Voice Code Verification System Supporting Multi-Modal Speech Interaction Using ASR and TTS

Heungkyu Lee[1] and Hanseok Ko[2]
[1]Dept. of Visual Information Processing, Korea University
[2]Dept. of Electronics and Computer Engineering, Korea University
hklee@ispl.korea.ac.kr, hsko@korea.ac.kr

Abstract

This paper proposes a voice code verification method for an intelligent surveillance guard robot, wherein a robot prompts for a code (i.e. word or phrase) for user entrance authentication. In the application scenario, the voice code can be changed every day for security reasoning and the targeting domain is unlimited. Thus, the voice code verification system not only requires the text-prompted and speaker independent verification but also it should not require an extra training model for speaker normalization. To resolve these issues, we propose to exploit the sub-word based anti-models for log-likelihood normalization model through reusing an acoustic model and competing with voice code model. In addition, the proposed system includes speech interaction tools for human and machine interaction on a same DSP board. The performance evaluation is achieved by using the PBW452DB, which consists of 63,280 utterances of 452 isolated word recorded in silent environment.

Key words: Voice code verification, Surveillance guard robot, and User entrance authentication.

1. Introduction

Recently, research works about human and machine interfacing on robots has been done. This includes efficient tools to facilitate control center design, to plan tasks in hazardous environments and to train works [1][2]. However, for security reasoning, some works [3][4][5] such as face recognition, fingerprint recognition and speaker verification have limitation. For surveillance task and the use of a robot, there exists user authentication tasks where the system does not need to know who the user is and this has only to check whether the user knows the authorized password or not on specific area and situation. This situation can be frequently occurred in artificial reality and telexistence technology.

This paper focuses on the security reasoning function that verifies the uttered speech code on a surveillance guard robot. For surveillance task, a lot of manpower at the sentry is placed on duty to guard the premise against unauthorized personnel for 24 hours. To lesson the time and overload of human guards at post, an intelligent surveillance guard robot is desirable. In this task, the one of a desirable password can be a prompted voice password (voice code).

This paper describes the security task where the system does not need to know the speaker and this has only to verify whether the uttered voice code is correct or not on specific area. To cope with this task, we could consider the utterance verification approach. Mostly, confidence measure (CM) for this task is used to verify the uttered observation sequences after or during calculating the probability of a word *W* being recognized by a speech recognition system. Besides the utterance verification, a filler model or garbage model can be used for these purposes. However, most algorithms require the extra models trained for a garbage model or anti-model. But limited memory size of our proposed embedded system prevents the algorithm from using and storing the extra alternative hypothesis model. Thus, the method that does not require the extra trained model and the re-use of the acoustic model is investigated for voice code utterance verification. To manage this problem, the anti-models that are re-usable from an acoustic model and can compete with a voice code model should be considered. In addition, the proposed system provides the human and machine interface using speech recognition and text-to-speech modules. The voice code verifier, speech recognition and text-to-speech engine are implemented on the same DSP board system.

The content of this paper is as follows. The voice code verification technique is presented in Section 2. In Section 3, we describe the multi-modal human and machine interaction for interacting between them. In Section 4, we conduct the experimental evaluation. Finally, in Section 5, we provide conclusive remarks.

2. Voice Code Verification

2.1 Utterance verification using anti-models

For the text-prompted and speaker independent verification method, we apply the utterance verification technique based on *N*-best rejection method using automatically produced anti-models against the statistical distance using the manner and place of articulation, and tongue advancement and aperture [6] when speaking a word as in Figure 1.

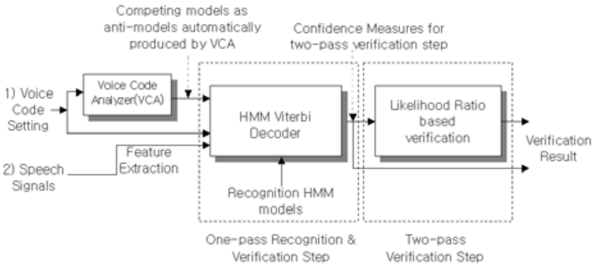

Fig 1. Block-diagram of the voice code verification

In general, sub-word based utterance verification and out-of-vocabulary rejection method [3] are based on likelihood ratio test (LRT) as follows.

$$LRT(X) = \frac{P(X/H_0)}{P(X/H_1)} = \frac{P(O_n/\lambda_n)}{P(O_n/\overline{\lambda_n})} \geq \eta \qquad (1)$$

where H_0 means that hypothesis is true and H_1 means that hypothesis is false. λ is the sub-word model and $\overline{\lambda}$ is the anti-model. In this paper, this is a word model that is connected with a phoneme model of a word. X is the uttered input observations that the number of sub-word is N.

To perform the LRT, we need an anti-model, hypothesis H_1 with respect to the hypothesis H_0. In case of utterance verification or speaker verification, an anti-phoneme model, which can affect the performance of LRT, is trained previously and then it used as a normalization model for LRT.

To reduce memory requirement on an embedded system, we reuse the acoustic model and apply two-pass strategy. Anti-models are automatically made according to the statistical distance of phonemes when the initial time of voice code verification is required. These anti-models are competed with the claimed voice code text during the decoding process as in equation (2), and used as a normalization model for LRT.

$$W_k = \arg\max_j L\left(O/\lambda_j, \overline{\lambda_{j+1,\ldots,N}}\right) \qquad (2)$$

where j is 1 and given voice code model, O is the observation sequence and W_k is the most likely word. If j is 1, voice code is recognized first. Then, for voice code verification, LRT is applied using N-best models that are the reused anti-models.

$$R_n = \frac{1}{l_n}\left[\log P(O_n/\lambda_n) - \frac{1}{nBest}\sum_{m=1}^{nBest}\log(O_n/\overline{\lambda_m})\right] \qquad (3)$$

The idea is to increase the likelihood of anti-models when someone speaks false voice code. Thus, construction of anti-models can affect the performance of proposed system.

For effective voice code verification, we need to define a function to combine the results of sub-word tests. A confidence measure (CM) for an input utterance O can be represented and applied as

$$CM(O) = f\left(CM_1, CM_2, \ldots, CM_N\right) \qquad (4)$$

where $f()$ is the function to combine the verification scores.

2.2 Sub-word based anti-models as competing models

The prompted voice text is automatically changed into phoneme string produced using grapheme to phoneme (G2P) converter through the text analysis. Then, the following rules for construction of anti-models are applied.

The voice code can be composed of concatenation of a syllable, S that is the set of phonemes. A voice code, W_0 is expressed by

$$W_0 = \{S_1, S_2, \ldots, S_N\} \qquad (5)$$

where N is the total number of syllable of a given voice code. At first, when a person says a similar word, this may result in a verification success. This is occurred when any person can say the word as follows.

$$\begin{aligned} \overline{W_1^1} &= \{\overline{S_1}, S_2, \ldots, S_N\}, \\ \overline{W_2^1} &= \{S_1, \overline{S_2}, \ldots, S_N\}, \ldots, \\ \overline{W_N^1} &= \{S_1, S_2, \ldots, \overline{S_N}\} \end{aligned} \qquad (6)$$

where N is the number of anti-syllable models for the first method and the variable, $\overline{S}$ is the anti-syllable. This sometimes results in a verification success. Thus, we can use the equation (6) as anti-models to prevent the false acceptance through competing with a voice code model when a person says a similar password. The anti-syllable model can be constructed using a concatenation of anti-phoneme against the each syllable unit as

$$\overline{S_N} = \{\overline{P_1}, \overline{P_2}, \ldots, \overline{P_M}\} \qquad (7)$$

At this time, each phoneme and anti-phoneme can be grouped using statistical distance of phonemes as in Table 1. In this paper, we use the 44 phonemes set for Korean voice code verification. The anti-phoneme is the one that is statistically far from the phoneme. Thus, the anti-phoneme becomes the one of phonemes according to the Table 1.

Table 1. the anti-model production rules according to the statistical distance of phonemes.

	Phoneme to Anti-phoneme		Standard
Consonant	Phoneme	g kh gg ng d th dd s ss j ch jj n r b ph bb m h	Manner and place of articulation
	Anti-phoneme	bb b b b gg g g bb b gg g g gg gg gg g g g b	
Vowel	Phoneme	a ya v i eui wi u yu eu e e we o yo yv wv wa ye yae	Tongue advancement and aperture
	Anti-phoneme	wi u wi a a a i a a wa wa a i a a e i a a	

To make the anti-model of each syllable, corresponding syllable in the prompted voice code is changed into anti-syllable using the anti-phoneme according to the Table 1 after the text is changed into the phoneme list using grapheme to phoneme converter, where it needs a parsing process to find the each syllable that is composed of consonant and vowel. In Korean language, syllable can be composed of "C+V", "C+V+C" and "V+C" where V is the vowel and C is the consonant [6]. Korean syllable can be classified into 6 groups as in Table 2. Using the rule of Table 2, a given text is classified into syllable lists.

Table 2. Korean syllable production rules

Syllable	Production rules	Group	Group number	Comments
CV	CV/CV	CV/CV (PART1)	1	
	CV/CVC			
	CV/VC	CV/V (PART2)	2	
	CV/V			
CVC	CVC/CV	CVC/C (PART3)	3	Part 4 follows the rule. part 1 according to the Korean utterance rule.
	CVC/CVC			
	CVC/VC	CVC/V (PART4)	1	
	CVC/V			
VC	VC/CV	VC/C (PART5)	4	Part 6 follows the rule, part 7 according to the Korean utterance rule.
	VC/CVC			
	VC/VC	VC/V (PART6)	5	
	VC/V			
V	V/CV	V/CV (PART7)	5	
	V/CVC			
	V/VC	V/V (PART8)	6	
	V/V			

Second, when any person utters the similar word that includes all parts of a prompted voice code, it often results in a verification success. It would be the time that any person utters a false text as follows

$$\begin{aligned} \overline{W_1^2} &= \{S_1, S_2, ..., S_N, \overline{S_{N+1}}\}, \\ \overline{W_2^2} &= \{S_1, S_2, ..., S_N, \overline{S_{N+1}}, \overline{S_{N+2}}\}, ..., \\ \overline{W_M^2} &= \{S_1, S_2, ..., S_N, \overline{S_{N+1}}, \overline{S_{N+2}}, ..., \overline{S_{N+M}}\} \end{aligned} \quad . \qquad (8)$$

where M is the umber of anti-syllable models to compete with a given voice code model, and anti syllable, $\overline{S_{N+M}}$ is matched to its syllable, S_N. To prevent this case, we use the equation (8) as anti-models. The anti-syllable model also can be constructed using Table 1.

Third, when any person says the similar word that is some part of the password text, this also often results in a verification success. It is the time that any person says a text as follows.

$$\begin{aligned} \overline{W_1^3} &= \{S_1\}, \\ \overline{W_2^3} &= \{S_1, S_2\}, ..., \\ \overline{W_{N-1}^3} &= \{S_1, S_2, ..., S_{N-1}\} \end{aligned} \qquad (9)$$

where $N\text{-}1$ is the number of anti-syllable models. To prevent this case, we use the equation (9) as anti-models. In addition, we can use the anti-models contrary to the equation (9) as follows.

$$\begin{aligned} \overline{W_1^4} &= \{\overline{S_1}\}, \\ \overline{W_2^4} &= \{\overline{S_1}, \overline{S_2}\}, ..., \\ \overline{W_{N-1}^4} &= \{\overline{S_1}, \overline{S_2}, ..., \overline{S_{N-1}}\} \end{aligned} \qquad (10)$$

After these anti-models are constructed through the analysis of a given voice code, all anti-models are used for competing with a voice code model. These models would increase the likelihood score of anti-models while the likelihood score is reduced when someone speaks a false word or phrase.

3. Human and Machine Interaction

Human-to-computer interfacing can be made more accessible and convenient if a conversational agent is applied. Conversation is one of the most important interactions that facilitate dynamic knowledge interaction. People can have a conversation with a conversational agent that can talk with people by using speech recognition and text-to-speech as a combined unit. This achieves command and control tasks while interacting with the human according to the given scenarios on the surveillance guard robot.

3.1 Speech recognition and text-to-speech

Whenever any person enters into the security area, some rules are followed. The intelligent surveillance guard robot performs these rules using speech recognition and text-to-speech interface. Speech recognition module [4] is designed to share the decoding routine with a voice code verifier because embedded system has a low memory and performance. Text-to-speech engine [4] has a role of broadcasting some information to communicate, and warning. This provides the human and machine interface for speech interaction when a person does a challenge.

Table 3. Priority control rules for scheduling; eAPV: embedded Automatic Password Verifier.

Next state Previous state	eASR(or eAPV) Request	eTTS request
EASR(or eAPV) running now	Previous eASR(or eAPV) is diable, requested eASR is enable	eASR continues to do and eTTS starts.
ETTS running now	Previous eTTS is pause, and then eASR (or eAPV) start. After eASR(or eAPV) is completed, eTTs resumes.	previous eTTS is stoped and requested eTTS starts.
Nothing is done	eASR(or eAPV) starts.	eTTS starts.

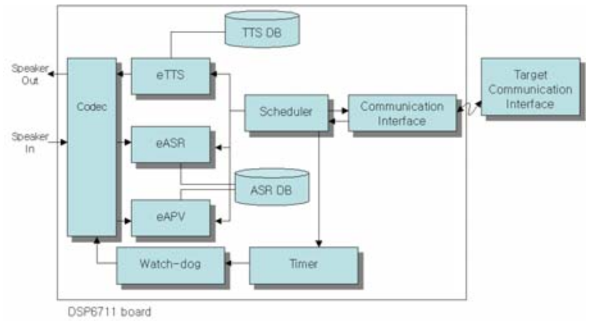

Fig 2. Block-diagram of the proposed system

3.2 Scenarios for voice code verification

Human can communicate with surveillance guard robot in order to enter a specific area by verifying the prompted voice code. For this task, the rule based speech interaction method using speech recognition and text-to-speech is required as in Table 3. To cope with unexpected events, the scheduler provides the priority control and resource management service [7].

The DSP board has a serial interface to communicate with central command and control center. Thus, the main application send the command code to this speech interactive tools, and then receive the responses from this via the serial interface. This includes the voice code verifier, speech recognition and text-to-speech modules on the same board. Thus, these engines should share the input and output channels as in Figure 2.

4. Experimental Evaluation

The voice code verification system is implemented on a DSP board for a surveillance guard robot. Training data set consists of about 120,000 utterances of 6,000 isolated words set recorded. Test data consist of 63,280 utterances of PBW452 isolated words recorded in silent environment.

We applied an utterance verification technique using *N*-best alternative hypothesis model [8] for likelihood normalization in LRT that is easy to implement and has low calculation time on a DSP board. The simulation is done using our proposed total anti-models using equations, (6), (8), (9) and (10). Figure 3 shows the simulation result. This result is improved by 16% than the one of utterance verification result in our previous work [8].

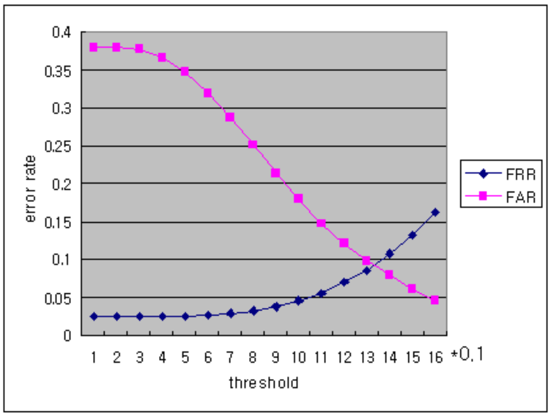

Fig 3. Simulation results of voice code verification

Table 4. EER under noisy environments.

		EER(Equal error Rate)			
		Clean	5dB	10dB	15dB
Clean DB	FRR	0.07673	-	-	-
	FAR	0.10962	-	-	-
Babble noise	FRR	-	0.08865	0.08806	0.08923
	FAR	-	0.09612	0.09592	0.09538
White noise	FRR	-	0.51826	0.37604	0.22187
	FAR	-	0.36229	0.32082	0.21829

This system can be usually utilized on outdoor surveillance region. Thus, This requires noise robustness for voice code verification to cope with environmental noise and other white noises. To resolve this problem, harmonics-based spectral subtraction algorithm [9] is applied for preprocessing the noise. The experiment is evaluated using PBW452 DB as in Table 4. In babble noise environment, EER (Equal Error Rate) did not show the rapid decrease of EER. However, in white noise, EER showed the rapid decrease in EER. But, It brought about 40% of relative improvement respectively than when there is no harmonics-based spectral subtraction algorithm.

Our proposed method for text-prompted and speaker independent verification showed that it could provide voice code verification function without extra trained model for likelihood normalization in *N*-best based LRT. The key point is to use the competing models that are anti-models using statistical distance of phonemes. This idea is due to the fact that the alternative model always follows the same state as the target model. Thus, if we can do modeling of the alternative hypothesis very well, we thought that voice code verification task could be solved by competing each other without extra trained models such as filler or garbage models. In some case, the traditional LLR does not find the most representative alternative hypothesis. Thus, the modeling of alternative model is a crucial issue for the voice code verification.

5. Conclusion

We proposed a voice code verification method using text-prompted speaker independent verification technique for a surveillance guard robot. In experiment, the result is improved by 16% than the result of general *N*-best utterance verification result. In addition, the performance showed the noisy robustness under noisy environment that brought about 40%, improvement.

References

1. Kaur K, Sutcliffe A, Maiden N, “Improving interaction with virtual environments,” In:Leevers DFA, Benest ID(eds), The 3D interface for the information worker, IEEE, London.
2. K. Oyama et al,. “Experimental Study on Remote Manipulation Using virtual Reality,” Presence, Vol. 2, No. 2, 1993, pp.112-124.
3. Tomoko Matsui, Sadaoki Furui, “Likelihood normalization for speaker verification using a phoneme- and speaker-independent model,” Speech Communication 17(1995) 109-116.
4. X. Huang, A. Acero and H. Hon, ***Spoken Language Processing***, Prentice Hall PTR, 2001.
5. R. Chellappa, C. Wilson, and S. Sirohey, “Human and Machine Recognition of Faces: A survey,” Proc. IEEE, vol. 83, no. 5, pp 705-740, 1995.
6. Willian J. Hardcastle and John laver, “The Handbook of Phonetic Sciences,” Blackwell publishers Ltd, 1997.
7. Gerard J. Holzmann, ***Design and Validation of Computer protocols***, Prentice Hall, 1991.
8. Taeyoon Kim and Hanseok Ko, "Uttrance Verification Under Distributed Detection and Fusion Framework", Eurospeech 2003, pp. 889~892, Sep, 2003.
9. Jounghoon Beh and Hanseok Ko, "A Novel Spectral Subtraction Scheme For Robust Speech Recognition: Spectral Subtraction using Spectral Harmonics of Speech," ICME 2003, III 633 ~ III 636, Jul, 2003.