

Implementasi Motion dan Gesture Pada Aplikasi Mobile Berbasis iOS Sebagai Wireless Remote

Steven Wijaya¹, Kristo Radion Purba²

Program Studi Teknik Informatika, Fakultas Teknologi Industri, Universitas Kristen Petra

Jl. Siwalankerto 121-131, Surabaya 60236

Telp. (031)-2983455, Fax. (031)-8417658

E-mail: m26413019@john.petra.ac.id¹, kristo@petra.ac.id²

ABSTRAK

Motion dan *gesture* dalam dunia teknologi semakin berkembang sehingga *smartphone* yang dipakai sehari-hari dan juga memiliki berbagai macam sensor. Sensor – sensor tersebut dapat memberi data yang dapat dikalkulasi sehingga diketahui sebuah gerakan tertentu.

Data dari *accelerometer* digunakan untuk mengetahui gerakan yang dilakukan oleh user. Accelerometer digunakan sebagai inputan data yang akan dipakai pada fitur air mouse dan gesture recognition. Selain itu juga akan mengimplementasikan *image recognition* untuk memindahkan kursor mouse ke objek terdekat.

Hasil dari penelitian ini merupakan sebuah aplikasi yang berfungsi untuk pengganti mouse fisik sehingga kursor mouse dapat digerakkan melalui *smartphone*. Sensor yang digunakan adalah *accelerometer* pada *smartphone*. Aplikasi ini akan sangat membantu bagi presenter untuk mengendalikan laptopnya jarak jauh. Keunggulan aplikasi ini dibandingkan mouse *wireless* adalah aplikasi ini tidak perlu alas yang datar dan dapat dijalankan di udara.

Kata Kunci: Accelerometer, motion, gesture recognition, image snapping.

ABSTRACT

Motion and gesture in the world of technology is growing so smartphones are used everyday and also has a variety of sensors. These sensors can provide data that can be calculated so that a certain movement is known.

Data from the accelerometer is used to determine the movements performed by the user. Accelerometer is used as input data to be used on the air mouse feature and gesture recognition. It will also implement image recognition to move the mouse cursor to nearest objects.

The results of this study is an application that works for a physical mouse replacement so that the mouse cursor can be driven through the smartphone. The sensor used is the accelerometer on the smartphone. This application will be very helpful for the presenter to control his laptop remotely. The advantages of this application than the wireless mouse is this application does not need a flat base and can run in the air.

Keywords: Accelerometer, motion, gesture recognition, image snapping.

1. PENDAHULUAN

Motion dan gesture dalam dunia teknologi semakin berkembang sehingga smartphone yang dipakai sehari-hari juga memiliki berbagai macam sensor. Sensor – sensor tersebut dapat memberi data yang dapat dikalkulasi sehingga diketahui sebuah gerakan tertentu. Salah satu sensor untuk membaca motion adalah sensor accelerometer. Sensor tersebut memberikan output percepatan dari ketiga axis yaitu sumbu X, sumbu Y, sumbu Z. Data tersebut dapat diolah untuk mendapat hasil pergerakan dari user.

Pada aplikasi yang dibuat akan menggunakan data dari *accelerometer* untuk mendeteksi gerakan yang dilakukan oleh pengguna. Untuk mendeteksi gerakan data yang dibaca dari *accelerometer* adalah data percepatan dari ketiga *axis* yaitu sumbu X, sumbu Y dan sumbu Z. Pada aplikasi juga akan menerapkan penggunaan *accelerometer* untuk menggantikan mouse fisik. Pada fitur ini yang akan dibaca hanya 2 *axis* yaitu sumbu X dan sumbu Y. Selain itu juga akan mengimplementasikan *image recognition* untuk memindahkan kursor *mouse* ke objek terdekat. Yang dilakukan agar dapat memindahkan *mouse* adalah mendeteksi dan membedakan antara latar belakang dan objek pada layar lalu menghitung jarak antara objek dan layar.

Hasil dari aplikasi ini merupakan sebuah aplikasi yang berfungsi untuk pengganti *mouse* fisik sehingga dapat menggerakkan *mouse* dengan pergerakan dari *handphone*. Sensor yang digunakan adalah *accelerometer* pada *smartphone*. Aplikasi ini akan sangat membantu bagi presenter untuk mengendalikan laptopnya jarak jauh. Keunggulan aplikasi ini dibandingkan *mouse wireless* adalah aplikasi ini tidak perlu alas yang datar dan dapat dijalankan di udara. Kemampuan dari aplikasi ini juga akan ditambahkan dengan beberapa fitur pendukung untuk mengontrol *media player* pada laptop.

2. TINJAUAN PUSTAKA

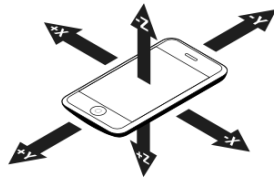
2.1 Motion Sensor

Accelerometer adalah sebuah sensor kecil yang dapat menghitung percepatan dari dirinya sendiri atau percepatan dari alat dimana sensor *accelerometer* tersebut dipasang.

Accelerometer yang ada pada iPhone adalah LIS302DL micro electro-mechanical system (MEMS). *Accelerometer* tersebut dapat membaca gerakan dari 3 sumbu dengan ketepatan kurang lebih 2g.

Didalam sensor gerak yang dimiliki iPhone sebenarnya juga terdapat berbagai macam sensor seperti magneto meter, Gyroscope, altitude sensor dan lain-lain.

Gambar 1 adalah gambar dari sumbu yang dapat dibaca oleh iPhone.



Gambar 1 Sumbu dari Accelerometer yang ada di iPhone

Dikarenakan data yang dibaca oleh *accelerometer* adalah percepatan yang dialami oleh sensor tersebut dan dalam keadaan diam di atas meja yang datar pada sumbu Z tetap terpengaruh oleh percepatan gravitasi bumi sehingga data yang ditunjukkan oleh *accelerometer* tersebut adalah sekitar 1g ke atas, $a = \{a_x = 0g, a_y = 0g, a_z = -1g\}$.

Selain *accelerometer* ada juga sensor *gyroscope* yang berfungsi untuk mendeteksi rotasi pada perangkat. Dengan sensor ini kita dapat membaca rotasi dari perangkat per detik. Namun sensor ini punya kelemahan tidak dapat membaca putaran pada sumbu Z dikarenakan pembacaan sensor ini dipengaruhi oleh gravitasi dan putaran pada sumbu Z tidak memberi dampak pada sensor. Hal ini bisa diatasi dengan sensor magnet yang biasanya terdapat di *smartphone* untuk aplikasi kompas yang memberi output berupa derajat yang selalu menunjuk ke arah utara.

2.2 CoreBluetooth

CoreBluetooth [1] adalah *framework* yang disediakan oleh Apple Inc untuk produk – produk Apple agar dapat berkomunikasi dengan perangkat lainnya yang memiliki fitur *Bluetooth*. Beberapa *class* yang ada di *framework* tersebut adalah *CBCentralManager* yang akan digunakan untuk penghubung dan menjaga hubungan antara 2 perangkat yang terhubung dan *CBPeripheralManager* yang akan digunakan untuk membaca atau menulis dari koneksi tersebut. Data yang dikirim dari 1 perangkat ke perangkat lain dapat dalam bentuk *string* ataupun sebuah *file*.

Pada penggunaan ini *device* akan dibagi menjadi 2 kategori. Yang pertama adalah *peripheral*. *Device* yang dijadikan *peripheral* akan berfungsi untuk melakukan *advertising* data. Data tersebut akan ditangkap oleh *device* yang dijadikan *central*. Yang kedua adalah *central* dimana *device* tersebut akan melakukan *scanning peripheral* yang ada dan meminta izin untuk *subscribe* kepada *peripheral* yang dipilih agar dapat mendapat data yang *advertised* dari sisi *peripheral*.

2.3 Swift

Bahasa pemrograman baru yang dibuat oleh Apple Inc untuk *macOS*, *iOS*, *watchOS*, dan *tvOS*. Bahasa pemrograman ini diluncurkan saat Apple Inc melakukan Apple World Wide Developer Conference (WDC) 2014. Hingga saat ini bahasa *SWIFT* telah dikembangkan hingga versi *SWIFT* 3.0.

Apple Inc menciptakan *swift* dengan tujuan agar *SWIFT* menjadi salah satu bahasa pemrograman yang paling lancar dan mudah dipahami. Kelebihan dari *SWIFT* adalah bahasa yang ada dibuat lebih simpel untuk digunakan. Namun *SWIFT* juga punya kelemahan yaitu hanya dapat digunakan untuk membuat program-program dari perusahaan Apple Inc dan tidak dapat digunakan di produk yang bukan buatan Apple Inc namun kabar dari Apple Inc

mereka akan terus mengembangkan bahasa ini agar semakin lengkap namun tetap sederhana dan akan dapat digunakan untuk produk-produk yang bukan buatan Apple Inc.

2.4 Perbandingan dengan Aplikasi Sejenis

Tabel 1 adalah daftar aplikasi penjadwalan tugas lain yang sejenis:

Tabel 1 Tabel perbandingan dari aplikasi sejenis

Atribut dan fitur	Remote Mouse	Air Keyboard	Remote controller Benzele inc	Mobile Mouse Remote R.P.A. Tech	Wi-Fi Mouse Shimeng wang	Aplikasi Ini
<i>Gesture Recognition</i>	-	-	-	-	-	✓
<i>Snap to Object</i>	-	-	-	-	-	✓
<i>Trackpad</i>	✓	✓	✓	✓	✓	-
<i>Mouse Button</i>	✓	✓	✓	✓	✓	✓
<i>Air Mouse</i>	-	-	-	-	-	✓
<i>MultiTouch</i>	✓	-	-	✓	✓	-
<i>Keyboard</i>	✓	✓	✓	✓	✓	✓
<i>Media Remote</i>	✓	-	✓	-	✓	✓
<i>Application Switcher</i>	-	-	✓	-	-	✓
<i>Special Function</i>	-	-	✓	-	-	-
<i>Connect via Wi-Fi</i>	✓	✓	✓	✓	✓	-
<i>Connect via Bluetooth</i>	-	-	-	✓	-	✓

3. ANALISIS DAN DESAIN

3.1 Desain aplikasi pada *smartphone*

Desain pada iPhone akan memiliki beberapa tombol. Tombol tersebut akan berguna sebagai *trigger* dari fungsi yang ada.

Daftar Halaman pada aplikasi :

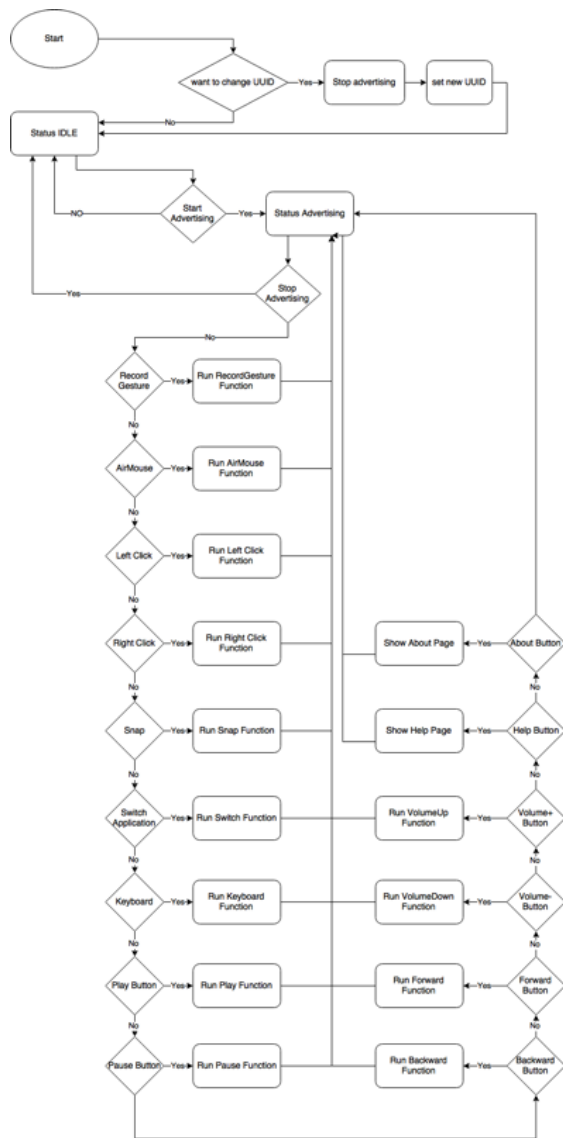
- 1) Halaman utama.
- 2) Halaman *help*.
- 3) Halaman *about*.

Daftar tombol pada aplikasi :

- 1) Tombol untuk *trigger* perekaman *gesture*.
- 2) Tombol untuk *trigger* *air mouse*.
- 3) Tombol untuk mengganti tombol pada *mouse*.
- 4) Tombol untuk *toggle snap*.
- 5) Tombol untuk menyimulasikan penekanan tombol *Command + Tab*.
- 6) Tombol untuk mengganti *keyboard*.
- 7) Tombol untuk mengontrol *media player* sederhana.
- 8) Tombol untuk Mengganti *UUID*.
- 9) Tombol untuk *start advertising*.
- 10) Tombol untuk *stop advertising*.
- 11) Tombol untuk menampilkan halaman *Help*.
- 12) Tombol untuk menampilkan halaman *About*.

Desain dan penjelasan tiap fitur dan tampilan tiap halaman akan dijelaskan di halaman selanjutnya.

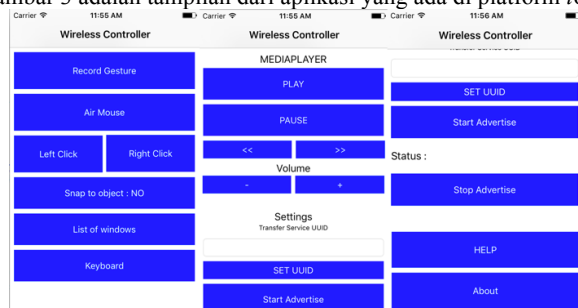
Gambar 2 adalah *flowchart* secara umum dari penggunaan aplikasi pada *iOS*



Gambar 2 Flowchart dari aplikasi pada mobile secara keseluruhan

Saat program dijalankan pertama-tama akan melakukan koneksi dengan *macintosh*. Lalu setelah itu program akan *idle* hingga user menekan salah satu tombol. Setelah pengguna menekan tombol program akan menjalankan fungsi sesuai tombol yang ditekan. Setelah menjalankan fungsi aplikasi akan kembali ke posisi *idle* hingga mendapatkan masukan baru atau aplikasi ditutup.

Gambar 3 adalah tampilan dari aplikasi yang ada di platform *iOS*:



Gambar 3 Tampilan Awal dari aplikasi

Pada tampilan ini terdapat 19 tombol yang dibagi menjadi tombol untuk merekam *gesture*, tombol untuk *mouse controller* sederhana, tombol untuk fitur *snap*, tombol untuk fitur *switch application*, dan fitur untuk *Keyboard*, tombol untuk fitur *Media Player* sederhana, Tombol untuk mengganti UUID, tombol untuk memulai dan memberhentikan *advertising*, tombol untuk menampilkan halaman *Help* dan *About*.

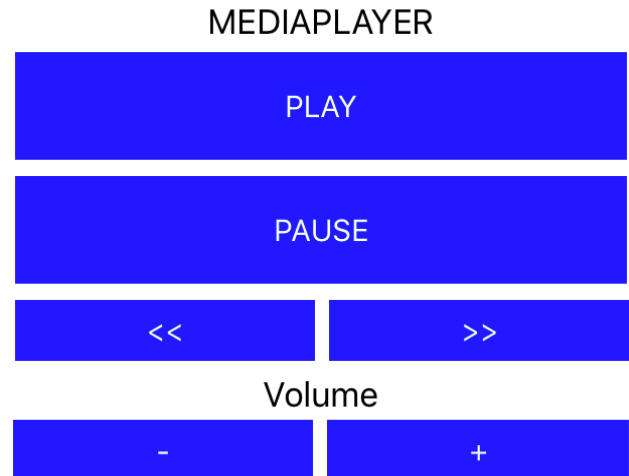
1) Fitur Keyboard.

Dengan mengaktifkan fitur *keyboard* maka *keyboard* virtual akan ditampilkan. Keyboard ini berfungsi untuk mengirim perintah kepada aplikasi pada *macbook* untuk menyimulasikan tekanan pada *keyboard* fisik. Fitur *keyboard* ini dimunculkan dengan *toggle button keyboard*.

2) Fitur Media Player

Fitur ini menyediakan tombol-tombol sederhana untuk mengendalikan media *player* seperti *play/pause*, *forward*, *backward*, *volume control*. Fungsi dari tombol-tombol tersebut ketika diklik akan mengirim perintah kepada aplikasi di *macintosh* untuk melakukan sesuatu sesuai dari tombol yang dipencet. Perintah yang dikirim untuk fitur *play/pause* adalah perintah untuk menyimulasikan tekanan tombol spasi karena tombol tersebut digunakan di banyak aplikasi media *player* untuk melakukan *play / pause*. Tombol *forward* dan *backward* akan mengirim perintah untuk menyimulasikan penekanan tombol panah kanan dan kiri. Sedangkan tombol *volume* akan mengirim perintah untuk menaikkan dan menurunkan *volume* sistem.

Gambar 4 adalah tampilan dari fitur *mediaplayer*



Gambar 4 Tampilan Dari fitur Mediaplayer

3) Fitur Settings

Fitur dari *settings* ini adalah untuk mengubah UUID yang akan di *advertise* oleh aplikasi ini. Dan tombol untuk memulai *advertising* dan tombol untuk menghentikan *advertising* dari aplikasi ini. Ada label yang menunjukkan status saat ini untuk mengetahui saat ini aplikasi ini sedang *advertising* atau telah berhenti *advertising*. UUID yang baru dapat langsung *dicopy* melalui fitur *Universal Keyboard* milik Apple.

4) halaman Help

Halaman ini menampilkan *list* dari *gesture* yang disediakan beserta penjelasan singkat dari fitur-fitur aplikasi ini.

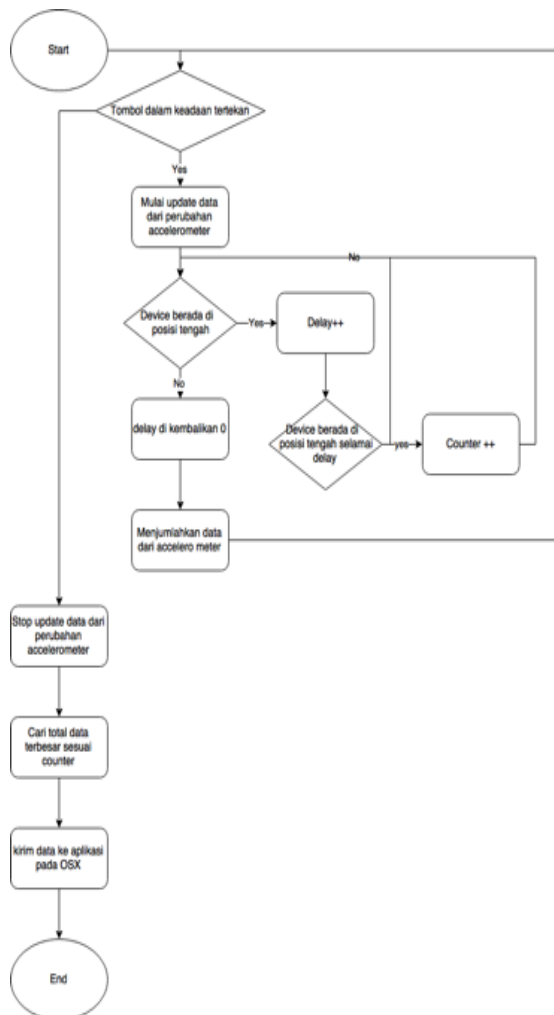
5) Halaman *About*

Tampilan dari halaman ini menunjukkan tombol untuk kembali ke halaman sebelumnya dan juga penjelasan singkat tentang aplikasi ini beserta hak cipta dari aplikasi tersebut.

3.2 Record Gesture

Fitur ini adalah fitur utama pada aplikasi ini yang membedakannya dengan aplikasi sejenis yang lain. Dengan fitur ini dapat membantu *presenter* untuk menjalankan *shortcut* tanpa perlu melihat laptop atau tombol pada *wireless controller*. Kebanyakan pada *wireless controller* pun hanya disediakan tombol *next previous* dan media *control* yang singkat. Terdapat 1 tombol untuk memulai fitur ini. Ketika tombol ditekan data dari accelerometer akan dibaca dan diolah sehingga gerakan^[5] dari pengguna dapat diketahui.

Gambar 5 adalah flowchart dari fungsi RecordGesture.



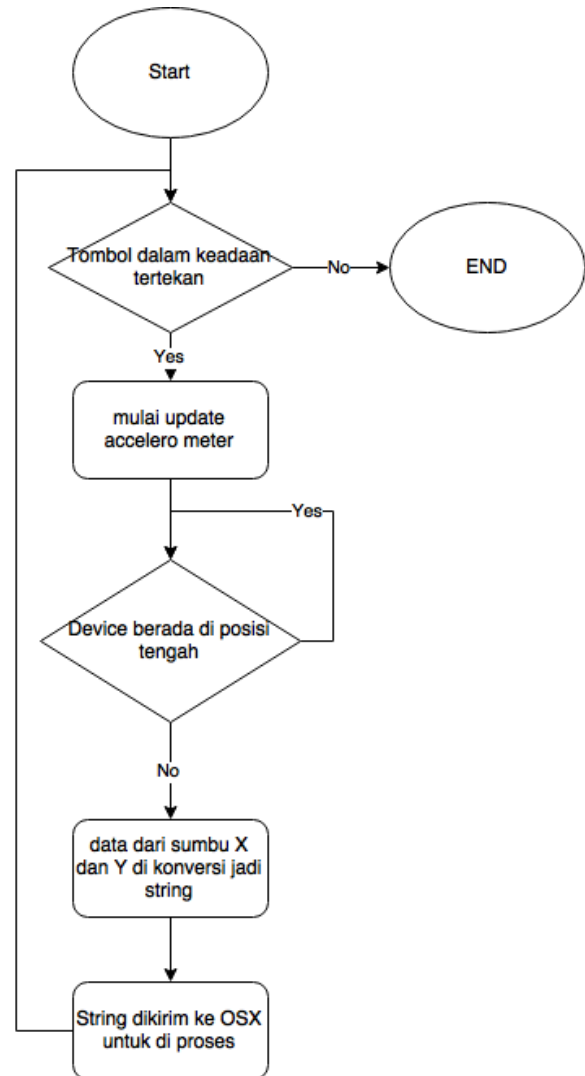
Gambar 5 Flowchart dari fungsi gesture recognition

3.3 Air Mouse

Fitur ini untuk menggantikan kinerja mouse sehingga *presenter* dapat menggerakkan kursor mouse hanya dengan *smartphone*. Fitur ini menggunakan *gyroscope*^[4] untuk mengambil rotasi dari *smartphone* dan mengubahnya menjadi *string* seperti “atas”,

“bawah”, “kiri”, “kanan”. Setelah mendapat *string* tersebut akan dikirim ke laptop untuk menggerakkan kursor.

Gambar 6 adalah flowchart dari fungsi airmouse.



Gambar 6 Flowchart dari fungsi Air Mouse

3.4 Snap to Object

Fitur ini dibuat untuk mempermudah *presenter* untuk menunjuk objek yang diinginkan tanpa harus menggerakkan kursor secara spesifik ke objek tertentu. Fitur ini dibuat dengan mengambil gambar^[2] pada layar dan mengolahnya dengan filter *threshold*^[3] sehingga menjadi hitam dan putih^[6]. Lalu melakukan perhitungan agar dapat diketahui mana yang merupakan latar belakang dan mana yang merupakan *object*^[7]. Setelah itu akan di kalkulasi jarak antara dua titik pada semua yang merupakan *object* sehingga didapat jarak terdekat antara *object* dan kursor mouse. Setelah itu kursor mouse akan dipindahkan ke *object* terdekat tersebut.

Dengan fitur ini *presenter* dapat dengan mudah menunjuk objek yang dimaksud. Fitur ini juga berfungsi terhadap tulisan dengan syarat warna objek dan latar belakang harus memiliki warna yang kontras. Hal tersebut diharuskan agar aplikasi ini dapat membedakan objek dan latar belakang menggunakan fungsi *threshold*.

Gambar 7 adalah flowchart dari fungsi snap.



Gambar 7 Flowchart dari fungsi snap

3.5 Keyboard

Fitur ini dibuat agar pengguna tidak perlu kembali ke laptop jika membutuhkan *keyboard* dan juga dapat menggantikan *keyboard* fisik jika dibutuhkan. Cara kerja fungsi ini hanya menampilkan *keyboard* di layar dan mengirim ke aplikasi laptop tombol yang di pencet dan aplikasi laptop akan menjalankan fungsi seakan tombol itu pada laptop sedang dipencet.

3.6 Media Player

Fungsi ini berguna sebagai *shortcut* untuk mengendalikan media seperti melakukan *play* atau *pause* pada video atau music. Melakukan *forward* atau *backward*. Dan juga dapat untuk menaikkan atau menurunkan volume. Pada aplikasi terdapat 1 bagian untuk tombol media player sehingga mempermudah pengguna untuk mencari tombol tersebut.

Tombol untuk media player :

- 1 tombol untuk *play* .
- 1 tombol untuk *pause*.
- 2 Tombol untuk *next* dan *prefious*.
- 2 tombol untuk mengatur *volume*.

3.7 Switch Application

Fitur ini untuk menampilkan *windows* yang sedang aktif sehingga pengguna dapat pindah *windows* dengan cepat. Ketika tombol ini diklik akan dikirim perintah untuk aplikasi yang berada di laptop untuk mensimulasikan command tab untuk berpindah ke aplikasi sebelumnya.

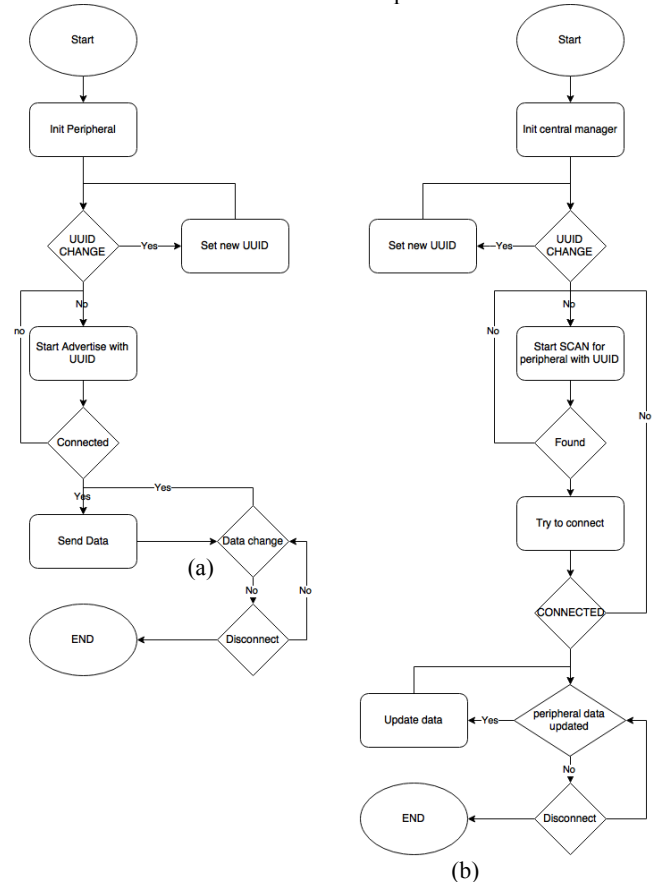
3.8 Koneksi Bluetooth pada iOS dan OSX

Koneksi Bluetooth ini dibuat dengan library CoreBluetooth bawaan dari XCODE. Pada iOS akan di inisialisasikan sebagai peripheral dimana fungsi dari peripheral adalah untuk melakukan advertising yang selanjutnya data yang di advertisingkan akan dapat diambil oleh central. Pada OSX akan di inisialisasikan sebagai Central dimana berfungsi untuk melakukan request kepada peripheral agar

dapat membaca data yang di advertising oleh peripheral manager. Dalam melakukan koneksi juga akan fungsi untuk melakukan pembersihan koneksi jika ada perubahan dan fungsi untuk mengirim data dari peripheral menuju central.

Berikut adalah flowchart untuk melakukan koneksi dari peripheral

Gambar 8 flowchart koneksi Bluetooth pada iOS dan OSX.

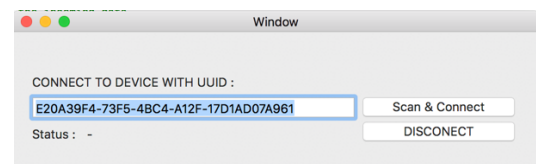


Gambar 8 Flowchart Koneksi Bluetooth (a) iOS (b) OSX

3.9 Desain aplikasi pada laptop

Desain aplikasi pada laptop hanya sebuah tampilan sederhana untuk melakukan koneksi dengan *smartphone*, untuk memutus hubungan dengan *smartphone*, mensetting UUID, dan juga melihat status koneksi terhubung dengan perangkat yang mana.

Gambar 9 tampilan aplikasi pada OSX.



Gambar 9 Tampilan awal dari aplikasi pada laptop

Aplikasi ini setelah melakukan koneksi dengan iOS akan selalu mendapat data dari iOS jika ada perubahan data. data yang didapat akan digunakan untuk memanggil fungsi” tertentu yang telah disediakan. Tombol scan & connect berfungsi untuk melakukan scan terhadap perangkat di sekitar dan akan mengajukan permintaan untuk mengambil data dari perangkat tersebut.

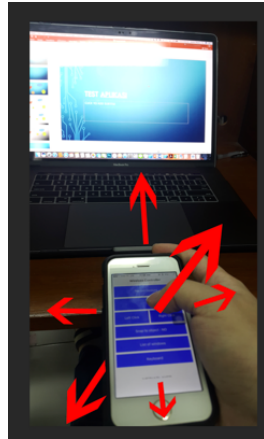
Sedangkan tombol disconnect akan memutus hubungan dari kedua perangkat tersebut.

4. HASIL DAN PEMBAHASAN

4.1 Pengujian Record Gesture

Pengujian untuk *gesture recognition* bermasalah pada sensor *accelerometer* dimana sensor tersebut dipengaruhi gravitasi^[8]. Selain itu ketika digerakkan dan berhenti sensor tersebut akan menampilkan data tumbukan dimanah data tersebut dapat memberi data berlebih dan mengacaukan fungsi *gesture recognition*^[9].

Gambar 10 Pengujian fungsi Record Gesture.



Gambar 10 pergerakan 6 axis yang dapat di record

dibaca oleh iPhone. Arah tersebut adalah atas, bawah, kiri, kanan, depan, belakang. Kendala dari aplikasi ini untuk menentukan gerakan dari iPhone adalah di *sensitivitas* dari *accelerometer*. Berikut adalah *table sensitivitas* iPhone dalam jangka waktu 0.5sec dan interval 0.01 dengan posisi *iphone* diam diatas meja yang datar.

Tabel 2 Update data dari *accelerometer* saat iPhone diam di atas meja datar

No.	NILAI PADA SUMBU			No.	NILAI PADA SUMBU		
	X	Y	Z		X	Y	Z
1	0.00011261	0.00087647	0.99472295	26	0.00012480	0.00019916	0.99620471
2	0.00019789	0.00033206	0.96570414	27	0.00058849	0.00016991	0.98765299
3	0.00022911	0.00058849	0.97587166	28	0.00074058	0.00031171	0.96635346
4	0.00083132	0.00028324	0.98116157	29	0.00026345	0.00084658	0.99676081
5	0.00065732	0.00027686	0.98167935	30	0.00060504	0.00089202	0.98773714
6	0.00020654	0.00087151	0.98885238	31	0.00016080	0.00094004	0.96269273
7	0.00058849	0.00033007	0.98765299	32	0.00053316	0.00052227	0.97776634
8	0.00098067	0.00012472	0.99322738	33	0.00087554	0.00058849	0.97948986
9	0.00083617	0.00025055	0.96292380	34	0.00090799	0.00058849	0.98946635
10	0.00036890	0.00077194	0.98956554	35	0.00069003	0.00047032	0.99938453
11	0.00045753	0.00061613	0.96073021	36	0.00029941	0.00045026	0.95624984
12	0.00058849	0.00013973	0.99940098	37	0.00032270	0.00027506	0.98454654
13	0.00057770	0.00059600	0.98285759	38	0.00031046	0.00088848	0.97414563
14	0.00019345	0.00045902	0.96489320	39	0.00016797	0.00058849	0.97723639
15	0.00085459	0.00058849	0.98128611	40	0.00056145	0.00067327	0.97177376
16	0.00067866	0.00043591	0.97828329	41	0.00047236	0.00060827	0.98981691
17	0.00076622	0.00015286	0.98200809	42	0.00077476	0.00061249	0.97680889
18	0.00088146	0.00012765	0.97551053	43	0.00052712	0.00088956	0.97805304
19	0.00076086	0.00047315	0.99032571	44	0.00051500	0.00033622	0.98987521
20	0.00083522	0.00079105	0.97652970	45	0.00025790	0.00058849	0.96376114
21	0.00058209	0.00087950	0.98584124	46	0.00038841	0.00098220	0.99242786
22	0.00044942	0.00030426	0.98510388	47	0.00024098	0.00058849	0.99000745
23	0.00040301	0.00065633	0.97787178	48	0.00058063	0.00039213	0.98546190
24	0.00066794	0.00021309	0.97557982	49	0.00073846	0.00029544	0.96045577
25	0.00010597	0.00051726	0.96198799	50	0.00058849	0.00059105	0.98394506

4.2 Pengujian Air Mouse

Pengujian untuk pergerakan *airmouse* untuk pergerakan ke 8 arah dapat berjalan dengan lancar. Kecepatan untuk *menggerakan* mouse juga beragam sesuai percepatan dari *accelerometer*.

4.3 Pengujian Snap to Object

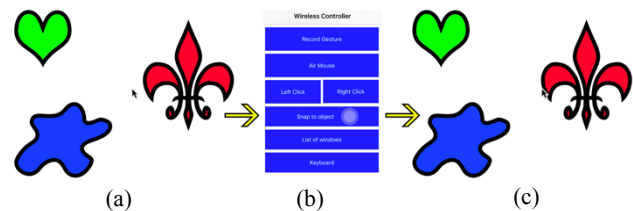
Pengujian untuk *snap to object* berjalan lancar. Kursor berhasil pindah ke *object* terdekat dengan *sensitivitas* 8 *pixel*.

Waktu yang diperlukan untuk mengambil *screenshot*, memproses gambar hingga memindah kursur mouse:

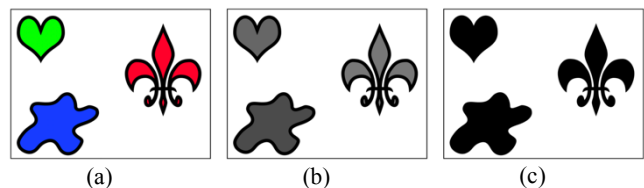
Tabel 3 Tabel waktu yang dibutuhkan jika menggunakan *sensitivitas* tertentu

<i>Sensitivitas</i>	Waktu Yang diperlukan
1 <i>pixel</i>	+ 4900.4 ms
2 <i>pixel</i>	+ 2819.6 ms
3 <i>pixel</i>	+ 1633.3 ms
4 <i>pixel</i>	+ 1225.4 ms
5 <i>pixel</i>	+ 948.6 ms
6 <i>pixel</i>	+ 646.6 ms
7 <i>pixel</i>	+ 587.4 ms
8 <i>pixel</i>	+ 540.6 ms

Dari sensitivitas 1 hingga 8 *pixel object* dapat ditemukan dengan waktu yang sangat berbeda sehingga sensitivitas 8 *pixel* masih dapat dipakai dan memberi hasil yang bagus.



Gambar 11 (a)posisi mouse awal, (b) tombol *snap* dipencet, (c)posisi mouse setelah di *snap*



Gambar 12 Gambar awal, (b)Filter *Grayscale*, (c)*Threshold*

Dari hasil pengolahan pada gambar 5.3 diketahui jumlah *pixel* hitam lebih sedikit dibanding *pixel* putih sehingga ditentukan bahwa *pixel* hitam adalah objek dan *pixel* putih adalah *background*.

4.4 Pengujian Keyboard

Pengujian terhadap keyboard berjalan lancar namun terkadang terjadi bug dimana tombol shift seakan tertekan terus.

Gambar 13 merupakan Pengujian terhadap fungsi keyboard Keyboard.



Gambar 13 Pengujian keyboard

4.5 Pengujian Media Player

Pengujian terhadap tombol – tombol untuk media player berjalan lancar. Kelemahan dari fitur ini hanya dapat digunakan di aplikasi yang memang mensupport play pause dengan tombol spasi, backward forward dengan tombol panah kanan atau kiri. Untuk pengaturan volume karena langsung terhubung dengan system tidak ada masalah yang terjadi.

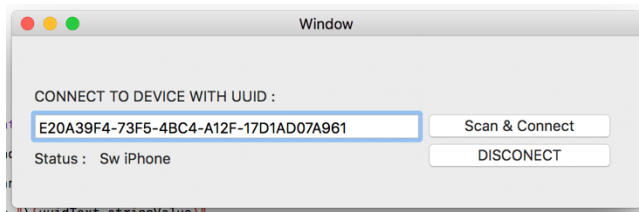
4.6 Pengujian Swich Application

Pengujian pada swich application dikarenakan karena menggunakan kombinasi tombol command + tab terkadang program menganggap command masih dalam keadaan tertekan sehingga tidak dapat berjalan sempurna. Namun ketika tombol command & tab fisik dipencet program akan bekerja seperti normal lagi.

4.7 Pengujian Koneksi Bluetooth

Bluetooth berjalan lancar ketika pertama kali melakukan advertise dan melakukan scanning. Namun ada kendala pada aplikasi OSX ketika hubungan di putus dan berusaha melakukan hubungan ulang. Kendala yang dimaksud adalah hubungan berhasil dilakukan namun data yang di advertise tidak berhasil di input seakan-akan data sisi peripheral tidak melakukan perubahan data. Untuk perubahan UUID berjalan lancar namun kesusahan untuk menyalin UUID dari 1 device ke device yang lain. Namun ada fitur universal clipboard dari iPhone dan MacBook dimana sesuatu yang di copy di 1 device dapat di paste di device lainnya dan hal ini sangat membantu dalam memindah UUID.

Gambar 14 adalah tampilan pengujian terhadap koneksi Bluetooth.



Gambar 14 koneksi dengan iPhone berhasil

5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil dari pengujian dapat disimpulkan hal sebagai berikut :

- Metode gesture recognition menggunakan motion dan gesture kurang bias dilakukan dengan hanya accelerometer meter dikarenakan data yang didapat oleh accelerometer meter dipengaruhi grafitasi bumi sehingga data terlalu sensitif.
- Metode air mouse yang diterapkan menggunakan accelerometer dapat berfungsi dengan baik namun butuh penyesuaian oleh user untuk terbiasa menggunakannya.
- Metode image snapping untuk memindahkan kursor mouse ke object terdekat membutuhkan waktu sangat besar jika layar memiliki resolusi yang tinggi namun dengan menurunkan sensitifitas dapat diperoleh waktu yang lebih cepat tanpa mempengaruhi hasil.
- Metode Bluetooth yang digunakan adalah advertising dimana metode ini kurang bagus karena semakin sering ada perubahan maka akan menimbulkan banyak glitch.
- Fitur keyboard karena mengirim banyak kombinasi dapat menimbulkan glitch pada aplikasi OSX.

5.2 Saran

Setelah mengevaluasi skripsi ini beberapa saran yang dapat diberikan adalah :

- Untuk menambah kemampuan dari gesture recognition dapat digunakan beberapa sensor lainnya seperti magnetic sensor atau gyroscope sensor.
- Koneksi dapat diganti dari Bluetooth menjadi local wifi.
- Untuk image snapping screenshot dengan resolusi yang tinggi dapat di resize menjadi lebih kecil dahulu.

6. DAFTAR REFERENSI

- [1] Apple.inc, 2016, Core Bluetooth Reference, <https://developer.apple.com/reference/corebluetooth>, retrieved 10 February 2017.
- [2] Apple.inc, 2016, Core Graphic Reference, <https://developer.apple.com/reference/coregraphic>, retrieved 22 February 2017.
- [3] Apple.inc, 2016, Core Image Reference, <https://developer.apple.com/reference/coreimage>, retrieved 22 February 2017.
- [4] Apple.inc, 2016, Core Motion Reference, <https://developer.apple.com/reference/coremotion>, retrieved 14 February 2017.
- [5] Klingman Marco, 2019, Accelerometer-Based Gesture Recognition with the iPhone, MSc in Cognitive Computing, Goldsmiths University of London.
- [6] Larson Brad, 2016, GPUImage2, <https://github.com/BradLarson/GPUImage2/>, retrieved 27 April 2017.
- [7] Park, Jung-Me, Carl G. Looney dan Hui-Chuan Chen, 2012, Fast Connected Component Labeling Algorithm Using Divide and Conquer Technique. University of Alabama, Tuscalona.
- [8] Tong Lina, Song Quanjun, Ge Yunjian, 2013, HMM-Based Human Fall Detection and Prediction Method Using Tri-Axial Accelerometer, IEEE Sensors Journal
- [9] Xie, Michael, David Pan, 2012, Accelerometer Gesture Recognition, Stanford University.