SOFTENING QUANTIZATION IN BAG-OF-AUDIO-WORDS

Stephanie Pancoast[1,2], *Murat Akbacak*[3]

[1] Speech Technology and Research Lab, SRI International, Menlo Park, CA
[2] Department of Electrical Engineering, Stanford University, Stanford, CA
[3] Microsoft, Sunnyvale, CA

ABSTRACT

The audio component of multimedia data can be crucial for multimedia content analysis. Bag-of-audio-words (BoAW) approach is one of the most frequently used methods to represent audio content in multimedia event detection and related tasks. The method, however, has numerous criticisms, amongst which is the loss of information in the "vector quantization" step which generates word-like units. In this work, we address this issue by employing a soft quantization representation where the distance to the nearest codeword is incorporated into the model, rather than only using the nearest codeword's index as is the case with hard quantization. We explore two techniques for soft quantization and apply it to the BoAW for multimedia event detection. We find the best setup yields a 13% improvement in mean average precision, improving performance for 27 of the 30 video events.

Index Terms: Bag-of-audio-words, soft quantization, multimedia event detection

1. INTRODUCTION

Over the past decade, research on content analysis in user-submitted videos has become an increasingly important area of focus. For this reason, Text Retrieval Conference Video Retrieval Evaluation (TRECVID) has created a multimedia event detection (MED) track. The goal of the TRECVID MED task is to allow users to define their own events and search for that event in a large collection of video clips [1]. Features in the video imagery play a significant role in determining the content; however, the audio component for a video can also be critical. Consider the case of detecting a home run in baseball game videos. Analysis of the frame-level imagery may determine that the setting is a baseball game, but without the capability to detect cheering in the audio, it would be significantly more difficult to discriminate between an uneventful game and one with a home run.

In this paper, we employ a variation of the popular *bag-of-audio-words* (BoAW) approach. In contrast to supervised techniques, which typically require annotations of specific sounds to model multimedia events as is done in our previous work [2], the BoAW method has the advantage of being completely unsupervised with the exception of training the final MED classifier. This method is inspired by the well-established techniques in the text document (*bag-of-words*) and image document (*bag-of-visual-words*) domains and has been recently used for audio document retrieval [3], song retrieval [4], copy detection [5], and MED tasks [6, 7].

The basic BoAW method is illustrated in Figure 1. There are numerous basic variations to the illustrated pipeline, such as the codebook size and classifier parameters. Results from exploring these variations are presented in our previous work [7]. The BoAW approach first generates a set of "words" (referred to as a codebook) via a clustering algorithm. This codebook is then used to quantize the features by replacing each feature with the index of the word it is closest to in the codebook. The histogram is then generated by counting the number of occurrences of each codeword in the document file.

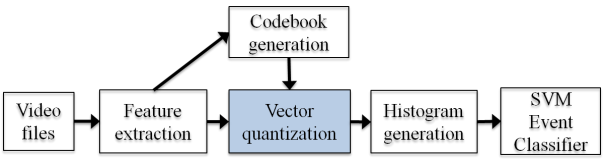

Fig. 1. Diagram of the basic Bag-of-Audio-Words pipeline. In this paper we focus on improvements in the vector quantization step.

This method is similar to the sister BoVW method. In the image and audio domains the words are generated via a clustering algorithm to best represent the original feature space, contrasting from the natural language word units in text documents.

but differs more drastically from the BoW method. When working with text documents, the units are words occurring in natural language while

One critique of the BoVW and BoAW method is that information from the original feature space is lost during the quantization step. Previous work in the image domain has sought to address this issue through what is often referred to as *soft quantization* [8, 9, 10]. The front-end feature space is very different in nature than those used for object classification and other image-related problems. We have also previously discovered that variations found optimal for the BoVW do not necessarily extend to the BoAW [7]. Before applying soft quantization to the BoAW, we therefore should first explore the methods and parameters that best fit the audio domain. To our knowledge, soft quantization has not yet been explored for the BoAW approach.

In our work, we apply soft quantization to the BoAW approach and find noticeable performance improvement over the original hard quantization variation. Section 2 provides an overview of the basic BoAW pipeline. This is followed with a discussion of the two soft quantization variations in Section 3. The experimental setup and results are presented in Sections 4 and 5 respectively. Finally we conclude with a discussion of soft quantization and future work.

2. OVERVIEW OF BASIC BAG-OF-AUDIO-WORDS APPROACH

In this section we provide an overview of the basic BoAW method. Documents, whether written, visual or audio, vary in length. Features representing these documents are often not fixed-length and as a result cannot be used directly with many classifiers. The "bag-of-words" approach resolves this issue by representing the variable-length file with a fixed-length histogram vector. The pipeline for the

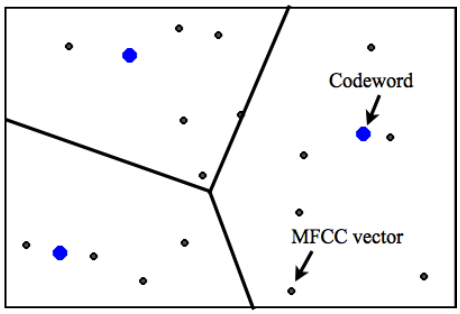

Fig. 2. Codewords and MFCC vectors mapped onto 2-dimensional space with PCA to illustrate the importance of soft quantization. The codewords are represented by the large blue dots while the MFCC vectors by the smaller dark gray dots.

BoAW approach is presented in Figure 1. When words do not exist naturally, like in text documents, a codebook is created using a clustering algorithm. Lloyd (k-means) clustering is a common choice for this step. The centroid of the resulting clusters are taken as the codewords, and the original feature vectors (Mel frequency cepstral coefficients in our case) are replaced by a single index representing the codeword nearest to the original vector. This process is called vector quantization [11]. The "bag" is then created by simply generating a histogram of codewords in the given file. At this point every document is represented by a fixed-length histogram and can be passed to the MED classifier to complete the system. We refer to this as the *histogram vector*. There are numerous variations to the basic pipeline such as the codebook size and classifier parameters which are explored in our previous work [7].

3. PROPOSED EXTENSIONS TO BASIC BoAW

As explained in the previous section, the traditional BoAW algorithm assigns an acoustic feature (MFCCs) to the nearest codeword and adds 1 to the histogram vector for that assignment. However, some features are closer to an assigned codeword than others which is not accounted for in the traditional approach. Consider the data visualization in Figure 2. We performed principle component analysis on the data, and plotted a sample of MFCC vectors (dark gray dots) against the codewords (large blue dots) on the axis of the two principal axis with the highest variance. The boundaries between the three displayed codewords is illustrated by the solid black lines.

One observation from this illustration is that some vectors are closer to the assigned codeword and therefore are a stronger assignment than those further away. *Soft quantization* accounts for this difference. Instead of adding "1" for each codeword assignment, we add some value proportional to the strength of the assignment. Numerous methods for soft quantization are explored for bag-of-visual-words in previous work [8, 9, 10]. We consider two of the methods for their simplicity and computational efficiency, and apply them to the BoAW approach to the MED task. In the remainder of this section we describe the two soft quantization methods in more detail. Here d is the Euclidean distance between the original MFCC vector and the nearest codeword.

3.1. Soft Thresholding

Sparse coding has been found to consistently outperform the traditional hard quantization encoding in BoVW [12, 13]. However, this method is computationally demanding. Authors [10] found the soft thresholding approach to perform on par with sparse coding and require significantly less computation. We therefore chose to examine soft thresholding as one method for soft quantization.

For a k-dimensional codebook, the histogram vector will be of length $2k$ in soft thresholding. If j is the index of the nearest codeword for a given MFCC vector, the update given to the histogram is determined by:

$$h_j \mathrel{+}= max(\alpha - d, 0)$$
$$h_{j+k} \mathrel{+}= max(d - \alpha, 0)$$

The fixed parameter α needs to be selected. Intuition suggests the weight from an MFCC vector should be inversely proportional to d; however in thresholding this is only the case for when $d < \alpha$. If the MFCC feature is further than α from the nearest codeword, the opposite is true. The reason this works is because the histogram vector generated by soft thresholding is then used as the MED feature for the support vector machine (SVM) classifier. The SVM can learn during MED training that greater values for bins one through k and lower values for bins $k + 1$ through $2k$ indicate a strong match with a video event.

3.2. Gaussian Encoding

Another variation for softening the VQ step is to apply a scaling function to d. The scaling function should be monotonically decreasing for increasing d in order to achieve the desired greater weight for stronger assignments. The Gaussian function a common choice for the scaling function and is used by authors [8, 9, 10] for the BoVW approach. We also explore this method when applied to the audio domain and refer to it as *Gaussian encoding*. Gaussian encoding scales the distance to the nearest codeword by the exponential function given by:

$$h_j \mathrel{+}= e^{\frac{-d^2}{2\sigma^2}}$$

Like with soft thresholding, this method also depends on a predetermined parameter (σ).

4. EXPERIMENTS

We ran our experiments using data from the National Institute of Standards (NIST) development set provided for the TRECVID 2011 and 2012 multimedia event detection track [1]. The videos were provided in MP4 format. We extracted the audio components with a sampling rate of 16 kHz. We used two separate datasets: one to select the best performing setup for the soft quantization methods (Set A) and another to do the final comparison (Set B).

We performed experiments using what is referred to as a *verification* or *one-against-all* setup. For each video event, a file is labeled as *in-class* or *out-of-class*. Examples include *Parade* and non-*Parade* as well as *Birthday party* and non-*Birthday party*. A 5-fold cross validation setup was used. Both datasets along with the total number of positive samples are presented in Table 1. Many of the samples do not contain a positive label for any of the video events, and these are accounted for in the "total" count at the bottom of the table.

We used mean average precision (MAP) scores to measure system performance. Precision is calculated as $\frac{tp}{tp+fp}$ where tp is the number of true positives and fp false positives. The output of a classifier often provides a confidence score. A threshold is set to determine the confidence needed to be considered a "positive". Average precision, as suggested by the name, is the average of the precision value at every threshold level. MAP is calculated as the average precision across all video event experiments.

Like in our previous work, [7], we used Mel frequency cepstral coefficients (MFCCs) for the front-end features. The MFCCs are

Event Name	Set A	Set B	Event Name	Set A	Set B
Attempting a board trick	126	160	Doing homework or studying	-	130
Feeding an animal	124	161	Hide and seek	-	132
Landing a fish	90	119	Hiking	-	149
Wedding ceremony	81	123	Installing flooring	-	125
Working on a wood. project	95	141	Writing	-	129
Birthday party	-	172	Attempting a bike trick	-	130
Changing a vehicle tire	-	110	Cleaning an appliance	-	129
Flash mob gathering	-	174	Dog show	-	128
Getting a vehicle unstuck	-	130	Giving directions to a location	-	130
Grooming an animal	-	138	Marriage proposal	-	130
Making a sandwich	-	125	Renovating a home	-	121
Parade	-	138	Rock climbing	-	130
Parkour	-	115	Town hall meeting	-	130
Repairing an appliance	-	139	Winning a race without a vehicle	-	130
Working on a sewing project	-	120	Working on a metal crafts proj.	-	129
			Total	10042	9009

Table 1. Video events for data in Set A and Set B. The number of positive samples and full name for each of the 30 events in shown. The total number of files, including those not belonging to any event, is included in the final row of the table.

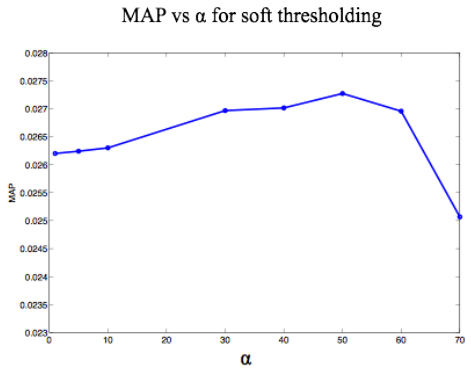

Fig. 3. MAP values when varying α for soft thresholding BoAW on Set A.

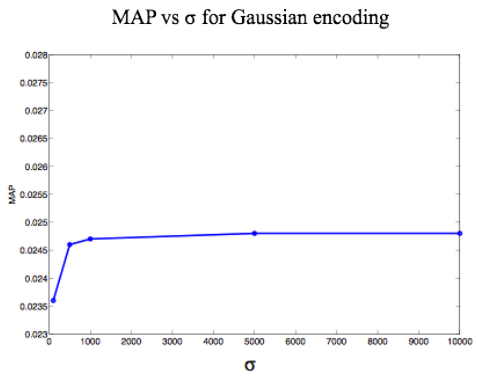

Fig. 4. MAP values when varying σ for Gaussian encoding BoAW on Set A.

computed for every 10-ms audio segment and are extracted using a hamming window with 50% overlap. The features consist of 12 coefficients as well as the log energy. The first and second derivatives of each coefficient as well as the log energy are concatenated with the original features to result in a 39-dimensional feature vector.

We considered L1 normalization as well as no normalization of the histogram vector. L1 normalization is computed as : $x_j \leftarrow \frac{x_j}{\sum_i |x_i|}$. Each term in the histogram vector is divided by the 1-norm of the entire vector. In our previous work we found that, for the basic BoAW setup, no histogram normalization showed better results than L1 histogram normalization. However, since we are changing the quantization step, the normalization needed to be re-addressed. Soft quantization and histogram normalization are the focus of this paper and we therefore fix the codebook size at 1000 codewords and use a support vector machine with a histogram intersection kernel for all of the MED experiments.

5. RESULTS

We first selected the best-performing setup for soft quantization using Set A. This includes selecting the technique-specific parameters as well as the histogram normalization. We first used L1 normalization for parameter selection. The best performing parameters were then fixed and the histogram normalization was varied. Each soft quantization method was then applied with the selected parameters and histogram normalization to Set B. Set B contains a more diverse range of video events and therefore not only serves as an independent test to prevent over-fitting on Set A, but also provides insight into the stability of the method.

5.1. Parameter Selection

Soft thresholding depends on the fixed parameter α. Fixing all other aspects of the method, we varied α and measured the MAP on Set A. Results are shown in the left plot of Figure 3. An α value of 50 performed the best with this setup with a MAP of 0.0272.

Gaussian encoding, like soft thresholding, depends on a fixed parameter. The parameter σ was selected again by varying the value and selecting the one that yielded the greatest MAP on Set A. As seen by Figure 4, the MAP score plateaus around 0.0271 for large values of σ. We therefore selected to use $\sigma = 5000$ for the remainder of the experiments.

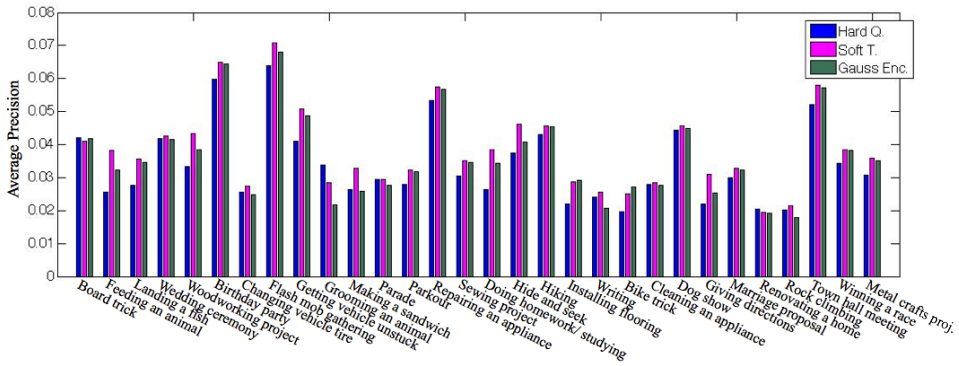

Fig. 5. Average precision by event for the original BoAW (left bar, blue), soft thresholding BoAW (middle bar, pink) and Gaussian encoding BoAW(right bar, green).

Soft Q. Method	Hist. Norm.	MAP
Soft thresholding	None	0.025
Soft thresholding	L1	**0.027**
Gaussian encoding	None	0.022
Gaussian encoding	L1	**0.025**

Table 2. Histogram vector normalization (Hist. Norm.) impact on the mean average precision (MAP) of the soft thresholding ($\alpha = 50$) and Gaussian encoding ($\sigma = 5000$) BoAW variations.

5.2. Histogram Normalization

Next we experimented on Set A to determine which histogram normalization would perform the best on each of the soft quantization approaches. Results are presented in Table2. L1-normalization outperforms no histogram normalization for both soft quantization methods. This is somewhat surprising since our previous work [7] found any normalization of the histogram vector to decrease performance, suggesting that the length of a video is correlated with the events.

The fact that L1-normalization outperforms no normalization indicates that the strength of assignment across the video's MFCCs outweighs the benefit gained by keeping the 1-norm in the model. Consider two videos: *X* and *Y*. Video *X* is longer than *Y* and therefore has more MFCC features vectors. In the original BoAW approach, the 1-norm of *X* would be greater than *Y* because the 1-norm is exactly equal to the number of original feature vectors in the file. However, when Gaussian encoding is applied, the 1-norm of *Y* becomes greater than that of *X*. This indicates that the assignments in *Y* are stronger than in *X*. This is an important factor that is not accounted for in the original BoAW approach.

5.3. Results by Event

Finally we applied the original (hard quantization) BoAW as well as the two soft quantization BoAW variations to Set B. The setup for the original BoAW is taken from our previous work [7] while the parameter values and histogram normalization for soft thresholding and Gaussian encoding are selected from the previously described experiments. MED results for the three BoAW methods across the 30 video events are shown in Figure 5. From these results it becomes clear that the soft quantization outperforms the hard quantization, especially when soft thresholding is used. The MAP score using the original BoAW is 0.034 where as Gaussian encoding and soft thresholding result in 0.036 and 0.038 respectively. Gaussian encoding improves the average precision in 20 of 30 events while soft thresholding improves performance in *27 of 30* events.

6. CONCLUSION AND FUTURE WORK

We presented our work on the soft quantization variation of the Bag-of-Audio-Words approach. Soft quantization accounts for the strength of code word assignments by scaling the histogram vector contribution proportional to the distance between that feature and the nearest codeword. The mean average precision, when evaluated on the TRECIVD datasets improves by 13%, showing an improvement in 27 of the 30 video events.

Future work will seek to further improve the BoAW quantization step by allowing for multiple assignments, also referred to as "soft assignment." Some MFCC vectors are nearly equidistance to more than one codeword. The original BoAW method assigns an MFCC feature vector to only one codeword, even if the difference in distance between the first and second best codeword in miniscule. Using the soft quantization techniques discussed in this paper, we can use the scaled values to select which and to what extent codewords should contribute to the histogram vector.

7. ACKNOWLEDGMENTS

We thank Greg Myers and Professor Robert M. Gray for their valuable discussions. This work was supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC0067. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes nonwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the U.S. Government.

This material is also based upon work supported by the National Science Foundation under Grant No. DGE-1147470.

8. REFERENCES

[1] TRECVID multimedia event detection 2011 evaluation. [Online]. Available: http://www.nist.gov/itl/iad/mig/med11.cmf

[2] S. Pancoast, M. Akbacak, and M. Sanchez, "Supervised acoustic concept extraction for multimedia event detection," in *Proceedings of the 2012 ACM international workshop on Audio and multimedia methods for large-scale video analysis*. ACM, 2012, pp. 9–14.

[3] G. Chechik, E. Ie, M. Rehn, S. Bengio, and D. Lyon, "Large-scale content-based audio retrieval from text queries," in *Proceedings of the 1st ACM international conference on Multimedia information retrieval*, ser. MIR '08. New York, NY, USA: ACM, 2008, pp. 105–112.

[4] M. Riley, E. Heinen, and J. Ghosh, "A text retrieval approach to content-based audio retrieval," in *Int. Symp. on Music Information Retrieval (ISMIR)*, 2008, pp. 295–300.

[5] Y. Uchida, S. Sakazawa, M. Agrawal, and M. Akbacak, "KDDI labs and sri international at TRECVID 2010: Conent-based copy detection," in *NIST TRECVID 2010 Evaluation Workshop*, 2010.

[6] Y. Jiang, X. Zeng, G. Ye, S. Bhattacharya, D. Ellis, M. Shah, and S. Chang, "Columbia-UCF TRECVID 2010 multimedia event detection: Combining multiple modalities, contextual concepts, and temporal matching," in *NIST TRECVID Workshop*, 2010.

[7] S. Pancoast and M. Akbacak, "Bag-of-audio-words approach for multimedia event detection," in *Proceedings of ICASSP*, 2012.

[8] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[9] J. van Gemert, C. Veenman, A. Smeulders, and J. Geusebroek, "Visual word ambiguity," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 7, pp. 1271–1283, 2010.

[10] A. Coates and A. Ng, "The importance of encoding versus training with sparse coding and vector quantization," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, 2011, pp. 921–928.

[11] A. Gersho and R. M. Gray, *Vector quantization and signal compression*. Norwell, MA, USA: Kluwer Academic Publishers, 1991.

[12] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 1794–1801.

[13] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce, "Learning mid-level features for recognition," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 2559–2566.