ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

A Review on Controlling Network Using Voice on Mobile Device with Enhanced Security

Radha Sharma, Dr. Dinesh Kumar

M.Tech Student, Department of Computer Science and Engineering, SRCEM at Palwal, MD University, Haryana, India.

Professor & HOD, Department of Computer Science and Engineering, SRCEM at Palwal, MD University, Haryana, India.

ABSTRACT: Imagine the scene, you've just left home or office and you've forgotten to turn your PC and network off, your local electricity company will be dancing with joy and your colleague could come to your desk to see that work you've been viewing or working on. What to do? Do you turn around and go all the way back? Well, now there's no need, because you can actually turn off your computer/network (and a whole lot more) by just clicking on the smart phone. There's no cost involved in all kind of this stuff. We only need an proposed application to shut down your machine, we are just one click away to shutting the computer down or execute another operations like playing music, accessing files, getting still screen shots, stopping the different applications that are running on the computer or network.

I.INTRODUCTION

Previously people used to go to each and every individual machine in the network and access resources on it. There is no way to do it using a single server which is connected in the same local area network. There were no ways to impose these rules for a remote server.

There is also the windows utility called remote desktop which gives us the ability to remotely connect to a computer/PC in the network, after getting connected to the computer the screen of the computer appears on the machine from where people are connecting. After the connection is successful people can control the PC as if it is their own PC and people are controlling it with their keyboard and mouse. But this windows utility is completely different from my project as it do not connect the machine but still can control the resources to lock and unlock them. This way is saves the processing power of both the server and the client computers, thus speeding up the process and also provides portability via a handheld phone.

In the proposed solution it will write an application in java with two different components as server component and client system with enhanced security. The client is basically the mobile application that has been created using Java & Android is installed in our mobile phone and the server is in a machine or laptops. It communicates among themselves through the wifi connection and allow the remote controller of the computer to control the machines. In the remote PC an application written in .Net works in the background that executes the control commands. A graphical user interface which is very user friendly and very easy to learn and understand for the end users is also developed.

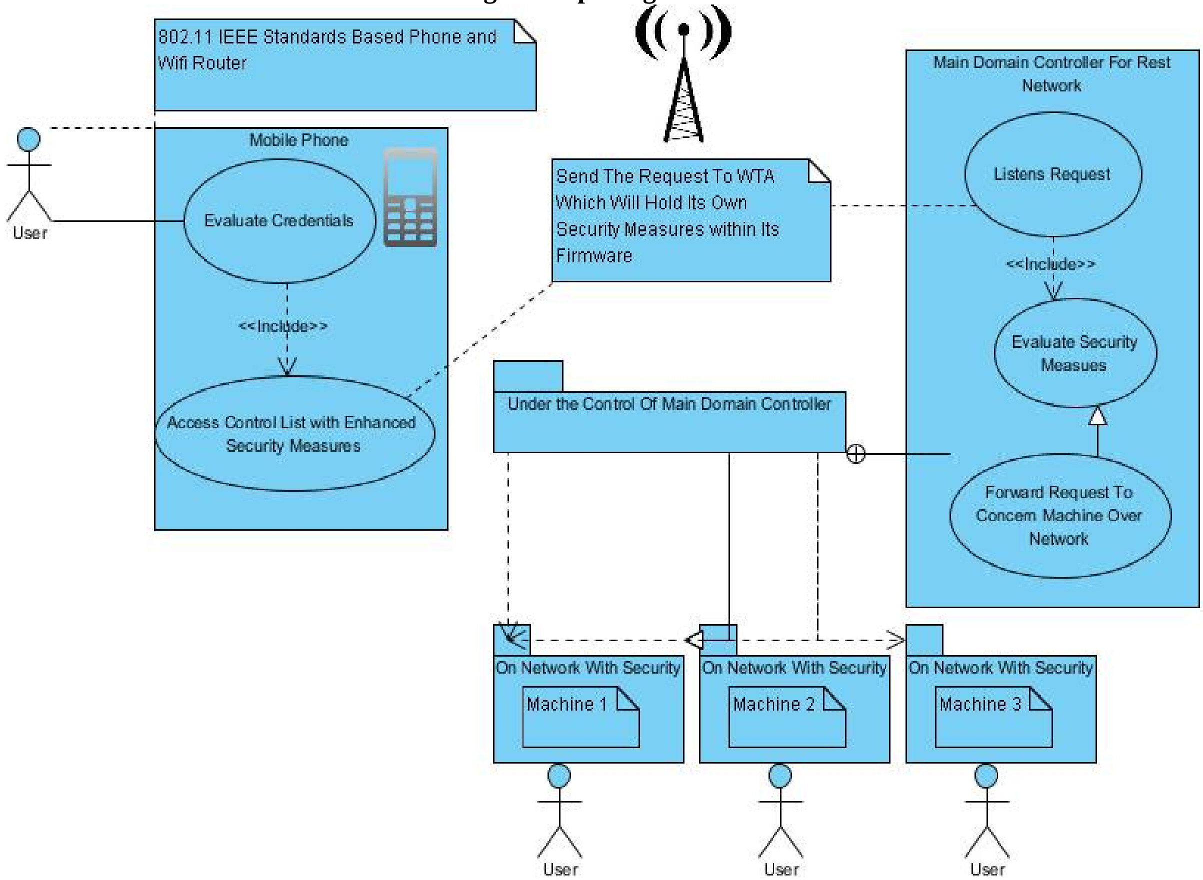
ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

Diagram depicting the scene



II.SPEECH RECOGONIZITION

Language IDs works as a single entity in many applications, but it is, in itself a set of three black boxes; *front-end processing system, phoneme recognizer, and language models*. Speech Data is given as an input to these set of boxes and then it flows into the system as shown in the figure. Implementation of every system is hidden from others; only interfaces are standardized as we do in case of OSI Layers of networking. By standardization, we mean the format of data, which will be passed from one system to another, is fixed.

Front-End Processing

Main purpose of front-end processing is the feature vector extraction. Many different algorithms exist for speech recognition and language identification. A common need between them is some form of parameterized representation (feature vectors) of the speech input. These feature vector streams may then be used to train or interrogate the language models which will follow the feature extraction module in a typical language identification system [6]. It is obvious that there exist an infinite number of ways to encode the speech, depending upon which particular numerical measures are deemed useful. Over the many years of speech recognition research, there has been a convergence towards a few (spectrally based) features that perform well. Of these, Linear Prediction (LP) and Cepstral measures are most widely used .

The final test for any such front-end is its effect on the accuracy of the overall language ID system. In this respect the system based on Cepstral compares favorably with any other we have come across in our investigation.

Phoneme Recognizer

The basic aim behind this system is to generate the phoneme sequences from the vector sequences. There are 56 phonemes, and their different combinations can represent all possible speeches in various languages. We used Hidden Markov Models (HMMs) for this purpose.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

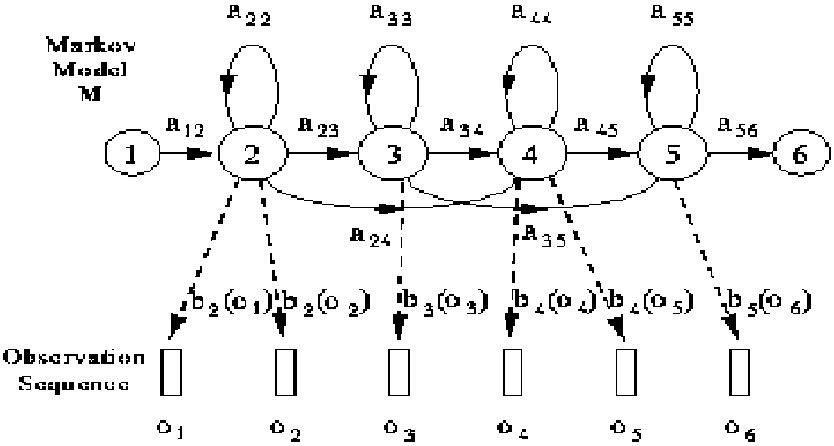
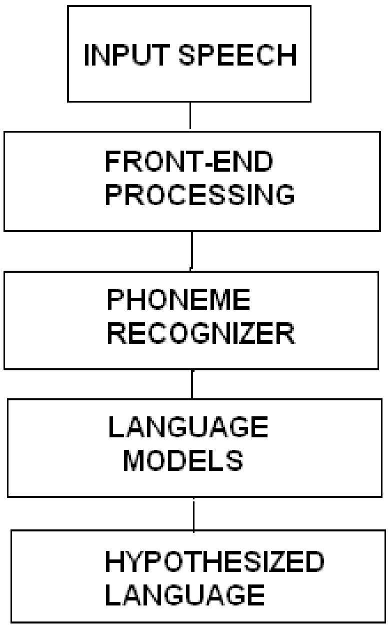


Fig. The Markov Generation Model

HMM models are primarily probabilistic state machines, in which each state represents a phoneme. Now there are two kinds of probabilities attached with each state. First, is the probability with which we can say which will be the next state (B_{xx}, as shown in fig.) and second, is the probability with which we can say what will be the output sound when this state is reached, which are represented by $A_x(O_y)$ in the figure. Basically HMMs are used for three problems, out of which one which we will be using it for is to find out the most probable state sequence given a sequence of output sound [1]. So basically what it does is that when processed speech vectors are passed through this system it gives sequence of phonemes. Usually you have phoneme recognizer specific to a language, depending on the training data used. We will discuss about this in later in white paper.

Diagram: Future Approach



Now before discussing each phase at implementation level lets talk about whole process of language recognition at an abstract level. In the first phase of language models large amount of training data is passed through the model along with the language information and no recognition takes place in this phase. Now based on training data various probabilities are calculated like probability of occurrence of given phoneme in given language. Now once all these

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

probabilities are calculated then next phase of recognition starts. In this phase also speech data is passed through models but now language information is not given to the system, in fact it's the system which predicts the language of the speech data by using the probabilities calculated in training phase. For example suppose speech data for English language was passed then in training phase what model will calculate is the probability like P(ph/E) which what is the probability of phoneme 'ph' occurring in English speech data. Now during recognition phase suppose some phoneme sequence ph1 ph2 ph3 occurred in the test data then the probability that this sequence is in English will be

P(ph1 ph2 ph3 | Eng) = P(ph1 | Eng) * P(ph2 | Eng) * P(ph3 | Eng)

Training Phase

Training phase is the most important phase among all the three, and decides how good or bad the whole system is going to perform. Main aim of this phase is to extract maximum possible information about a language from its training data. There are number of ways to do it. One of them is by finding the probability with which a given phoneme from a set occurs in that language. There are two issues to deal with in this probabilistic approach which are as follows:

i. **Method of finding the probabilities**: There are number of ways in which probabilities related to a phoneme can be found. One of them is *P(ph|X)*, which represent the probability of phoneme ph occurring in language X. This value is found by counting the number of times a given phoneme occurs in training data and then dividing it by the total number phonemes in the whole data and is calculated for every phoneme.

ii. **Type of probabilities**: Now there are various ways of capturing the language specific information from the training data. While selecting the appropriate method there are some parameters you should consider. First of them is the kind of application, language models are designed for. And second is the computational resource available. Like in our case where we are trying to design a language ID, we know that it's the order in which phonemes occur, makes one language different from other and then there are some phonemes which are specific to some languages and never occur in others.

So considering these factors a training model was designed in which when data is passed three types of probabilities are calculated.

1. **Unigram**: These are the probability of occurrence of single phoneme in language. These basically try to capture the distinct phonemes which are special to particular language like phonemes ending with \h\ are more probable in Hindi than in English.

   Uni_Prob(ph|X) =

   No. of time phoneme 'ph' occur in the training data

   Count of total no of phonemes in the training data for 'X'

2. **Bigram & Trigram**: This is the probability of a phoneme being followed by a given phoneme or pair of phonemes in a given language. This basically tries to capture the sequential information related to phonemes which is also specific to a language. This method is called as n-Gram approach, and we can go till any value of n but as you increase value of n complexity increases exponentially. Therefore we have calculated till n=3.

   Bi_Prob(ph2 | X, ph1) =

   No. of time phoneme ph1 is followed by phoneme ph2 in the data
   ----
   No of times phonemes ph1 occurs in the training data of language X

   Tri_Prob(ph3 | X, ph1, ph2) =

   No. of time phoneme ph1 is followed by ph2 and then ph3 in the data
   ----
   No. of time phoneme ph1 is followed by ph2 in the training data of language X

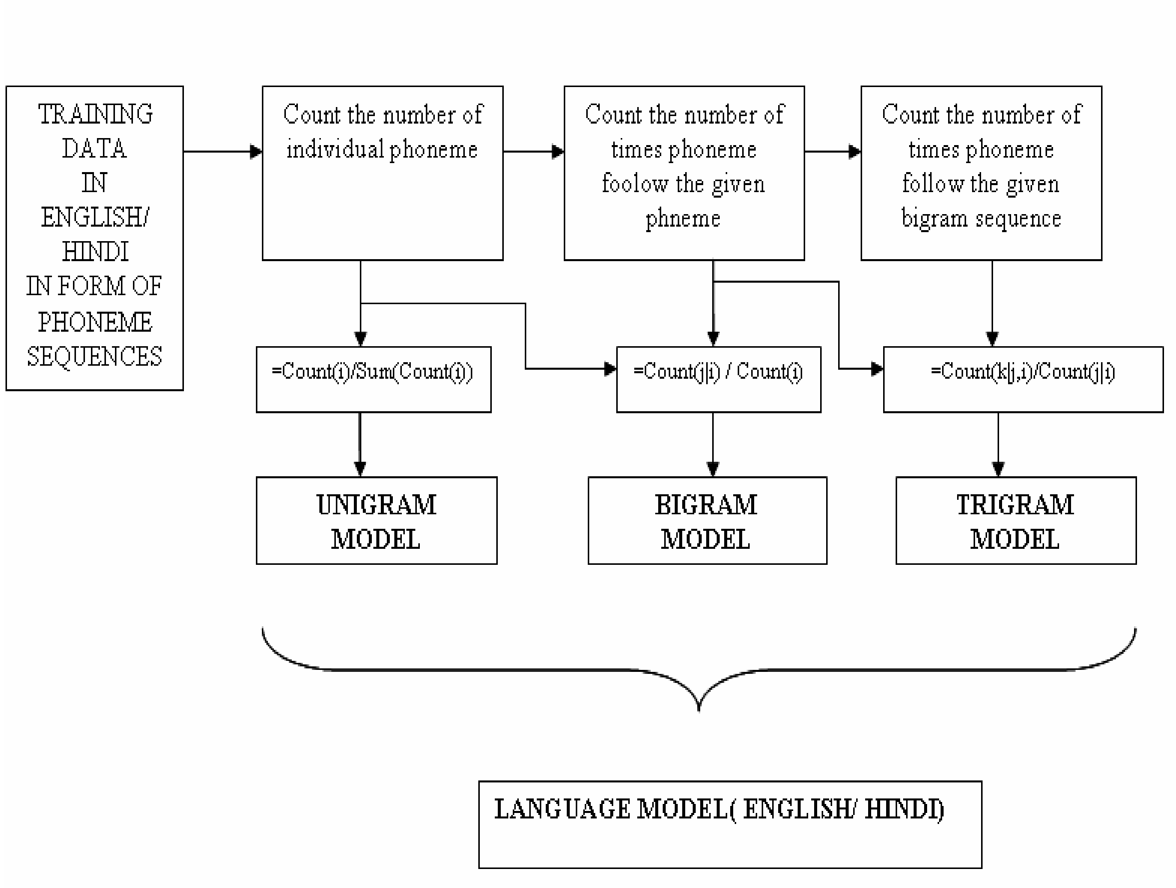
ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

So in a simple language in case of trigrams what you do is that instead of considering phoneme as single unit you consider sequence of three phonemes as a single unit to calculate the probability. Following diagram gives a pictorial view of the whole process:



So this was all about training phase of language models. The outputs of this phase are three probability files for each language as described above. Now before discussing about other phases lets discuss briefly how these are going to use these probability files. Let's suppose "a b c d e f "is a phoneme sequence, where a, b, c, d, e and f are different phonemes, came for recognition during testing phase, then different probabilities related to it will be:

Unigram_Prob = $P(a|X)*P(b|X)*P(c|X)*P(d|X)*P(e|X)*P(f|X)$……………..(1)

Bigram_Prob = $P(b|X, a)*P(c|X, b)*P(d|X, c)*P(e|X, d)*P(f|X, e)$………..(2)

Trigram_Prob = $P(c|X, a, b)*P(d|X, b, c)*P(e|X, c, d)*P(f|X, e, f)$………….(3)

III.SECURITY ALGORITHM

Accept voice input to start, stop, play, forward the songs from voice via to server next phase is to implement the below mentioned security algorithm:

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

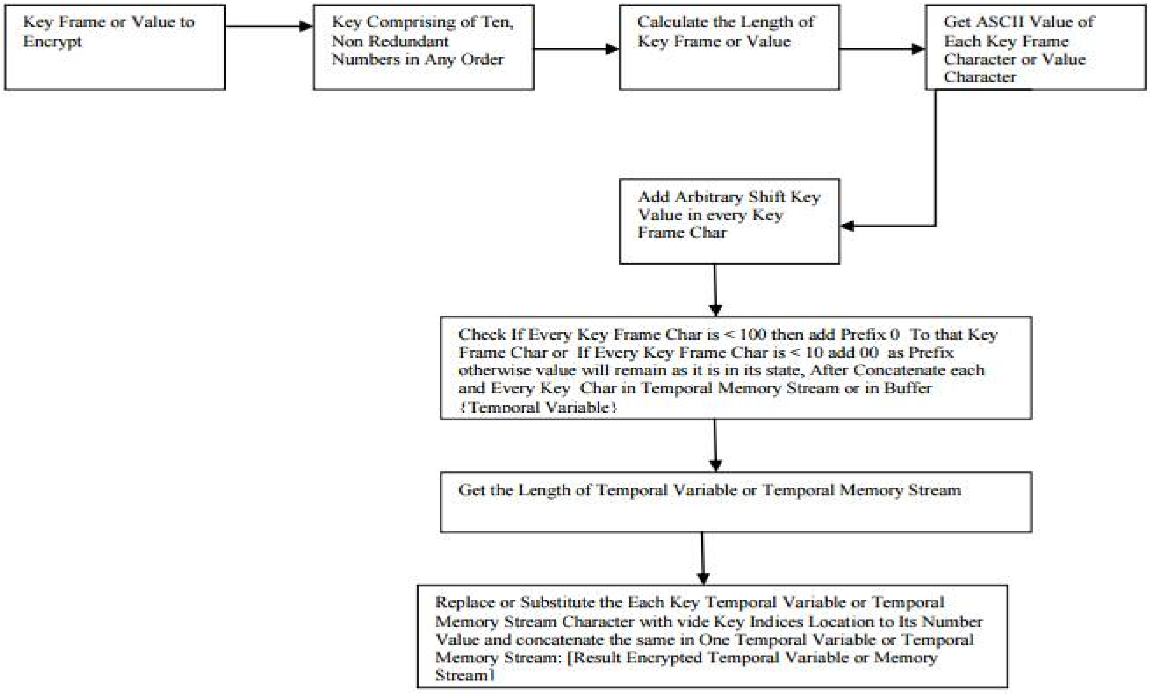


Figure3.1 Flow Graph of encryption algorithm

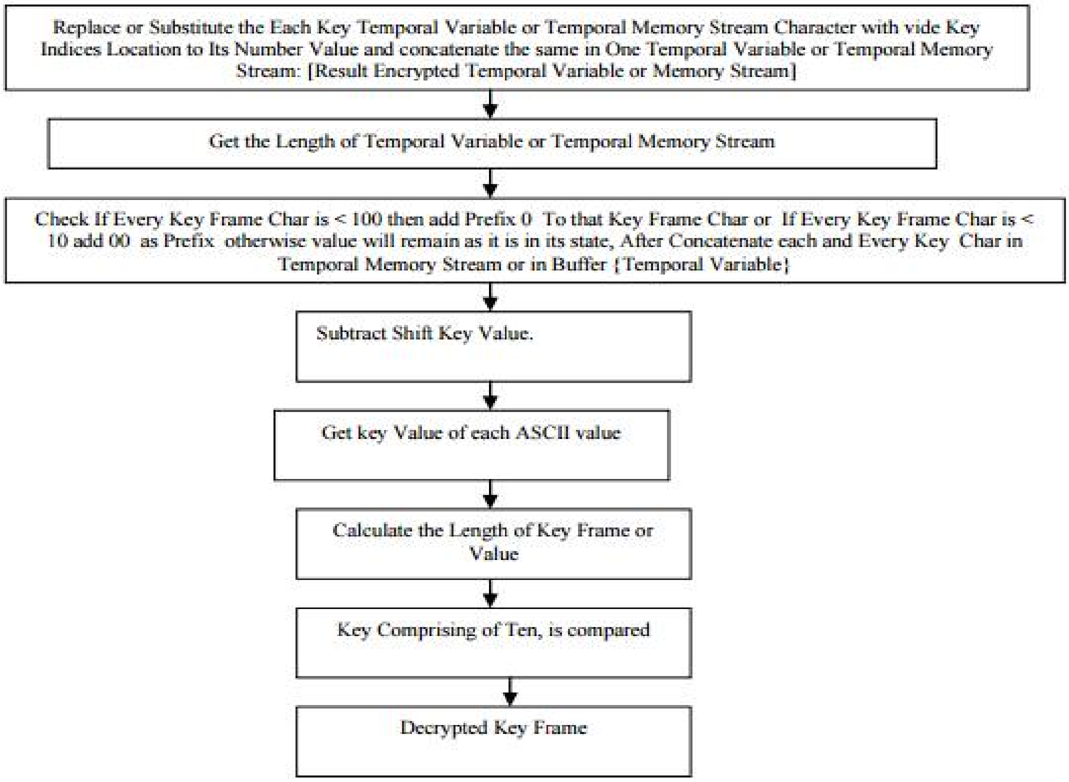


Figure3.2 Flow chart of Decryption

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

IV. CONCLUSION

With the above proposed scheme we can achieve high speed and performance, anywhere, anytime access of remote machine by the user, secured transfer of information via encryption decryption algorithm, single instruction on most modules allows the administrator to access the remote computer or network without doing anything on the remote side, Intrusion detection mechanism to provide improved security to the user to restrict access.

REFERENCES

1. A Survey on Wireless Security protocols (WEP, WPA and Wpa2/802.lli), ArashHabibiLashkari and Mir Mohammad seyedDanesh, September 2l, 2009 from IEEE Xplore
2. ASR Context- Sensitive Error Correction Based on Microsoft N-Gram Dataset, Youssef Bassil and Paul Semmaan, Journal of Computing, Volume 4, Issuel, January 20l2, ISSN 2l52-96l7
3. Discussion and a new attack of the optical asymmetric cryptosystem based on phase-truncated Fourier transform by Xiaogang Wang, Yixiang Chen, Chaoqing Dai, and Daomu Zhao in Sept 20l4
4. An Effective & Coherent Speech Recognition System-Review, Copyright (c) 20l5 CH. Suneetha. Kodell, Vijaya Kumar. Vadladi
5. Language and Dialect Identification: A survey, Etman, A. ; DSPRL, ECE Department, Virginia Tech, Blacksburg, USA ; Beex, A.A.Louis, SAI Intelligent Systems Conference (IntelliSys), 20l5
6. An analysis and comparison for popular video encryption algorithms, Saleh, M.A. ; Fac. of Electr. Eng., Univ. Teknol. MARA, Shah Alam, Malaysia ; Tahir, N.M. ; Hisham, E. ; Hashim, H., Computer Applications & Industrial Electronics (ISCAIE), 20l5 IEEE Symposium
7. Emotion recognition and affective computing on vocal social media Copyright © 20l5 Elsevier B.V. Published by Elsevier B.V. All rights reserved
8. In defense of stylistic diversity in speech researchCopyright © 20l4 Elsevier Ltd. All rights reserved.
9. Eardrum-Inspired Active Sensors for Self-Powered Cardiovascular System Characterization and Throat-Attached Anti-Interference Voice Recognition Authors Jin Yang, Jun Chen, Yuanjie Su, Qingshen Jing, Fang Yi, First published: l2 January 20l5
10. Emotion recognition from speech: a reviewKoolagudi, Shashidhar G. · Rao, K. Sreenivasa in *International Journal of Speech Technology (20l2)*
11. Robust emotion recognition in noisy speech via sparse representationZhao, Xiaoming · Zhang, Shiqing · Lei, Bicheng in *Neural Computing and Applications (20l3)*