

APLIKASI PERINTAH SUARA UNTUK MENGENDALIKAN TAYANGAN PRESENTASI PADA MICROSOFT POWER POINT

Andre Rianda

*Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Gunadarma.
riandaandre@gmail.com*

Abstrak

Seiring dengan berkembangnya teknologi ponsel pintar, para pengembang aplikasi pada ponsel pintar berbasis sistem operasi Android mulai menciptakan remote PC. Untuk menggunakannya, ponsel Android pengguna harus terhubung dengan perangkat PC melalui koneksi nirkabel, bluetooth, atau QR Code. Konfigurasi untuk saling terhubung antar perangkat tidaklah mudah. Solusi lain untuk mengatasi masalah di atas adalah dengan bantuan operator. Ketika tayangan yang dikendalikan operator dengan informasi yang disampaikan presenter tidak sinkron, ini akan membuat presentasi kurang baik. Untuk memecahkan masalah tersebut, aplikasi perintah suara dibuat untuk mengendalikan tayangan presentasi. Data suara diproses melalui beberapa langkah pemrosesan sinyal yang meliputi pencuplikan, kuantisasi, pengkodean, lalu ekstraksi fitur. Ekstraksi fitur merupakan proses mendapatkan sederetan besaran pada bagian sinyal masukan untuk menetapkan pola pembelajaran atau pola uji. Setelah itu dilakukan proses pengenalan pola dari hasil ekstraksi fitur. Kemudian pola model akan dicocokkan dengan pola referensi. Jika memiliki kesamaan pola, maka perintah tersebut akan mengeksekusi perintah sesuai dengan yang telah ditentukan. Aplikasi perintah suara mempermudah dan membuat presenter hanya perlu mengucapkan perintah layaknya fungsi keyboard untuk mengendalikan tayangan seperti perintah untuk full screen, next, back, dan lain-lain.

Kata Kunci: *Perintah Suara, Power Point 2013, Visual Studio 2013, C#(Sharp).*

CONTROLLING MICROSOFT POWER POINT PRESENTATION BY USING VOICE COMMAND APPLICATION

Abstract

As the development of smartphone technology, Android application developers begin to create PC remote. To use the PC remote, android users have to connect between the devices using wireless connection, bluetooth, or QR Code. Connecting these devices is not an easy thing to do. One of the solution for that problem is by using operator. When the slides, which is controlled by the operator, is out of sync, it will make the presentation ineffective. The voice command application is made to solve such problem, which mean it is created to control the slides of presentation. Voice data is processed through several signal-processing steps that includes sampling, quantization, coding, and feature extraction. Feature extraction is the process of getting magnitude series on the input signal part to establish the learning pattern or test pattern. Then, the pattern recognition process is done from feature extraction results. Next, the model pattern will be matched with the reference pattern. If they have the same pattern, then the command will execute the orders according to the predetermined command. The voice command will simplify the presenters and make the presenters only need to say some commands like the keyboard functions such as the command to control slides like fullscreen mode, next, back, and others.

Keywords: *Voice Command, Power Point 2013, Visual Studio 2013, C#(Sharp).*

PENDAHULUAN

Komputer merupakan media teknologi penunjang untuk kegiatan presentasi. Presentasi dapat dibuat secara menarik menggunakan yang tersedia pada perangkat lunak Microsoft Power Point untuk sistem operasi Windows. Perkembangan versi Microsoft Office 2010 ke 2013 sangat berbeda dalam antar muka pengguna. Bagi pengguna yang sudah terbiasa dengan versi lama paket Microsoft Office seperti Microsoft Office 2003, Microsoft Office 2007, maupun pengguna yang baru menggunakan aplikasi paket Microsoft Office akan kesulitan dan membutuhkan waktu untuk beradaptasi dengan tampilan fungsi-fungsi tombol maupun menuturutama untuk Microsoft Power Point. Seiring dengan berkembangnya teknologi ponsel pintar, para pengembang aplikasi pada ponsel pintar berbasis sistem operasi Android mulai menciptakan *remote PC*. Ponsel itu sendiri juga dapat digunakan untuk mengendalikan tayangan presentasi. Konfigurasi untuk saling terhubung antar perangkat tidaklah mudah dan harus melalui beberapa tahap. Solusi lain untuk mengatasi masalah di atas adalah dengan bantuan operator. Ketika tayangan yang dikendalikan operator dengan informasi yang disampaikan presenter tidak sinkron, ini akan membuat presentasi kurang baik. Untuk memecahkan masalah tersebut, aplikasi perintah suara dapat dijadikan solusidalam mengontrol tayangan presentasi.

Penelitian sebelumnya tentang aplikasi perintah suaramembahas sulitnya pengguna komputer menggunakan sejumlah *button* yang terdapat dalam sebuah sistem operasi Windows (Kisumal, 2010). Hal ini menimbulkan keinginan membuat aplikasi yang dapat memudahkan pengguna dalam mengoperasikan sistem operasi tersebut. Solusi yang ditempuh yaitu membuat aplikasi perintah

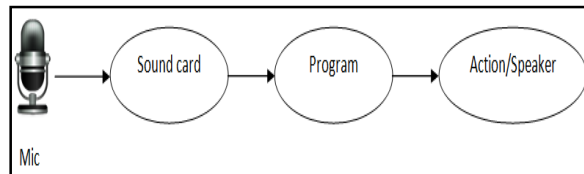
suara untuk pemanggilan program aplikasi di Windows. Penelitian tentang sistem pengendalian pintu pagar secara otomatis menggunakan perintah suara pada *smartphone* berbasis Android (Alpurqon, 2014). Penelitian ini dikembangkan karena sulitnya membuka pintu pagar secara manual, terlebih ketika turun dari kendaraan dalam keadaan hujan cukup merepotkan. Aplikasi ini menggunakan *microcontroller* untuk mengolah masukan berupa suara. Pintu akan terbuka secara otomatis setelah mengucapkan perintah suara.

METODE PENELITIAN

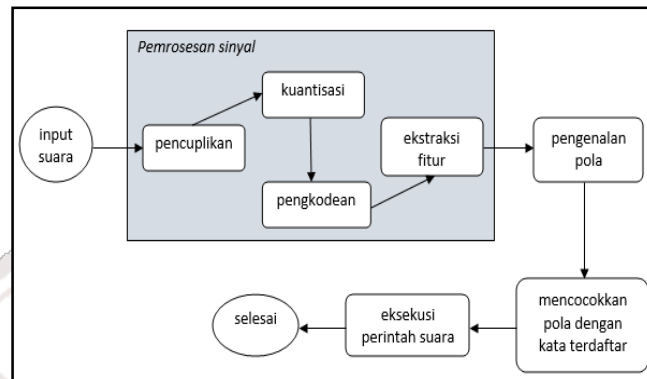
Tahapan yang ditempuh dalam penelitian diantaranya adalah metode pengumpulan data, analisis kebutuhan sistem, perancangan aplikasi, dan perancangan antar muka pengguna. Tahapan pengumpulan data dan informasi meliputi tiga metode pengumpulan data yaitu studi pustaka, studi literatur, dan observasi.

AnalisaKebutuhanSistem

Microphone digunakan sebagai alat masukan sinyal yang diteruskan ke *sound card* untuk diolah. *Sound card* memiliki empat fungsi utama, yaitu sebagai *synthesizer*, sebagai *MIDI interface*, *converter* data analog ke digital (misalnya merekam suara dari mikrofon) dan *converter* data digital ke bentuk analog. Hasil proses di dalam *sound card* akan menghasilkan format suara *wave*. Selanjutnya akan diproses kedalam suatu proses *sampling* dan lain-lain untuk selanjutnya dapat dilakukan proses *feature extraction*. Ekstraksi fitur adalah sebuah proses yang mengkonversi sinyal suara menjadi beberapa parameter yang dapat diambil untuk proses selanjutnya yaitu identifikasi pola suara. Gambar 1 menunjukkan analisa kebutuhan sistem aplikasi perintah suara untuk mengendalikan tayangan presentasi.



Gambar 1. Proses berjalannya sistem berdasarkan alur komponen



Gambar 2. Flowchart sistem pengenalan suara secara umum

Program akan mencocokkan pola suara dengan kata yang terdaftar. Fungsi dari *shortcut keyboard* akan dipanggil saat kata perintah yang diucapkan sesuai dengan kata yang terdaftar dan juga *grammar*. Hasil dari masukkan berupa sinyal analog suara akan diubah menjadi eksekusi perintah *shortcut keyboard*. Program dibuat tanpa menggunakan *database* karena fungsi *shortcut keyboard* yang bekerja terbatas hanya tombol *arrows*, *esc*, *home*, *end*, dan *F5*.

Perancangan Sistem

Proses perancangan sistem diartikan sebagai kumpulan aktivitas, metode, *best practice*, *deliverable*, dan *tools* otomatis yang digunakan *stakeholders* untuk merancang sistem informasi dan perangkat lunak secara kontinu. Artinya perancangan dilakukan bertahap, mulai dari hal yang menjadi kendala sistem sampai hal yang menjadi kebutuhan sistem (Aziz, 2009). Alur kerja dari pengenalan suara ditunjukkan pada Gambar 2.

Data suara diproses melalui beberapa langkah pemrosesan sinyal yang meliputi pencuplikan, kuantisasi, pengko-

dean, lalu ekstraksi fitur. Ekstraksi fitur merupakan proses mendapatkan sederetan besaran pada bagian sinyal masukan untuk menetapkan pola pembelajaran atau pola uji (Saksono M.W.T, Hidayatno A, Ajulian A, 2011). Setelah itu dilakukan proses pengenalan pola dari hasil ekstraksi fitur. Kemudian pola model akan dicocokkan dengan pola referensi. Jika memiliki kesamaan pola, maka perintah tersebut akan mengeksekusi perintah sesuai dengan yang telah ditentukan. Sistem dirancang untuk dapat mengenali kata perintah yang diucapkan pengguna yang kemudian diteruskan dengan pemanggilan fungsi *Sendkeys.Send()* sesuai dengan kata yang terdaftar. Fungsi ini akan mengeksekusi *shortcut keyboard* secara otomatis. Program tidak menggunakan *database* karena fungsi *shortcuts* yang terbatas pada *Power Point*. Pengguna mengaktifkan tombol 'Say Command' untuk dapat menggunakan sistem ini. Kemudian dalam keadaan aktif, sistem siap menerima masukan perintah suara. Setelah selesai menggunakan program, pengguna dapat keluar dari program dengan tombol *Exit*.

PEMBAHASAN

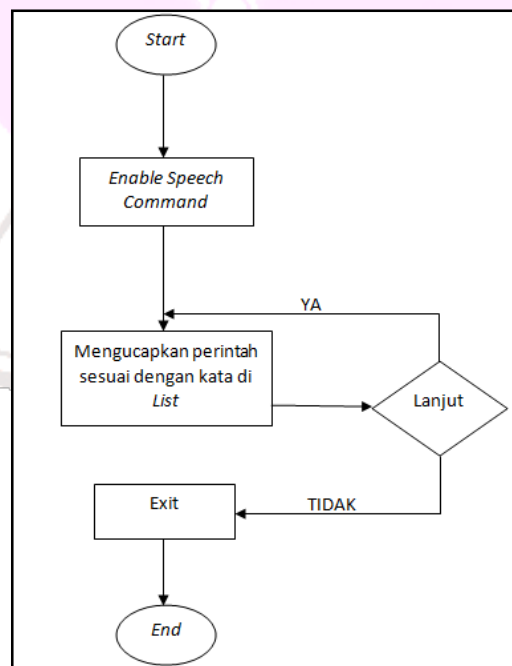
Dalam pembuatan aplikasi dibutuhkan kriteria standar baik dari segi perangkat lunak, maupun perangkat keras yang digunakan untuk menunjang keberhasilan pembuatan aplikasi perintah suara. Perancangan program aplikasi ini dibuat dengan spesifikasi komputer serta sistem operasi Hewlett-Packard G42 Notebook PC, Intel Core i3 CPU M380 2.53GHz, Windows 8.1 Pro, 64-bit OS, x64-based processor, build 9600 dengan Memory RAM: 4.00 GB, Display Intel HD Graphics, mikrofon SonicGear DM120, dan speaker ALTEC - Realtek High Definition Audio

Pembuatan aplikasi ini menggunakan perangkat lunak Microsoft Visual Studio 2013 dengan bahasa pemrograman Visual C# 2013. Aplikasi ini dijalankan pada sistem operasi Windows yang mendukung *.NET Framework* versi 4.5.51641.

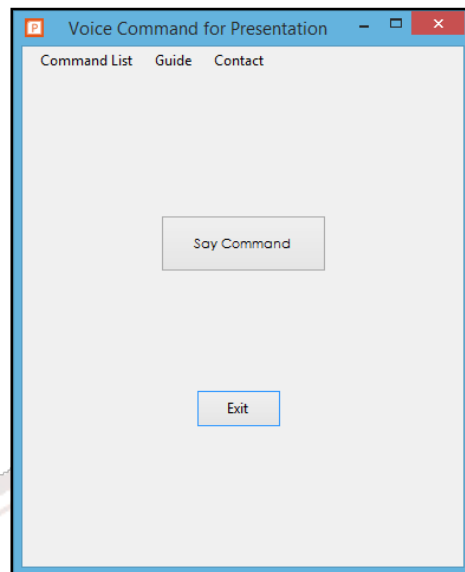
AntarMukaAplikasi

Grafis antar muka memberikan kemudahan dalam berkomunikasi antara pengguna dengan sistem. Antar muka pengguna juga berfungsi sebagai keterangan yang dapat membantu pengguna menemukan solusi dari masalah yang dihadapi. Secara umum, alur kerja program utama aplikasi terlihat seperti pada Gambar 3.

Pada *form* awal, tersedia dua tombol yang masing-masing berfungsi untuk mengaktifkan perintah suara dan untuk keluar dari aplikasi. Ada tiga *menu bar* pada *form* menu utama. *Command List* berisi daftar perintah yang pengguna bisa gunakan dalam mengoperasikan aplikasi ini. *Guide* berisi cara-cara penggunaan aplikasi. *Contact* berisi informasi dari pembuat aplikasi. Penyusunan antar muka dan pengkodean aplikasi dilakukan pada perangkat lunak Visual Studio 2013. Saat program dijalankan akan muncul tampilan awal aplikasi seperti pada Gambar 4.



Gambar 3. Diagram Alur Program Utama



Gambar 4. Tampilan *form* Menu Utama aplikasi

```
public void loadVoice(){
//list kata yang dapat bekerja pada aplikasi
_recognizer = new SpeechRecognitionEngine();
_recognizer.LoadGrammar(new Grammar(new GrammarBuilder("next")));
_recognizer.LoadGrammar(new Grammar(new GrammarBuilder("back")));
_recognizer.LoadGrammar(new Grammar(new GrammarBuilder("up")));
_recognizer.LoadGrammar(new Grammar(new GrammarBuilder("down")));
_recognizer.LoadGrammar(new Grammar(new GrammarBuilder("close")));
_recognizer.LoadGrammar(new Grammar(new GrammarBuilder("fullscreen")));
_recognizer.LoadGrammar(new Grammar(new GrammarBuilder("end")));
_recognizer.LoadGrammar(new Grammar(new GrammarBuilder("home")));
}
```

Gambar 5. Kode fungsi *loadVoice()*

Untuk dapat menggunakan aplikasi, pengguna harus terlebih dahulu mengklik tombol *Say Command*. Tombol ini berisi kode pemanggilan fungsi utama yaitu *loadVoice()*. Kode *functionloadVoice()* seperti pada Gambar 5.

Tombol *Exit* digunakan untuk mengakhiri program. Sebelum program diakhiri, akan muncul pesan untuk verifikasi apakah pengguna yakin untuk keluar dari program atau tidak. Di atas *form* terdapat komponen *menuStrip1*. Komponen ini berfungsi layaknya *Menu Bar* pada Microsoft Word. Komponen ini berisi tiga *ToolStripMenuItem*. Menu yang pertama yaitu *Command List*. Menu ini berisi perintah apa saja yang dapat pengguna katakan untuk perintah.

Pengguna juga dapat mendengarkan masing-masing cara pengucapan perintah tersebut dengan cara mengklik tombol yang bergambar *speaker*. Fungsi *speech Synthesizer. Speak()* adalah untuk memunculkan suara pada *speaker*. Keluaran suara tersebut berdasarkan pola kata yang terdapat di dalam kurung fungsi tersebut. Misalnya "say, next", program akan memunculkan suara tersebut ke pengguna melalui *speaker*. Konsep utama dari program ini cukup sederhana. Untuk menggunakan fitur ini, profil suara PC atau laptop pengguna harus dalam keadaan tidak *Mute*. Pengguna dapat mendengarkan cara pengucapannya dengan benar. Tampilan menu *Command List* terlihat seperti pada Gambar 6.



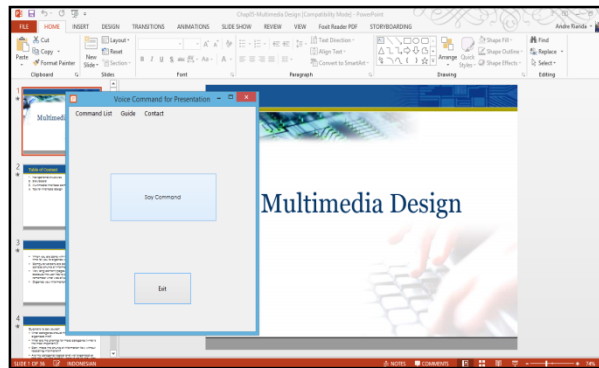
Gambar 6. Tampilan Menu *Command List*

Ketika pengguna mengucapkan perintah, misalnya “slide”, program selanjutnya akan menjalankan fungsi *SendKeys.Send("{F5}");*. Di dalam program Power Point, secara manual pengguna dapat menekan tombol F5 secara langsung untuk keyboard PC untuk mode tayangan penuh presentasi. Sementara untuk beberapa laptop, harus menekan tombol *Fn* bersamaan dengan F5. Fungsi *SendKeys.Send()* hanya mengeksekusi satu tombol, sehingga untuk menerapkannya pada aplikasi ini, dibutuhkan konfigurasi untuk membuat tombol *Fn* menjadi *disable* melalui menu BIOS, setelah masuk ke BIOS, langkah selanjutnya adalah memasuki *System Configuration* selanjutnya *Action Keys Mode [Enable]*. Dalam keadaan *default* akan *Enable*. Mode *Enable* dapat diganti ke mode *Disable* untuk mematikan tombol *Fn* pada keyboard. Fungsi

SendKeys.Send() khusus untuk perintah mode layar penuh “Slide” sudah dapat digunakan.

UjiCobaAplikasi

Setelah program selesai dibuat dan telah melalui tahap *debug*, selanjutnya adalah tahap pengujian aplikasi. Untuk menggunakan aplikasi ini, pengguna dapat langsung mengoperasikan melalui file *executable/application* yang ada pada direktori *Debug* dengan nama file *Voice*. Sebelum program dijalankan, pengguna dapat mengaktifkan file presentasinya. Setelah file presentasi terbuka, pengguna siap menggunakan perintah suara dengan mengklik dua kali *Voice.exe* lalu aktifkan fitur perintah suara dengan klik tombol *Say Command*. Program telah siap mendengarkan perintah dari pengguna. Ilustrasi saat program aplikasi berjalan adalah seperti Gambar 7.



Gambar 7. Ilustrasi Saat Program Dijalankan

Tabel 1. Uji Keberhasilan Jumlah Pengucapan Kata Perintah Pada Kondisi Berbeda oleh Pengguna

Percobaan Ke-	Perintah	Jumlah Pengucapan Kata Perintah	
		Tenang	Bising
1	<i>Up</i>	5/5	5/5
2	<i>Down</i>	5/5	5/5
3	<i>Home</i>	2/5	4/5
4	<i>End</i>	5/5	5/5
5	<i>Fullscreen</i>	5/5	5/5
6	<i>Next</i>	5/5	3/5
7	<i>Back</i>	4/5	5/5
8	<i>Close</i>	5/5	4/5

Evaluasi

Dari hasil uji coba aplikasi yang dilakukan, terlihat bahwa dengan menggunakan aplikasi perintah suara ini, pengguna dapat melakukan pengendalian tayangan presentasi tanpa menggunakan *keyboard* ataupun *remote PC*. Perintah yang terdaftar merupakan fungsi *keyboard* yang dapat berfungsi pada Power Point. Jadi, terbatas hanya beberapa perintah saja yang digunakan. Berdasarkan pengamatan selama dan setelah uji coba, terdapat kelemahan dan kekurangan dari aplikasi ini. Masih terdapat sedikit *bug/error* saat berada pada mode layar penuh. Tayangan masih sering kembali ke tayangan pertama sehingga saat pengguna sudah berada pada tayangan tertentu, terkadang kembali ke tayangan awal.

Aplikasi tidak dapat mengeksekusi fungsi ***loadEksekusiVoiceGagal***. Untuk tingkat pengenalan suara, terdapat beberapa faktor yang berpengaruh terhadap keakuratan pemanggilan perintah. Faktor tersebut yaitu tingkat kebisingan, ketepatan pengucapan kata. Tingkat keakuratan pengucapan perintah oleh pengguna terlihat pada Tabel 1. Dari data yang diperoleh pada Tabel 1, dapat diketahui rata-rata keberhasilan pengucapan masing-masing kata perintah dalam kondisi tenang dan bising. Pengambilan suara dalam kondisi bising saat uji coba yaitu berada di dekat *speaker* PC yang sedang memainkan lagu dengan volume maksimal. Kemudian di sekitar pengguna terdapat tiga rekan yang sedang mengobrol untuk menciptakan *noise*. Setiap kata

perintah diucapkan sebanyak lima kali. Pada pengucapan kata perintah *Up* dalam kondisi tenang, pengguna berhasil mengucapkan kata sesuai dengan fungsinya dengan tingkat keberhasilan 100 persen. Dalam kondisi bising, keberhasilan yang didapat juga 100 persen untuk pengucapan kata *Up*. Pada pengucapan kata perintah *Down*, dalam kondisi tenang dan bising didapat hasil yang sama dalam tingkat keberhasilan pengucapan kata perintahnya. Pada kata perintah *Home* dalam kondisi tenang, pengguna berhasil sebanyak dua kali dari lima kali percobaan. Fungsi perintah suara *Home* adalah untuk kembali ke tayangan pertama.

Jika pengguna dalam presentasinya sudah mencapai tayangan akhir, pengguna dapat kembali ke tayangan paling awal dengan fungsi *Home*. Tayangan berhasil dikendalikan sesuai fungsinya sebanyak dua kali. Kegagalan sebanyak tiga kali. Tayangan yang seharusnya kembali ke tayangan pertama akan tetapi mengeksekusi fungsi perintah *Up* atau tayangan naik (bukan dalam keadaan *fullscreen mode*). Kegagalan eksekusi perintah *Home* dalam kondisi bising sebanyak satu kali. Kemudian untuk fungsi *Next* dalam kondisi bising, kegagalan aplikasi mengeksekusi perintah sebanyak dua kali dari lima kali percobaan. Pada perintah *Back*, pengguna juga gagal sebanyak satu kali. Kegagalannya yaitu program tidak merespon perintah *Back*.

Sensitifitas pada masing-masing perintah juga mempengaruhi keberhasilan uji coba. Kata perintah yang hampir sama pengucapannya yaitu *Home* dengan *Up*. Pada Tabel 1 terlihat kegagalan eksekusi dari kata perintah *Home* sebanyak tiga kali dari lima kali percobaan. Hasil dari kegagalan tersebut adalah tayangan yang seharusnya dapat berpindah ke tayangan awal namun hanya berpindah satu tayangan ke atas atau mengeksekusi fungsi perintah *Up*. Untuk sensitifitas

perangkat *microphone* juga berpengaruh terhadap keberhasilan aplikasi dalam mengeksekusi perintah. *Microphone* yang terlalu sensitif juga tidak disarankan. Aksi seperti mengetuk *microphone* adalah salah satu contoh yang dapat berpengaruh terhadap perangkat *microphone* yang sangat sensitif.

SIMPULAN DAN SARAN

Ujicoba aplikasi perintah suara untuk mengendalikan tayangan aplikasi pada Microsoft Power Point berhasil. Aplikasi perintah suara ini menggantikan peran *remote* untuk mengendalikan tayangan presentasi. Berdasarkan uji coba keberhasilan pengucapan kata perintah masih terdapat kegagalan eksekusi pada kata perintah tertentu dan dengan kondisi tertentu. Pemilihan kata perintah harus tepat agar tayangan bergerak sesuai dengan perintah.

Aplikasi ini masih terdapat kekurangan. Pengguna tidak dapat mengganti kata perintah sesuai keinginan sendiri karena kata perintah yang dapat digunakan sudah ditentukan. Kemudian tidak adanya tombol untuk menghentikan perintah suara (*enable/disable command*). Pengguna dengan fasilitas laptop harus melakukan konfigurasi fungsi tombol *Fn* untuk dapat melakukan perintah *fullscreen*. Terdapat beberapa *bug* pada saat aplikasi digunakan.

Untuk pengembangan kedepan, diharapkan pengembang aplikasi dapat menyediakan fitur *database*, sehingga dapat mengganti kata perintah sesuai keinginan. Pengembang dapat menambahkan petunjuk aktif atau tidaknya perintah suara saat dijalankan, seperti menggunakan *radio button* yang terintegrasi dengan tombol *Say Command*. Kemudian pengembang dapat menambahkan notifikasi status ucapan perintah yang tidak dapat dikenali aplikasi. Pengembang dapat memperbaiki perintah suara yang terkadang tereksekusi dua kali

dalam satu kali ucap. Hal ini menyebabkan tayangan yang tidak dimaksudkan muncul. Penggunaan mikrofon dengan kabel menyebabkan jarak dengan laptop terbatas. Pengembang diharapkan menggunakan *wireless microphone* agar jarak presentasi tidak terbatas pada kabel.

DAFTARPUSTAKA

- Alpurqon, A. 2014. *Sistem Pengendali Pintu Pagar Secara Otomatis Menggunakan Aplikasi Voice Command pada Smartphone Android*. Skripsi. STMIK RaharjaTangerang.
- Andriana., A., D. 2013. *Perangkat Lunak Untuk Membuka Aplikasi Pada Komputer Dengan Perintah Suara Menggunakan Metode Mel Frequency Cepstrum Coefficients*. *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*. Vol. 2, No. 1, pp 21 - 26
- Aziz, A., A. 2009. *Aplikasi Tanda Tangan Digital Menggunakan Metode Ong-Schnorr-Shamir Dan Euclidean Pada Teks*. Skripsi. Universitas Islam Negeri Syarif Hidayatullah, Fakultas Sains dan Teknologi, Jakarta.
- Kisumal., H. 2010. *Aplikasi Perintah Suara Pada Windows*. Skripsi. Universitas Islam Negeri Syarif Hidayatullah, Fakultas Sains dan Teknologi, Jakarta.
- Saksono, M., W., T, Hidayatno., A, Ajulian., A. 2011, *Aplikasi Pengenal Ucapan Sebagai Pengatur Mobil Dengan Pengendali Jarak Jauh*. Jurusan Teknik Elektro, Fakultas Teknik, Universitas Diponegoro, Semarang.