Multiclass Neural Network for Codec Classification

Seungwoo Wee

Department of Electronics and
Computer Engineering
Hanyang University
Seoul, South Korea
Email: slike0910@hanyang.ac.kr

Jechang Jeong

Department of Electronics and
Computer Engineering
Hanyang University
Seoul, South Korea
Email: jjeong@hanyang.ac.kr

***Abstract*—In this paper, we suggest to remove or modify the denoted class of codec in a bitstream for military purposes in image communication. In that case, a decoder first needs to determine the codec type to restore the original data. This paper proposes a codec classification method which has not been studied much yet. For extracting the feature of a bitstream, Recurrent Neural Network (RNN) model is used since it is suitable for time series data used for training on classification. Video codecs have their own distinctive header structures, which can be considered features in the encoded bitstreams. The proposed method extracts the feature of an encoded bitstream and classifies the bitstream into the specific codec. Three standard codecs, MPEG-2, H.263, and H.264/AVC, are used to generate the training and the test data set in the experiment. We analyze several components affecting the performance and compare to conventional algorithm. The performance degrades when two kinds of bitstreams generated by H.263 and H.264/AVC are trained together. However, when the training data includes both H.263 and H.264/AVC, performances improved with increasing training data set sizes.**

Keywords–Feature extraction; Classification; Bitstream; Multiclass neural network; Recurrent neural network.

I. Introduction

In image communication, decoders decode images by parsing the received bitstreams. Since the class of codec is denoted in the header of the bitstream, a decoder does not need additional processing to determine the codec type of received bitstreams. Therefore, codecs have been developed focused on compression rate and complexity. For this reason, codec classification methods have not been studied yet. However, if the class of codec written in the header is removed or modified for military purposes, a decoder first needs to classify the codec type to restore the original data.

Classification algorithms, usually, have been applied to image and language [2][4]. To determine the codec type using partial bitstreams, Recurrent Neural Network (RNN) is used for codec classification [13]. This method exploits the fact that standard codecs are hierarchically structured, which can be used to extract features of the specific codec.

In this paper, we propose a multiclass neural network model for codec classification considering MPEG-2, H.263, and H.264/AVC. Our proposed method classifies an unknown bitstream into a specific codec utilizing the fact that the encoded bitstream consists of its unique data form. Through the experimental results, we show the different tendencies of the performance according to the size and composition of the training data set.

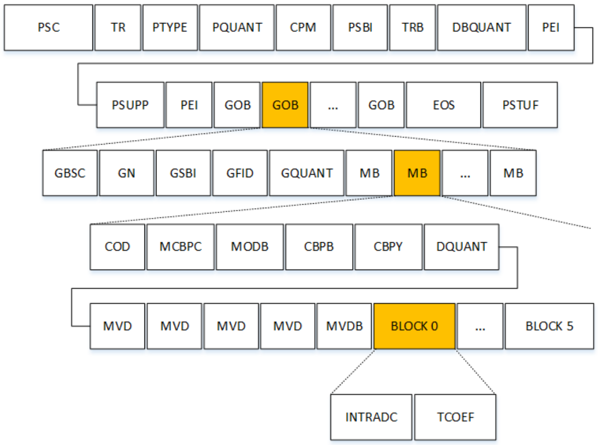


Figure 1. Hierarchical structure of H.263.

We organize the rest of the paper as follows. First, we introduce some backgrounds in Section II to help to understand the proposed algorithm about multiclass neural network. Section III introduces the proposed codec classification algorithm based on RNN model. In Section IV, experimental results are shown. Finally, this paper is concluded in Section V.

II. Background

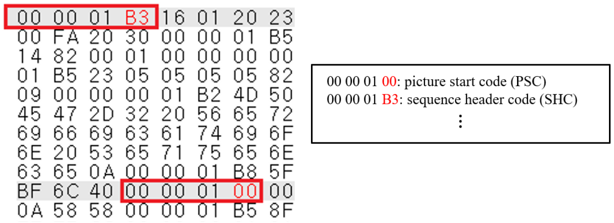
In this section, we give some backgrounds before introducing the proposed algorithm. After introducing the characteristics of the encoded bitstreams, we describe a neural network which extracts features of each codec for classification.

A. Video Coding

Video coding is an essential part to store or transmit video data efficiently. Two main organizations of video coding standards, Moving Picture Experts Group (MPEG) [14] and Video Coding Experts Group (VCEG) [15], try to compress data size while keeping the quality of decoded contents as high as possible. Each codec has hierarchical structure.

Figure 1 shows the hierarchical structure of H.263 to compress data efficiently. Likewise, the header structures of MPEG-2 and H.264/AVC are hierarchical, too [8] [9].

Based on each structure, codecs have their own start bit codes, which occur rarely at general data. Figure 2 represents bitstream generated by MPEG-2 encoder, and each element is expressed in hexadecimal. The bitstream begins with ’00 00



```
00 00 01 B3 16 01 20 23
00 FA 20 30 00 00 01 B5
14 82 00 01 00 00 00 00
01 B5 23 05 05 05 05 82
09 00 00 00 01 B2 4D 50
45 47 2D 32 20 56 65 72
69 66 69 63 61 74 69 6F
6E 20 53 65 71 75 65 6E
63 65 0A 00 00 01 B8 5F
BF 6C 40 00 00 01 00 00
0A 58 58 00 00 01 B5 8F
```

00 00 01 00: picture start code (PSC)
00 00 01 B3: sequence header code (SHC)
⋮

Figure 2. An example of the MPEG-2 bitstream in hexadecimal format.

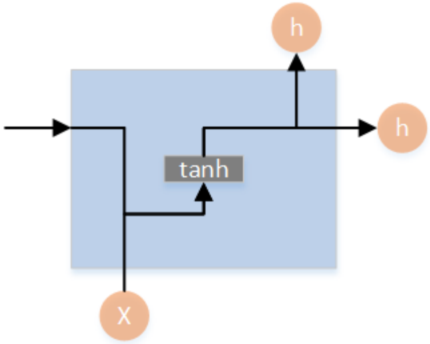


Figure 3. Basic RNN layer

01 B3', Sequence Header Code (SHC), which indicated that the following parts include sequence information.

Likewise, other standard codecs, such as H.263 and H.264, include their own start bit codes, respectively. These codes can be regarded as unique features of each codec.

B. Neural network

A Deep Neural Network (DNN) has layers between the input and output layers [1]. The inputs of DNN pass through the layers to calculate the probability of each output. The appearance of AlexNet which uses DNN improved the image classification performance substantially [2].

Convolutional Neural Network (CNN, or ConvNet) is a deep, feed-forward Artificial Neural Network (ANN) and it uses convolution for feature extraction. Therefore, it is usually utilized in image recognition and natural language processing[3][4][10][11].

Recurrent Neural Network (RNN) is an artificial neural network and has connections between nodes. The nodes can indicate time dynamic behavior for the time sequence data like handwriting or voice signals. RNN stores the internal state, which handles inputs of the network and influences the near layers. For tasks such as handwriting and speech recognition, this network is used [5]–[7]. RNN is known as a model suitable for processing data that appears sequentially, such as voice and text.

Figure 3 represents a basic RNN layer. The output *h* of the current state is updated by receiving previous output and current input x. The activation function of the hidden state is *tanh*, a nonlinear function. The hidden node stores a state and is connected to the next layer.

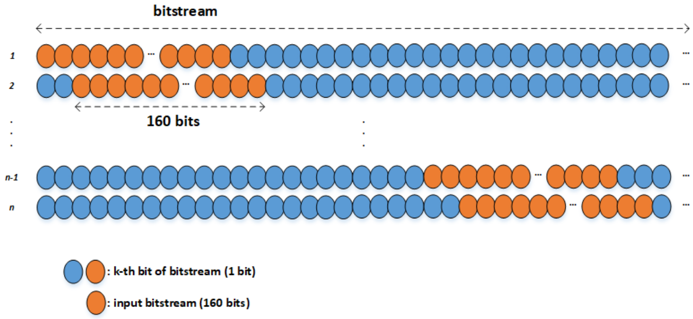


Figure 4. Composition of training and test data set.

As shown in Figure 3, RNN has an advantage that it can create various flexible structures according to needs, because it can accept inputs and outputs regardless of the sequence length. Each RNN layer stores a state and the state affects the input of the next layers. The closer the distance between the two states, the greater the impact on each other. This time dependent characteristic motivates our proposed method for applying RNN to codec classification. Since the bitstream generated by a codec is time series data and each codec has its own structure, the RNN based model is suitable to extract those features. We describe how the data set is organized and the structure of the neural network in the next section.

III. Proposed Algorithm

Our proposed algorithm depends on the following two assumptions. First, the RNN based model can be trained to extract the features of an unknown bitstream. In our previous work [13], we have already shown that an RNN based model with a simple RNN structure is trainable. Second, the longer the length of each input or the deeper the layer of the model, the higher the accuracy of codec classification. We explain the second assumption with analyze the experimental results in the next section

As mentioned in Section II, here we describe the construction of data set in detail Figure 4 presents the composition of the training and test data set. All rows represent the same bitstream of a codec. The orange circles of the *n*-th row are labeled as the codec class, which is a basis for training our network. *n* is the number of items in the data set, and the consecutive orange circles, 160 bits, are the *n*-th input of the proposed network.

Since the characteristics of each codec are concentrated in the header section, the data set is constructed at the beginning of the bitstream. As shown in Figure 4, the data set is created from the beginning of the bitstream by shifting because the characteristics of each codec are concentrated in the header part. We analyzed the results according to *n* values and the composition of the data set.

The overall network structure of the proposed multiclass neural network based on RNN is shown in Figure 5. Before the input bitstream passes through RNN layers for the training process, we apply an embedding process to improve performance [7]. Since the bitstream consists of binary data, the difference between input data that will be considered feature is too subtle to discriminate.

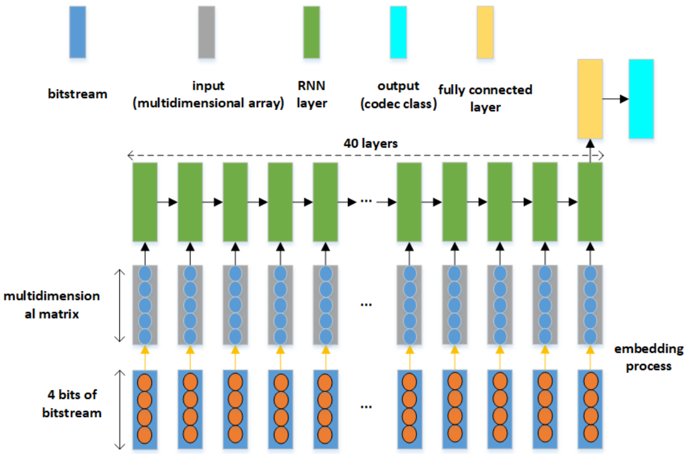


Figure 5. Network structure of the proposed algorithm.

Through the embedding process, 4 bits of the bitstream are transformed into a multidimensional matrix, which is useful to extract features and to be updated. The proposed algorithm converts each input to a 100-dimensional matrix whose elements consist of rational numbers. We compose 40 RNN layers for training and a fully connected layer estimates a specific codec class of the current input.

IV. Experimental Results

In this section, we present experimental results and the analysis of the results. Through the experimental results, we found the different tendencies of the performance according to the size and composition of the training data set.

We used RNN layer to classify the unkown bitstream into the specific codec. Three standard video codecs, MPEG-2, H.263, and H.264/AVC, were used in the experiment to generate bitstreams and test the performance. To construct the training and test set, Common Intermediate Format (CIF) video sequences were used for a total of 3,000 labeled data items. The experiments were performed using Python 3.6, PyTorch and Window 10 Pro x64 environment.

The hidden layer and the batch size were 100 and 256, respectively. We created the training data set by shifting, as shown in Figure 4. Each input was composed of 4-bit units and an input passed through each RNN layer. Before inputs passed through the RNN layer, the inputs was transformed into a 100-dimensional matrix by the embedding process. 40 matrices passed through 40 RNN layers at once.

We compared the results according to the composition and size of the training data set. The organization of the training data set was composed to calculate the accuracy of binary codec classification and multiple codec classification. The size of the test data set was the fixed length of 3,000. The sizes of training data sets were of 3,000, 6,000, 9,000, and 12,000, respectively.

Table I shows the accuracy of our proposed algorithm according to the composition and size of the training data set. Performance degradation occurred when both bitstreams generated by H.263 and H.264/AVC trained together. Based on this, we can assume that H.263 and H.264/AVC have relatively similar header structure compared to MPEG2. As shown in Table I, performances improved with larger training data set sizes except when the training data includes both H.263 and H.264/AVC.

TABLE I. Codec classification accuracy according to the composition and size of training data set.

composition of training data set	*training data set size*			
	3,000	*6,000*	*9,000*	*12,000*
MPEG-2 and H.263	0.82	0.85	0.85	0.87
MPEG-2 and H.264	0.76	0.79	0.81	0.82
H.263 and H.264	0.77	0.78	0.77	0.76
MPEG-2, H.263, and H.264	0.70	0.71	0.72	0.72

V. Conclusion

In this paper, we proposed an RNN based multiclass neural network for codec classification. Unlike the previous work, this work used three standard video codecs to test the performance of the method. Besides, further analysis is done considering the size and composition of the training data set.

We experimentally verify that the size and composition of the training data set affected performance and the proposed method is trainable for extracting the features of each codec. Experimental results show that increasing the size of the training dataset improves performance because the various structures in the training dataset help to generalize the model.

Acknowledgment

This work was supported by the Brain Korea 21 plus Project in 2014.

References

[1] J. Schmidhuber, “Deep Learning in Neural Networks: An Overview,” *Neural Networks*, vol. 61, 2015, pp. 85–117.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” in *Proc. Conf. Adv. Neural Inform. Process. Syst.*, 2012, pp. 1097–1105.

[3] A. Van den Oord, S. Dieleman, and B. Schrauwen, “Deep Content-Based Music Recommendation,” in *Advances in neural information processing systems*, 2013, pp. 2643–2651.

[4] R. Collobert and J. Weston, “A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning,” *Proceedings of the 25th International Conference on Machine Learning, ACM*, 2008, pp. 160–167.

[5] R. Bertolami et al., “A Novel Connectionist System for Improved Unconstrained Handwriting Recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, 2009, pp. 855–868.

[6] H. Sak, A. Senior, and F. Beaufays, “Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling,” in *15th annual conference of the international speech communication association*, 2014, pp. 338–342.

[7] X. Li and X. Wu, “Constructing Long Short-Term Memory based Deep Recurrent Neural Networks for Large Vocabulary Speech Recognition,” 2014.

[8] B. G. Haskell, A. Puri, and A. N. Netravali, Digital video: an introduction to MPEG-2, Springer Science Business Media, 1996.

[9] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, “Overview of the H. 264/AVC video coding standard,” *IEEE Transactions on circuits and systems for video technology*, vol. 13, no. 7, 2003, pp. 560–576.

[10] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–13.

[11] C. Szegedy et al., “Going Deeper with Convolutions,” in *Proc. IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[12] K. Jack, Video demystified: a handbook for the digital engineer, Elsevier, 2011.

ISBN: 978-1-61208-731-3

[13] S. Wee and J. Jeong “RNN-based bitstream feature extraction method for codec classification,”in *Proc. SPIE 11049, International Workshop on Advanced Image Technology (IWAIT)*, vol. 11049, 2019, p. 110493N.

[14] The MPEG website. [Online]. Available: https://mpeg.chiariglione.org/

[15] The VCEG document archive site. [Online]. Available: https://www.itu.int/wftp3/av-arch/video-site/