

## SISTEM PENGENALAN SINYAL UCAPAN BAHASA INDONESIA PADA PERANGKAT ANDROID

Suyanto<sup>1)</sup>, Jeffry Adityatama<sup>2)</sup>

Fakultas Informatika, Institut Teknologi Telkom (IT Telkom), Bandung  
email: 1) suy@ittelkom.ac.id, 2) jeffry.adityatama@yahoo.co.id

**Abstrak** – Makalah ini membahas pengembangan sistem pengenalan sinyal ucapan berbahasa Indonesia yang bisa digunakan untuk beragam aplikasi pada perangkat berbasis Android. Sistem ini dibangun menggunakan dua alat bantu, Pocketsphinx dan Java. Terdapat tiga aplikasi yang dijadikan studi kasus dengan tata bahasa dan kosakata berbeda, yaitu Pendiktean pesan singkat atau SMS, Layanan Direktori, dan Kontrol Perangkat. Pengujian terhadap ketiga aplikasi menunjukkan bahwa sistem pengenalan sinyal ucapan memiliki akurasi sangat tinggi untuk lingkungan dengan kebisingan rendah. Tetapi, akurasi akan sedikit menurun saat digunakan pada lingkungan dengan kebisingan yang relatif tinggi.

**Kata kunci:** pengenalan sinyal ucapan, bahasa Indonesia, perangkat Android, Pocketsphinx

### 1. PENDAHULUAN

Masyarakat Indonesia memiliki kebiasaan buruk dalam menggunakan perangkat komunikasi. Salah satunya adalah melakukan aktivitas pesan singkat dalam kondisi berbahaya dan bisa mengganggu kegiatan orang lain. Misalkan, saat sedang menyetir mobil atau motor, di kendaraan dan lokasi umum, sambil berjalan, di ruang rapat, saat mengajar di kelas atau melakukan pekerjaan yang menjadi profesinya. Kebiasaan buruk ini sulit dihilangkan meskipun sudah ada himbauan dari pemerintah daerah dan kepolisian. Sudah banyak kecelakaan lalu lintas yang terjadi akibat kebiasaan ini.

Salah satu solusi untuk mengurangi efek negatif dari kebiasaan tersebut adalah membangun sistem pengenalan suara untuk mempercepat aktivitas pembuatan SMS. Teknologi *handphone* dan sistem komunikasi bergerak saat ini memungkinkan untuk hal tersebut. Ada dua skema sistem pengenalan suara yang bisa digunakan. Pertama, *online system*, dimana sistem pengenalan suara berada di sisi *server* dan untuk menggunakannya pengguna harus selalu melakukan koneksi ke *server*. Ke dua, *offline system*, dimana sistem pengenalan suara diletakkan di *handset* sehingga pengguna tidak perlu terhubung ke *server*.

Salah satu contoh *online system* adalah Siri yang merupakan teknologi yang dibuat oleh Apple. Teknologi Siri berhasil meningkatkan penjualan iPhone secara signifikan. Fitur Siri ini sangat diminati karena akurasinya cukup tinggi. Siri menggunakan jaringan VoIP [7]. Suara dikirim dalam bentuk paket-paket data dan pengenalan suara diproses di *server*. Dengan cara ini, maka Siri memiliki dua kelebihan. Pertama, dengan menggunakan paket data, maka data suara tersebut bebas derau pada saat transmisi di jaringan komunikasi. Kedua, proses pengenalan di *server*, dengan kemampuan prosesor dan *memory* yang besar, memungkinkan penggunaan sistem pengenalan suara yang berkualitas bagus. Oleh karena itu, Siri memiliki akurasi sangat tinggi. Tetapi, satu kekurangannya adalah pengguna harus selalu mengaktifkan koneksi internet ketika menggunakan Siri. Hal ini tentu saja membuat pengguna

mengeluarkan biaya lebih mahal. Oleh karena itu, pengguna teknologi Siri di Indonesia kurang begitu banyak.

Masyarakat Indonesia lebih tertarik pada *handphone* berbasis Blackberry dan Android. Kedua teknologi ini saling berebut pasar Indonesia. Jumlah pengguna Blackberry masih paling banyak. Disusul dengan Android di posisi ke dua. Tetapi, ada satu kecenderungan menarik dimana masyarakat Indonesia mulai banyak tertarik pada *handphone* berbasis Android. Dua kelebihan Android adalah banyaknya aplikasi yang bisa dibangun dan harga yang relatif terjangkau. Pada kuartal terakhir tahun 2011, pengguna *smartphone* berbasis Android di Indonesia meningkat secara signifikan [2]. *Associate market analyst for Client Devices Research IDC Indonesia* menyatakan bahwa Android akan mendominasi pasar *smartphone* di Indonesia pada 2012 [2]. Beberapa pengembang di Indonesia telah mengembangkan banyak aplikasi Android seperti games, seni, musik, utilitas, dan sebagainya [3].

Kedua hal di atas, kebiasaan masyarakat Indonesia dan kondisi teknologi yang populer, menjadi motivasi kuat untuk mengembangkan aplikasi pendiktean SMS untuk perangkat berbasis Android. Aplikasi ini bersifat *offline* sehingga lebih murah dan cepat. Pengguna tidak perlu selalu terhubung ke internet. Tetapi, tentu saja jumlah kalimat SMS yang bisa dibuat hanya terbatas mengingat kapasitas *memory* yang relatif rendah dan prosesor yang tidak terlalu cepat. Untuk itu, aplikasi ini dibatasi pada kalimat dengan topik yang sangat penting ketika kondisi darurat. Misalkan, “Sorry, lagi di jalan. Nanti saya kontak lagi. Terima kasih.”, “Aku lagi rapat. Tunggu 5 menit ya. Ok.”, “Gue lagi makan. Setengah jam lagi ya. Makasih.”, dan kalimat-kalimat lain yang sering digunakan oleh pengguna ketika berada dalam kondisi darurat.

Selain pendiktean SMS, banyak aplikasi lain yang bisa dibuat untuk perangkat Android. Diantaranya adalah yang berhubungan dengan perjalanan (*traveling*) seperti layanan direktori, *voice command* untuk mengontrol perangkat pada mobil atau rumah,

permainan (*games*), rekomendasi resep masakan, alat bantu pendukung kegiatan bisnis, dan lain-lain.

Makalah ini membahas tiga aplikasi Android yang berhubungan dengan pengenalan sinyal ucapan, yaitu Pendiktean SMS yang diberi nama IVOMES (*Indonesian Voice Message Services*), Layanan Direktori yang diberi nama IVODES (*Indonesian Voice Directory Services*), dan Kontrol Perangkat pada mobil yang diberi nama IVOCOM (*Indonesian Voice Command*). Ketiga aplikasi dibangun menggunakan dua alat bantu, Pocketsphinx dan Java. Data suara (*speech corpus*) yang digunakan disini adalah yang dibangun oleh Telkom RDC (*Telkom research & development center*) [1]. Terdapat 43780 .wav yang diucapkan oleh 398 pembicara. Data teks (*text corpus*) dibangun dengan cara mengumpulkan pesan-pesan SMS dari sejumlah pengguna *handphone* dan *grammar* dibuat berdasarkan kebutuhan pengguna.

Makalah ini difokuskan pada investigasi penggunaan sistem pengenalan sinyal ucapan dalam hal akurasi dan tingkat kebisingan untuk ketiga aplikasi menggunakan sebuah *smartphone* dengan spesifikasi cukup tinggi.

## 2. ANDROID DAN POCKETSPHINX

Saat ini popularitas Android sangat cepat meningkat sejalan dengan makin banyaknya aplikasi yang disediakan di *market* secara gratis atau berbayar sangat murah. Apa dan bagaimana Android itu? Android adalah sistem operasi berbasis kernel linux 2.6 yang telah dimodifikasi sehingga bisa digunakan untuk perangkat bergerak.

SDK atau *software developmet kit* Android dapat diunduh secara gratis dari [5]. Oleh karena itu, kebanyakan aplikasi Android diimplementasikan menggunakan Java yang juga *open source* dan gratis. Android memanjakan para pengembang untuk mewujudkan segala ide inovatif dengan memberikan akses penuh ke *APIs framework* yang juga digunakan oleh aplikasi inti. Arsitektur Android dibangun sangat fleksibel sehingga komponen-komponennya bisa digunakan kembali (*reuseable*). Sejumlah aplikasi berbeda dapat mengakses suatu komponen dan mengubahnya, seperti menambah daftar kontak, SMS, atau mengatur alarm [5].

Sphinx adalah alat bantu (*toolkit*) untuk pengenalan suara yang juga *open source* yang dikembangkan oleh *Carneige Mellon University* (CMU), Amerika Serikat [4]. Sphinx memiliki sejumlah versi yang bisa digunakan pada *server*, komputer pribadi atau perangkat bergerak. Pocketsphinx adalah salah satu versi Sphinx yang dibuat khusus untuk perangkat bergerak [4]. Pocketsphinx menggunakan *Hidden Markov Model* (HMM) untuk membangun model akustik karena HMM memberikan performansi tinggi dan juga sangat tahan terhadap derau [6]. [7].

CMU Sphinx toolkit menyediakan alat bantu yang sangat umum digunakan oleh pengembang, yaitu sphinxbase-0.7, Pocketsphinx-0.7, CMU Sphinx4, cmuclmtk and sphinxtrain. Sphinxbase menyediakan sejumlah *libraries* yang akan dimanfaatkan oleh Pocketsphinx, CMU Sphinx dan sphinxtrain. Pocketsphinx dan CMU Sphinx berfungsi sebagai

mesin pengenalan suara (*speech recognition engine*) untuk mengkonversi suara masukan menjadi teks. Sedangkan cmuclmtk dan sphinxtrain digunakan untuk membangun model bahasa dan model akustik.

## 3. DESAIN APLIKASI

Ketiga aplikasi didesain untuk bisa mengakomodasi kebutuhan masyarakat Indonesia yang mempunyai beragam dialek berbeda dan dapat digunakan untuk *smartphone* kelas rendah hingga tinggi. Dua isu penting yang sangat dipertimbangkan adalah akurasi dan tingkat kebisingan. Oleh karena itu, desain ini dibuat agar bisa diimplementasikan menggunakan alat bantu Pocketsphinx dan Java. Pocketsphinx dibangun untuk *embedded platforms* yang menyediakan *Hidden Markov Model* (HMM) untuk membangun *acoustic model* dan sejumlah fungsi untuk membangun *language model*. Dekoder Pocketsphinx bekerja sangat cepat karena menggunakan HMM semi kontinyu [6]. Berikut ini adalah desain ketiga aplikasi yang berupa kebutuhan pengguna, data suara, dan *Graphical User Interface* (GUI).

### 3.1 Aplikasi IVOMES

Aplikasi pendiktean SMS ini ditujukan untuk pengguna *handphone* di Indonesia, khususnya yang sedang berada dalam kondisi darurat. Misalkan, sedang menyetir mobil atau berada dalam acara penting, sehingga pengguna merasa kesulitan menulis SMS menggunakan *keyboard*. Berhubung kemampuan perangkat Android yang masih terbatas, baik prosesor maupun *memory*, maka aplikasi ini membutuhkan alat bantu yang berukuran kecil dan kompleksitas komputasinya relatif rendah.

Tiga hal yang dibutuhkan pengguna dalam melakukan aktivitas SMS adalah, mudah, akurat, dan cepat. Untuk memenuhi hal itu, IVOMES didesain dengan tiga prinsip, yaitu:

**Simpel** : lebih sederhana dan lebih mudah digunakan dibandingkan mengetik menggunakan *keyboard*.

**Akurat** : bisa mengenali beragam kalimat pesan yang diharapkan pengguna dengan akurasi tinggi.

**Cepat** : waktu yang diperlukan untuk mendikte menggunakan suara harus lebih cepat dibandingkan mengetik menggunakan *keyboard*.

Desain aplikasi IVOMES difokuskan pada dua hal: 1) bagaimana membangun *dictionary* dan *grammar* yang bisa menangani kalimat-kalimat pesan yang sering digunakan pada kondisi darurat. Di sini akan diteliti seberapa besar ukuran *dictionary* dan *grammar* yang masih memberikan akurasi yang bisa diterima; 2) bagaimana membangun kamus ejaan sehingga bisa, secara akurat, mengenali suara pengguna yang memiliki beragam dialek: Batak, Betawi, Jawa, Sunda, Padang, Bali, dan sebagainya.

Terdapat dua *grammar* yang digunakan untuk aplikasi IVOMES, yaitu yang terdiri dari 100 kalimat dan yang

Terdapat dua *grammar* yang digunakan untuk aplikasi IVOMES, yaitu yang terdiri dari 100 kalimat dan yang berbasis aturan (*rule-based grammar*) dengan jumlah kalimat yang lebih banyak. Akan diteliti sampai

seberapa besar *grammar* yang masih bisa diterima secara akurasi.

### 3.2 Aplikasi IVODES

Aplikasi layanan direktori ini diharapkan bisa memenuhi kebutuhan pengguna dalam mencari lokasi penting yang diinginkan. Pada tahap awal ini, aplikasi akan difokuskan pada 100 nama lokasi yang diambil secara acak, baik itu hotel, restoran, rumah sakit, kantor polisi, dan sebagainya.

Dengan demikian, *grammar* yang digunakan untuk aplikasi ini hanya berisi 100 kalimat yang merupakan nama lokasi. Jumlah kata pada kamus ejaan tentunya lebih sedikit dibandingkan dengan aplikasi IVOMES.

### 3.3 Aplikasi IVOCOM

Aplikasi pengontrolan perangkat ini difokuskan pada kebutuhan pengguna untuk mengontrol perangkat pada mobil, misalkan AC, CD, DVD, radio, lampu, pintu dan jendela. Kalimat yang bisa dibangun hanya terbatas pada pengontrolan perangkat tersebut. Misalkan, “Nyalakan AC”, “Matikan DVD”, “Buka pintu”, “Tutup jendela”, dan sebagainya.

*Grammar* dan kamus ejaan yang digunakan untuk aplikasi ini tentu saja jauh lebih kecil dibandingkan *grammar* untuk aplikasi IVOMES.

### 3.4 Data suara

Ketiga aplikasi dikembangkan menggunakan data suara (*speech corpus*) yang dibangun oleh *Telkom research and development center* (Telkom RDC) [1]. Terdapat 43.780 file .wav yang diucapkan oleh 398 pembicara, 199 pria dan 199 wanita, dimana setiap pembicara mengucapkan 110 kalimat. Mereka memiliki usia yang beragam dari remaja hingga lansia dan menggunakan empat dialek berbeda, yaitu Batak, Betawi, Jawa dan Sunda.

### 3.5 Perangkat *smartphone*

Saat ini terdapat banyak sekali *smartphone* yang menggunakan sistem operasi Android dengan berbagai versi berbeda. Banyak produsen *smartphone* yang telah meluncurkan produk Android-nya. Setiap perangkat bisa memiliki beragam prosesor dan memori. Kualitas perangkat, khususnya *microphone*, juga sangat beragam.

Tabel 1. Spesifikasi perangkat *smartphone*

Prosesor	1.4 GHz
Sistem Operasi	Android 2.3 (Ginger Bread)
Memori	512 MB
External store SD card	2 GB

Aplikasi akan diuji menggunakan *smartphone* yang sangat populer di masyarakat Indonesia yang dibuat oleh sebuah produsen sangat terkenal. Spesifikasi *smartphone* diilustrasikan pada tabel 1. Tentu saja akurasi aplikasi bisa saja menurun atau bahkan meningkat jika diuji menggunakan *smartphone* dengan kualitas perangkat keras yang berbeda.

### 3.6 Desain GUI

Pada pengembangan ini, desain GUI lebih difokuskan pada aplikasi IVOMES. Untuk sementara, dua aplikasi lainnya menggunakan GUI yang sama dengan IVOMES. Terdapat enam menu yang dibuat sangat sederhana, yaitu:

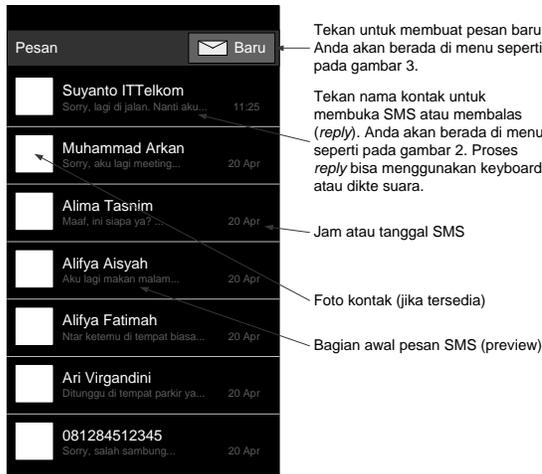
- Menu Utama;
- Buka dan Balas SMS;
- Buat SMS Baru;
- Cari Kontak;
- Dikte SMS; dan
- Pengaturan Tingkat Kebisingan.

Untuk aplikasi ini, GUI sengaja didesain menggunakan bahasa Indonesia, bukan bahasa Inggris, agar mudah bagi pengguna. Sejumlah tombol interaksi dibuat berukuran besar untuk tingkat kesalahan relatif rendah. Berikut ini adalah sejumlah gambar GUI untuk semua menu aplikasi IVOMES.

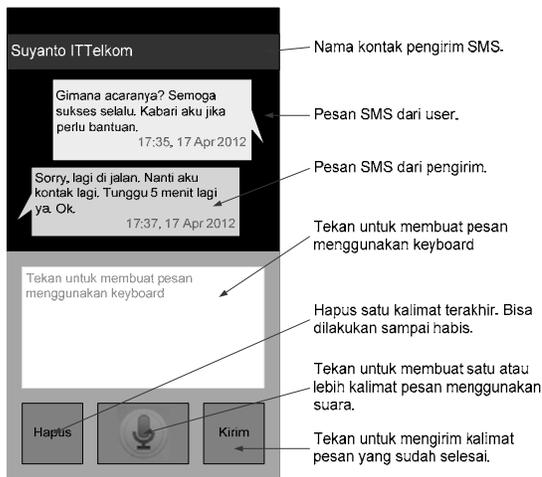
Penjelasan dari semua desain GUI adalah sebagai berikut. Aplikasi IVOMES ditujukan untuk menggantikan aplikasi *Messaging* yang menjadi *default* dari *smartphone* Android saat ini. Dengan demikian, semua menu yang ditawarkan IVOMES minimal sama dengan aplikasi *Messaging* tersebut. Menu utama IVOMES berisi Kotak Pesan dan Pesan Baru seperti pada gambar 1. Saat pengguna menekan nama kontak pada menu utama tersebut, maka pengguna akan berada pada menu Buka dan Balas seperti gambar 2 di atas. Sejumlah dialog yang tersimpan di dalam basisdata bisa ditampilkan dalam format seperti pada gambar 2 tersebut. Untuk membalas (*reply*) SMS, pengguna bisa menggunakan *keyboard* atau suara. Tekan *textbox* besar berwarna putih untuk membuat pesan menggunakan *keyboard*. Tekan tombol *microphone* abu-abu pada bagian tengah bawah untuk membuat pesan dengan cara mendikte menggunakan suara. Pengguna juga bisa menggunakan gabungan suara dan *keyboard* secara fleksibel. Misalkan, kalimat pertama dan ke dua menggunakan suara, sedangkan Kalimat ke tiga menggunakan *keyboard*.

Menu Buat SMS Baru didesain sangat simpel seperti pada gambar 3. Terdapat dua bagian utama, yaitu 1) **Cari Kontak**, yang penjelasan detailnya diilustrasikan oleh gambar 4; dan 2) **Buat Pesan**, yang penjelasannya dapat dilihat pada gambar 5.

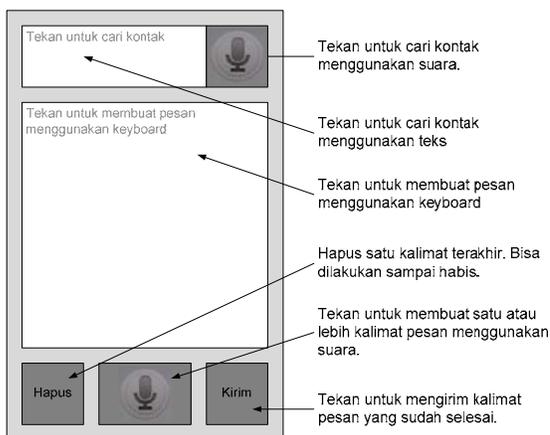
Pada bagian pertama, pengguna bisa melakukan pencarian kontak menggunakan suara atau *keyboard*. Tekan tombol *microphone* sebelah kanan atas untuk mencari kontak menggunakan suara. Tekan *textbox* sebelah kiri atas untuk mencari kontak menggunakan *keyboard*.



Gambar 1. Menu Utama



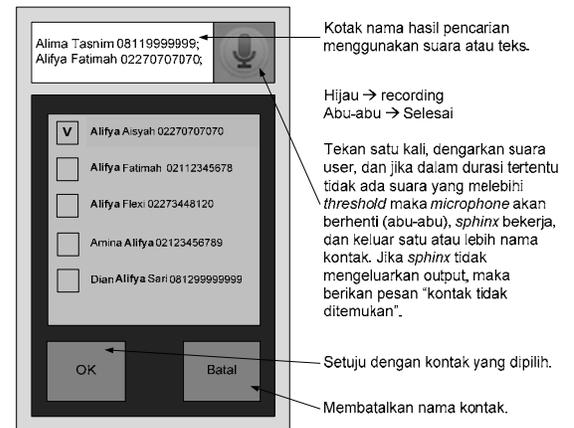
Gambar 2. Menu Balas SMS



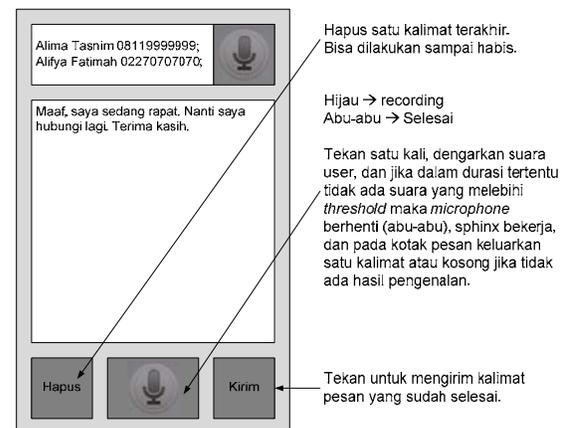
Gambar 3. Menu Buat SMS Baru

hijau; 3) diktekan satu kalimat pesan; 4) tunggu hingga satu kalimat pesan tersebut tertulis di *textbox* dengan tanda titik atau tanda tanya. Selanjutnya, pengguna bisa mendiktekan kalimat ke dua dan seterusnya. Pengguna bisa mendiktekan sejumlah kalimat pesan yang diinginkan. Untuk menghentikan proses pendiktean pesan, pengguna cukup menekan tombol *microphone* hingga berwarna abu-abu. *Textbox* besar bagian tengah untuk menulis pesan menggunakan *keyboard*. Jika pengguna sudah mendiktekan pesan dan ingin menambahkan pesan memakai *keyboard*, pengguna bisa menekan *textbox* besar bagian tengah dan menambahkan pesan menggunakan *keyboard*.

Saat mendiktekan kalimat pesan menggunakan suara, pengguna bisa melakukannya sesuai dengan tingkat kebisingan lingkungan dan kecepatan suara yang bisa diatur pada menu pengaturan seperti pada gambar 6. Jika tingkat kebisingan rendah (lingkungan tenang), pengguna bisa berbicara pelan (bahkan berbisik). Tetapi, untuk lingkungan yang sangat bising, pengguna harus berbicara dengan suara keras (melebihi suara bising di sekitar pengguna) dan dekat dengan *microphone* pada *handphone* pengguna.

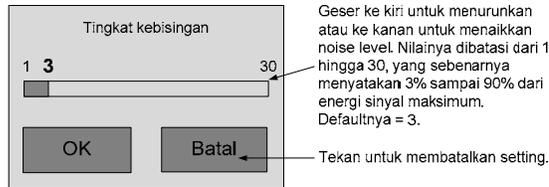


Gambar 4. Menu Cari Kontak



Gambar 5. Menu Buat Pesan

Pada bagian ke dua, pengguna bisa membuat kalimat pesan menggunakan suara atau *keyboard* atau gabungan keduanya. Caranya adalah: 1) tekan tombol *microphone* abu-abu di bagian tengah bawah; 2) diktekan kalimat pesan saat *microphone* berwarna



Gambar 6. Menu Pengaturan Tingkat Kebisingan

#### 4. PENGUJIAN DAN DISKUSI

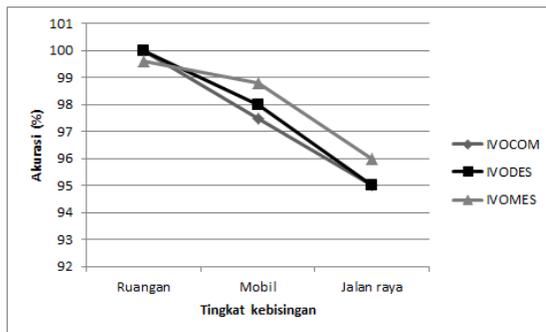
Sistem pengenalan ucapan diuji di dalam banyak kondisi dengan tingkat kebisingan berbeda untuk melihat kelayakan penggunaannya. Di sini, pengujian difokuskan pada tiga kondisi saja, yaitu: 1) di dalam ruangan tertutup dengan tingkat kebisingan di bawah 5%; 2) di dalam mobil berjalan di keramaian lalu lintas dan suara radio dengan tingkat kebisingan sekitar 20%; dan 3) di pinggir jalan raya dengan tingkat kebisingan sekitar 40% dari suara pengguna.

Gambar 7 menunjukkan grafik pengujian sistem pengenalan ucapan pada ketiga tingkat kebisingan tersebut. Akurasi sistem mengalami penurunan yang relatif rendah, hanya sekitar 5%, meskipun tingkat kebisingan mencapai 40%.

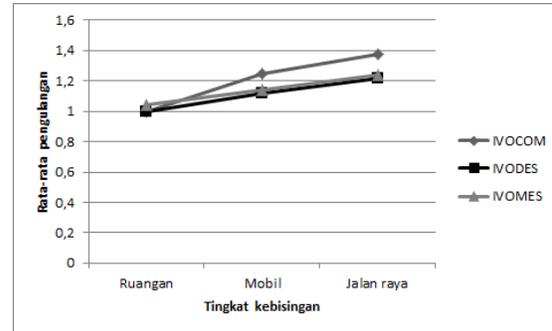
Pengujian juga dilakukan untuk mengetahui berapa kali pengulangan yang harus dilakukan pengguna sampai suaranya bisa dikenali oleh sistem dengan benar. Gambar 8 menunjukkan bahwa diperlukan jumlah pengulangan lebih banyak ketika pengguna berada di lingkungan yang semakin bising.

Ketika di dalam ruangan, pengguna tidak memerlukan pengulangan ucapan. Tetapi, ketika di pinggir jalan raya, pengguna memerlukan pengulangan untuk sejumlah kalimat tertentu sehingga rata-rata pengulangannya mendekati 1,4 kali. Tetapi hal ini relatif masih bisa diterima dibandingkan pengguna harus menggunakan *keyboard* untuk menulis SMS.

Kedua hasil pengujian di atas menunjukkan bahwa sistem ini layak digunakan saat pengguna berada dalam kondisi darurat.



Gambar 7. Pengaruh tingkat kebisingan terhadap akurasi sistem pengenalan sinyal ucapan



Gambar 8. Pengaruh tingkat kebisingan terhadap jumlah pengulangan sampai suara pengguna bisa dikenali dengan benar

#### 5. KESIMPULAN

Akurasi sistem pengenalan sinyal ucapan untuk ketiga aplikasi sangat layak digunakan mengingat akurasi yang relatif tinggi pada lingkungan dengan tingkat kebisingan rendah. Akurasi sistem hanya menurun sedikit meskipun digunakan pada lingkungan dengan tingkat kebisingan yang relatif tinggi.

#### DAFTAR REFERENSI

- [1] Suyanto, "Signal Energy-Based Automatic Speech Splitter: a tool for Developing Speech Corpus", IEEE TENCON 2007, 30 October - 02 November 2007, Taipei, Taiwan.
- [2] Aditya Panji and Reza Wahyudi, "Android Grows Fast in Indonesia", <http://tekno.kompas.com/read/2012/03/26/09584235/Android.Tumbuh.Cepat.di.Indonesia>, 26 March 2012.
- [3] Iqbal Muhtarom, "Google Joins Android Developers in Indonesia", <http://www.tempo.co/read/news/2012/03/30/072393682/Google-Gandeng-Developer-Android-di-Indonesia>, 30 March 2012.
- [4] <http://www.speech.cs.cmu.edu>
- [5] <http://developer.android.com/index.html>
- [6] Juang, B. H. and L. R. Rabiner, "Hidden Markov Models for Speech Recognition", Speech Research Department AT&T Bell Laboratories Murray Hill, 2007.
- [7] <http://tekno.kompas.com/read/2011/11/15/12341520/Developer.Ungkap.Rahasia.Siri.di.iPhone.4S>