

PEMODELAN HUMAN COMPUTER INTERACTION PADA MULTI TOUCH TABLE SYSTEM

Broto Poernomo, Hendrik Tukunang

STMIK ASIA Malang

ABSTRAK

Perkembangan teknologi yang sangat pesat sangat berdampak dalam kebutuhan manusia. Hal yang menjadi kekurangan yang muncul dari perkembangan teknologi adalah harga *hardware* yang semakin naik dari waktu ke waktu, namun tidak diimbangi dengan efisiensi dalam segi penggunaan teknologi itu sendiri. Salah satu optimasi yang dapat dilakukan untuk menekan biaya pengadaan adalah menggunakan konsep *multi user* untuk tiap komputer sehingga komputer didapatkan efisiensi dimana komputer dapat melayani hingga empat *user* dengan memanfaatkan pengolahan citra sebagai penerima input sentuhan. Konsep *multi user* ini diwujudkan dalam produk *multi touch table system* dimana *user* dapat mengakses sistem bersamaan tanpa kehilangan kestabilan pelayanan, tren *hardware* dan yang paling penting dapat menekan biaya pengadaan bagi sebuah instansi atau organisasi. Dengan konsep *multi user* ini komputer dapat dioptimasi dari segi penggunaan hingga 400%, dan juga dapat menghemat biaya pengadaan fasilitas pada instansi. Kelebihan lain yang dapat ditemukan adalah konsep interaksi yang lebih menarik dibandingkan penggunaan komputer secara tradisional dengan *mouse* dan *keyboard*, yakni dengan basis layar sentuh yang saat ini sangat populer penggunaannya dalam *gadget* seperti *smartphone* maupun *tablet computer* serta juga sering ditemukan pada perangkat mini komputer saat ini.

KATA KUNCI: *Muti touch table, Layar Sentuh dengan pengolahan citra*

ABSTRACT

The rapid development of technology greatly impact the human needs. The shortcomings part of it arise from the increasingly rising hardware's prices but it does not commensurate with the efficiency in terms of the use of the technology itself. There is some optimization that can be done to reduce the cost of procurement, one of it is by use multi-user concept for each computer in order to obtain the efficiency with which to serve up to four user by using image processing as the recipient of touch input. This concept is embodied in multi-touch table system which users can access the system simultaneously without losing the stability of services, trends and the most important it can reduce the cost of procurement for an agency or organization. By using this concept computers can be optimized in terms of the use of up to 400%, and also can save the cost of procurement of facilities in the institution. Another advantage that can be found is this concept is about the interaction of human and computer that is more attractive than the traditional use of a computer with a mouse and keyboard, which is the basis based on a touch screen system which is currently very popular use in gadgets such as smartphones and tablet computers, and also often found on mini-computer devices in nowadays.

KEYWORDS: *Muti touch table system, Touch screen based on image processing*

PENDAHULUAN

Latar Belakang

Perkembangan teknologi informasi tidak dapat dipungkiri menjadi salah satu kebutuhan bagi tiap orang saat ini. Berbagai konsep teknologi ditawarkan, mulai dari perangkat *portable* seperti *smartphone* dan *tablet computer* menjadi bagian dari kebutuhan dari tiap individu saat ini. Namun konsep yang tidak dapat dirubah walaupun perkembangan teknologi sangat pesat adalah efisiensi sebuah produk, dimana tiap produk hanya dapat melayani tidak lebih dari satu *user* saja, hal ini dapat dilihat dari produk-produk saat ini seperti *tablet computer*, *smartphone*, bahkan *PC*

(*Personal computer*) yang memiliki spesifikasi *hardware* yang kurang berfungsi jika hanya dipakai tidak lebih dari satu *user*. Selain biaya pengadaan fasilitas yang sangat besar, juga dapat diperkirakan besarnya biaya perawatan untuk tiap komputer. Banyak instansi dan organisasi yang membutuhkan konsep *multi user* dari sebuah produk, Salah satunya instansi yang membutuhkan konsep *multi user* adalah universitas dalam efisiensi komputer untuk urusan akademik, nilai dan perpustakaan yang dapat diwujudkan dalam pemodelan *human computer interaction* pada *multi touch table system*.

Rumusan Masalah

Perumusan masalah dalam penelitian ini adalah bagaimana membangun sebuah sistem *multi user* pada sebuah produk tanpa mengabaikan kenyamanan user dalam mengoperasikan produk tersebut. Hadirlah inovasi baru dalam konsep interaksi manusia dan komputer yang dapat menangani kebutuhan *multi user* pada sebuah produk yakni *multi-touch table system*. Produk yang diwujudkan dalam bentuk meja sentuh berbasis pengolahan citra pola untuk menangani titik sentuh yang diakses oleh *user*. Konsep ini bisa tetap berjalan tanpa mengganggu stabilitas antar *user* saat digunakan bersamaan dan dapat menekan biaya pengadaan fasilitas hingga 400%, dikarenakan tiap produk dapat menangani hingga empat *user* yang mengakses *multi-touch table system* dalam waktu yang bersamaan.

Tujuan dan Manfaat

Adapun manfaat dari penelitian ini antara lain adalah Instansi dapat melakukan penghematan pengadaan barang melalui konsep *multi user* yang terdapat pada produk ini. Konsep awalnya satu *PC* hanya untuk 1 *user* dapat dioptimasi menjadi 1 *PC* dapat digunakan hingga 4 *user*, Mahasiswa dapat dengan mudah mengakses berbagai *software* aplikasi yang sangat membantu dalam perkuliahan seperti rekomendasi mata kuliah yang harus diambil tiap semester, rekomendasi judul skripsi dan *search engine* untuk perpustakaan yang diintegrasikan dalam sebuah *software* induk

LANDASAN TEORI

Multi Touch Table System

Penggunaan teknologi layar sentuh sangat banyak digunakan pada produk elektronik *portable* maupun bersifat *home computer*. Pada awalnya Hugh Le Caine dan Robert Moog bereksperimen dengan menggunakan sensor kapatisasi untuk mengontrol suara yang keluar dari instrumen mereka. IBM mulai membangun sistem sentuh pertama pada akhir 1960 dan pada tahun 1972, *Control Data* merilis komputer PLATO IV yang difungsikan untuk tujuan pendidikan yang mempelajari titik tunggal sentuh dalam *array* 16x16 sebagai *user interfacenya*. (Stumpe, 2010:3) Teknologi *multi-touch* secara substansial dimulai pada tahun 1982, ketika *University of Toronto Input Research Group* mengembangkan sistem *multi-touch* pertama, meskipun saklar-matriks layar sentuh sistem sebelumnya dari MIT juga mampu deteksi *multi-touch*. Sistem di *University of Toronto* menggunakan panel kaca baur dengan kamera ditempatkan di belakang kaca. Ketika jari atau beberapa jari menekan pada kaca, kamera akan mendeteksi tindakan sebagai salah satu atau lebih titik hitam pada latar belakang jika tidak

putih, yang memungkinkan untuk dianggap sebagai *input*. Karena ukuran dari titik tergantung pada tekanan (seberapa keras orang itu menekan pada kaca). (Mehta, 1982:19)

Human Computer Interaction

Interaksi manusia-komputer (HCI) melibatkan studi, perencanaan, desain dan penggunaan antarmuka antara orang-orang (pengguna) dan komputer. Hal ini sering dianggap sebagai gabungan ilmu komputer, ilmu perilaku, desain, studi media, dan beberapa bidang studi lainnya. Istilah ini dipopulerkan oleh Stuart K. Card, Thomas P. Moran dan Newell Allen pada tahun 1983 buku mereka, *The Psychology of human computer interaction*, meskipun penulis pertama menggunakan istilah pada tahun 1980, penggunaan pertama pada tahun 1975, istilah berkonotasi bahwa, tidak seperti alat-alat lain dengan batasan. Komputer memiliki banyak *affordances* untuk digunakan dan ini terjadi dalam sebuah dialog terbuka yang berakhir antara pengguna dan komputer. (Carlisle, 1976:14)

UX Design

User experience design (UXD atau UED) adalah proses meningkatkan kepuasan dan loyalitas pelanggan dengan meningkatkan kegunaan, kemudahan penggunaan, dan kesenangan yang tersedia dalam interaksi antara pelanggan dan produk. (Kujala, 2011:3) Pengalaman pengguna desain meliputi desain *HCI*, dan meluas dengan mengatasi semua aspek produk atau jasa seperti yang dirasakan oleh pengguna. Pengalaman pengguna adalah setiap aspek dari interaksi seseorang dengan sistem IT yang diberikan, termasuk antarmuka, grafis, desain industri, interaksi fisik, dan manual. (Kujala, 2011:6) Bidang *User experience design* berakar dari faktor manusia dan ergonomi. Sejak akhir 1940-an, telah difokuskan pada interaksi antara manusia pengguna, mesin, dan lingkungan kontekstual untuk merancang sistem yang membahas pengalaman pengguna. Dengan perkembangan komputer di tempat kerja pada awal 1990-an, pengalaman pengguna menjadi perhatian penting bagi para desainer. Itu Donald Norman, pengalaman pengguna arsitek, yang menciptakan dan membawa pengalaman pengguna istilah pengetahuan yang lebih luas. Istilah ini juga memiliki koneksi yang lebih baru untuk pengguna berpusat desain, interaksi manusia-komputer, dan juga menggabungkan unsur-unsur dari sejenis bidang desain yang berpusat pada pengguna. (Kujala, 2011:23)

Citra

Citra adalah gambar dua dimensi yang dihasilkan dari gambar analog dua dimensi yang kontinu menjadi gambar diskrit melalui proses sampling. Gambar analog dibagi menjadi N baris dan M kolom sehingga menjadi gambar diskrit dimana setiap pasangan indeks baris dan kolom menyatakan nilai kecerahan titik tersebut. Titik-titik tersebut dinamakan sebagai elemen citra, atau *pixel (picture element)*. Dalam kamus komputer, gambar atau foto diistilahkan sebagai citra digital mempunyai representasi matematika berupa matriks $C_m \times n = (C_{ij})$. Secara sederhana dapat diartikan sebagai fungsi intensitas cahaya dua dimensi $f(x,y)$ dimana x dan y menunjukkan koordinat spasial dan nilai f pada suatu titik (x,y) sebanding dengan level keabuan dari citra di titik tersebut. (Munir, 2004:7)

Citra Digital

Citra digital adalah citra yang dapat diolah oleh komputer dimana x dan y adalah koordinat bidang datar, dan harga fungsi f di setiap pasangan koordinat (x,y) disebut intensitas atau level keabuan dari gambar di titik itu. Jika x,y dan f semua berhingga (*finite*), dan nilainya diskrit, maka gambarnya disebut citra digital. (Fajar, 2013:3)

Citra RGB

Setiap piksel pada citra warna mewakili warna yang merupakan kombinasi dari tiga warna dasar (RGB = *Red Green Blue*). Setiap warna dasar menggunakan penyimpanan 8 bit = 1 *byte*, yang berarti setiap warna mempunyai gradasi sebanyak 255 warna. Berarti setiap piksel mempunyai kombinasi warna sebanyak 16 juta warna lebih. Itulah sebabnya format ini dinamakan *true color* karena mempunyai jumlah warna yang cukup besar sehingga bisa dikatakan hampir mencakup semua warna di dunia nyata. (fajar, 2013:97)

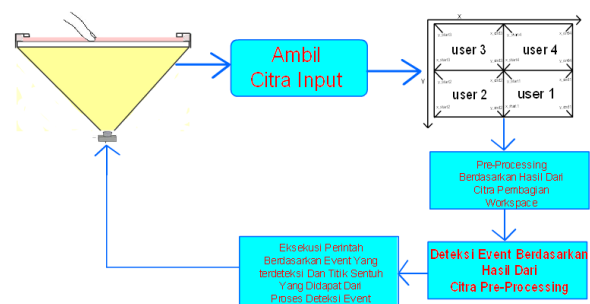
Citra Biner

Citra biner adalah citra yang hanya mempunyai dua nilai derajat keabuan: hitam dan putih. Meskipun saat ini citra berwarna lebih disukai karena memberi kesan yang lebih kaya daripada citra biner, namun tidak membuat citra biner mati. Pada beberapa aplikasi citra biner masih tetap dibutuhkan, misalnya citra logo instansi (yang hanya terdiri atas warna hitam dan putih), citra kode batang (*bar code*) yang tertera pada label barang, citra hasil pemindaian dokumen teks, dan sebagainya. (Fajar, 2013:16)

METODOLOGI PENELITIAN

Analisis Data

Pada Perancangan *multi touch table system* dapat dirincikan beberapa permasalahan dan solusi yang akan dibahas yakni Kamera yang merupakan sensor utama sistem untuk pendeteksian jenis sentuh serta koordinat sentuh. Pemilihan kamera sangat penting dikarenakan ada beberapa hal yang harus diperhatikan antara lain resolusi dan *fps* kamera itu sendiri. Resolusi merupakan ukuran citra yang dihasilkan kamera dan *fps* adalah kecepatan kamera melakukan *update* citra dalam satu detik yang berguna mengikuti gerakan jari pengguna saat mengoperasikan meja. Kamera yang dipilih adalah jenis *ps3 eye* karena memiliki resolusi yang tinggi yakni 640×480 dan memiliki *fps* yang tinggi yakni 60 *fps*. Selanjutnya adalah tentang metode pengolahan citra yang diperlukan untuk mendapatkan kualitas citra yang layak sebelum dilakukan deteksi *event*. Metode pengolahan citra yang dipakai dibagi menjadi dua yakni *filtering warna* dan *erosi citra*. Metode *filtering warna* dipakai agar sistem dapat membedakan antara pixel objek dan bukan pixel objek karena sistem hanya membutuhkan pixel objek yang akan dideteksi jenis sentuhnya. Metode *erosi citra* dilakukan setelah proses *filtering warna* selesai yang berfungsi untuk menghilangkan titik-titik kecil yang dianggap *noise* sehingga citra yang dihasilkan adalah citra dengan yang memiliki kualitas baik untuk dilakukan pendeteksian jenis sentuh. Gambaran pengolahan sistem secara global dapat dilihat pada gambar 1.



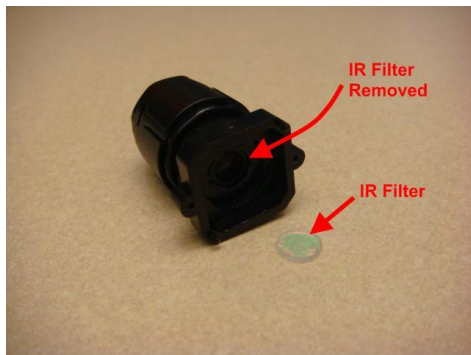
Gambar 1. kinerja Sistem Secara Global

PEMBAHASAN DAN HASIL

Konsep Hardware

Rancangan hardware *multi touch table system* membutuhkan beberapa komponen pembantu seperti kamera, lampu LED inframerah, meja, pijakan kaki, cermin, akrilik sebagai *surface* dan proyektor yang akan dibahas selanjutnya. Pertama adalah Kamera digunakan sebagai sensor untuk menangkap saat pengguna melakukan sentuhan pada *surface* sehingga didapat titik-titik yang disentuh oleh pengguna dan selanjutnya akan diproses untuk deteksi *click event*. Kamera yang

dipakai adalah jenis PS3 eye, karena memiliki spesifikasi yang cukup baik saat digunakan sebagai sensor sentuh antara lain resolusi mencapai 640 x 480 dan memiliki *framerate* hingga 60 *frame per second* sehingga dapat menangkap perpindahan sentuh real time dengan sangat baik. Sebelum digunakan kamera harus dimodifikasi terlebih dahulu yaitu dengan melepas *filter* inframerah dan menambahkan negatif *film* pada lensa bagian dalam sehingga kamera hanya dapat menangkap objek yang terkena cahaya inframerah dan akan mengabaikan objek lain yang tidak terkena cahaya inframerah. Pada gambar 2 dapat dilihat filter inframerah yang telah dilepas pada kamera PS3 eye.



Gambar 2. Melepas IR filter

Tahap selanjutnya adalah menambahkan negatif *film* yang telah hangus atau dalam kata lain berwarna hitam pekat pada bagian dalam lensa. Tanpa negatif *film* kamera akan tetap menangkap objek yang tidak terkena cahaya inframerah dan jika negatif *film* yang digunakan tidak berwarna hitam pekat maka hasil citra yang didapatkan akan hampir sama jika saat tidak diberi negatif *film* sama sekali. Pemilihan resolusi kamera dan *framerate* menjadi hal penting agar *image processing* dapat lebih efisien dan tidak memakan proses yang terlalu lama. Proses menambahkan negatif *film* dapat dilihat pada gambar 3.



Gambar 3 Menambahkan Negatif *film*
Acrylic digunakan sebagai media sentuh sekaligus sebagai media *display*. *Acrylic* yang digunakan berbentuk persegi panjang berukuran 120cm x 100cm, pada bagian bawah *Acrylic* dilapisi dengan

skotlet berwarna buram yang berfungsi menahan *display* dari *projektor* agar dapat ditampilkan pada *acrylic*. *Acrylic* ditempatkan diatas rangka meja yang terbuat dari almunium, tebal *Acrylic* yang digunakan adalah 6mm, *Acrylic* dipilih karna tidak rapuh seperti kaca, tahan banting serta memiliki pembiasan cahaya yang lebih baik dari pada kaca, namun kekurangan *Acrylic* terdapat pada mudahnya tergores jika terkena benda runcing. Secara tekstur juga *Acrylic* memiliki kelebihan dari pada kaca yakni lebih lentur dari pada kaca. Tampilan *surface acrylic* dapat dilihat pada gambar 4.



Gambar 4. Surface

Projector berfungsi sebagai penampil *display* pengganti *output* monitor yang ditampilkan pada *surface*. *Projector* yang dipakai adalah *projector* merek epson tipe rx-5, *projector* diletakan secara horizontal berhadapan dengan cermin dengan jarak 90cm dari cermin yang dipantulkan ke *surface* oleh cermin sehingga *ouput* dari *projector* dapat tampil pada *surface* secara *fullscreen*. penampilan dari epson tipe rx-5 dapat dilihat pada gambar 5.



Gambar 5. Proyektor

LED berfungsi sebagai pemancar cahaya inframerah yang diletakan diatas *surface* secara *horizontal* agar saat pengguna menyentuh *surface* maka bagian jari yang menyentuh bagian *surface* akan terkena cahaya inframerah sehingga kamera

dapat menangkap titik yang disentuh oleh pengguna. LED dirangkai secara paralel dan diletakan secara mengitari *surface* seperti pada gambar 6. Setiap rangkaian diberi tegangan 3 volt dari sumber tegangan, serta bagian atas LED ditutupi agar pancaran cahaya tidak mengenai lengan dari pengguna. Jarak antara penutup LED dan *surface* adalah 0.3cm sehingga bagian jari yang menyentuh akrilik saja yang terkena cahaya inframerah.



Gambar 6. Rangkain LED

Rangka meja berfungsi sebagai tulang yang menyangga beban dari meja sehingga dibutuhkan bahan yang kuat untuk membuat rangka ini. Bahan yang digunakan adalah rangka almunium. Ukuran dari rangka meja adalah untuk panjang adalah 127cm, lebar adalah 110cm dan tinggi adalah 100cm, karena *surface* memiliki ukuran cukup besar maka tinggi harus disesuaikan agar kamera dapat menangkap seluruh bagian *surface*. Rangka meja memiliki alas sebagai tempat hardware yang dipakai dalam proyek ini seperti PC, kamera dan *mirror*. Pada bagian sisi rangka ditambahkan pipa untuk menutupi kabel arus yang dilingkarkan pada bagian samping meja dan pada bagian sisi tengah ditutupi dengan tripleks karena perlu diingat bahwa kamera inframerah sangat sensitif terhadap cahaya matahari. Penampakan rangka meja dapat dilihat pada gambar 7.



Gambar 7. Rangka Meja

Pijakan kaki diperlukan karena dilihat dari tinggi meja yakni 100cm sangat tidak nyaman untuk digunakan oleh pengguna, karena pada umumnya tinggi meja adalah sekitar 70cm sampai 80cm. Selain itu pijakan kaki juga digunakan sebagai tempat *projector*, karena *projector* membutuhkan jarak yang cukup panjang untuk dipantulkan oleh *mirror* agar dapat memenuhi semua bagian *surface*. Pijakan kaki diletakan pada empat sisi bagian bawah meja sehingga menutupi seluruh bagian bawah meja. Pijakan kaki terbuat dari kayu dengan panjang 160cm, lebar 50cm dan tinggi 20cm pada sisi depan dan belakang, sedangkan untuk sisi kiri dan kanan memiliki panjang 120cm, lebar 50cm dan tinggi 20cm. Pada bagian belakang tempat *projector* diberi dua buah *fan* agar *projector* tidak *overheat* saat digunakan. Pijakan kaki dapat dilihat pada gambar 8.



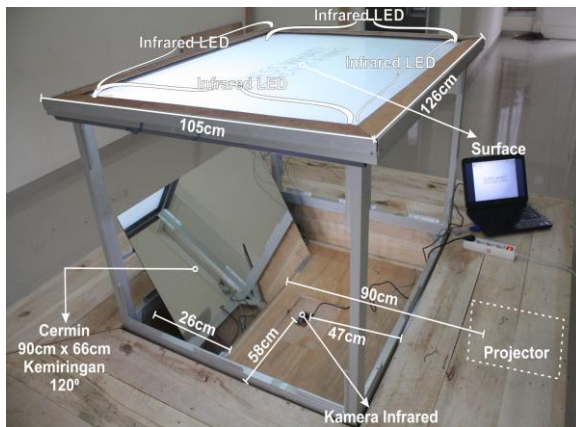
Gambar 8. Pijakan Kaki

Cermin difungsikan sebagai pemantul *display* dari *projector* menuju *surface*. Cermin diletakan berlawanan arah dengan *projector* dan berjarak 26.5cm dari sisi belakang meja dan diletakan dengan kemiringan 120 derajat. Posisi ini menyebabkan gambar yang dihasilkan berbentuk trapesium namun cukup untuk memnugi seluruh bagian *surface*. Penampakan cermin dapat dilihat pada gambar 9.



Gambar 9. Cermin

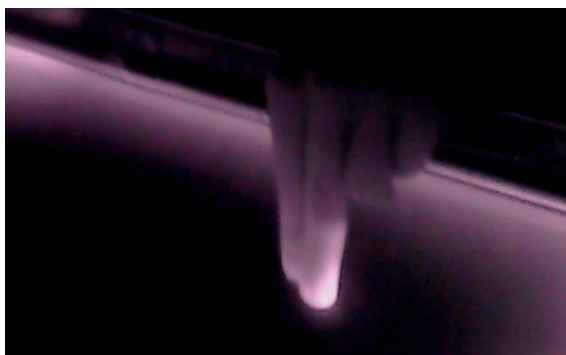
Secara keseluruhan Detail meja dapat dilihat pada gambar 10.



Gambar 10. Detail Meja

Sistem Secara Global

Seperti yang dapat dilihat pada gambar 1 sistem secara global dapat dirincikan sebagai berikut. Pada tahap pengambilan citra input, kamera akan menangkap objek yang terkena cahaya inframerah yang berasal dari LED inframerah pada tiap bagian sisi dari meja. Jika ada objek yang terkena cahaya inframerah maka dapat diartikan jari pengguna sedang menyentuh *surface* yang kemudian keadaan ini akan ditangkap oleh kamera berupa file citra yang akan diproses pada tahap selanjutnya. Saat jari pengguna menempel pada *surface* terjadi perpotongan cahaya yang terhadang oleh jari pengguna sehingga menyebabkan jari pengguna berwarna putih. Contoh gambar perpotongan cahaya dapat dilihat pada gambar 11.

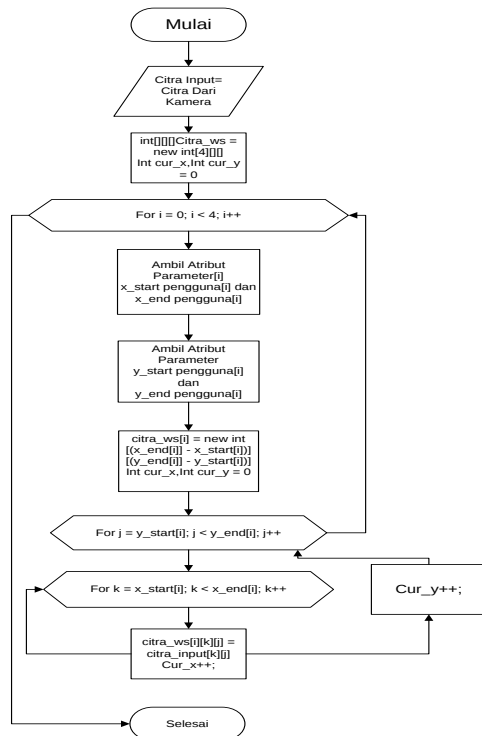


Gambar 11. Perpotongan cahaya jari pengguna Setelah citra input didapatkan, selanjutnya akan dilakukan pembagian *workspace* berdasarkan beberapa atribut parameter yakni x_{start} , x_{end} , y_{start} dan y_{end} untuk tiap pengguna. Penjelasan atribut parameter secara detail dapat dilihat pada sub-bab berikutnya. Setelah proses pembagian *workspace* selesai didapatkan empat buah citra yang merupakan pembagian wilayah dari citra

input menurut atribut parameter yang telah disebutkan sebelumnya. Proses pembagian *workspace* menghasilkan empat buah citra yang siap untuk dilakukan *pre-processing*. Tiap citra akan melalui dua buah proses *pre-processing* yakni *filtering* warna yang berfungsi membedakan antara objek yang menyentuh *surface* dan yang tidak menyentuh *surface* dan erosi citra yang berguna untuk menghilangkan titik-titik kecil yang dianggap *noise* yang dapat mengganggu proses deteksi event. Setelah mendapat citra yang layak hasil dari *pre-processing* sekarang dapat dilakukan proses deteksi *event*. Pada tahap deteksi *event* terdapat dua buah proses yang harus dilalui oleh citra yakni proses deteksi *event* dasar dan proses proses deteksi *event* lanjutan. Proses ini menghasilkan jenis sentuh yang di-input-kan oleh pengguna dan titik yang disentuh oleh pengguna. Penjelasan secara lengkap dapat dilihat pada sub-bab selanjutnya. Proses akhir yakni eksekusi perintah. Pada tahap sebelumnya kita telah mendapatkan jenis sentuh yang di-input-kan pengguna dan titik yang disentuh pengguna. Dengan hasil yang telah didapat sistem dapat mengeksekusi perintah yang telah diberikan sesuai jenis sentuh dan titik sentuh yang telah dideteksi.

Pembagian Workspace

Sebagian Fokus utama adalah bagaimana mengoptimasi konsep *multi user* dimana sebuah PC yang pada umumnya hanya dapat digunakan untuk satu orang, dapat dioptimasi hingga empat orang dalam sebuah PC. Proses *multi user* memerlukan beberapa atribut parameter dari sistem antara lain. X_{start} adalah variabel penyimpanan yang dimiliki tiap *workspace* sebagai posisi mulainya *workspace* tiap pengguna pada sumbu x, X_{end} adalah variabel penyimpanan yang dimiliki tiap *workspace* sebagai posisi berakhirnya *workspace* tiap pengguna pada sumbu x, Y_{start} adalah variabel penyimpanan yang dimiliki tiap *workspace* sebagai posisi mulainya *workspace* tiap pengguna pada sumbu y dan yang terakhir adalah Y_{end} adalah variabel penyimpanan yang dimiliki tiap *workspace* sebagai posisi berakhirnya *workspace* tiap pengguna pada sumbu y. Sistem telah menentukan ke-empat atribut parameter untuk tiap pengguna sesuai kebutuhan dari tiap pengguna. Tiap *workspace* memiliki tinggi dan lebar yang tidak sama dikarenakan harus menghindari area blind spot pada tiap sisi meja karena pantulan cahaya inframerah yang sangat terang. Pembagian penanganan untuk tiap pengguna dapat dilihat dalam *flowchart* yang terdapat pada gambar 12.



Gambar 12. Flowchart Pembagian Workspace Filtering Warna

Proses *filtering* warna dimaksudkan untuk memisahkan antara objek dan bukan objek. Objek yang berarti jari pengguna yang tersentuh *surface* memiliki nilai RGB yang berbeda dengan yang hanya tidak menempel pada *surface*.

Sistem memerlukan atribut parameter untuk melakukan *filtering* warna antara lain:

A. MIN_R

MIN_R merupakan representasi batas minimum nilai *red* dari tiap pixel yang diproses *filtering* warna. Nilai MIN_R pada sistem adalah 36.

B. MIN_G

MIN_G merupakan representasi batas minimum nilai *green* dari tiap pixel yang diproses *filtering* warna. Nilai MIN_G pada sistem adalah 40.

C. MIN_B

MIN_B merupakan representasi batas minimum nilai *blue* dari tiap pixel yang diproses *filtering* warna. Nilai MIN_B pada sistem adalah 44.

D. MAX_R

MAX_R merupakan representasi batas maksimum nilai *red* dari tiap pixel yang diproses *filtering* warna. Nilai MAX_R pada sistem adalah 118

E. MAX_G

MAX_G merupakan representasi batas maksimum nilai *green* dari tiap pixel yang diproses *filtering* warna. Nilai MAX_G pada sistem adalah 122.

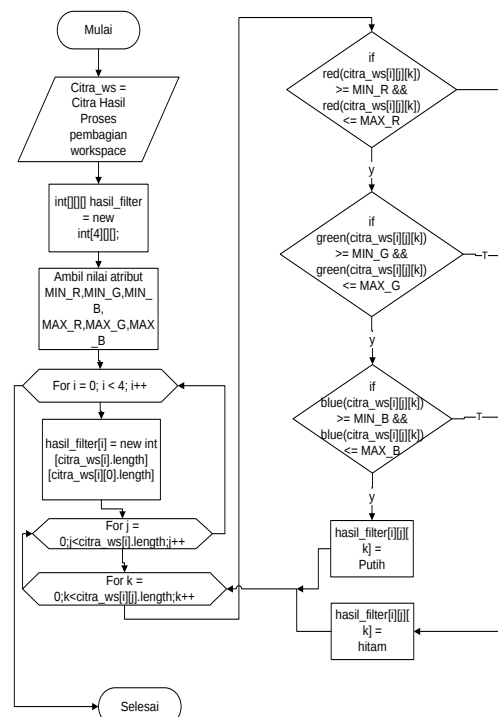
F. MAX_B

MAX_B merupakan representasi batas maksimum

nilai *blue* dari tiap pixel yang diproses *filtering* warna. Nilai MAX_B pada sistem adalah 131.

Secara singkat jika warna pixel dari citra tidak berada dalam *range* atribut yang telah disebutkan maka pixel tersebut dianggap sebagai pixel bukan objek yang diberi warna hitam, sedangkan jika nilai pixel berada dalam *range* atribut yang telah disebutkan maka pixel tersebut dianggap sebagai pixel objek yang diberi warna putih. Dari hasil *filtering* warna sistem memiliki citra yang terdiri dari objek yakni jari pengguna yang menempel pada *surface* yang diwakili dengan warna putih dan *background* bagian yang berwarna hitam. Namun ada beberapa kendala didapatkan dalam proses ini yakni adanya *blind spot* yakni area yang terkena pantulan cahaya inframerah yang sangat terang hingga saat jari pengguna berinteraksi dengan *surface* tidak terdapat perbedaan warna antara objek dan area *blind spot*, dan *noise* atau titik-titik kecil pada citra.

Proses *filtering* warna juga tetap akan menangkap objek benda padat selain jari yang menempel pada *surface*. Beberapa hal yang harus diperhatikan adalah *filtering* warna tetap akan mengambil objek yang melayang sedikit diatas *surface* karena cahaya inframerah yang dipancarkan oleh LED inframerah terpantul pada *surface*. Flowchart *filtering* warna dapat dilihat pada gambar 13.

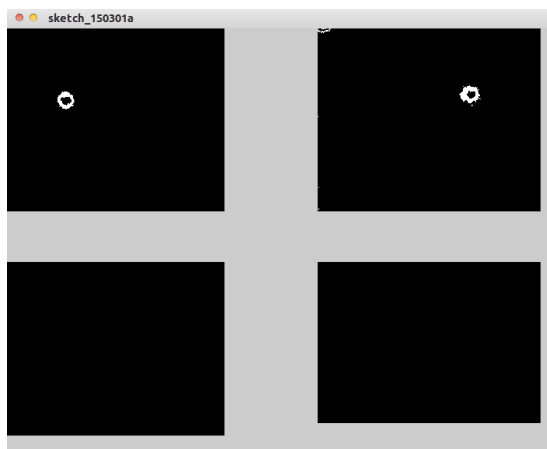


Gambar 13. Flowchart Filtering Warna
Seperti contoh sistem menerima input citra seperti pada gambar 14.



Gambar 14. Input Citra

Akan dilakukan proses filtering warna dengan nilai MIN_R, MAX_R, MIN_B, MAX_B, MIN_G dan MAX_G seperti yang telah ditentukan sebelumnya Sesuai dengan hasil pembagian *workspace* nilai *red*, *green* dan *blue* tiap pixel akan difilter sesuai atribut parameter yang telah disebutkan sebelumnya. Hasil dari *filtering* warna dapat dilihat pada gambar 15. Dapat dilihat bahwa ditemukan objek pada *user* ketiga dan *user* empat. Untuk *user* ketiga dapat dilihat citra setelah *filtering* warna sudah baik sedangkan untuk *user* keempat terdapat noise pada bagian kiri dan atas citra. Nois ini akan dihilangkan dalam proses berikutnya yakni proses erosi citra.

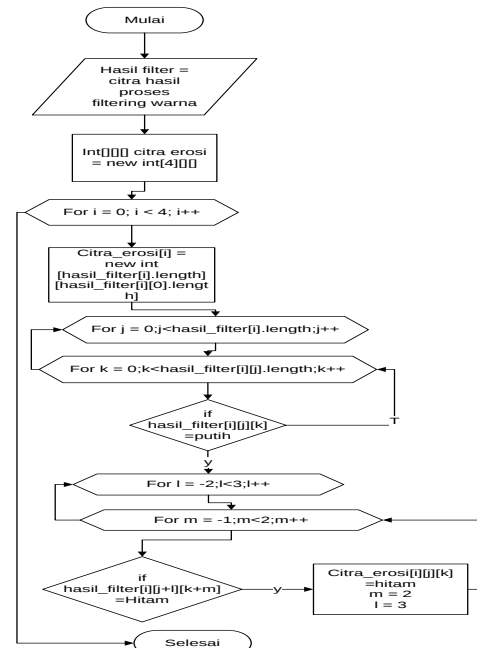


Gambar 15. Citra Hasil Filtering Warna

Erosi Citra

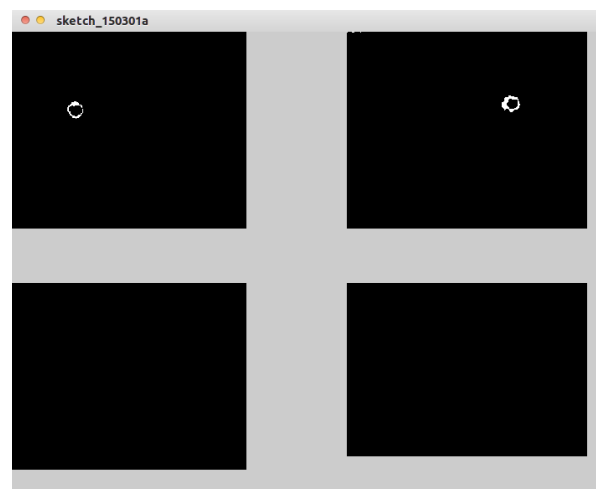
Erosi citra berfungsi untuk menghilangkan titik-titik kecil yang dianggap sebagai *noise* yang tidak dibutuhkan untuk proses deteksi jenis sentuh. Cara kerja erosi citra adalah dengan melakukan pengecekan terhadap pixel tetangga secara *vertical* dan *horizontal*. Secara *vertical* dengan jumlah empat pixel yakni dua pixel tetangga pada bagian atas dan dua pixel tetangga pada bagian bawah. Sedangkan secara *horizontal* berjumlah dua pixel

yakni satu pixel pada bagian kiri dan satu pixel pada bagian kanan. Setelah dilakukan pengecekan andaikan salah satu dari pixel tetangga berwarna hitam atau bukan objek maka pixel yang sedang diproses akan bernilai hitam dan jika tidak ada pixel tetangga yang bernilai hitam maka pixel yang sedang diproses akan bernilai putih atau pixel objek. Detail proses erosi citra dapat dilihat pada *flowchart* yang terdapat pada gambar 16.



Gambar 16 Flowchart Erosi Citra

Misalkan sistem akan melakukan proses erosi citra terhadap citra hasil *filtering* warna yang terdapat pada gambar 15, maka akan dihasilkan seperti pada gambar 16.

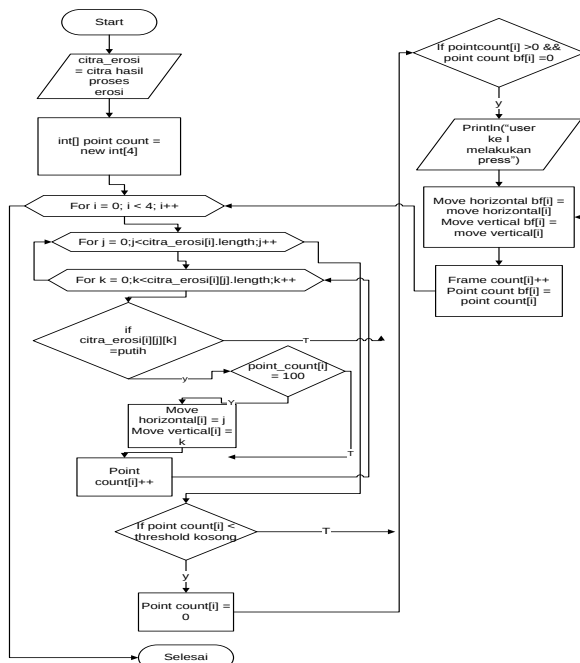


Gambar 17. Hasil Proses Erosi Citra

Dapat dilihat noise pada *user* kanan atas sebelumnya memiliki noise pada citra bagian kiri atas setelah dilakukan proses erosi citra noise tersebut berkurang.

Press

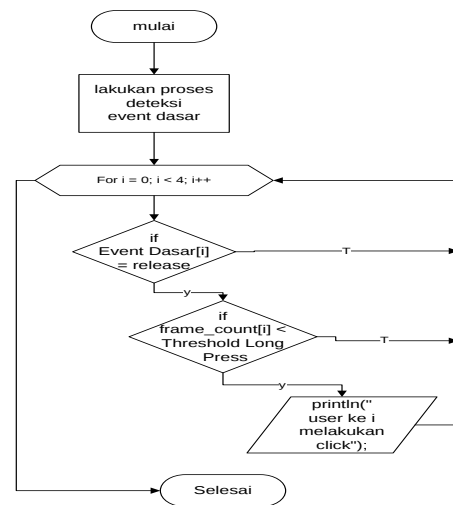
Event press adalah saat dimana pengguna menempelkan jari pada suface setelah sebelumnya telah melakukan *event release*. Pada *event* ini akan dijalankan beberapa perintah seperti menyimpan posisi x dan y untuk *flick vertical* PF dan *flick horizontal* PF. Detail pendeteksian *event press* dapat dilihat pada *flowchart* yang terdapat pada gambar 18.



Gambar 18 Flowchart Event Press

Hold

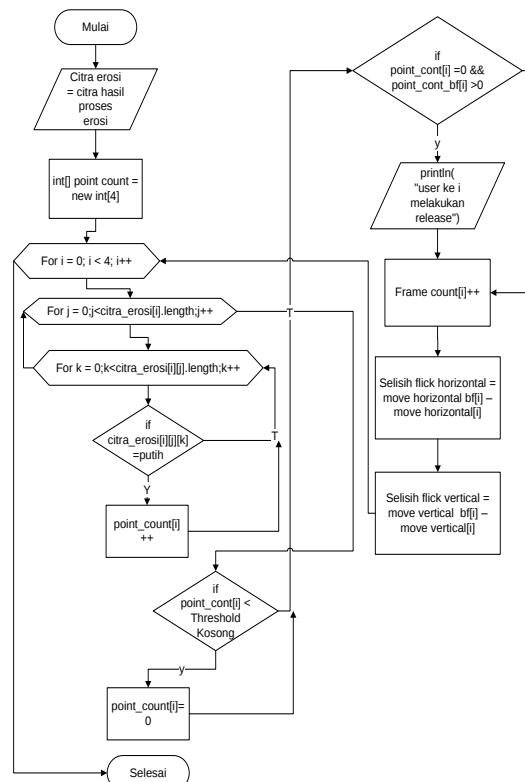
Event hold adalah *event* dimana pengguna menahan jari tetap menempel pada *surface*. Kebanyakan *event* ini digunakan untuk fungsi *drag and drop*. Pada *event* ini dilakukan fungsi counter frame yang telah dilewati oleh sistem selama *event hold*. Detail pendeteksian *event hold* dapat dilihat pada *flowchart* yang terdapat pada gambar 19. Pada *event* ini akan dilakukan pendeteksian *event* dengan jenis *long press*. *Event hold* terdiri dari beberapa tahap yang hampir sama dengan *event press* yakni mengambil tiap pixel hasil dari citra erosi, untuk tiap pixel object atau pixel yang berwarna putih disimpan keseluruhannya pada variabel *Point Count*. Setelah perhitungan pixel objek selesai dilakukan pengecekan jika jumlah total pixel objek kurang dari *Threshold* Kosong, jika tidak maka variabel *Point Count* akan diset menjadi nol.



Gambar 19 Flowchart Event Hold

Release

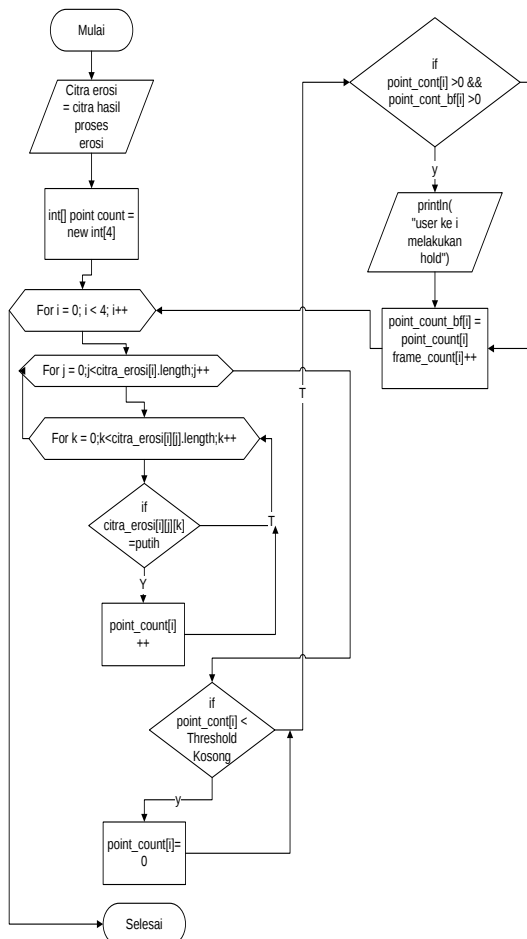
Event press adalah *event* dimana pengguna melepaskan sentuhan terhadap *suface* setelah melakukan *event press* atau *event hold* sebelumnya. *Event* ini adalah dasar untuk mendeteksi lima *event* lanjutan yakni *click*, *flick* kiri, *flick* kanan, *flick* atas dan *flick* bawah. *Flowchart event release* dapat dilihat pada gambar 20.



Gambar 20 Flowchart Event Release

Tap

Event click/tap adalah gabungan antara dua *event* dasar yakni *event press* dan *event release* yang dilakukan dalam kurun waktu kurang dari atribut nilai *Threshold Long Press* seperti yang telah dijelaskan pada sub bab sebelumnya. Implementasi kita mengambil input hasil erosi dan akan menampilkan beberapa gambar selama proses *event click* berlangsung. *Flowchart event tap* dapat dilihat pada gambar 21.



Gambar 21 Flowchart Event Tap
Kinerja tan dapat dilihat pada gambar 22.



Gambar 22 Kinerja Tap

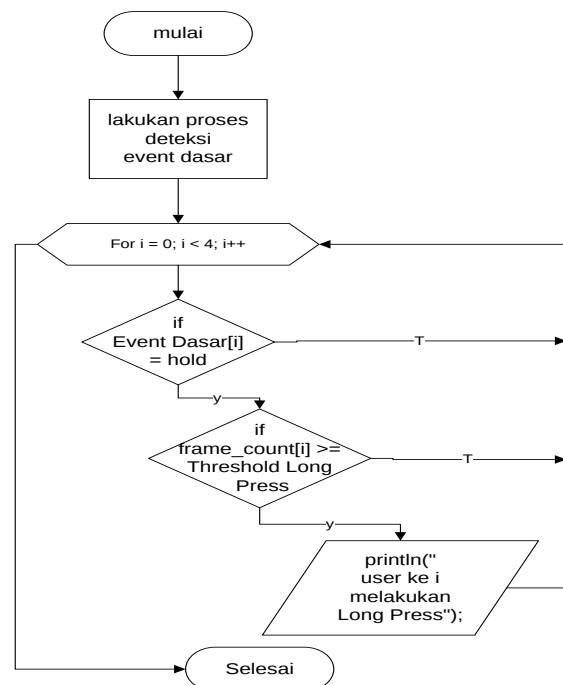
Long Press

Long press berfungsi saat pengguna melakukan *event hold* pada sebuah menu atau objek seperti halnya *drag and drop* yang dilakukan pada *personal computer* dan kinerja *long press* pada *tablet computer*. Kinerja *long press* pada user dapat dilihat pada gambar 23.



Gambar 24 Kinerja Long Press

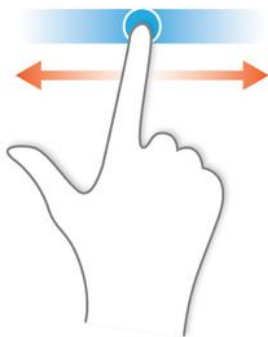
Detail dari kinerja *long press* dapat dilihat pada *flowchart* yang terdapat pada gambar 25. *Long press* memiliki beberapa atribut yang hampir sama seperti tap antara lain *threshold* pixel kosong dimana berfungsi untuk melakukan filter terhadap citra yang jumlah pixel objectnya dibawah minimum jumlah pixel object yang telah ditentukan dalam kata lain jika ditetapkan minimum jumlah pixel object yang telah ditentukan adalah 30 maka jika ada citra dengan jumlah pixel object kurang dari 30 maka citra tersebut dianggap tidak memiliki objek. *Event long press* akan terus berlanjut selama *event hold* juga terus terdeteksi. Saat *event release* terdeteksi maka kondisi *long press* akan berakhir. Berikut *flowchart event long press*.



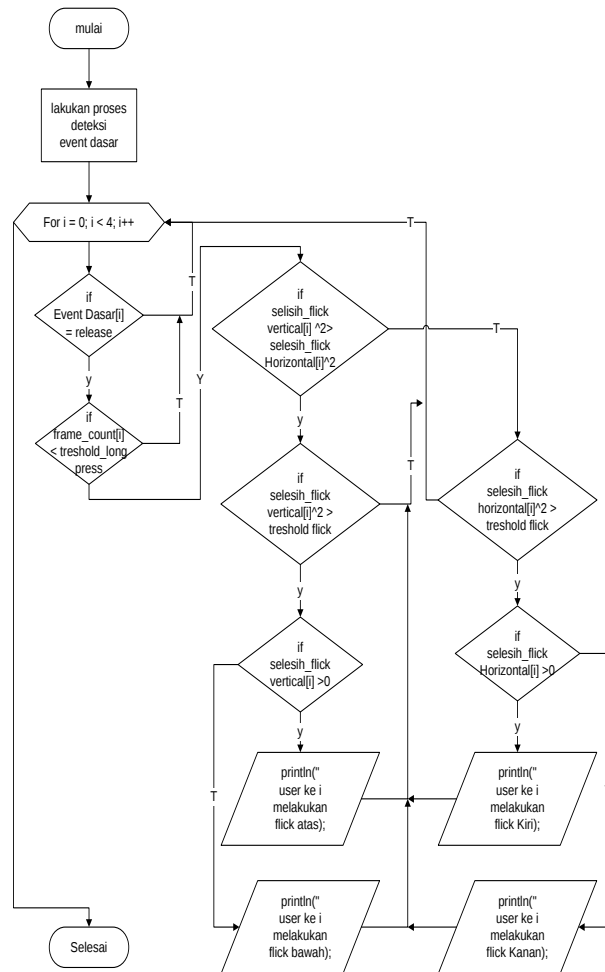
Gambar 25 Flowchart Long Press

Flick

Fitur *flick* berfungsi sebagai navigasi dalam sistem *pagination* pada hasil pencarian dokumen maupun menu yang dapat diarahkan ke arah kanan, kiri, atas dan bawah sebagai pemindah halaman. Secara umum penggunaan *flick* sama seperti yang ada pada *table computer* atau *smart phone* saat ini. Kinerja *flick* pada *user* dapat dilihat pada gambar 26 *Flick* memiliki beberapa jenis sesuai dengan arah yang diinginkan pengguna antara lain *flick* ke arah kanan, *flick* ke arah kiri, *flick* ke arah atas dan *flick* ke arah bawah. Semua jenis *flick* memiliki satu atribut parameter yang sama yakni *threshold* atau batas ambang yang merupakan jarak antara titik pertama saat *event press* dan titik kedua saat *event release*. Jika jarak antara kedua titik tersebut melebihi *threshold* maka akan dianggap sebagai *event flick* namun jika tidak akan dianggap sebagai *tap/ click*. Cara membedakan jenis *flick* adalah dengan atribut selisih *flick horizontal* dan selisih *flick vertical*, dimana selisih *flick horizontal* akan menghitung jarak *horizontal* antar titik x dan selisih *flick vertical* akan menghitung jarak *vertical* antar titik y. Perhitungan jarak *horizontal* untuk mendapatkan jenis *flick* kiri atau *flick* kanan dengan mengurangi titik pertama dan titik kedua lalu melakukan pengecekan jika selisih yang dihasilkan bukan nilai *absolute* maka akan dianggap sebagai *flick* kanan dan jika bukan nilai *absolute* maka akan dianggap sebagai *flick* kiri, akan tetapi kondisi ini hanya berlaku untuk *user* pertama dan *user* kedua karena posisi kedua *user* tersebut sama sedangkan untuk *user* ketiga dan *user* keempat yang memiliki posisi berlawanan maka selisih dengan nilai *absolute* akan dianggap sebagai *flick* kiri dan jika bukan nilai *absolute* akan dianggap sebagai *flick* kanan. *Flowchart* deteksi *flick* dapat dilihat pada gambar 27



Gambar 26 Kinerja Flick

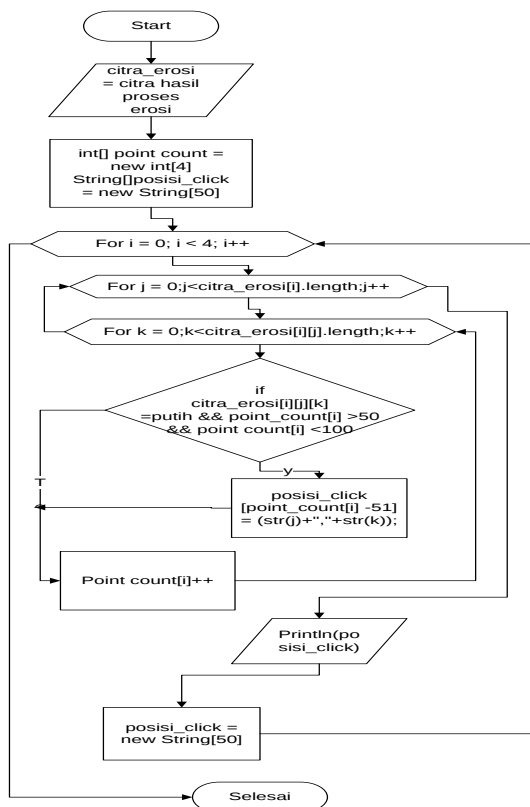


Gambar 27 Flowchart Flick

Pengambilan Koordinat Sentuh

Setelah mendapatkan *event* yang di-inputkan oleh pengguna hal yang terakhir dilakukan adalah sistem harus mengetahui koordinat-koordinat yang disentuh oleh pengguna. Koordinat tersebut dibutuhkan untuk mengetahui menu atau objek apa yang disentuh oleh pengguna. Konsep pengambilan koordinat sentuh memiliki langkah-langkah yang hampir sama saat mengambil nilai point count akan tetapi untuk mengambil koordinat sentuhan sistem mengambil atribut point count ke 50 sampai point count ke 100 sehingga total koordinat sentuh dalam sebuah frame terdapat 50 kombinasi antara sumbu x dan sumbu y. Setelah didapatkan titik-titik yang disentuh oleh pengguna tinggal melakukan nilai minimal dari objek yang mendekati titik sentuh karena titik sentuh bersifat linear untuk tiap sumbu, jadi sistem cukup melakukan *check-ing* nilai minimum dan maximum dari tiap sumbu. Akan tetapi hal ini tidak akan dibahas dalam

flowchart karena pembahasan hanya terbatas pada deteksi event dan pengambilan koordinat sentuh. Flowchart pengambilan koordinat sentuh dapat dilihat pada gambar 28.



Gambar 28 Flowchart Pengambilan Koordinat

Penutup

Kesimpulan yang dapat diambil adalah Sistem ini sangat *powerfull* karena walaupun secara kinerja penanganan dibagi menjadi empat *user* akan tetapi kestabilan pelayanan untuk tiap *user* sangat terjaga karena tidak terlalu banyak proses pada pendeteksi inputan sentuh namun tidak mengurangi kenyamanan dan fungsional untuk tiap *user*. Konsep ini juga dapat digunakan bukan hanya dalam lingkup akademik, tetapi dalam berbagai organisasi jenis apapun, tergantung *module* yang dipakai. Akan tetapi didalam konsep dari laporan ini hanya dikhususkan dalam dunia akademik dikarenakan *module* yang dipakai dikhususkan untuk kegiatan mahasiswa serta pengajaran. Tidak melalui banyak proses pengolahan citra yang dipakai antara lain hanya berupa pembineran *input* citra dan pencocokan koordinat menu yang aktif dan beberapa pengenalan *gesture* seperti *tap*, *long press* dan *flick*. Sedangkan untuk *hardware* *user* untuk membuat *MTTS* dibutuhkan *cost* cukup banyak, tumpuan

terberat *cost* ada pada *projector*, *acrylic* dan rangka meja. Beberapa hal yang bisa ditambahkan dalam konsep dari produk ini adalah Penambahan *add-ons* untuk tiap menu sehingga jika ada *module-module* baru yang dibutuhkan dapat ditambahkan tanpa harus *re-design* program, dengan begitu produk menjadi lebih *flexible* dalam penggunaannya, Dapat ditambahkan pembagian *workspace* untuk *user* secara otomatis dalam kata lain jika *MTTS* hanya dipakai untuk satu *user* maka seluruh meja menjadi *workspace user* itu sendiri, dan saat ada *user* baru yang masuk secara otomatis *workspace* akan terbagi menjadi dua dan seterusnya hingga ada empat *user* maksimal yang masuk, LED inframerah dapat diganti dengan laser inframerah yang ditempatkan pada tiap sudut meja agar dapat mengurangi banyak noise dikarenakan arah pembiasan cahaya lurus datar tidak seperti led yang melebar secara *vertical*.

Datar Pustaka

1. Bent Stumpe, A new principle for x-y touch system, CERN, Meyrin. 2010
2. Carlisle, James H, Evaluating the impact of office automation on top management communication, National Computer Conference and Exposition. pp. 611-616. 1976
3. Fajar Astuti, Pengolahan Citra Digital Konsep & Teori, Andi, Yogyakarta. 2013
4. Kujala Sari, Roto, Väänänen-Vainio-Mattila, Kaisa Karapanos, Evangelos Sinnelä, Arto, UX Curve: A method for evaluating long-term user experience. Interacting With Computers, Partridge, New Delhi. 2011
5. Mehta Nimish, A Flexible Machine Interface, M.A.Sc. Thesis, Department of Electrical Engineering University of Toronto supervised. 1982
7. Rinaldi Munir, Pengolahan Citra Digital dengan Pendekatan Algoritmik, Informatika, Bandung. 2004