Cover Song Recognition Based on MPEG-7 Audio Features

Mochammad Faris Ponighzwa R, Riyanarto Sarno, Dwi Sunaryono
Department of Informatics Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
ponighzwa13@mhs.if.its.ac.id, riyanarto@if.its.ac.id, dwi@if.its.ac.id

Abstract—Lately, song industry has developed rapidly throughout the world. In the past, there were many applications which used song as their main themes, such as Shazam and Sound hound. Shazam and Sound hound could identify a song based on recorded one through the application. These applications work by matching the recorded song with an original song in the database. However, matching process is only based on the particular part of the spectrogram instead of an entire song's spectrogram. The disadvantages of this method arise though. This application could only identify the recorded original song. When application recorded a cover song, it cannot identify the title of the original song's since the spectrogram of a cover performance's and its original song's is entirely different. This paper exists to discuss how to recognize a cover song based on MPEG-7 standard ISO. KNN was used as classification method and combined with Audio Spectrum Projection and Audio Spectrum Flatness feature from MPEG-7 extraction. The result from this method identifies an original song from recorded cover of the original one. Result for experiment in this paper is about 75–80%, depends on testing data; whether the testing data is a dominant vocal song or dominant instrument song.

Keywords—cover song recognition; MPEG-7; KNN

I. INTRODUCTION

Basically, the song is a sound/audio that has a tone. Tone can't be watched directly but can be observed as spectrogram through software as a signal. Song can be identified based on their particular part of the spectrogram. This method is only suitable to identify a recorded song to its original because the spectrogram of recorded song must have the same spectrogram part to its original. This method cannot be applied to identify a cover song to its original since sometimes a cover song has the same tone as the original song, but it still feels "different" from the original. This "different" sense occurs because cover artist sang the same tone as the original artist, but played on a different note from the original artist. It can be lower or higher from the original song, depending on the capability of the cover artist. Another problem with this method is sometimes there is a few cover song sung by a specific gender, but the original song was sung by opposite gender. If this happens, it is very unlikely that the cover songs can be identified to its original song due to the fact that the spectrogram of male and female is diverse even though they sang the same song.

Song recognition application such as Shazaam and Sound hound used identification method that is based on particular song's spectrogram. Shazam and Sound hound only used a certain part of the spectrogram, called fingerprint, and matched its fingerprint with their database. So, in another word they cannot identify a cover song to its original. This experiment proposes cover song recognition method based on MPEG-7, using 2 features from extraction. Audio Spectrum Projection and Audio Spectrum Flatness will be used as a feature in this method. Audio Spectrum Projection was chosen because in MPEG-7 standard, Audio Spectrum Projection was an MPEG-7 feature that is described as a classification feature. Audio Spectrum Projection can distinguish a sound from many sources, such as woman and man or sound identification. While Audio Spectrum Flatness is an MPEG-7 feature that served its purpose to describe flatness properties of the power spectrum's [1]. In general, Audio Spectrum Flatness was used to calculate the similarity between signals. Based on this feature, this experiment proposes a cover song recognition method with modified KNN (K-Nearest Neighbors) algorithm. KNN classification will be combined along with signal processing on previous experiment about electroencephalogram's signal processing [2-3].

In previous work, sliding method has been proposed to identify a piece of the song's to its original. Sliding method is a method that slides through every sub-band of a song [4]. In this case, an original song was compared each sub-band with entire sub-band of pieces of the original song. But this method is only suitable to compare the original song to its clip. If the testing data is a cover song, the result will not be accurate with the original song since cover song has different spectrogram to its original song and different spectrogram will produce different sub-band. Another cover song recognition method has been proposed in previous paper. In the paper, a chrome of spectrogram is produced from a cover song and its original song, of which was compared. However, feature extraction was not using MPEG-7 standard but used raw signal as a feature [5].

The remainder of this paper is organized as follows, Section II describes materials and methods used in this

978-1-5090-5864-8/17/$31.00 ©2017 IEEE

experiment. Section III explains result and discussion including detail of dataset used in this experiment, MPEG-7 Feature Extraction detail, and modified KNN with its result. Finally, section IV is the conclusion of this work along with future work.

II. MATERIALS AND METHODS

This section discusses both material and method used in this experiment. There are 3 major components used in this experiment, i.e. View Component, Model Component, and Controller Component. Each of these 3 major components has its own process to fulfill their role. View component is responsible for displaying and receiving input from the user. Model component is responsible for handling all processes related to signal processing and feature extraction. Finally, Controller component is responsible as a "connector" between Model component and View component. So, in general, this experiment used MVC architecture to unify each used component. The detail of MVC architecture which is applied in this experiment explained in Fig. 1 and will be discussed in detail in each sub-section of this paper.

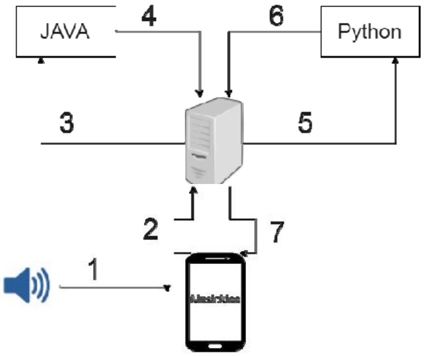

Fig. 1. MVC architecture applied in this paper.

A. MPEG-7

MPEG-7 is a multimedia content description standard in a video or audio [6]. This experiment focused on MPEG-7 related to audio content because the input data and purpose of this experiment are processing audio. MPEG-7 has many features known as DDL (Description Definition Language), such as Audio Spectrum Projection, Audio Signature Type, Audio Spectrum Spread, etc. DDL described the richness content of audio's in metadata form. All metadata of an audio from feature extraction is based on MPEG-7 saved in XML document. Each DDL has N x M dimension, N value means the duration of extracted audio and M value means collected sub- band of each N. So, the longer the duration of an audio is, the larger N value will be. As for M value, it depends on DDL used in MPEG-7; for example, Audio Spectrum Projection has value 9, but other feature could be greater of less than 9. Detail of DDL's dimension can be seen in Fig. 2.

$$\begin{bmatrix} A_{1,1} & A_{1,2} & A_{1,3} & \dots & A_{1,M-1} & A_{1,M} \\ A_{2,1} & A_{2,2} & A_{2,3} & \dots & A_{2,M-1} & A_{2,M} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ A_{N-1,1} & A_{N-1,2} & A_{N-1,3} & \dots & A_{N-1,M-1} & A_{N-1,M} \\ A_{N,1} & A_{N,2} & A_{N,3} & \dots & A_{N,M-1} & A_{N,M} \end{bmatrix}$$

Fig. 2. NxM dimension of DDL's metadata.

B. Dataset

For audio datasets, this experiment used 5 labels for training data. Each label represents a title from the original song and each label has 10 songs, with 5 songs from male cover and 5 songs from female cover. So the total of training data is 50, but selected song only contains domain vocal song. Cover song for dataset is downloaded from YouTube with .wav extension. Each song was cut 1 minute only. Each 1-minute cut version of a cover song's was extracted and went through feature extraction and feature selection process. Selected feature from cut-version was uploaded to database. This experiment used 5 titles that serve as labels; the titles are Pillowtalk original by Zayn Malik, Sky Full of Star (shortened "Sky") original by Coldplay, Heathens original by Twenty One Pilots, Treat You Better (shortened "Treat"), and Stitches original by Shawn Mendes.

C. Android Application

Android application was developed using Android Studio application. Android application has role as View component and the sound of cover song was recorded here. When the cover song was successfully recorded, this application will upload it to the database through controller component. When Controller component receives recorded cover song, it will call Model component to begin feature extraction. Android application use audio encoding PCM 16 bit and sample rate 44100 kHz in order to get best quality of recorded audio.

D. Php Server (XAMPP)

XAMPP has a major role in this architecture as controller component. XAMPP server will connect both model component with view component and between model component (Java server and Python server). Communication between model component and view component or between servers was done by using a string variable. Communication between model components was performed by using string parameter. When a server has finished doing its job, it will return a string value to XAMPP. Then, XAMPP will use this string value as input to another server and begin its process. Communication between model component and view component is also performed using string parameter. Meaning, once the final calculation is done, it will return string value and parse it to view component to be displayed as result.

E. Java Server

Java server was developed using Play framework application. Play framework was used because its flexibility to handle some request operation. Play framework handles feature extraction from wav audio extension. This operation was triggered when XAMPP called routing address of this application. When routing address was called, feature extraction process will be implemented. Java server responsible to handle feature extraction from .wav audio extension to XML document MPEG-7 standard.

F. Python Server

Python server was developed using Flask web application. Both Flask and Play framework has identic architecture. Both has flexibility routing and handle request operation. But Flask application was used, because both calculation and classification need to be done in python programming language, due to the richness of libraries such as numpy, sklearn, scipy, and etc. Main role of Flask web application was to calculate and classify data on the database. This experiment using SQL database to save dataset. Detail of saved dataset has been explained in the previous section. Python server responsible both pre-processing stage and processing stage. Pre-processing stage involve implementation of wavelet method and processing stage involve KNN classification process.

III. RESULT AND DISCUSSION

A. A. Feature Extraction

For feature extraction, MPEG7AudioEncApp was used as java library. MPEG7AudioEncApp java library takes a song with .wav extension as input and the output was a document with .xml extension [7]. XQuery (query for XML document) was applied to XML document to select Audio Spectrum Projection and Audio Spectrum Flatness feature [8]. Only 2 feature selected because Audio Spectrum Projection and Audio Spectrum Flatness feature are relevant for cover song recognition.

1) Audio Spectrum Projection (ASP)

Audio Spectrum Projection is used to represent low-dimensional features of a spectrum after projection against a reduced rank basis. Audio Spectrum Projection represents spectrogram that used as sound classification from many sources. MFCC was common feature extraction for audio classification. However MFCC was not MPEG-7 ISO standard, so this experiment using ASP instead.

From Table I, the differences between MFCC and Audio Spectrum Projection was explained [1]. This experiment using Audio Spectrum Projection because ASP was equivalent feature as MFCC but ASP was MPEG-7 ISO standard. From Table I, the result of classification between Audio Spectrum Projection and MFCC extraction was explained. This result was taken from previous work [1]. The best result from the previous experiment was Audio Spectrum Projection with 23 feature dimension and can be observed on Table II.

TABLE I. DIFFERENCES BETWEEN MFCC AND AUDIO SPECTRUM PROJECTION

Steps	**MFCC**	***Audio Spectrum Projection***
1	Converted to frames	Converted to frames
2	Each frame, obtain the amplitude spectrum	Each frame, obtain the amplitude spectrum
3	Mel-scaling and smoothing	Logarithmic scale octave bands
4	Take the logarithm	Normalization
5	Take the DCT	Perform basis decomposition using PCA, ICA, or NMF for projection features

TABLE II. TOTAL CLASSIFICATION ACCURACY (%15) OF 15 CLASSES

Feature extraction method	**Feature Dimension**		
	7	*13*	*23*
PCA–ASP	82.9	90.2	95.0
ICA–ASP	81.7	91.5	94.6
NMF–ASP	74.5	77.2	78.6
MFCC	90.5	93.2	94.2

2) Audio Spectrum Flatness

Audio Spectrum Flatness describes the flatness properties of the spectrum of an audio signal within a given number of frequency bands. Means, each value in Audio Spectrum Flatness expressing the deviation of the signal’s power spectrum from a flat shape inside a predefined frequency band. Finally, with this measure Audio Spectrum Flatness was used to calculate how similar one signal to another.

B. Discrete Wavelets Transform

To recognize a title of an original song from a cover song’s, it needs to compare singer spectrogram of the original and the cover song. A singer’s spectrogram can be retrieved by applying wavelet method to the entire song’s spectrogram. To retrieve a vocal’s spectrogram from a song, it needs to obtain low-pass of wavelet method. Low-pass of wavelet method is represented by approximation value of spectrogram. This approximation value was compared with approximation value of dataset in the database.

Fig. 3, shows the plotting of normal cover song’s spectrogram. It means that Fig. 3 shows mixed spectrogram between instrument and vocal. Fig.4 show Low-pass filter spectrogram of Fig. 3. It means that Fig. 4 only shows vocal spectrogram of a cover songs. It is important to apply Low-pass filter to both testing and training data spectrogram because if the spectrogram was directly compared with testing and training data, the spectrogram will contain the instrument’s information. This experiment uses Discrete

Wavelets Transform to denoise a signal so that a spectrogram was matched against vocal dominant only.

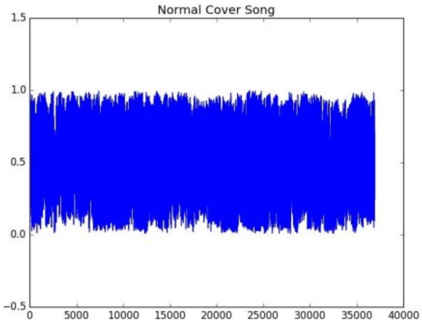

Fig. 3. Normal cover song spectrogram (with instrument).

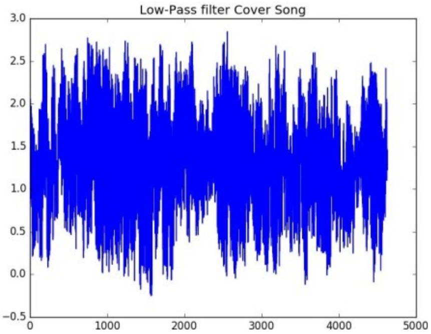

Fig. 4. Low-pass filter cover song spectrogram (vocal only).

Fig. 5, shows the plotting of Low-pass filter of cover song's spectrogram. Differences between Fig. 4 and Fig. 5 are in Fig. 4, using correct decomposition level to apply wavelet method. When Low-pass filter is obtained from wavelet method, it returns spectrogram that contains information of the vocal artist. While in Fig. 5 it uses incorrect decomposition level to apply wavelet method. When Low-pass filter is obtained from wavelet method, it returns spectrogram that contains the information of vocal artist's though some information were lost due to incorrect decomposition level.

To find correct decomposition level of wavelet method, applied (1) [9]. First, calculate the mean value of spectrogram. Each value minus mean of spectrogram then absolute every value, and applies the Fast Fourier Transform (FFT) method. FFT is a method that transforms time domain spectrogram into frequency domain spectrogram [10]. maxIndex is the index of the highest value of FFT spectrogram. Fs is the default value of audio frequency (1024 kHz). Finally, length of spectrogram from Fast Fourier Transform method is the value of L. Frequency value as resulted from (1), was compared with Table III [11] to determine whether it is in range level 1, level 2, etc. Rule to determine each value of Table III could be obtained by applied (2), where is the sampling frequency, is the dominant frequency, and L is decomposition level for Discrete Wavelet Transform.

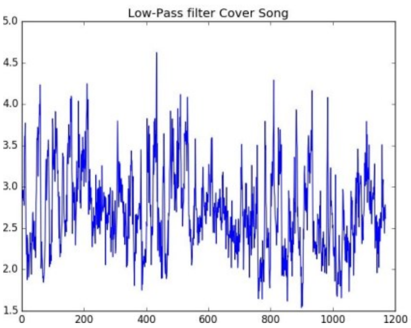

Fig. 5. Defect Low-pass spectrogram.

$$Fr = \frac{Fs\ x\ maxIndex}{L} \tag{1}$$

$$\frac{Fq}{2^{L+1}} \leq F_{dom} \leq \frac{Fq}{2^{L}} \tag{2}$$

TABLE III. DECOMPOSITION LEVEL OF WAVELET BASED ON FREQUENCY RANGE

Decomposition level (L)	Frequency range (Fr) (Hz)
1	256-512
2	128-256
3	64-128
4	32-64
5	16-32
6	8-16
7	4-8
8	2-4
9	1-2
10	0.5-1
11	0.25-0.5
12	0.125-0.25
13	0.0625-0.125

C. *Modified KNN*

This cover song recognition experiment, used a modified KNN. KNN classification was used because refers to previous EEG (Electroencephalogram) experiment [12]. In previous EEG experiment, data was in signal form and used KNN as classification method [13]. This experiment also data was in signal form, so this experiment used KNN as classification refers to previous paper. General KNN uses single data for nearest neighbor, such as iris flower classification problem. Each testing data was computed against training data to find its distance value. The minimal distance was assumed as the correct class of testing data [14]. It is essential to modify KNN algorithm because in this case both training and testing data are in matrix form instead of single data. So, it needs to compute matrix against matrix, not single data against single data. The matrix of a piece of cover’s from wavelet method was compared against matrix inside the database. Matrix from a piece of cover song has dimension [1 x m] and matrix from database has dimension [1 x n], where m < n. To calculate distance, it needs to apply sliding algorithm.

Fig. 6 explains modified KNN used in this experiment. Fig. 7 explains general KNN classification problem. The differences between modified and general KNN is in the modified KNN use sliding algorithm to compute distance between training and testing data. Fig. 9 explains iris data classification problem. From Fig. 9, iris data is a single data, not in matrix form. Iris data classification can be calculated directly by distance equation, such as Manhattan distance, Euclidian distance, etc. From Fig. 10, signal classification cannot be calculated directly because testing data (piece of cover song) has a different dimension from training data (full cover song).

```
Begin:
  Feature extraction for KNN
  classification Begin Loop:
      Begin Loop:
    Apply Sliding Algorithm:
    Compute distance of each object
    Shortest distance assumed as distance to object
  Sort each object ascending based on distance
  Shortest distance assumed as “class truth”
```

Fig. 6. Modified KNN Pseudo-code

```
Begin:
  Feature extraction for KNN
  classification Begin Loop:
      Compute distance of each object
    Sort each object ascending based on
  distance Shortest distance assumed as “class
  truth”
```

Fig. 7. KNN Classification Pseudo-code

To calculate distance with different dimensions such as Fig. 10, sliding algorithm needs to be applied. Fig. 8 described how sliding algorithm works [4]. Sliding algorithm was applied in every training data inside the database. Return value of Sliding algorithm is a minimum distance from training data. This minimum value represents similarity distance testing data to its training data. After obtaining each value of similarity distance, nearest neighbor from all training data was observed. Nearest K neighbor was assumed as the original song to its pieces.

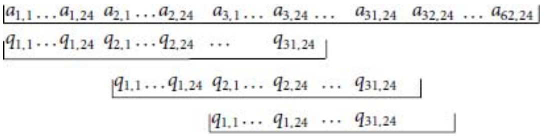
$a_{1,1} \ldots a_{1,24}\ a_{2,1} \ldots a_{2,24}\ a_{3,1} \ldots\ a_{3,24} \ldots\ a_{31,24}\ a_{32,24} \ldots a_{62,24}$
$q_{1,1} \ldots q_{1,24}\ q_{2,1} \ldots q_{2,24}\ \ldots\ q_{31,24}$
$q_{1,1} \ldots q_{1,24}\ q_{2,1} \ldots\ q_{2,24}\ \ldots\ q_{31,24}$
$q_{1,1} \ldots\ q_{1,24}\ \ldots\ q_{31,24}$

Fig. 8. Sliding Algorithm

Training Iris Data

Sepal Width	Sepal Length	Petal Width	Petal Height	Iris Class
5.1	3.5	1.4	0.2	I.Setosa
4.9	3.0	1.4	0.2	I.Setosa
….	….	….	….	….
….	….	….	….	….
….	….	….	….	….

Testing Iris Data

Sepal Width	Sepal Length	Petal Width	Petal Height	Iris Class
4.0	3.3	2.0	0.5	??

Fig. 9. Iris Data Classification

Training Signal

Feature A	Feature B			Label
				Label A
				Label B
….	….	….	….	….
….	….	….	….	….
….	….	….	….	….

Testing Signal

Feature A	Feature B	Label
		???

Fig. 10. Signal Classification

Scenario for testing data was divided into two groups: male testing and female testing. Both male and female testing share the same dataset containing mixed male and female artist of cover songs’. Result for each group was observed to find out if there is a difference between male and female cover song recognition. The accuracy for testing scenario was computed by following equation.

$$Accuracy\ (\%) = \frac{True\ Label}{Total\ Testing}\ x\ 100 \qquad (3)$$

True label from (3) means the correct label that system predict. Correct label could be obtained from result of modified KNN. If there were a correct label in nearest K compared label with testing data, then it would be counted as success. Value of K in this experiment was 5. Total testing was the total of testing data but separated between male testing and female testing.

1) Male Result

Table IV contains the result of male cover song recognition from the experiment. The result was pretty good because each original song was recognized by the system. The only one which got a bad result was Treat You Better cover song. This result occured because of many factors, one of them due to the fact that the recorded sound (testing data) was instrument dominant song; so when wavelet method was applied to testing data, the instrument spectrogram still contains instrument, not vocal only. But overall, the result of male cover song artist has 80% of accuracy.

TABLE IV. RESULT OF MALE COVER SONG, HIGHEST 5 RANKING

Title of cover song	**Highest 5 ranking**				
	Rank 1	***Rank 2***	***Rank 3***	***Rank 4***	***Rank 5***
Sky	Sky	Sky	Sky	Sky	Sky
Stitches	Stitches	Pillowtalk	Sky	Stitches	Heathens
Treat	Stitches	Pillowtalk	Pillowtalk	Pillowtalk	Pillowtalk
Heathens	Heathens	Treat	Heathens	Sky	Stitches
Pillowtalk	Pillowtalk	Sky	Sky	Sky	Pillowtalk

2) Female Result

Table V contained the result of female cover song recognition from this experiment. The result was good, but not good as male cover song recognition's result. From 5 testings, only 3 were accurate while the rest got a pretty bad result. The good result for a cover song from Table V was "Sky", "Heathens", and "Pillowtalk". The conclusion of good result was based on how many songs were recognized in the highest 5 ranking. This result occurs because of many factors. If the male result of cover song (Table IV) Treat You better was the bad result because of dominant instrument, the female artist of cover song was bad because there were some possibilities that female artist did some "improvement" on a few tones of the original artist's. This "improvement" will produce high spectrogram and it can be totally different from the original song. Different spectrogram can produce a different result for classification. Overall female cover song artist has 60% of accuracy.

TABLE V. RESULT OF FEMALE COVER SONG, HIGHEST 5 RANKING

Title of cover song	**Highest 5 ranking**				
	Rank 1	***Rank 2***	***Rank 3***	***Rank 4***	***Rank 5***
Sky	Sky	Sky	Pillowtalk	Stitches	Sky
Stitches	Sky	Heathens	Pillowtalk	Heathens	Sky
Treat	Sky	Sky	Heathens	Pillowtalk	Heathens
Heathens	Heathens	Sky	Sky	Heathens	Heathens
Pillowtalk	Sky	Sky	Pillowtalk	Sky	Pillowtalk

IV. CONCLUSION

Discrete Wavelet Transform only helps to de-noise a spectrogram of balance instrument (not dominant vocal or instrument). If the training or testing data was a dominant instrument, result from Discrete Wavelet Transform still, contain instrument. This instrument will produce spectrogram along with vocal and when matching process apply the result was not good because vocal dominant matched against vocal dominant with instrument remaining. The result of modified KNN for classifying spectrogram was 80% for male artist and 60% for a female artist, so the average accuracy was 70% for 10 testing data again 50 training data of cover song.

For future work, calculation for sliding algorithm can take parallel process not linear. This parallel process can be implemented by using thread programming. Thread programming will calculate the distance of testing data against each data in the dataset but in a parallel process. This method will reduce total duration for classification process, so the time of Cover Song Recognition system will be shortened.

REFERENCES

[1] H.G. Kim, N. Moreau, and T. Sikora, "MPEG-7 Audio and Beyond Audio Content Indexing and Retrieval," .

[2] B.T. Nugraha, R. Sarno, D.A. Asfani, T. Igasaki, and M.N. Munawar, "Classification of Driver Fatigue State Based on Eeg Using Emotiv Epoc +," J. Theor. Appl. Inf. Technol. Islamabad, vol. 86, no. 3, pp. 347–359, Apr. 2016.

[3] R. Sarno, M.N. Munawar, B.T. Nugraha, M.N. Munawar, and B.T. Nugraha, "Real-Time Electroencephalography-Based Emotion Recognition System," Int. Rev. Comput. Softw. IRECOS, vol. 11, no. 5, pp. 456–465, May 2016. doi : https://doi.org/10.15866/irecos.v11i5.9334

[4] S.D. You, W.H. Chen, and W.K. Chen, "Music Identification System Using MPEG-7 Audio Signature Descriptors," Sci. World J., vol. 2013, pp. 1–11, Mar. 2013.

[5] T. Bertin-Mahieux and D.P.W. Ellis, "Large-scale cover song recognition using hashed chroma landmarks," in 2011 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), 2011, pp. 117–120.

[6] "ISO/IEC 15938-1:2002 - Information technology -- Multimedia content description interface -- Part 1: Systems," ISO. [Online]. Available: http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=34228. [Accessed: 24-Dec-2016].

[7] "Programmers' Manual." [Online]. Available: http://mpeg7audioenc.sourceforge.net/programmer.html. [Accessed: 18-Dec-2016].

[8] "XQuery Tutorial." [Online]. Available: http://www.w3schools.com/xml/xquery_intro.asp. [Accessed: 18-Dec-2016].

[9] R.W. Dedy, R. Sarno, and Z. Enny, "Sensor Array Optimization for Mobile Electronic Nose: Wavelet Transform and Filter Based Feature Selection Approach," Int. Rev. Comput. Softw. IRECOS, vol. 11, p. 659, 20160831. doi : https://doi.org/10.15866/irecos.v11i8.9425

[10] C. Van Loan, Computational Frameworks for the Fast Fourier Transform. Society for Industrial and Applied Mathematics, 1992.

[11] D.R. Wijaya, R. Sarno, and E. Zulaika, "Information Quality Ratio as a novel metric for mother wavelet selection," Chemom. Intell. Lab. Syst., vol. 160, pp. 59–71, 2016. doi : http://dx.doi.org/10.1016/j.chemolab.2016.11. 012

[12] M.N. Munawar, R. Sarno, D.A. Asfani, T. Igasaki, and B.T. Nugraha, “Significant preprocessing method in EEG- Based emotions classification,” J. Theor. Appl. Inf. Technol., vol. 87, no. 2, pp. 176–190, May 2016.

[13] R. Sarno, B.T. Nugraha, M.N. Munawar, R. Sarno, B.T. Nugraha, and M.N. Munawar, “Real Time Fatigue-Driver Detection from Electroencephalography Using Emotiv EPOC+,” Int. Rev. Comput. Softw. IRECOS, vol. 11, no. 3, pp. 214–223, Mar. 2016. doi : doi.org/10.15866/irecos.v 11i3.8562

[14] N.S. Altman, “An Introduction to Kernel and Nearest- Neighbor Nonparametric Regression,” Am. Stat., vol. 46, no. 3, pp. 175–185, Aug. 1992