UNSUPERVISED FEATURE EXTRACTION FOR MULTIMEDIA EVENT DETECTION AND RANKING USING AUDIO CONTENT

Ehsan Amid[⋆†] *Annamaria Mesaros*[⋆] *Kalle J. Palomäki*[⋆] *Jorma Laaksonen*[†] *Mikko Kurimo*[⋆]

[⋆] Department of Signal Processing and Acoustics
[†] Department of Information and Computer Science
Aalto University, Espoo, Finland 02150
firstname.lastname@aalto.fi

ABSTRACT

In this paper, we propose a new approach to classify and rank multimedia events based purely on audio content using video data from TRECVID-2013 multimedia event detection (MED) challenge. We perform several layers of nonlinear mappings to extract a set of unsupervised features from an initial set of temporal and spectral features to obtain a superior presentation of the atomic audio units. Additionally, we propose a novel weighted divergence measure for kernel based classifiers. The extensive set of experiments confirms that augmentation of the proposed steps results in an improved accuracy for most of the event classes.

Index Terms— Multimedia Event Detection, Unsupervised Feature Extraction, Stacked Denoising Autoencoders, Bag of Words, Term Weighting, Weighted Jensen-Shannon Divergence.

1. INTRODUCTION

Event recognition in multimedia has attained rapidly increasing research interest in the past few years [1, 2, 3]. With the emergence of online multimedia datasets, there is growing demand for fast and accurate methods to obtain a ranking based on the query of the user. Most approaches focus on developing sophisticated visual features while the scope of the audio features is limited to using Mel-frequency cepstral coefficients (MFCC) [4]. Despite the success of the MFCC features in speech and music recognition tasks [5], their insufficiency in handling more general sounds has been shown by several authors (see [6] and [7]). This can results in a considerable loss of information which resides in the audio content.

On the other hand, unsupervised feature extraction methods have been announced as a promising approach in many pattern recognition schemes including object recognition [8], speech recognition [9] and music analysis [10]. These methods can be categorized into two main groups based on the type of the input signal: first, those that tend to extract a set of unsupervised features from the raw signal without any intervention by the user and, second, those that are built on a set of manually designed features as the input. Examples of both approaches can be found in [11].

In this paper, we develop an unsupervised feature extraction method which is utilized for multimedia event detection using the audio content. Our method is based on extracting an extensive set of temporal and spectral features in the first stage and then, finding a higher order presentation using several stages of nonlinear mappings. We also propose a novel approach to define a similarity measure between videos based on detecting event specific clues. The method is computationally efficient making it suitable for large scale retrieval tasks such as TRECVID multimedia event detection challenge [12].

2. SYSTEM DESCRIPTION

The final goal of our system is to recognize or rank a set of predefined events in a video based on the audio content. The block diagram of the system is shown in Figure 1. Each block is discussed in more details as follows.

2.1. Feature Extraction

A fundamental step in the system is to find an extensive set of features which can represent a broad range of audio signals in a proper manner. We describe our extensive feature extraction scheme in the following.

2.1.1. Preprocessing

In many video retrieval tasks, it can happen that the whole audio has been removed or replaced by a music track. In this case, the audio aspect may not provide any important clues to recognize the underlying event. Therefore, the first step in our system is detecting the music in audio content and discarding those instances which mainly contain music.

Music contains stable spectral peaks more often than speech or any non-harmonic sound [13]. This can be employed to distinguish music from other non-harmonic sounds. We adopted a similar approach by considering the spectrum as a gray-scale image and calculating the ratio of the stable

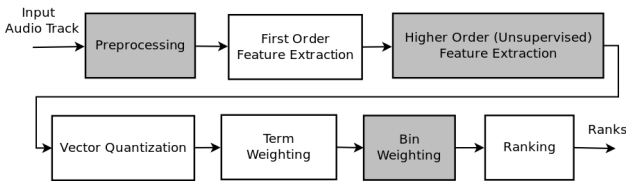

Fig. 1. Block diagram of the proposed system. The blocks shown in gray can be switched on or off.

spectral peaks in a short segment. After a (high-pass) pre-emphasis filter, we modeled the signal in a short-time (30 ms) overlapping window by a high order (600) autoregressive filter obtained using linear prediction with sampling frequency of 44100 Hz. Then, we calculated a frequency smoothed version of the spectrogram by finding the transfer function of the filters. Additionally, we performed a differentiation in the frequency direction, using a mask operator similar to Sobel gradient filter [14]. In this manner, we obtained the edges (peaks) of the spectrogram that show up as lines along the time axis. We further normalized the edges and discarded the short pieces of lines by applying a morphological opening operator with a line structuring element in the time direction. Finally, we summed up the edge intensities in short-time segments (0.5 second) and compared the value with a predefined threshold chosen based on experiments on a set of pure music instances. We labeled a segment as music if its edge intensity value exceeds the threshold. All the videos having music ratio above a certain value were discarded. Additionally, we omitted the videos with no audio content, i.e. silence. The remaining instances were processed further using the following steps.

2.1.2. Elementary Feature Extraction

As the videos contain a wide variety of sub-events and activities, generally overlapping, an extensive set of features, including temporal, spectral as well as cepstral, is required to achieve an appropriate elementary representation. The first-order features help the system focus on the desired aspects of the signal in the later steps. In our approach, we evaluated the mean and variance of the following features in a short-time segment (0.5 second) and considered it as the feature vector for the corresponding segment: *loudness, brightness, zero-crossing rate, spectral flux, spectral roll-off, MFCC, spectral sub-band energies* [15]. Moreover, we considered three additional features: mean and variance of *mean pitch frequency* which is defined as the mean value of the stable pitch frequency of the harmonic sounds over a short-time window and, *zero ratio (ZR)* which is the ratio between the number of frames containing short-time fundamental frequency (harmonic frames) and the total number of frames in a segment. ZR tends to be high for harmonic sounds, e.g. music, but lower for sounds with some harmonic portions, e.g. speech, and almost zero for non-harmonic sounds [16]. Concatenating all the features together, we obtained a feature vector of length 291 for each 0.5 second audio segment.

2.1.3. Higher order presentation using Stacked Denoising Autoencoder

To extract a higher level representation, we built several layers of nonlinear transformations over the initial set of features using a stacked denoising autoencoder.

Stacked denoising autoencoder (SDAE) is a generalized version of denoising autoencoder (DA) with several stacked hidden layers [17]. DA is an extension of the classical autoencoder which tries to reconstruct the clean input $\mathbf{x}$ from a noise corrupted version of it through a single nonlinear hidden layer. The encoding step can be presented as a mapping from the noise corrupted input $\tilde{\mathbf{x}}$ to the presentation in the hidden layer,

$$\mathbf{y} = f_\theta(\tilde{\mathbf{x}}) = s(\mathbf{W}\tilde{\mathbf{x}} + \mathbf{b}) \tag{1}$$

where $\theta = \{\mathbf{W}, \mathbf{b}\}$ represents the parameters of the encoder, namely the weight matrix $\mathbf{W}$ and the bias vector $\mathbf{b}$ and $\mathbf{s}(\cdot)$ is a squashing nonlinearity. The reconstruction step consists of a similar linear transformation followed by a squashing nonlinear function

$$\mathbf{x} = f_{\theta'}(\mathbf{y}) = s'(\mathbf{W}'\mathbf{y} + \mathbf{b}') \tag{2}$$

where the prime indicates the parameters of the decoder. A squared error loss or a cross-entropy loss can be considered in case of a general continuous input or an input which is either binary or bounded to the interval $[0, 1]$, respectively. As shown in [17], minimizing the reconstruction error amounts to finding a fine nonlinear structure in the input which is equivalent to learning a stochastic operator that maps the noise corrupted input to the lower dimensional nonlinear manifold that captures the variations in the input space, discarding the effect of noise. The hidden layer presentation is used as an effective feature extractor for classification [18] using general classifiers e.g. SVM. A higher level of nonlinearity can be obtained by considering several hidden layers where each layer tends to learn a higher level structure from whatever representation it receives from the layer immediately below. By using a set of manually designed features in the input, we encourage focus on the provided aspects of the signal. In our case, we trained an SDAE with 3 hidden layers containing 200, 100 and 10 hidden units, respectively, where the initial learning rate was set to 5000 with a batch size of 1000 and 500 epochs over a subset of 100000 randomly selected feature vectors. This way, we obtained a higher order representation of the features while reducing dimensionality.

2.2. Classification

2.2.1. Vector Quantization and Bag of Words

The dataset consists of videos of different lengths; therefore the sets of features for videos have different cardinalities. To obtain a fixed length representation of the videos, we adopted an approach based on vector quantization (VQ) which consists of first, clustering the feature space into a large number of cells and then, forming a feature vector for each video based on the histogram of the corresponding atomic audio units. This scheme is also referred as *bag of words*.

First, we defined a codebook as the set of all possible instances using kmeans++ algorithm [19]. Each feature vector in a video was then encoded by the index of the cluster which it belongs to. In the end, we formed a histogram vector for each video, containing the frequency of occurrence of each cell (word) in that video.

2.2.2. Term Weighting

Following our bag of words scheme, the histogram of the cluster indexes in each video can also be considered as the frequency of the terms (atomic audio units) occurring in an audio track. The term frequency approach enables us to adopt text mining methods to obtain an enhanced representation for the set of new features.

Instead of using the normalized histogram of the atomic audio events in an audio track, a better result can be obtained by highlighting the atomic units which are only frequent in a particular event class and discounting the effect of those that are frequent in most of the classes. This can be accomplished by considering the frequency of the units not only in each audio track but in the whole corpus, as well.

Similar to the approach in [20], a collection of n audio tracks indexed by m atomic audio units can be presented by an $m \times n$ matrix $\mathbf{A}$ in which A_{ij} entry corresponds to the weighted frequency of atomic audio unit i in audio track j. We chose $A_{ij} = l_{ij} g_i$ where l_{ij} is the local weight for unit i occurring in track j, g_i is the global weight for unit i in the collection. Let f_{ij} be the frequency of unit i in track j and $p_{ij} = f_{ij} / \sum_j f_{ij}$ denote the normalized frequency. We considered log-entropy term weighting

$$l_{ij} = \log(1 + f_{ij}) \tag{3}$$

and

$$g_i = 1 + (\sum_j p_{ij} \log(p_{ij})) / \log(n) \tag{4}$$

where a logarithm of base 2 is assumed. The resulting weighted frequency vectors, which are columns of $\mathbf{A}$, were used as the final features for ranking of the events.

2.2.3. Divergence Measure

The normalized histogram vectors can be viewed as probability vectors representing the weighted frequency of the different atomic audio units. This allows usage of several divergence measures over non-negative vectors as distance measure between two histograms. However, we note that if the atomic events are clearly separated, there will be larger peaks in particular bins in the histograms of a specific event compared to the background videos corresponding to the unique clues which are peculiar to that event. For this reason, we propose a novel weighting method for the weighted Jensen-Shannon divergence (WJSD) [21]. WJSD is defined by

$$D^w_{\text{JS}}(\mathbf{p} \parallel \mathbf{q}) = \frac{1}{2} \left(D^w_{\text{KL}}(\mathbf{p} \parallel \mathbf{m}) + D^w_{\text{KL}}(\mathbf{q} \parallel \mathbf{m}) \right) \tag{5}$$

$$\mathbf{m} = \frac{1}{2}(\mathbf{p} + \mathbf{q}) \tag{6}$$

where $\mathbf{p}$ and $\mathbf{q}$ are histogram vectors and

$$D^w_{\text{KL}}(\mathbf{u} \parallel \mathbf{v}) = \sum_i w_i \cdot u_i \log \left(\frac{u_i}{v_i} \right) \tag{7}$$

is a weighted KL-divergence between histogram vectors $\mathbf{u}$ and $\mathbf{v}$ which emphasizes on different bins when calculating the distance. The new weights are defined by first calculating the mean $\boldsymbol{\mu}^k$ and variance $\boldsymbol{\nu}^k$, $\quad k = 1, 2, \ldots, K$, vectors of the histograms for each class from the positive examples, where K is the number of classes. For the background videos, the mean and variance is denoted by $\boldsymbol{\mu}^b$ and $\boldsymbol{\nu}^b$, respectively. Then, the weight vector for class k is defined by

$$w^k_i = |\mu^k_i - \mu^b_i| \exp \left(-\frac{1}{2} \frac{\nu^k_i + \nu^b_i}{\max_j \{\nu^k_j\}} \right), i = 1, 2, \ldots, d \tag{8}$$

where d is the number of bins. The effect of the exponential term is to reduce the effect of peaks which are comparatively large but have a considerable variance. Additionally, it further smoothens the weight values. The WJSD can be used as the kernel function in a kernel based classifier e.g. SVM.

Table 1. Number of instances in each set before and after applying the preprocessing step

Dataset	Before	After
Positive (DEV)	3000	2613
Background	4992	4090
Research	10163	8335

3. EXPERIMENTAL RESULTS

The dataset used in the experiments is part of the TRECVID-2013 MED challenge. It consists of a set of positive example videos (MED2013 DEV) from 30 event classes, containing equal number of instances from each class plus a set of background videos which do not belong to any of those classes. We also considered another set of videos (TRECVID-2013 research set) for learning the unsupervised parts. The name list

Table 2. Top and bottom 3 performance values as well as the average over 30 event classes by means of area under ROC curve.

Dataset	ID	#1	#2	#3	#4	#5
Top 3 Values	29	0.64	0.74	0.74	0.71	**0.81**
	14	0.60	0.75	0.59	0.61	**0.80**
	8	0.59	0.66	0.57	0.54	**0.79**
Bottom 3 Values	10	0.47	0.53	0.42	0.49	**0.55**
	24	0.52	0.55	**0.56**	0.56	0.55
	7	0.47	**0.50**	0.48	0.46	0.50
Avg. over 30 Events	-	0.55	0.60	0.59	0.58	**0.64**

Table 3. Top and bottom 3 performance values as well as the average over 30 event classes by means of mean average precision (%).

Dataset	ID	#1	#2	#3	#4	#5
Top 3 Values	29	7.85	9.65	9.97	9.08	**19.62**
	14	7.61	10.68	11.51	8.17	**17.87**
	23	9.58	8.83	9.64	10.14	**15.09**
Bottom 3 Values	10	3.73	3.44	2.83	3.62	**3.80**
	7	**4.46**	3.78	3.80	3.17	3.67
	3	2.92	3.68	3.10	**4.51**	3.65
Avg. over 30 Events	-	4.76	5.58	5.90	5.79	**7.62**

of the events and a more detailed description of the dataset can be found in [12]. The total number of videos in each set before and after applying the preprocessing step and discarding the tracks with no audio or with a high music ratio is shown in Table 1. After the feature extraction step and normalization, an SDAE with 3 hidden layers was trained on the research set, as mentioned in Section 2. Next, kmeans++ algorithm with $k = 100$ was applied on the features obtained from the binary hidden units in the bottleneck layer of the SDEA and the weighted term frequency histogram vector was formed.

We performed a 10-fold cross validation. For each class, we considered a set of 100 support vector regression (SVR) [22] models each trained using all the positive instances and a proportional set of negative instances randomly drawn from the background set. We considered a one-against-all approach on the validation set for each class. The final results were averaged over the outputs of the models.

We conducted four different experiments to show the effect of the non-linear mapping using SDAE and the weights (8) in WJSD as well as the preprocessing step on the performance, by switching the gray blocks in Figure 1 on or off:

#1. No SDAE + (unweighted) JSD

#2. No SDAE + WJSD

#3. Non-linear mapping using SDAE + (unweighted) JSD

#4. Non-linear mapping using SDAE + WJSD but without the preprocessing step

#5. Non-linear mapping using SDAE + WJSD (proposed method)

We evaluated the results with two measures: 1) Area under ROC curve and, 2) mean average precision (MAP). As it can be seen in Table 2 and Table 3, incorporation of the highlighted steps in the proposed method leads to a significant improvement by means of both area under ROC curve and MAP, respectively, for most of the event classes as well as the average performance over all classes. However, it should be noted that the dataset that we utilized for our experiments is extremely hard to be analyzed by using only the audio content. This is because there are many instances which convey only a few or even no audio clues to be considered for detection. Our results can be compared with those reported in [3]. The experiments in [3] are performed on a subset of 15 events using 100 semantic audio models trained using additional annotated data. Our dataset contains the same 15 events but different instances, and for those we obtained 7.25% MAP compared to about 22% reported in [3]. However, our system builds atomic audio units in an unsupervised manner, avoiding the need for any annotations for supervised training of the semantic concepts. Additionally, applying the semantic concept models is computationally much heavier than our feature extraction approach.

The poor results for some event classes are due to lack of audio clues which makes it difficult to recognize without incorporating additional information. For instance, event E007-Changing a vehicle tire, which is among low performance results, mainly contains narration of the process, making it intractable to detect without incorporating visual clues or an ASR system. More details about the full multimodal system incorporating the proposed approach can be found in [23].

4. CONCLUSION

We proposed an unsupervised method of feature extraction using SDAE for event ranking based on the audio content. The results show that the proposed method is capable of extracting more applicable features for classification from a set of initial features which are commonly used in speech and audio recognition. Additionally, we presented a weighted Jensen-Shannon divergence measure which we found useful for kernel based classifiers.

5. ACKNOWLEDGEMENTS

This work was supported by the Academy of Finland in projects 251170 and 136209, by KAUTE and Emil Aaltonen foundations, and by TEKES FuNeSoMo project. The authors would like to thank Kyunghyun Cho for providing the code.

6. REFERENCES

[1] M. Sjöberg, S. Ishikawa, M. Koskela, J. Laaksonen, and E. Oja, "Picsom experiments in TRECVID 2011," in *Proc. of the TRECVID 2011 Workshop*, Gaithersburg, MD, USA, 2011.

[2] Y. Jiang, X. Zeng, G. Ye, and et al., "Columbia-UCF TRECVID2010 Multimedia Event Detection: Combining Multiple Modalities, Contextual Concepts, and Temporal Matching," in *NIST TRECVID Workshop*, Gaithersburg, MD, November 2010.

[3] L. Cao, S.-F. Chang, N. Codella, C. Cotton, and et al., "IBM Research and Columbia University TRECVID-2012 Multimedia Event Detection (MED), Multimedia Event Recounting (MER), and Semantic Indexing (SIN) Systems," *TREC Video Retrieval Evaluation Workshop*, 2012.

[4] S. Chaudhuri, M. Harvilla, and B. Raj, "Unsupervised learning of acoustic unit descriptors for audio content representation and classification.," in *INTERSPEECH*. 2011, pp. 2265–2268, ISCA.

[5] J. Martinez, H. Perez, E. Escamilla, and M.M. Suzuki, "Speaker recognition using Mel frequency cepstral coefficients (MFCC) and vector quantization (VQ) techniques," in *CONIELECOMP, 22nd Int. Conf. on*, 2012, pp. 248–251.

[6] B. Ghoraani and S. Krishnan, "Time-frequency matrix feature extraction and classification of environmental audio signals," *Audio, Speech, and Language Proc., IEEE Trans. on*, vol. 19, no. 7, pp. 2197–2209, 2011.

[7] E. Tsau, S. Chachada, and C.-C.J. Kuo, "Content/context-adaptive feature selection for environmental sound recognition," in *Sig. Info. Processing Association Annual Summit and Conf. (APSIPA ASC), Asia-Pacific*, 2012, pp. 1–5.

[8] V. Nair and G. E. Hinton, "3-D object recognition with deep belief nets," in *Advances in Neural Information Processing Systems 22*, 2009.

[9] H. Lee, P. Pham, Y. Largman, and A. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," in *Adv. in Neural Information Proc. Sys. 22*, pp. 1096–1104. 2009.

[10] P. Hamel and D. Eck, "Learning features from music audio with deep belief networks," in *Proc. of ISMIR*, Aug. 2010, pp. 339–344.

[11] Yoshua Bengio, Aaron Courville, and Pascal Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[12] P. Over, G. Awad, Ma. Michel, and et al., "TRECVID 2013 – an overview of the goals, tasks, data, evaluation mechanisms and metrics," in *Proceedings of TRECVID 2013*. NIST, USA, 2013.

[13] K. Minami, A. Akutsu, H. Hamada, and Y. Tonomura, "Video handling with music and speech detection," *MultiMedia, IEEE*, vol. 5, no. 3, pp. 17–25, 1998.

[14] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2nd edition, 1992.

[15] S. Chu, S. Narayanan, and C.-C.J. Kuo, "Environmental sound recognition with time-frequency audio features," *Audio, Speech, and Language Processing, IEEE Trans. on*, vol. 17, no. 6, pp. 1142–1158, 2009.

[16] T. Zhang and C.-C.J. Kuo, "Audio content analysis for online audiovisual data segmentation and classification," *Speech and Audio Processing, IEEE Trans. on*, vol. 9, no. 4, pp. 441–457, 2001.

[17] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P. A. Manzagol, "Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion," *J. Mach. Learn. Res.*, vol. 11, pp. 3371–3408, Dec. 2010.

[18] P. Vincent, H. Larochelle, Y. Bengio, and P. A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. of the 25th int. conf. on Mach. learn.*, NY, USA, 2008, ICML '08, pp. 1096–1103.

[19] D. Arthur and S. Vassilvitskii, "k-means++: the advantages of careful seeding," in *Proc. of the 18th annual ACM-SIAM symp. on Discrete algorithms*, 2007, pp. 1027–1035.

[20] M. W. Berry, N. Gillis, and F. Glineur, "Document classification using nonnegative matrix factorization and underapproximation," in *Circuits and Systems, 2009. ISCAS 2009. IEEE Int. Symp. on*, 2009, pp. 2782–2785.

[21] J. Lin, "Divergence measures based on the shannon entropy," *Information Theory, IEEE Transactions on*, vol. 37, no. 1, pp. 145–151, 1991.

[22] C. C. Chang and C. J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. on Intelligent Sys. and Tech.*, vol. 2, pp. 27:1–27:27, 2011.

[23] S. Ishikawa, M. Koskela, M Sjöberg, J. Laaksonen, E. Oja, E. Amid, K. Palomäki, A. Mesaros, and M. Kurimo, "Picsom experiments in TRECVID 2013," in *Proc. of the TRECVID 2013 Workshop*, Gaithersburg, MD, USA, 2013.