Voice Recognition menggunakan RIPEMD-128

Hasanul Hakim / NIM : 13504091 [1)]

1) Program Studi Teknik Informatika ITB, Bandung,
email: if14091@students.if.itb.ac.id, haha_3030@yahoo.com

Abstract *– Fungsi hash adalah suatu fungsi dalam bidang kriptografi yang merupakan alat yang sangat penting dalam berbagai aplikasi kriptografi, sebagai contoh dalam pembentukan tanda-tangan digital, otentikasi, dan sebagainya. Semenjak ditemukannya algoritma hash MD4 telah banyak algoritma-algoritma hash yang lain yang telah dibentuk berdasarkan prinsip-prinsip dalam algoritma ini. Salah satunya adalah algoritma RIPEMD-128.*
Karena fungsi hash memiliki sifat memetakan berbagai pesan menjadi pesan ringkas yang sama dan sulit untuk menemukan pasangan input sedemikian sehingga keluarannya ekuivalen, maka fungsi hash dapat digunakan untuk pengenalan suara. Voice Recognition adalah isu modern yang rumit dan masih terus berkembang hingga proposal ini dibuat. Oleh karena itu, tulisan yang akan dibuat dibatasi berupa penggunaan RIPEMD-128 untuk voice recognition (speaker recognition) .

Kata Kunci*: Voice Recognition, Speaker Recognition RIPEMD-128.*

1. PENDAHULUAN

Di dalam kriptografi, terdapat fungsi yang berguna untuk aplikasi keamanan untuk menjaga integritas pesan dan otentikasinya. Fungsi tersebut adalah fungsi hash. Fungsi ini menerima masukan string yang panjangnya sembarang dan mengkonversinya menjadi string keluaran yang panjangnya tetap dan umumnya berukuran jauh lebih kecil daripada ukuran string semula.

Fungsi hash yang sering dan aman digunakan adalah fungsi yang termasuk ke dalam jenis fungsi satu arah (*one-way function*). Fungsi hash yang satu arah berarti pesan yang diubah menjadi *message digest* oleh fungsi tersebut tidak akan dapat dikembalikan lagi.

Bentuk persamaan umum fungsi hash adalah:
$h=f(X)$ (1)
f(X) atau h adalah hasil dari fungsi hash. Inputnya adalah X.

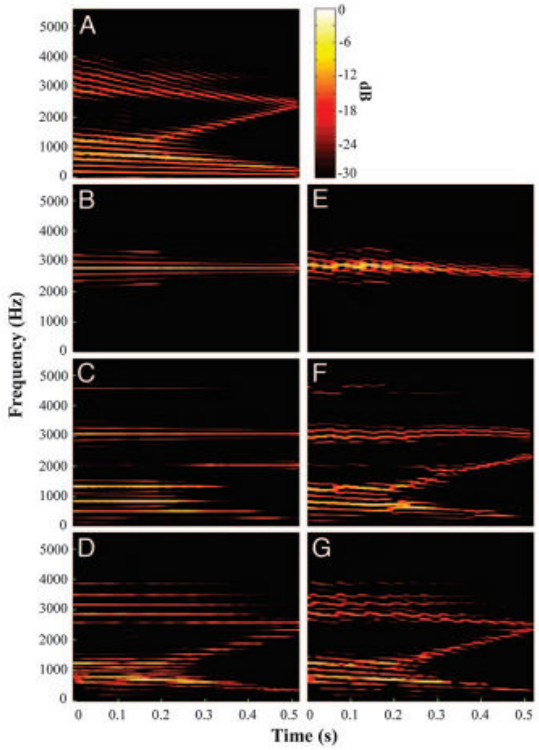
Prinsip-prinsip fungsi hash yang baik adalah sebagai berikut.

1. Fungsi f dapat diterapkan pada blok data berukuran berapa saja.
2. f menghasilkan nilai (h) dengan panjang tetap (*fixed-length output*).
3. f(X) mudah dihitung untuk setiap nilai X yang diberikan.
4. Untuk setiap h yang diberikan, tidak mungkin menemukan X sedemikian sehingga f(X) = h. Sifat ini menyatakan bahwa fungsi f harus merupakan fungsi satu arah. Sifat ini disebut juga dengan *preimage resistance*.
5. Secara komputasi tidak mungkin mencari pasangan nilai X dan Y sehingga f(X) = f(Y). Sifat ini disebut juga dengan second *preimage resistance*.
6. Untuk setiap X yang diberikan, tidak mungkin mencari $Y \neq X$ sedemikian sehingga f(Y)=f(X). Sifat ini disebut juga *collision resistance*.

RIPEMD diajukan konsorsium RIPE sebagai hasil realisasi dari hasil analisis terhadap MD4 dan MD5. Fungsi hash RIPEMD (RIPE *Message Digest*) adalah algoritma kriptografi hash yang ditujukan untuk implementasi software pada mesin berarsitektur 32-bit. Algoritma ini dikembangkan dari algoritma hash MD4 varian 256-bit yang pertama sekali diperlihatkan pada tahun 1990 oleh Ron Rivest. Fitur utama dari algoritma RIPEMD ini adalah adanya dua rantai komputasi yang berbeda, independen, dan paralel, yang hasil kedua komputasi ini kemudian digabungkan pada akhir prosesnya. Varian RIPEMD, RIPEMD-160 menghasilkan nilai hash 160-bit. Algoritma ini ditujukan untuk memberikan tingkat keamanan yang tinggi selama 10 tahun atau lebih. Algoritma varian RIPEMD yang lain, RIPEMD-128 adalah varian yang memiliki performa yang lebih cepat daripada RIPEMD-160. Oleh karena itu, pada masalah ini, RIPEMD-128 dipilih untuk digunakan sebagai alat untuk pembanding pada *voice recognition*, meskipun berpeluang lebih besar menimbulkan kolisi daripada RIPEMD-160

Voice recognition (speaker recognition) merupakan bagian dari ilmu komputer yang berkaitan dengan pendesainan sistem komputer yang dapat mengenali suara. *Voice recognition* adalah suatu proses untuk mengenali seseorang dengan mengenali suara dari orang tersebut.

Voice recognition adalah alternatif lain dari pengetikan pada *keyboard*. Sederhananya, seseorang berbicara ke komputer kemudian kata-kata muncul di layar.



Gambar 1 Spectogram dari sound /ai/ untuk berbagai pita amplitude modulation (AM)

Proses voice recognition dibagi menjadi dua tahap yaitu:

1. Verifikasi
2. Identifikasi.

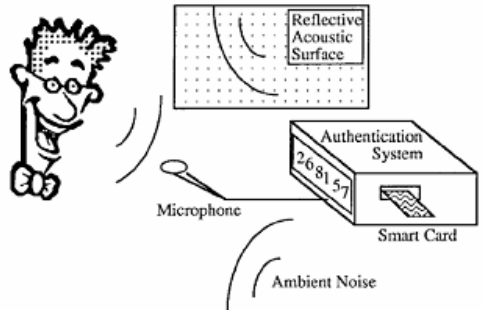
2. KONFIGURASI DAN TAHAPAN

Automatic speaker verification (ASV) adalah penggunaan dari sebuah mesin untuk membuktikan identitas yang diklaim oleh seseorang dari suaranya. Beberapa literatur menggunakan istilah yang berbeda untuk *speaker verification* termasuk *voice verification*, *speaker authentication, voice authentication, talker authentication* dan *talker verification*. Dalam *automatic speaker identification* (ASI), tidak ada pembuktian identitas yang diklaim dari sistem menentukan siapakah orang, anggota dari kelompok manakah orang tersebut, atau dalam kasus ini orang tersebut tidak diketahui.

Speaker verification didefinisikan sebagai proses penentuan jika seorang speaker adalah orang yang mengklaim dirinya. Hal ini berbeda dengan masalah *speaker identification*, yang berupa proses penentuan jika seseorang *speaker* adalah orang yang spesifik atau bagian dari sebuah yang terdiri dari beberapa orang. Dalam *speaker verification*, seseorang membuat sebuah klaim identitas (misalnya dengan memasukkan sebuah nomor karyawan atau smart card yang dimilikinya). Dalam *text-dependent recognition*, frasa diketahui oleh sistem dan dapat berupa frasa yang tetap atau dapat berubah. Orang yang mengklaim (*claimant*) mengucapkan suatu frasa ke dalam microphone. Sinyal ini dianalisis oleh sebuah sistem verifikasi yang membuat keputusan biner untuk menerima atau menolak klaim identitas user atau mungkin untuk melaporkan kepercayaan yang tidak cukup dan meminta input tambahan sebelum membuat keputusan.

Gambar 2 *speaker* memasukkan suaranya

Sebuah konfigurasi ASV terlihat pada Gambar 2. Claimant, yang sebelumnya direkam oleh sistem, memasukkan smart card yang mengandung informasi identitasnya. Kemudian berusaha untuk dikenali dengan mengucapkan sebuah frasa ke dalam microphone. Hal ini secara umum mencocokkan antara akurasi dan waktu pelaksanaan tes (*test-session duration*) sebagai tambahan dari suaranya, *ambient room noise* dan suara yang tertunda masuk ke microphone melalui permukaan reflektif akustik (*reflective acoustic surface*). Hal utama untuk sebuah sesi verifikasi, user harus merekam dalam sistem (di bawah kondisi yang diperhatikan). Selama proses perekaman, model suara dihasilkan dan disimpan (mungkin dalam sebuah smart card) untuk digunakan dalam sesi verifikasi berikutnya. Dalam hal ini juga mencocokkan antara akurasi dan durasi serta jumlah dari sesi perekaman.



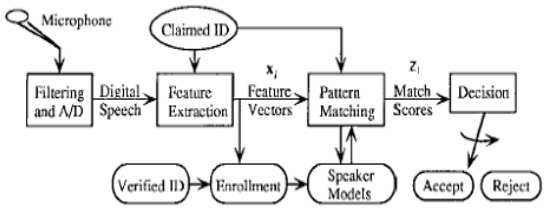
Gambar 3 konfigurasi ASV

Beberapa faktor dapat menyebabkan kesalahan dalam proses verifikasi dan identifikasi antara lain:

1. Kesalahan dalam pengucapan (*misspoken*) dan pembacaan (*misread*) frasa
2. Keadaan emosional yang ekstrim (misalnya stress)
3. Pergantian penempatan microphone (*intrasession* atau *intersession*)
4. Kekurangan atau ketidak-konsistenan akustik dari ruangan (misalnya *multipath* dan *noise*)
5. *Channel mismatch* (misalnya penggunaan microphone yang berbeda dalam perekaman dan verifikasi)

6. Sakit (misalnya flu yang dapat merubah *vocal tract*)
7. *Aging* (*model vocal tract* dapat berubah berdasarkan usia).

Pendekatan umum untuk ASV terdiri dari 5 tahap:
1. Digital speech data acquisition
2. Feature extraction
3. Pattern matching
4. Pembuatan keputusan: diterima atau ditolak
5. Perekaman untuk mendapatkan model speaker referensi.

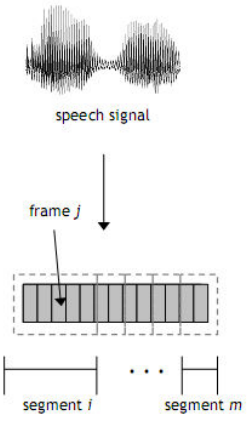


Gambar 4 sistem *speaker verification* yang umum

Awalnya, gelombang suara akustik diubah ke sebuah sinyal digital sesuai untuk *voice processing*. Sebuah microphone atau telephone handset dapat digunakan untuk merubah gelombang akustik ke dalam sebuah sinyal analog.

Sinyal analog ini dikondisikan dengan *antialiasing filtering* (dan mungkin filter tambahan untuk mengimbangi untuk setiap perusakan channel). *Antialiasing filter* membatasi *bandwidth* sinyal menjadi kira-kira *Nyquist rate* (setengah sampling rate) sebelum sampling. Sinyal analog terkondisikan kemudian diubah ke dalam bentuk sebuah sinyal digital oleh sebuah analog-to-digital (A/D) converter.

Dalam aplikasi *local speaker verification*, *channel analog* secara sederhana berupa microphone, kabelnya, dan *analog signal conditioning*. Kemudian, hasil sinyal digital dapat mempunyai kualitas yang sangat tinggi, tidak cukupnya distorsi dihasilkan oleh transmisi sinyal analog melalui jaringan telephone jarak jauh.



Gambar 5 Pemetaan sinyal suara menjadi segmen-segmen

Tugas dari *pattern matching* dari *speaker verification* meliputi perhitungan sebuah *match score*, yang menyatakan sebuah pengukuran dari kesamaan dari *input feature vector* terhadap beberapa model. Model speaker dibangun dari *feature* yang diekstrak dari sinyal suara. Untuk merekam user ke dalam sistem, sebuah model suara, tergantung pada *feature* yang diekstrak, dihasilkan dan disimpan (mungkin dalam sebuah *smartcard* yang berkode). Kemudian, untuk mengenali seorang user, *matching algorithm* membandingkan *score* sinyal suara yang baru masuk dengan model yang diklaim seseorang.

Fungsi hash ripemd-128 sepertinya dapat digunakan hampir disetiap tahapan *voice recognition*. Tahapan yang paling membutuhkan dukungan adalah pada pembentukan *feature vector* dan *auxiliary data.*

Ada dua tipe model yaitu *stochastic model* dan *template model*. Pada *stochastic model*, *pattern matching* adalah probabilistik dan hasil dalam sebuah pengukuran dari kemungkinan (*likelihood*), atau probabilitas keadaan, dari observasi diberikan model. Untuk *template model, pattern matching* adalah deterministik. Observasi diasumsikan menjadi sebuah replika yang tidak sempurna dari *template*, dan *alignment* dari *frame* yang diobservasi terhadap *frame template* dipilih untuk meminimalkan sebuah pengukuran perbedaan (*distance*) d. Kemungkinan L dapat diaproksimasi dalam model *template-based* oleh eksponensial *match score* yang diungkapkan

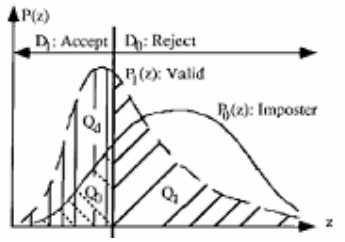
$$L = exp\ (-ad) \qquad (2)$$

Parameter a adalah sebuah konstanta positif (secara ekuivalen, score diasumsikan proporsional terhadap log kemungkinan). Perbandingan kemungkinan dapat digunakan menggunakan model *speaker global* atau kelompok (*cohost*) untuk normalisasi L.

Template model dan korespondensi pengukuran perbedaannya adalah model yang paling intuitif. Metoda *template* dapat tergantung (*dependent*) atau tidak tergantung (*independent*) terhadap waktu. Sebuah contoh dari semuah *time-independent template model* adalah VQ *modeling*. Semua variasi temporal diacuhkan pada model ini, dan *global average* (misalnya *centroid*) dari semua itu digunakan. Sebuah model *time-dependent* lebih rumit karena hal ini memperhatikan variasi pada *human speaking rate*.

Setelah menghitung *match score* antara *input speech-feature vector* dan sebuah model suara dari speaker yang diklaim, keputusan verifikasi dibuat untuk menerima atau menolak speaker atau meminta ungkapan lain (atau, tanpa sebuah identitas yang diklaim, sebuah keputusan identifikasi dibuat). Proses keputusan menerima atau menolak dapat berupa sebuah masalah penerimaan, kelanjutan, *time-out*, atau penolakan terhadap suatu pengujian hipotesis. Dalam

masalah ini, pembuatan keputusan, atau klasifikasi, prosedurnya adalah masalah pengujian hipotesis.



Gambar 6 Bentuk *match score* dari data yang valid dan *impostor* (penipu)

Nama dari area probabilitas pada gambar 7 diberikan pada tabel 1. Untuk mencari sebuah area performa probabilitas yang diberikan, hipotesis menerangkan melalui pdf (*probability density functions*) untuk menggabungkan, dan *threshold* menerangkan area keputusan membentuk batas integrasi.

Performa probabilitas	Keputusan D	Hipotesis H	Nama probabilitas	Hasil Keputusan	
Q_0	1	0	Ukuran test "signifikan"	Type I error	False acceptance atau alarm
Q_1	0	1		Type II error	False rejection
$Q_d = 1 - Q_1$	1	1	Power of test		True acceptance
$1 - Q_0$	0	0			True rejection

Tabel 1 Definisi dan keadaan probabilitas

3. HASIL PENGGUNAAN RIPEMD-128

Sifat dari fungsi hash adalah meskipun input yang diberikan berbeda sedikit, hasil yang diberikan sangat berbeda. Padahal, data biometrik terutama dalam hal pengenalan suara mengandung unsur ketidak-pastian. Pencocokannya tidak pasti/probabilistik sehingga penggunaan hash tidak cocok untuk masalah ini.

Berikut adalah beberapa hasil ekstrak suara untuk beberapa fonetik yang berkualitas jernih (sedikit *noise*).

Fonetik	Hasil ekstraksi (RIPEMD-128)
uh	4f1c64084f0ca5cfffec348732217c7c
er	bc6116641191e8d2e4c9b3b28343a2e9
aw	79c6762da58a48bed18027c44e9c0b46
eh	b388aa213a90a17d0383a51ef1450f40
ae	1096fe5fe0855bdd2bae4c1e9bfd8221
uw	bb5e0c20c47a566ce8ab60cecedaaa69
iy	b6d6674dffc52573298dfa359b77938d
ey	3833b5ead90cc0c2e0de1d484b21158f
ay	c24979b2591ab1a5e17403b2717e0f0e
ah	4e8554eb33d6b95d41505f01a3bd3cc0
aa	f4f44c1a01e8ed1a0490ab2671c2c907
ng	6fc6b41b597bdb52f978ff3101226a7e
n	85d0b37deb350aedf875d6d68c047078
w	df7b8236c762690a4e3ab318aa972a98
ih	141555c9407cdf3201c8b03c2ea8fefa
ow	6e2d2f5e30eab7a2ca95ebc13afd2a6e
y	0d300dec9b0f9e552341e20f6a4ec37d
l	cec7344cef7aeafafd12a4d60ec198e2
ao	22fdf17873d9f8560dea86af9fe747dd
m	4eeb2f91d51fc05c0b963f690c604143
ax	5f0f5a55c802de3c0d12d3cc7dbd4750
el	bb7b9f98df91c813bcda3397daa09f28
r	17b1436248bf9fe3170a94af8cb4f633
oy	ad447ae3d7181ff03812600e40e20bac
en	1f69f3c3be173b4026cf7e885f8dba0a

Tabel 2 hasil ekstraksi menggunakan RIPEMD-128

Pada saat yang lainnya, hasilnya berbeda (dengan kualitas yang sama).

Fonetik	Hasil ekstraksi (RIPEMD-128)
uh	9189d3afea2007a9d222a6cdc50933e2
er	0531a68b23b7354784c8a699a4ff26d0
aw	c5ac9b4d3391e389a90d400a677a28ec
eh	5e36c66d90a42762ce342c6113039173
ae	646454c4c3449f26c8233b3a48cbc330
uw	2da9cb003ef536ef1a06f7807781287a
iy	524e6816e918ffa79aaf3117eba70bc9
ey	4642f119218ef60944111ac8a257d9ca
ay	9572c4129798ae1010a0de6208fc6a10
ah	84e6f3075e8a127f51dcd54e619c5d17
aa	81c356e87321ba72c67e7c66cd4130ec
ng	2b0583fd335a0957b13155c7ca9634b2
n	69ed450c5b816c7958942c0daf17c1f6
w	4c4dd11c4d7a3d66e8aee82ad9bf62e7
ih	c92557609cb7b76c3bd8979133435c9b
ow	574b3277c8e3ace6457d1ab1631c3047
y	b9699d623dab2443fd22d7d5911eeb10
l	f7716916f940f70f4f9b2772afde5309
ao	401120717d03c2787614a0b185a78a3b
m	36e6b03fd59c72297cef4f60d39acb9a
ax	32c2f77278bf918509294f3ebd990ec3
el	e240fb7b763c39f1ff021b631baa5e66
r	4848e4f87d296ec450256e0e0f7f6cd8
oy	6ed47da2204fb694d31143d594139b42
en	e828cdbdf74cd040e4e6808bdc33b7e4

Tabel 3 hasil ekstraksi menggunakan RIPEMD-128

Berikut adalah hasil kualitas *voice recognition* ideal untuk beberapa fonetik dengan menggunakan suatu metode yang paling efektif.

Phones		Accuracy	Phones		Accuracy
uh	uh_cr	74.47%	w	w_cr	69.91%
er	er_cr	73.26%	ih	ih_cr	69.75%
aw	aw_cr	73.26%	ow	ow_cr	69.09%
eh	eh_cr	71.93%	y	y_cr	68.45 %
ae	ae_cr	71.52%	l	l_cr	68.23 %
uw	uw_cr	71.42%	ao	ao_cr	68.04 %
iy	iy_cr	70.51%	m	m_cr	67.79 %
ey	ey_cr	70.50 %	ax	ax_cr	67.24 %
ay	ay_cr	70.37 %	el	el_cr	66.85 %
ah	ah_cr	70.14 %	r	r_cr	66.36 %
aa	aa_cr	70.13 %	oy	oy_cr	63.24 %
ng	ng_cr	70.05 %	en	en_cr	58.19 %
n	n_cr	70.03 %			

Tabel 4 kualitas *voice* untuk setiap fonetik, cr menandakan fonetik berkeriuk-keriuk

4. KESIMPULAN

RIPEMD-128 merupakan fungsi hash yang digunakan untuk pembentukan *feature vector*. Meskipun input yang diberikan berbeda sedikit, hasil yang diberikan sangat berbeda.

Data biometrik terutama dalam hal pengenalan suara mengandung unsur ketidak-pastian. Pencocokannya tidak pasti/probabilistik sehingga penggunaan hash tidak cocok untuk masalah ini.

DAFTAR REFERENSI

[1] Munir, Rinaldi. (2006). "Diktat Kuliah IF5054 Kriptografi", Program Studi Teknik Informatika Sekolah Teknik Elektro dan Informatika , Institut Teknologi Bandung.

[2] Bishop, David, "*Introduction to Cryptography with Java Applets*", Grinnell college, 2003.

[3] Monrose, Fabian c.s., "*Using Voice to Generate Cryptographic Keys",* Bell Labs, Lucent Technologies.

[4] Campbell, Joseph P., *Voice Recognition*, in proceedings of IEEE, Vol. 85, No. 9, September 1997.