Audio Codec Identification Through Payload Sampling

Samet Hiçsönmez , Husrev T. Sencar , Ismail Avcibas [*]

[#] *Computer Engineering Department,*

TOBB University of Economics and Technology, Ankara, Turkey

{shicsonmez,hsencar}@etu.edu.tr

[*] *Electrical and Electronics Engineering Department*

Turgut Ozal University, Ankara, Turkey

iavcibas@turgutozal.edu.tr

Abstract— **W0065 present a new method for audio codec identification that does not require decoding of coded audio data. The method utilizes randomness and chaotic characteristics of coded audio to build statistical models that represent encoding process associated with different codecs. The method is simple, as it does not assume knowledge on encoding structure of a codec. It is also fast, since it operates on a block of data, which is as small as a few kilobytes, selected randomly from the coded audio. Tests are performed to evaluate the effectiveness of the technique in identification of the codec used in encoding on both singly coded and transcoded audio samples**

I. INTRODUCTION

Digital audio is encoded and decoded using a variety of codecs that are primarily designed to compress sound and music for more compact storage, stream audio over the Internet, and transmit voice communications over PSTN's, cellular networks and VoIP networks. With more than hundred audio codecs available for use, the ability to quickly identify the codec used in coding of an audio without relying on any encoding metadata could provide new ways to tackle some existing challenges. For instance, all network-monitoring tools try to characterize network traffic flows to analyze network performance. As multimedia becomes a larger part of network traffic, accurate and fast characterization of audio traffic in its different forms (e.g., streaming, file transfers, VoIP applications, etc.) becomes more and more important [1].

Another field where codec identification or, more specifically, determining encoding history of (multiply transcoded) audio streams might be needed is the telephony system [2]. In today's vastly diversified and non-centralized telephony infrastructure, there are no reliable mechanisms to determine or verify the origin of an incoming call as the voice signal might have been routed over many networks. Lack of such capabilities serves as an enabler for new forms of malicious activity like voice spam and voice phishing attacks. Similarly, evaluation of quality of audio files and detecting fake-quality ones, i.e., low bit rate audio files transcoded at higher bit rates pretending to be in high quality, is another application area that can benefit from the ability to identify audio codec, and corresponding coding parameters, involved in generation of an audio [3].

In all these application scenarios, it may seem file or packet metadata is sufficient to identify the encoder used in generation of the audio. Although this is true, in some cases it may simply not be viable to access this information. In network monitoring, for example, payload of packets at lower layers of the protocol stack has to be accessed. In a high-speed network with many active network flows this level of inspection will require significant computational resources. In cases of call origin identification and audio quality evaluation, however, where not only the most recently used codec but also transcoding history is needed such a simple approach won't help at all. These two cases require statistical analysis of audio packets and audio content, which can be computationally intensive.

In this work, we introduce a new technique for codec identification that can be utilized either as a complementary or an independent tool in all these application scenarios. By codec identification we mean the ability to classify a given audio byte stream as being generated by one of the fixed number of possible audio codecs. While a large number of codecs exist, essentially, codecs simply differ in the trade-off they make between numbers of competing design objectives. These objectives include a codec's ability to balance compression with sound quality, to provide robustness and error correction against noise and network glitches, and to adapt to varying transmission bandwidth. The crux of the paper lies in the fact that the net effect of these design choices inherent to encoding process reflects on the encoded byte stream. The method exploits this by sampling a small window of data, i.e, a byte vector, from the audio byte stream to characterize the codec in terms of the inherent statistical properties, randomness vs. determinism, entropy, and chaotic properties of the byte stream. Since the technique operates on the encoded audio byte stream and decoding is not required, it is naturally fast.

There are a few research works that use statistical characterization schemes to obtain information from encoded data. In [4], considering audio steganalysis application,

Böhme *et al.* proposed a procedure to determine the encoder of MP3 files (i.e., a certain implementation of the MP3 format) so that at later stages of steganalysis appropriate detection algorithm can be run. For this purpose, a set of 10 features is designated to capture implementation specific particularities. These features are then used in conjunction with a machine learning classifier to discriminate between 20 MP3 encoders. Although high identification accuracy is reported (87% on a random sample of MP3's), generalization of this approach to other encoders is not trivial as feature extraction process relies on the knowledge of MP3 structure. Alternatively in [1], authors proposed a method, which forms this basis of our approach, to characterize the content of a network flow based on its statistical properties. Rather than determining the codec used in encoding of a given data, the primary goal of that study is to classify network packet content as belonging to one of a set of data types like text, image, audio, video, encrypted data, etc. The underlying idea of the approach is that, regardless of the codec used in encoding of one type of data, the degree of randomness and redundancy in the corresponding byte stream will differ. Using a set of 6-25 features and randomly sampling 32 kb of data from each flow in the test dataset, an average accuracy of 90% was achieved in distinguishing 7 types of data, which included WMA and MP3 audio file formats as two different types.

Although inspired by these techniques, our approach differs from both of them in two aspects. First, byte vectors generated by different audio codecs will exhibit very similar statistical properties as compared to those of different data types; therefore, a more distinguishing set of features is needed. Second, we don't assume the knowledge of file structure associated with any of the codecs. In our tests, we utilize 16 of the most popular audio codecs used for audio compression, PSTN's, cellular networks, and VoIP networks.

The rest of this paper is organized as follows. In the next section, we present an overview of the system and introduce features that are used to build classification models capable of distinguishing between byte streams generated by different codecs. Main characteristics of selected audio codecs and the differences between them are described in Section 3. Experimental results are given in Section 4, and we conclude with further discussion in Section 5.

II. Methodology

The method described in this paper is developed for identifying the codec used in generation of an audio byte stream. It neither utilizes any coding metadata nor assumes the knowledge of structure of coded data. The technique is based on characterization of encoding process, which involves striking a balance between compression, quality and robustness. This is realized by measuring the inherent randomness and chaotic characteristics of coded data and using these characteristics as a part of a classification system.

The system works in two phases: the offline phase and the online phase. In the offline phase, the system is initially built from scratch through a process called training. Operation during this phase can be broken down into three stages. First one is the data sampling, which is performed by collecting a block of data from a random location in the coded audio file. In other words, a fixed-size vector of bytes (i.e., a vector of eight bit unsigned integers that take values from 0 to 255) represents each audio file. In the second stage, features needed for statistical characterization are extracted from the block data. Features are obtained from a training set of audio that is coded by different codecs. In the third state, extracted feature data is used to train machine learning algorithms needed for classification. During the online phase, the system is tested against given audio files, and its performance is measured as average accuracy in discriminating between audio files encoded by the same codec.

Obviously, the most important aspect of the technique is the selection of features that will be used for statistical characterization. Features we used to build the model can be grouped into two broad categories as follows.

A. Randomness Features

Randomness characteristics are determined by statistical analysis of a block of encoded data. These features are primarily inspired by the randomness tests devised by NIST to evaluate randomness properties of cryptographic applications [8], and they can be broadly categorized as time or frequency domain depending on how features are computed.

In time domain, we first compute simple statistics, like mean, variance, auto correlation, and entropy, and higher order statistics, like bicoherence, skewness and kurtosis. Since each codec uses a different encoding algorithm, it is expected that these differences would be reflected in the encoded data. To give an example, even a physical inspection of WMA encoded data will reveal frequent occurrence of long sequence of zeros. Similarly, one can observe that G.721 and G.726 coded data have higher mean values as compared to other outputs of other codecs.

Entropy is another discriminative feature we used in experiments. Since entropy quantifies the degree of randomness in the data, it is very reliable in differentiating non-encoded data like WAV and PCM from compressed data. Autocorrelation feature on the other hand is used to find repeating patterns in the encoded data which relates to coding structure. Therefore, first 20 coefficients of autocorrelation function are included as features.

We also computed higher order statistics like bicoherence, which is a measure of non-linearity and non-Gaussianity in the byte vector. Bicoherence takes values bounded between 0 and 1, ranging from non-significant to significant, and it is useful in discriminating between different levels of compression. We computed average bicoherence and power of the bicoherence amplitude as features. Skewness and kurtosis of each byte vector are also included as two features that denote two possible ways how the distribution of data deviates from the normal distribution. In frequency domain, however, we examine distribution of energy in different bands. For this, we divide the frequency band into four equal sub-bands and compute mean, variance and skewness in each band as features.

B. *Chaotic Features*

There is theoretical and experimental evidence for the existence of chaotic phenomena in speech signals that is left uncovered by linear models [7]. Assuming that the speech signal is produced by a chaotic system, different compression algorithms will change the chaotic structure of the speech signal and therefore the chaotic features such as Lyapunov Exponents (LE) and false neighbour fraction (FNF) and correlation dimension will be different for each of the compressed version of the speech signal. In our method, we included a total of 26 features out of which 11 are related to LE and 15 to FNF.

The main concept of the LE and FNF is based on the neighbourhood of the speech signal vectors in the phase space. Compression process changes the neighbourhood distances between each of the compressed signal vectors in the phase space. Compression algorithms exploit the redundancy present in the signals and their performance is measured by the amount of correlation left at their output. As there is no perfect (practical) compression algorithm, there still remain unexploited correlations at their output stream. Chaotic type features obtained in the phase space simply measures the leftover multidimensional correlations in the output stream. Our main hypothesis is that these features are different in a statistical sense for each of the compression algorithm, as they leave unique leftover correlations at their output stream to such an extent that we can design successful classifiers.

III. AUDIO CODECS

When designing our system and conducting experiments, we used 16 widely deployed audio codecs used in compression and transmission of audio [5] [6]. Below, we briefly describe these codecs and their main characteristics by grouping them into four categories depending on their main purpose of use. It should be noted that except for Flac codec, all codecs perform lossy compression when encoding data.

A. *Voip Codecs*

- Speex [10] is a wideband speech codec based on CELP (Code Excited Linear Prediction) that is designed for VoIP applications and voice file compressions. It provides three sampling rate options 8 kHz as narrow band, 16 kHz as wide band and 32 kHz as ultra-wide band and variable bit rate options (from 2 kbps to 44 kbps). It is widely used in teleconferencing, VoIP systems and also in video games.
- iLBC [11] is a narrow band speech codec that is very robust against packet losses. iLBC has 8 kHz sampling rate and operates on two bit rates, 13.33 kbps for 30 ms frames and 15.2 kbps for 20 ms frames. This codec uses a block independent LPC (Linear Predictive Coding) algorithm. iLBC is a compulsory standard for VoIP over cable and is also used in applications like Google Talk, Yahoo Messenger and Skype.
- G.721 is an ADPCM (Adaptive Differential Pulse Code Modulation) codec and is standardised in the mid 80s by CCITT. It has a sampling rate of 8 kHz and operates on 32 kbps bit rate. Its speech quality is as good as 64 kbps PCM codec (i.e., G.711).
- G.726 is another ADPCM codec that uses 8 kHz sampling rate and operates on four bit rates, i.e., 16, 24, 32 and 40 kbps. Main application of 16 and 24 kbps bit-rates is for overload channels carrying voice in digital circuit multiplication equipment (DCME). This codec is also the standard codec used in DECT wireless phone system.

B. *Cellular Network Codecs*

- AMR (Adaptive Multi-Rate) codec, also known as Gsm 6.90, is a narrow band (200-3400 Hz) codec, which is mandatory standard codec for 2.5G/3G wireless networks based on GSM (WDMA, EDGE, GPRS). AMR uses different encoding techniques like ACELP (Algebraic CELP), DTX (Discontinuous Transmission), VAD (Voice Activity Detection) and CN (Comfort Noise). The only sampling rate is 8 kHz but 7 different bit rates (from 4.75 to 12.20 kbps) are available.
- AMR-WB or AWB (G.722.2) [12] is a wide band (50-7000 Hz) audio codec that provides excellent speech quality. This codec has 16 kHz sampling frequency and operates on 9 different bit rates from 6.60 to 23.85 kbps. This codec uses ACELP as encoding technique and is used in a variety of applications like VoIP, Mobile Communication (UMTS, CDMA) and ISDN wide band telephony.
- GSM 6.10 or GSM-FR (full rate) [13] is the first digital speech-coding standard used in GSM mobile phone system that is still widely used in networks around the world. This codec uses RPE-LTP (Regular Pulse Excitation - Long Term Prediction) as speech coding scheme. 8 kHz sampling rate results in an average bit rate of 13 kbps.
- GSM 6.10 (WAV) is known as Microsoft GSM 06.10 and has a sampling rate of 11025 Hz and a bit rate of 18 kbps.

C. *PSTN Codecs*

- PCM (Pulse Code Modulation) is the process of converting analog signals into digital via sampling and quantization. It is the standard storage format used in computers and digital telephony system.
- A-law and μ-law [14] are two other commonly used PCM codecs deployed in telephony system. They both use a fixed sampling rate of 8 kHz and quantization resolution of 8 bits/sample, resulting in a 64 kbps bit rate. The difference between the two is that μ -law takes 14 bit signed linear PCM samples whereas A-law takes 13 bit PCM samples before conversion 8 bit samples.

D. *Compression Codecs*

- MP3 (MPEG layer III) [6] is the standard format of digital audio compression for the transfer and playback of music in almost all digital audio players. MP3 uses psychoacoustic compression techniques and is based on

MDCT (Modified Discrete Cosine Transform). MP3's are encodes using Huffman algorithm and can be encoded at different sampling rates, ranging from 8 kHz to 48 kHz and bit rates of 32-320 kbps.

- AAC (Advanced Audio Coding) is a lossy compression and encoding algorithm for digital audio that is designed to be the successor of MP3 format. Like MP3, AAC uses MDCT as encoding scheme. This codec has variable bit rate and fixed bit rate options. AAC supports various bit rates from 16 to 256 kbps and sampling rates of 8-96 kHz. It is the default audio format used in Apple products.
- Ogg Vorbis [15] is an open source audio codec designed specifically for digital music. Although it provides better sound quality than other MDCT based codecs, encoding and decoding operations are very computation intensive. Ogg Vorbis operates on sampling rates from 8 kHz to 192 kHz and up to 255 discrete channels (e.g., monaural, polyphonic, stereo, quadraphonic, ambisonic). Since Ogg Vorbis is a variable bit rate codec, its quality is not represented in bit rates but instead expressed on a quality scale where 1 corresponds to highest compression level and 10 to least compression.
- WMA (Windows Media Audio) is a lossy audio codec created by Windows. It uses a psychoacoustic model similar to MP3. WMA can encode signals to a sample rate of 48 kHz. WMA supports encoded streams at both constant bit rate and variable bit rate. In constant bit rate mode, WMA supports bit rates from 5 kbps to 384 kbps. Similar to AAC and Ogg Vorbis, WMA is also based on MDCT.
- Flac (Free Lossless Audio Codec) [16] is an audio compression codec uses lossless compression algorithm. Flac achieves almost 50% compression level for most music. To guarantee the encoding is fully lossless Flac uses fixed-point samples instead of floating-point. Flac manages sampling rates from 1 Hz to 655,350 Hz with 1 Hz steps and from 4 to 32 bits per sample.

IV. Experiments and Results

The data set used in experiments consisted of 1000 audio samples taken from 50 music CDs covering different genres such as classical, pop, folk, rock, and instrumental. Each sample is approximately 5 seconds long and 850 KB in size. During experiments, audio samples are encoded with 16 different codecs described in Section III, which resulted with 16K coded audio samples. Corresponding coding parameters for each codec are given in Table I. Except for AMR and AWB codecs, whose parameters are selected to provide the highest quality encoding, all others are default parameters used by the encoders.

For statistical characterization, from each coded audio sample, a randomly selected 1 KB of byte vector is extracted and features are computed. (This is realized by reading 1 KB of data starting at a randomly determined offset relative to start of by byte stream.) Our feature vector has 65 elements of which 39 represent the randomness properties, 15 the chaotic characteristic false nearest neighbour fraction and 11 the Lyapunov exponents. We then use a standard machine learning technique, a support vector machine (svm) implemented using Libsvm package [9], with a radial basis function kernel, for classifying feature vectors associated with different codecs. In all experiments, half the feature vectors from audio samples are used for training and the other half for testing.

In our experiments, two different scenarios are considered. In the first scenario, the goal is to quantify the performance of the method in identification of codec used in encoding of raw audio samples. Whereas in the second scenario, coded audio are transcoded with another codec (i.e., audio sample is decoded with the first codec and coded again with another codec), and the goal is to identify the first codec used in coding of the transcoded audio sample.

TABLE I
Encoding Parameters for Each Codec

	Sampling Rate	Bit Rate
Speex	8 kHz	18 kbps
iLBC	8 kHz	13,33 kbps
G.726	8 kHz	16 kbps
AMR	8 kHz	12,20 kbps
AWB	16 kHz	12,65 kbps
GSM 6.10	8 kHz	13 kbps
GSM (wav)	11,025 Hz	18 kbps
PCM	22,050 Hz	16 bits per sample
MP3	44,1 kHz	192 kbps
AAC	44,1 kHz	128 kbps
WMA	44,1 kHz	128 kbps

A. *Single Coding Scenario*

In our first test, we determine the ability of the technique in discriminating between 16 encoders. In the experiments, the set of un-coded (raw) audio samples is also included as a separate class. Feature vectors obtained from 8500 audio samples are used to train a 17-class svm, which is then tested on the remaining 8500 audio samples. The confusion matrix (i.e., error matrix) showing the classification results is given in Table II.

Overall classification accuracy, which is obtained by averaging the success in correctly classifying test samples in each class, is measured to be 85.1%. As can be seen from the confusion matrix, resulting classifier is quite successful in identification of most codecs but has difficulty in discriminating certain codecs from each other. Most notably, G.721 and G.726 couldn't be distinguished from each other accurately. This is primarily due to the similarity of the two encoding algorithms. Same reasoning applies for classification between MP3 and Ogg Vorbis codecs both of which utilize MDCT based encoding.

In experiments, we also tested how well codecs in each category can be differentiated from each other. For this

purpose different classifiers are built only considering four codecs used in cellular networks, six in compression, three in PSTN's and four in VoIP networks. Average identification accuracy within categories of cellular, compression, PSTN and VoIP codecs are obtained, respectively, as 96.2%, 83.8%, 98.5% and 80%. Identification of codecs used in compression and VoIP communication has the lowest accuracy. This can be attributed to centrality of compression in these codecs, which causes encoded audio to exhibit similar characteristics.

B. *Transcoding Scenario*

In practice, when two parties using different communication platforms communicate audio (i.e., cell phones, landlines, Internet phones) the audio has to be transcoded at the boundary of the two communication networks. (For example, when calling a landline from a VoIP service provider, audio coded by a VoIP codec has to be first decoded and re-encoded by the appropriate PSTN codec before it reaches its destination.) In such cases, the question will be about getting knowledge on the first codec used for encoding. To determine this, in this scenario, we investigate whether our proposed model can distinguish between singly- and multiply-coded audio, and moreover, whether it can identify the codec used during initial encoding. We consider three experimental scenarios to represent transcoding from one network to other.

In the first experiment, we consider transcoding from a cellular network to PSTN. That is, the audio is initially encoded with one of the four codecs mentioned in Section III-B. Then, depending on the choice of PSTN codec these audio samples are transcoded with one of three different codecs mentioned in Section III-C. Therefore in the experiments, raw audio samples are first encoded with four cellular network codecs, resulting with 4K audio samples, followed by decoding and encoding with each of the three different PSTN codecs, yielding 12K samples. Because there are three different PSTN codecs, three tests are performed. (Due to lack of space, resulting confusion matrices are not given.)

In the first test, audio encoded with AMR, AWB, GSM and GSM 6.10 (WAV) codecs are converted to A-law coded audio. Then a five-class classifier is trained to differentiate between four forms of A-law coded data and singly encoded audio using A-law codec. Overall accuracy of this classification is measured to be 81.72%. Results show that singly coded data and GSM 6.10 coded data can be identified with accuracy above 97% and confusions are mainly due to inability to discriminate between GSM 6.10 and AWB encoded data. In the second test, μ-law PCM codec is used instead. The overall accuracy in this case is measured as 79.68%. Similar to previous test, singly encoded data and GSM 6.10 codec has been identified with highest accuracy. In the third case, where PCM codec is used for transcoding, overall accuracy is measured as 74.60%. The most noteworthy result here is that singly encoded data is classified with 100% success.

Our second experiment is based on transcoding from GSM network to VoIP network. In this case, data is first encoded with one of the four GSM codecs and then converted with one of the four VoIP codecs mentioned in Section III-A. Therefore, four different tests are conducted. Similar to previous experiment, singly coded data is also included in the classification. In the first test, transcoding is done using iLBC codec. Corresponding classification results show that an accuracy of 80.40% is achievable. Furthermore, singly encoded data and GSM 6.10 coded data can be distinguished with 99% and 92%, accuracy, respectively. In the following tests, Speex, G.726 and G.721 codecs are used for transcoding and overall identification accuracy is obtained, respectively, as 78.7%, 73% and 63.2%. One interesting result here is that both with G.721 and G.726 codecs singly encoded data couldn't be distinguished from transcoded data as good as in the previous cases (72% and 56%). This is likely to be due to the compression algorithms used in these codecs, which are effective in suppressing traces of earlier encoding.

Our last experiment considers transcoding from VoIP network to PSTN. Similar to previous experiments, audio samples are first encoded with the four VoIP codecs, decoded and encoded again with each of the PSTN codecs. The classification accuracy in identifying whether a given PSTN coded audio is transcoded or not and, if so, identifying the codec is obtained for A-law codec, u-law codec, and PCM are obtained, respectively, as 78.75%, 78.3%, and 83.45%. In all cases, singly coded data was discriminated from transcoded data with accuracy close to 100%.

V. Discussion and Conclusions

A fast and simple method is introduced to identify codec used for the encoding of a given audio sample. The method uses a multi-class classification system based on features, which characterize randomness and chaotic behavior of coded data, and support vector machines. Two sets of experiments are performed. In the first one, identification among 16 audio codecs is considered. Results show that except for G.726 codec, which was mainly confused with the similar G.721 codec, most codecs can be identified quite accurately, with an average accuracy of 85%. In the second set of experiments, encoded audio samples are transcoded with another codec and the technique is used to identify the first codec. Results in this case show that singly coded and transcoded audio codecs can be discriminated from each other with an accuracy close to 100%, and in each category of codecs the codec before transcoding can be identified approximately with 80% accuracy. Overall, results show that as compression gets more severe, it becomes more difficult to discriminate among codecs. Further experiments will be performed to take into account different codec parameters and an increased number of codecs will be considered in the experiments.

Acknowledgment

This material is based upon work supported by the Scientific and Technological Research Council of Turkey under research grant number 110E049.

REFERENCES

[1] Shanmugasundaram, K., Kharrazi, M., Memon, N., "Nabs: a system for detecting resource abuses via characterization of flow content type", *Computer Security Applications Conference, 2004. 20th Annual*, vol., no., pp. 316- 325, 6-10 Dec. 2004.

[2] V. Balasubramaniyan, A. Poonawalla, M. Ahamad, M. Hunter and P. Traynor, "PinDr0p: Using Single-Ended Audio Features to Determine Call Provenance", 2010.

[3] R. Yang, Y. Q. Shi and J. Huang "Defeating fake-quality MP3", *Porc. ACM Multimedia and Security'09*, 2009.

[4] Rainer B., AndreasW., "Statistical characterisation of MP3 encoders for steganalysis", Proc. of the 2004 workshop on Multimedia and security, pp. 25-34, 2004.

[5] S. Karapantazis and F.N. Pavlidou, "VoIP: a comprehensive survey on a promising technology", *Computer Networks* 53 (12) (2009), pp. 2050–2090.

[6] D. Avelar, B. Morrissette, D. Forbes, G. Albert, "Audio Codecs: Evaluation and Comparison of Popular Format", 2008.

[7] O. H. Kocal, E. Yuruklu, I. Avcibas, "Speech steganalysis using chaotic-type features", IEEE Transactions on Information Forensics and Security (2008) 651-661.

[8] A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptogrpahic Applications, NIST Special publication 802-22,2001 http://csrc.nist.gov/groups/ST/toolkit/rng/documents/SP800-22b.pdf

[9] C.-C. Chang and C.-J. Lin, LIBSVM: A library for support vector machines, http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[10] The Speex Codec. http://www.speex.org/, 2003.

[11] Global IP Solutions. The Internet Low Bitrate Codec (ILBC). http://tools.ietf.org/html/rfc3951, 2004.

[12] ITU-T Recommendation G.722.2, Wideband Coding of Speech at Around 16 kbit/s using Adaptive Multi-Rate Wideband (AMR-WB), 2002.

[13] ITU-T Recommendation G.722.2, Wideband Coding of Speech at Around 16 kbit/s using Adaptive Multi-Rate Wideband (AMR-WB), 2002.

[14] The International Telecommunication Union. G.711: Pulse Code Modulation (PCM) of Voice Frequencies. http://www.itu.int/rec/T-REC-G.711/e, 1972.

[15] Xiph.org Foundation. http://xiph.org/vorbis/doc/Vorbis_I_spec.html, 2007.

[16] FLAC. FLAC: Free Lossless Audio Codec. http://flac.sourceforge.net/documentation_format_overview.html, 2008.

TABLE II
CONFUSION MATRIX FOR 17-CLASS CODEC IDENTIFICATION PROBLEM

Actual		Predicted																	
		PSTN			VoIP				Cellular				Compression						
		A-law	u-law	PCM	AMR	AWB	GSM	GSM (WAV)	G 721	G 726	iLBC	Speex	AAC	MP3	OGG	FLAC	WAV	WMA
PSTN	A-law	97,8	0,6	0,4	0	0	0	0	0	0,2	0,4	0	0	0	0	0,2	0,2	0,2
	u-law	3	95,8	1	0	0	0	0,2	0	0	0	0	0	0	0	0	0	0
	PCM	0,4	0,6	98,2	0	0	0	0	0	0	0	0	0	0	0	0	0	0,8
VoIP	AMR	0	0	0	92,2	3,8	0	0	0	0	3	0	0	0	0	1	0	0
	AWB	0	0	0	4	92,2	0	0	0	0	1,6	0,2	0	0,2	0	1,8	0	0
	GSM	0	0	0	0	0	88	11,8	0	0	0	0,2	0	0	0	0	0	0
	GSM (WAV)	0	0	0	0,6	0,4	1	94,6	0	0	0,2	0,8	0	0	0	2,4	0	0
Cellular	G 721	0,2	0	0	0	0	0	0	77	22,8	0	0	0	0	0	0	0	0
	G 726	0,2	0	0	0	0	0	0	49,8	49,6	0	0	0	0	0	0	0	0,4
	iLBC	0,2	0	0	0,8	1,2	0	0,2	0	0	85	2,2	0,8	2,4	1,6	5,2	0	0,4
	Speex	0	0	0	0	0	0	4	0	0	4	82,8	0	0	4	5	0	0,2
Compression	AAC	0	0	0	0	0,4	0	0	0	0	0,8	2,4	85,6	3,2	4,4	2,6	0	0,6
	MP3	0	0	0	0,4	0,2	0	0	0	0	3,8	1,6	7	65,8	12,4	6	0	2,8
	OGG	0	0	0	0,2	0	0	0	0	0	1	2,6	3,4	13	77,6	1	0	1,2
	FLAC	0,2	0	0	0,2	1,4	0	1,2	0	0	1	3	1,4	4,6	0,4	85	0	1,6
	WAV	0,4	0	0	0	0	0	0	0,6	0,6	0	0	0	0	0	0,2	97,4	0,8
	WMA	0	0	0	0,2	0	0	0	0	0	0,8	0,2	3	7,4	4,8	1,2	0	82,4