m) / N, x \in (0, mN), \omega \in (0, 4)$$
 (10)

Where $N = 2^k$, $m = 2^k$, K and k are both integers. When $\mu \ge 3.569$, the sequence generated by this map will be chaotic.

An example code to produce Logistic map is like below.

"
$$x(1,1)=a*x0*(1-x0);$$

for i=2:n
 $x(i,1)=a*x(i-1,1)*(1-x(i-1,1));$
end"

E Chebyshev Degree-L maps

(3)

Berber [7] uses Chebyshev degree-L maps in their chaos physical design of WSN. It is defined by this recursive equation

$$c(k+1) = \cos[L\cos^{-1}c(k)]$$
 (11)

Where discrete random samples in the sequence $c_{11}(k)$ have the values between -1 and +1. If L=2 the logistic map is obtained, and with L=3 the cubic map is defined. The discrete random samples will be called chips. The density function of these maps is

$$c_{i1}(k+1) = \cos[L\cos^{-1}c_{i1}(k)] \tag{12}$$

III. THE CHAOS OCCUR IN A COMPLEX WSN NETWORK

A Formation of problem: Data map to chaos

Suppose in WSN, data transmitting in the network include measure data and network manager data.

Then there is non linear observation univariate time series x(n), supposing the delay time of communication is almost varying, and then it creates a multivariate time series (13).

$$X(n) = (x(1), x(2 + \Delta), ..., x(n + \Delta'))$$
 (13)

In fact, a complex WSN, when nodes loss its connection or be attacked randomly, n and Δ is problematic.

There are two kinds of measurement data processing in the network, one is that the network characteristic data (test sequence, time ,and so on); another is the real time test data. And obviously the next one is the special status of the first one, when the characteristic data is data itself. Let look the temperature test data, the temperature itself is parameter, in commonplace, the change rate of temperature is also a parameter.

In applications, transmitting data often use test sequence number to ensure the order and delay of test data.

Then:

$$X_i(n) = (x_i(1), x_i(2+\Delta), ..., x_i(n+\Delta')); i \in \{1, 2, ..., n\}$$
 (14)

And
$$if(x_i(k) - miss), x_i(k + w \cdot \Delta) = 0; k \in \{1, ..n\}$$
 (15)
 $else, x_i(k + w \cdot \Delta) = x_i(k);$

First, let look how chaos can be occurred in complex WSN.

In Internet work, there are some researches that have indicated that the network delay time series process existed chaotic identity [17][18].

To describe the delay in complex WSN, here use a Logistic map to analyze its complex properties.

$$x_{n+1} = \mu x_n (1 - x_n), x \in (0,1), \mu \in [0,4]$$

Here x=1 means the time interval of transmitting data in one node. Even there may have different transmitting rate, the '1' is normalization value.

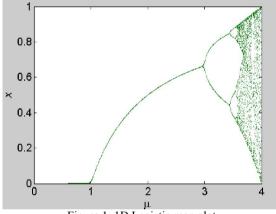


Figure 1. 1D Logistic map plot

Suppose the complexity of network is control by parameter μ , in a certain range, the larger the value, the more complexity of network.

Firstly, when μ is smaller than 1, its means the network is always in its normal operating status, the ideal mean delay time is often supposed to be zero.

Secondly, when μ is larger than 1 but smaller than 3, its means the network have collisions, the multi hop delay the transmitting time, and even some simple faults occurs, make the route of network become long and time cost. But the delay time may be determined by the network status, so the delay time becomes large. For example, it is varies from 0 to 0.6.

Thirdly, when network is more complex, even the node failure is random or period, then the more delay time in a network may exist, even in same network when network is complex. When μ is larger than 3.

In the end, when μ is larger enough, the chaos may occurs. The simple explanation for it is if there is some delay time (x value) can not be achieved, then we can always change the route path of the node to cover this value, when the network were complexity enough.

And more import one is there may be no effective way to calculate its statistic value (16) for delay time. For even although its time average value may be certain, its relationship with control parameter λ is complicated; and its space average value is uncertain due to its different initial node place.

$$\overline{x} = \lim_{N \to \infty} \frac{1}{N} \sum_{i=1}^{N} x_i \tag{16}$$

So, if the delay time is uncertain, even we know the physical or empirical model for a test dynamic parameter (have relationship with time), we can not calculate the best precise value of the parameter, even more, we cannot predict precisely with long time term value with a certain model in some simple network.

Here we get theorem 1.

Theorem 1: the network parameters can have the chaos character in a complex WSN network.

Its prove is in subsection B.

B Some Data map produce chaos

As discussed in Section II, (c) to (e), the network transmitting itself may can be chaos, as the character parameter estimation have relationship with time, there may produce global chaos in the network.

The phenomenon in subsection A and B, have blow properties.

Prove (of theorem 1)

(1) It is sensitive to initial conditions;

These mean in complex WSN, may have multi chaos in it, what chaos the nodes may have, it is determined by its initial value.

That is, if D is compact space: $f: D \rightarrow D$ is map, and satisfy

$$\exists \delta > 0, \forall x \in D, \forall U \in \mathfrak{I}(x), \exists y \in U, \exists n \in N,$$

$$\Rightarrow \rho(f^n(x), f^n(y)) > \delta$$
 (17)

(2) It is topologically mixing:

Topological mixing (or topological transitivity) means that the system will evolve over time so that any given region oropen set of its phase space will eventually overlap with any other given region.

That is,

$$\exists x_0 \in D \Rightarrow \overline{Orb_f(x_0)} = D, and,$$

$$Orb_f(x_0) = \{ f^n(x_0) | n = 0, 1, 2, ... \}$$
(18)

(3) It is dense periodic orbits. $\overline{Per(f)} = D$

A dense periodic orbit means that every point in the space is approached arbitrarily closely by periodic orbits.

Like Logistic map and other maps.

And follow the (1), (2), (3). We get there may have Devaney Chaos in complex WSN.

C. complexity degree v.s chaos

It is well known, if a system have two important property, that is (a) everywhere locally unstable (positive Lyapunov exponent) and (b) globally mixing (positive entropy), it may be a chaotic dynamical system.

Basis on network operating parameter, for example, delay time, throughput, Qos of WSN and so on, we define

complexity degree of WSN network. First, the delay time in network should be defined.

Definition 1: Simple Delay time. In a not congesting network, when transmitting x byte data, simple delay time is equal to

$$T_{simple_delay} = a \cdot x + b \tag{19}$$

Where 'a' stand for the delay needed for sending 1 data byte, parameter b is the time needed for the protocol overhead for sending one packet.

Definition 2: Multi value Delay time. In a normal network, there are often connection changes by random attack or random lose function of nodes. So the delay time of node may be multi value.

$$T_{w_delay} = \begin{cases} \sum_{i=1}^{k_w} a \cdot x_i + b & \text{where } w \in (1, 2, ..., n) \\ ax + b & \end{cases}$$
 (20)

Definition 3: here define w is complexity degree of network.

Obviously, longer run time and different placed network have more \boldsymbol{w} value. Complex topology network have big complexity degree value than simple topology network.

So owing to the definition of chaos, we get Lemma 1.

Lemma 1: The chaoses occur only when complexity degree is approach infinity. And more fragile network the more difficult the chaos produces.

D. error bound of estimation of test data under the complexity of network

Unlike normal signal processing, in WSN data process, often deal with characteristic data of signal processing.

So it is important to discuss a limited value (error bound) like Fisher information matrix for statistic data, even in chaos character in these data.

(1) Demand Analysis: calculated in fields node v.s. calculated in monitoring center.

TABLE I.
THE ADVANTAGE AND DISADVANTAGE OF DIFFERENT CALCULATE

	Calculate in fields node	Calculate in monitoring	
		center	
advantage	If physic or empirical	More reliable and	
	model were simple and	global precise than first	
	clear, calculation is more	method, but network	
	simple and precise.	topology varying may	
		be one influence factor.	
disadvantage	Add communication data	Network time base is	
	length, clock or timer drift	complex	
	may make calculate not		
	inconformity of all nodes		
	in the network.		
If missing	Not easy to recover	Hard to recover.	
data	-		
note	Distribute optimal	Global optimal	
	Fault diagnose and		
	prognosis are often		
	complex		

For example, the temperature change rate and vibration change rate, is calculated in the fields. But they can not achieve global optimal for its limited data store and processing ability. The compare of caculation in fields node wth calculated in monitoring center is shown in

Table.1. So how to control the complexity of network should be analyzed in some case.

(2) Model of test data

Sensor nodes i at time t, its observation is

$$z_i(t) = C_i(t)\theta^* + v_i(t), t = 0,1,...$$
 (21)

 $z_i(t) \in \Re^{n_i}$ is observation vector, $v_i(t) \in \Re^{n_i}$ is random noise. θ^* is the parameter need be test.

 $C_i(t) \in \Re^{n_i \times n}$ is time varying observation matrix.

(3) Model of communication in complex network

$$r_i(t + \Delta_i) = z_i(t), t = 0,1,...$$
 (22)

$$f_i(r_i(t),t_i) = d[(r_{i+1} - r_i)/(t_{i+1} - t_i)]$$
(23)

 $r_i(t)$ is receiving data, Δ_i delay time. $f_i(r_i(t), t_i)$ is a model of time processing function.

(4) Influence factors

Influence factors include time base drift, hop number, network reform, topology manager, and so on.

$$\frac{\partial r_i}{\partial t} = \frac{\partial r_i}{\partial T_{i_draf_t}} \frac{\partial T_{i_draf_t}}{\partial t} + \frac{\partial r_i}{\partial T_{i_hop}} \frac{\partial T_{i_hop}}{\partial t} + \dots$$
 (24)

(5) Error bound

As discussed in (3) and (4), the error bound is very complex, here use complexity degree to describe its error bound.

Lemma 2: if w is complexity degree of network, a function which have chaos related parameter (for example delay time in this paper), then the error bound of the function is proposed to be proportion with $\underline{w-1}$.

IV THE SIMULATION AND RESLUTS

In wireless equipment condition monitoring system, the temperature test is simple and import one. Without loss of generality, a heating and cooling process is often gone with the temperature change of metal container. In these applications, the limit (highest and lowest) temperature value is no meaning, but the plot and the heating rate are useful to monitor its operating status.

To model the heating process, here give a plot as (25).

$$T = \begin{cases} 3+1.5 \times t; t \in [0-10] \min utes \\ 18+0.467 \times t; t \in [18,40] \min utes \\ 32+0.75 \times t; t \in [40,80] \min utes \\ 62+0.09 \times t; t \in [80,90] \min utes \end{cases}$$
 (25)

Without loss of generality, the temperature test process has Gauss noise, with different sensor, the precise of temperature is different, but normally the test error can be proposed to be lesser than 10%.

In simulation, here use formula (21), the field test plot and model analyze picture is to be drawn like figure.2.

The color bold figure is test error may be existed in the fields and green thin line is physic model as formula (25).

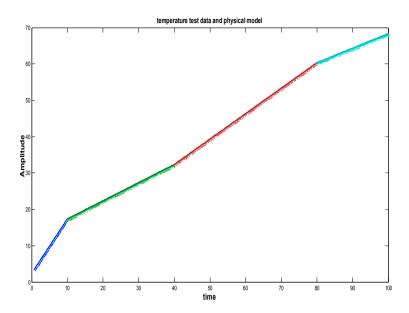


Figure 2. A simple metal containment heating processing and its normal test data

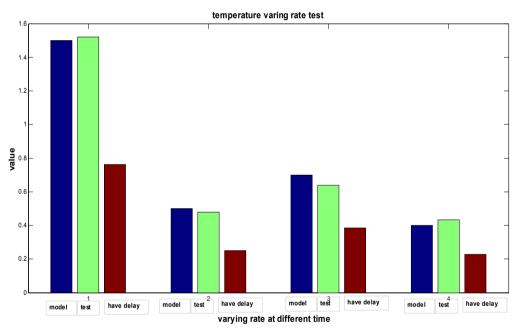


Figure 3. The simulation of temperature varying rate test in WSN (worst case)

Then the data should be transmitted wirelessly. The transmitted data may be temperature and temperature chance data. Without loss of generality, here use a worst changed delay time to calculate the heating rate in the figure (2). Here use Matlab to simulate delay time.

In figure 3, the blue one is model value as (25), the green one is received temperature rating data, and the red one is the data that receive temperature data and temperature rating data is calculated by the network delay time. It shows that if the network has complexity degree of two, the worst error may be 0.5.

Obviously, in a simple model, and when the throughput of network is small, calculating data at the fields and transmitting temperature chance data is more precise.

So, when the physic or empirical models were complicated, (1) the multi sink or micro network should be used to decrease complexity of network; (2) spatial redundancy should be used if complexity of network should not be added.

V CONCLUSIONS

The phenomenon that chaos occurs when nodes loss its connection or be attacked randomly in very complex WSN are discussed in the paper, for example, the delay time is one parameter that may have chaotic character.

Based on this, the paper has gained following results:

(1) The network parameters can have the chaos character in a complex WSN network.

- (2) The chaos occurs only when complexity degree is approach infinity. And more fragile network the more difficult the chaos produces
- (3) If w is complexity degree of network, a function which have chaos related parameter (for example delay time in this paper), then the error bound of the function is proportion with $\frac{w-1}{w}$.

w

ACKNOWLEDGMENT

This article is supported by "China-Canada Joint Research Project" (project number: 2009DFA12100) and Major Project of Education Department in Sichan (13Z204). For support, contributions to discuss we would like to thank the rest of MCC(mobile computer center) of UESTC.

REFERENCES

- [1] T. Xiang, X. F. Liao, and P. Guo. "A novel block cryptosystem based on iterating a chaotic map", *Physics Letter A.*, vol. 349, pp. 109-120, 2006.
- [2] D. R. Frey. "Chaotic digital encoding: An approach to secure communication", *IEEE Trans. Circuits Syst. II*, vol.40, pp. 660–666, 1993.
- [3] G. Guimaraes, E. Souto, D. Sadok, and J. Kelner. "Evaluation of security mechanisms in wireless sensor networks," in Systems Communications, 2005.
- [4] T. Kavitha1, D. Sridharan. "Security vulnerabilities in wireless sensor networks: A survey," Journal of Information Assurance and Security, vol.5, pp. 31 - 44, 2010.
- [5] H. Nakano, A. Utani, A. Miyauchi, H. Yamamoto. "Data Gathering Scheme Using Chaotic Pulse-Coupled Neural Networks for Wireless Sensor Networks," *IEICE Trans. Funds.*, vol.E92-A, 2, pp459-466, 2009.
- [6] Q. Fang, T. Jiang, X. Zhao, L. Hou. "A Chaos-Based Secure Cluster Protocol for Wireless Sensor Networks". *Kybernetika*, 2008, 44(4): 522~533.
- [7] S. Berer, S. Feng, "Chaos-Based Physical Layer Design for WSN Applications", Recent Advances in Telecommunications and Circuit Design, 2012, pp.157– 162.
- [8] Nen Saito, Macoto Kikuchi. "Robustness leads close to the edge of chaos in coupled map networks: toward the understanding of biological networks". New Journal of Physics. 15 (2013) 053037. doi:10.1088/1367-2630/15/6/063038.
- [9] Kauffman, S, A. The Origins of Order: Self Organization and Selection in Evolution .Oxford: Oxford University Press. 1993.
- [10] Lehnert J, H"ovel P, Flunkert V, Yu P, Guzenko Fradkov A,L and Sch"oll E. *Adaptive tuning of feedback gain in time-delayed feedback control Chaos*.21,043111, 2011
- [11] Saito N, Iba Y. "Probability of graphs with large spectral gap by multicanonical Monte Carlo Compute". *Phys. Commun.* 2011(5):182-223.
- [12] H. Nakano and T. Saito, "Basic dynamics from a pulse coupled network of autonomous integrate-and re chaotic circuits", *IEEE Trans. on Neural Networks*, vol.13, no.1, pp.92-100, 2002.
- [13] H. Nakano and T. Saito, "Grouping synchronization in a pulse-coupled network of chaotic spiking oscillators",

- *IEEE Trans. on Neural Networks*, vol.15, no.5, pp.1018-1026, 2004.
- [14] L. A. Bunimovich. "Short- and long-term forecast for chaotic and random systems". *Nonlinearity*. 27 (2014) R51–R60.
- [15] L. Kocarev, G. Jakimoski. "Logistic map as a block encryption algorithm". *Phys Lett A* 2001;289:199 206.
- [16] Chen S, Zhong XX, Wu ZZ, Block Chaos. "Cipher for wireless sensor network". Sci China Ser F-Inform Sci 2008;51:1055 - 63.
- [17] C. Li, H. Zhao, X. Zhang, S. Q. Yuan. "Chaotic analysis of Internet traveling time". Computer Science, 2008, 35 (5):85 - 86. (in Chinese)
- [18] LI HongYan , WANG Hong , GUI Chao. Internet time delay prediction based on autoregressive and neural network model. Proceedings of International Conference on Communications, Circuits and Systems [C]. Guilin. Guangxi: IEEE , 2006.1758 1761.

ChangJian Deng was born in 1968. He received his B.S.



degree in Lanzhou University in 1985 and M.S. degree in University of Electronic Science and Technology of China in 2005. He is currently pursuing the Ph.D. degree in University of Electronic Science and Technology of China (major is measure technology and

automation equipment). His research interests include harsh environment equipment, wireless communication and wireless sensor network. E-mail: chengli dcj@163.com

He is also associate professor in Chengdu University of Information Technology.

Instructions for Authors

Manuscript Submission

All paper submissions will be handled electronically in EDAS via the JMM Submission Page (URL: http://edas.info/newPaper.php?c=7325). After login EDAS, you will first register the paper. Afterwards, you will be able to add authors and submit the manuscript (file). If you do not have an EDAS account, you can obtain one. Along with the submission, Authors should select up to 3 topics from the EDICS (URL: http://www.academypublisher.com/jmm/jmmedics.html), and clearly state them during the registration of the submission.

JMM invites original, previously unpublished, research papers, review, survey and tutorial papers, application papers, plus case studies, short research notes and letters, on both applied and theoretical aspects. Submission implies that the manuscript has not been published previously, and is not currently submitted for publication elsewhere. Submission also implies that the corresponding author has consent of all authors. Upon acceptance for publication transfer of copyright will be made to Academy Publisher, article submission implies author agreement with this policy. Manuscripts should be written in English. Paper submissions are accepted only in PDF. Other formats will not be accepted. Papers should be formatted into A4-size (8.27" x 11.69") pages, with main text of 10-point Times New Roman, in single-spaced two-column format. Authors are advised to follow the format of the final version at this stage. All the papers, except survey, should ideally not exceed 12,000 words (14 pages) in length. Whenever applicable, submissions must include the following elements: title, authors, affiliations, contacts, abstract, index terms, introduction, main text, conclusions, appendixes, acknowledgement, references, and biographies.

Conference Version

Submissions previously published in conference proceedings are eligible for consideration provided that the author informs the Editors at the time of submission and that the submission has undergone substantial revision. In the new submission, authors are required to cite the previous publication and very clearly indicate how the new submission offers substantially novel or different contributions beyond those of the previously published work. The appropriate way to indicate that your paper has been revised substantially is for the new paper to have a new title. Author should supply a copy of the previous version to the Editor, and provide a brief description of the differences between the submitted manuscript and the previous version.

If the authors provide a previously published conference submission, Editors will check the submission to determine whether there has been sufficient new material added to warrant publication in the Journal. The Academy Publisher's guidelines are that the submission should contain a significant amount of new material, that is, material that has not been published elsewhere. New results are not required; however, the submission should contain expansions of key ideas, examples, elaborations, and so on, of the conference submission. The paper submitting to the journal should differ from the previously published material by at least 30 percent.

Review Process

Submissions are accepted for review with the understanding that the same work has been neither submitted to, nor published in, another publication. Concurrent submission to other publications will result in immediate rejection of the submission.

All manuscripts will be subject to a well established, fair, unbiased peer review and refereeing procedure, and are considered on the basis of their significance, novelty and usefulness to the Journals readership. The reviewing structure will always ensure the anonymity of the referees. The review output will be one of the following decisions: Accept, Accept with minor changes, Accept with major changes, or Reject.

The review process may take approximately three months to be completed. Should authors be requested by the editor to revise the text, the revised version should be submitted within three months for a major revision or one month for a minor revision. Authors who need more time are kindly requested to contact the Editor. The Editor reserves the right to reject a paper if it does not meet the aims and scope of the journal, it is not technically sound, it is not revised satisfactorily, or if it is inadequate in presentation.

Revised and Final Version Submission

Revised version should follow the same requirements as for the final version to format the paper, plus a short summary about the modifications authors have made and author's response to reviewer's comments.

Authors are requested to use the Academy Publisher Journal Style for preparing the final camera-ready version. A template in PDF and an MS word template can be downloaded from the web site. Authors are requested to strictly follow the guidelines specified in the templates. Only PDF format is acceptable. The PDF document should be sent as an open file, i.e. without any data protection. Authors should submit their paper electronically through email to the Journal's submission address. Please always refer to the paper ID in the submissions and any further enquiries.

Please do not use the Adobe Acrobat PDFWriter to generate the PDF file. Use the Adobe Acrobat Distiller instead, which is contained in the same package as the Acrobat PDFWriter. Make sure that you have used Type 1 or True Type Fonts (check with the Acrobat Reader or Acrobat Writer by clicking on File>Document Properties>Fonts to see the list of fonts and their type used in the PDF document).

Copyright

Submission of your paper to this journal implies that the paper is not under submission for publication elsewhere. Material which has been previously copyrighted, published, or accepted for publication will not be considered for publication in this journal. Submission of a manuscript is interpreted as a statement of certification that no part of the manuscript is copyrighted by any other publisher nor is under review by any other formal publication.

Submitted papers are assumed to contain no proprietary material unprotected by patent or patent application; responsibility for technical content and for protection of proprietary material rests solely with the author(s) and their organizations and is not the responsibility of the Academy Publisher or its editorial staff. The main author is responsible for ensuring that the article has been seen and approved by all the other authors. It is the responsibility of the author to obtain all necessary copyright release permissions for the use of any copyrighted materials in the manuscript prior to the submission. More information about permission request can be found at the web site.

Authors are asked to sign a warranty and copyright agreement upon acceptance of their manuscript, before the manuscript can be published. The Copyright Transfer Agreement can be downloaded from the web site.

Publication Charges and Re-print

The author's company or institution will be requested to pay a flat publication fee of EUR 360 for an accepted manuscript regardless of the length of the paper. The page charges are mandatory. Authors are entitled to a 30% discount on the journal, which is EUR 100 per copy. Reprints of the paper can be ordered with a price of EUR 100 per 20 copies. An allowance of 50% discount may be granted for individuals without a host institution and from less developed countries, upon application. Such application however will be handled case by case.

More information is available on the web site at http://www.academypublisher.com/jmm/authorguide.html.