



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

METODOLOGI DAN PERANCANGAN

3.1 Metode Penelitian

Metode penelitian yang digunakan dalam penelitian dapat dijabarkan sebagai berikut.

1. Tahapan studi literatur

Dalam tahapan ini dilakukan studi dan pemahaman terhadap literatur-literatur mengenai teori yang menunjang dalam pelaksanaan penelitian. Beberapa teori yang dipelajari antara lain teori mengenai tanda tangan, normalisasi, *Discrete Wavelet Transform*, *Multiscale Fourier Descriptor*, *Distance Measure*, Android, dan rekayasa piranti lunak. Referensi yang digunakan antara lain buku-buku mengenai teori yang bersangkutan, jurnal-jurnal ilmiah, artikel, dokumentasi, dan sumber-sumber lainnya.

2. Tahapan perancangan dan analisis

Tahapan perancangan merupakan tahapan di mana sistem dirancang secara keseluruhan dengan menggunakan diagram alir. Perancangan terhadap tampilan antar muka juga dilakukan pada tahapan ini. Tahapan analisis dibahas lebih mendetail pada subbab 3.2.

3. Tahapan implementasi

Pembangunan aplikasi dilakukan pada tahap ini. Implementasi terhadap metode-metode yang didefinisikan sebelumnya dan rancangan diagram alir

pada tahapan perancangan ke dalam bahasa pemrograman yang telah ditentukan.

4. Tahapan uji coba dan evaluasi

Uji coba dilakukan baik itu secara internal (dilakukan sendiri) maupun dengan menggunakan kuisioner. Hasil evaluasi didapatkan dengan mengambil kesimpulan dari kuisioner yang telah diisi oleh responden.

3.2 Analisis

Penelitian ini menggunakan dua buah algoritma utama yakni *Discrete Wavelet Transform* dan *Multiscale Fourier Descriptor*. Kedua algoritma tersebut didukung dengan algoritma sederhana untuk melakukan normalisasi dan *distance measure*. Normalisasi yang dilakukan adalah *cropping*, *rotation* dan *scaling*. Rotasi hanya dilakukan pada fase otentikasi sehingga pengguna dapat menyesuaikan kemiringan tanda tangan pada fase otentikasi agar mendekati tanda tangan pada fase registrasi. Algoritma *distance measure* yang digunakan adalah Manhattan Distance atau yang biasa dikenal dengan Taxicab Geometry. *Distance measure* ini digunakan untuk menghitung jarak antara tanda tangan pada fase registrasi dan tanda tangan pada fase otentikasi.

Aplikasi yang dibuat dalam penelitian ini merupakan aplikasi *mobile* untuk sistem operasi Android. Aplikasi android ini dibuat dengan menggunakan *Eclipse Integrated Development Environment* dan *Android SDK*. Bahasa pemrograman yang digunakan adalah Java dengan versi *Java Development Kit (JDK) 7* untuk perangkat komputer 64-bit. Aplikasi android ini sendiri dibuat agar

dapat digunakan oleh perangkat android mulai dari sistem operasi Gingerbread (2.3, API 9) hingga Jelly Bean (4.2.2, API 17). Seluruh penyimpanan data dilakukan pada internal memori perangkat Android, baik itu gambar tanda tangan maupun data *username* dan *password*. Perbedaannya, gambar tanda tangan di simpan dalam sebuah *folder* khusus aplikasi yang di *generate* secara otomatis oleh sistem operasi Android dengan *path* `/data/data/com.thesis.securenotepad/app_Sign Storage`, sedangkan untuk data seperti *username* dan *password* disimpan di sebuah *Shared Preference*. Aplikasi android ini akan memiliki nama Secure Notepad.

3.3 Perancangan Sistem

Sistem yang dikembangkan dapat dijelaskan melalui diagram sistem, fungsionalitas, serta masukan dan keluaran yang dibutuhkan yaitu sebagai berikut.

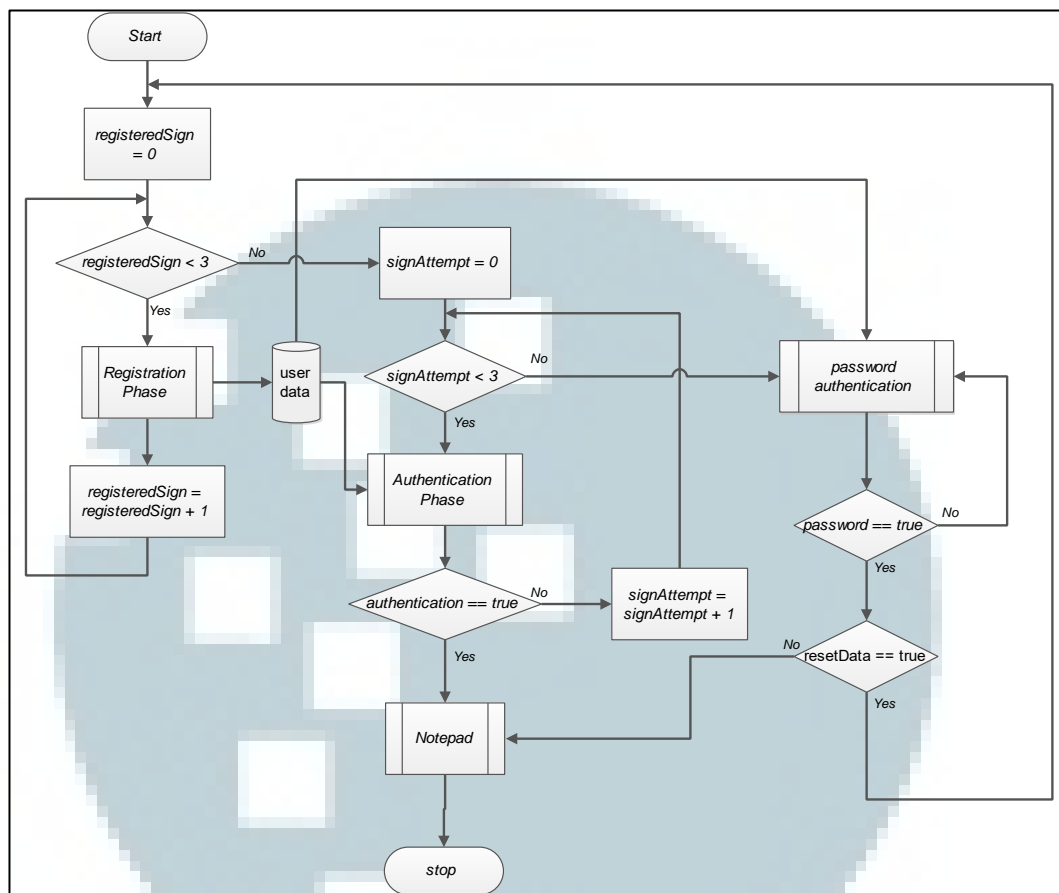
3.3.1 Diagram Sistem

Alur kerja sistem akan dijabarkan dengan menggunakan diagram. Proses kerja sistem secara umum terbagi ke dalam beberapa fase. Fase pendaftaran, verifikasi, dan pengolahan catatan. Fase pendaftaran terbagi menjadi beberapa bagian kecil yakni pengambilan tanda tangan dan penyimpanan kode sandi. Fase otentikasi terdiri atas otentikasi menggunakan tanda tangan dan verifikasi menggunakan kode sandi. Otentikasi menggunakan kode sandi merupakan fase yang hanya dilakukan apabila otentikasi menggunakan tanda tangan gagal sebanyak tiga kali. Pada fase pengambilan tanda tangan, baik itu bagian registrasi maupun otentikasi terdiri atas beberapa tahap antara lain, fase normalisasi, fase

pemrosesan dengan menggunakan wavelet, dan fase pemrosesan dengan menggunakan fourier descriptor. Hal yang membedakan adalah pada fase otentikasi terdapat fase penyesuaian rotasi yang dilakukan oleh *user* agar mendekati kemiringan tanda tangan pada fase otentikasi dan fase *distance measure* yang dilakukan setelah fase pemrosesan dengan menggunakan fourier descriptor. Fase penyesuaian ini termasuk ke dalam bagian *subroutine normalization*.

Data tanda tangan yang disimpan pada fase registrasi akan dibandingkan dengan data tanda tangan yang diambil dari fase otentikasi. Data tanda tangan pada fase otentikasi akan disimpan sebagai gambar pada memori internal perangkat yang bersangkutan. Penjelasan di atas dijabarkan lebih lanjut pada gambar 3.1.

UMMN



Gambar 3.1 Diagram alir aplikasi

3.3.2 Fungsionalitas Sistem

Fungsionalitas yang dimiliki oleh sistem adalah sebagai berikut.

1. Melakukan *cropping* dan *scaling* sebagai bagian dari proses normalisasi pada setiap gambar tanda tangan baik itu yang didapat pada fase registrasi ataupun otentikasi. Pada fase otentikasi terdapat proses dimana *user* melakukan penyesuaian rotasi agar tanda tangan pada fase otentikasi mendekati kemiringan tanda tangan pada fase registrasi.

2. Mengubah tanda tangan tersebut dari domain spasial menjadi domain frekuensi dengan menggunakan Discrete Wavelet Transform dengan jenis Symlet 8.
3. Melakukan pemrosesan Multiscale Fourier Descriptor terhadap tanda tangan yang telah melalui proses Discrete Wavelet Transform .
4. Membandingkan hasil pemrosesan Multiscale Fourier Descriptor pada setiap gambar tanda tangan dengan menggunakan Distance Measure. Metode Distance Measure yang digunakan adalah Manhattan Distance atau yang biasa disebut Taxicab Geometry.

3.3.3 Masukan dan Keluaran Sistem

Masukan yang diperlukan oleh sistem adalah sebagai berikut.

1. Untuk fase registrasi adalah sebagai berikut.
 - a. Tanda tangan yang dilakukan pada layar.
 - b. *Username* dan kode sandi yang akan digunakan apabila otentikasi dengan menggunakan tanda tangan gagal sebanyak tiga kali.
2. Untuk fase otentikasi adalah sebagai berikut.
 - a. Tanda tangan yang dilakukan pada layar.
 - b. Lokasi gambar tanda tangan yang dilakukan pada fase registrasi.
 - c. Kode sandi yang dimasukkan pada saat registrasi. Masukan ini dibutuhkan ketika otentikasi tanda tangan gagal sebanyak tiga kali.

Keluaran dari sistem adalah sebagai berikut.

1. Pada fase registrasi dihasilkan gambar tanda tangan yang dijadikan pembanding pada fase otentikasi.

2. Pada fase otentikasi dihasilkan jarak antara tanda tangan pada fase registrasi dan fase otentikasi.

3.4 Desain Sistem

Sistem yang dirancang akan diimplementasikan sebagai sebuah aplikasi yang akan dijalankan pada sistem operasi Android. Aplikasi ini akan memiliki fase registrasi dan fase otentikasi.

Penjelasan dari masing-masing fase ini akan digambarkan dalam diagram alir, hirarki menu aplikasi dan rancangan tampilan antarmuka aplikasi.

3.4.1 Desain Modul dan Subroutine

Desain modul dan *subroutine* yang digunakan pada *predefined process* modul-modul tersebut dijelaskan menggunakan diagram alir sebagai berikut.

1. Desain fase registrasi

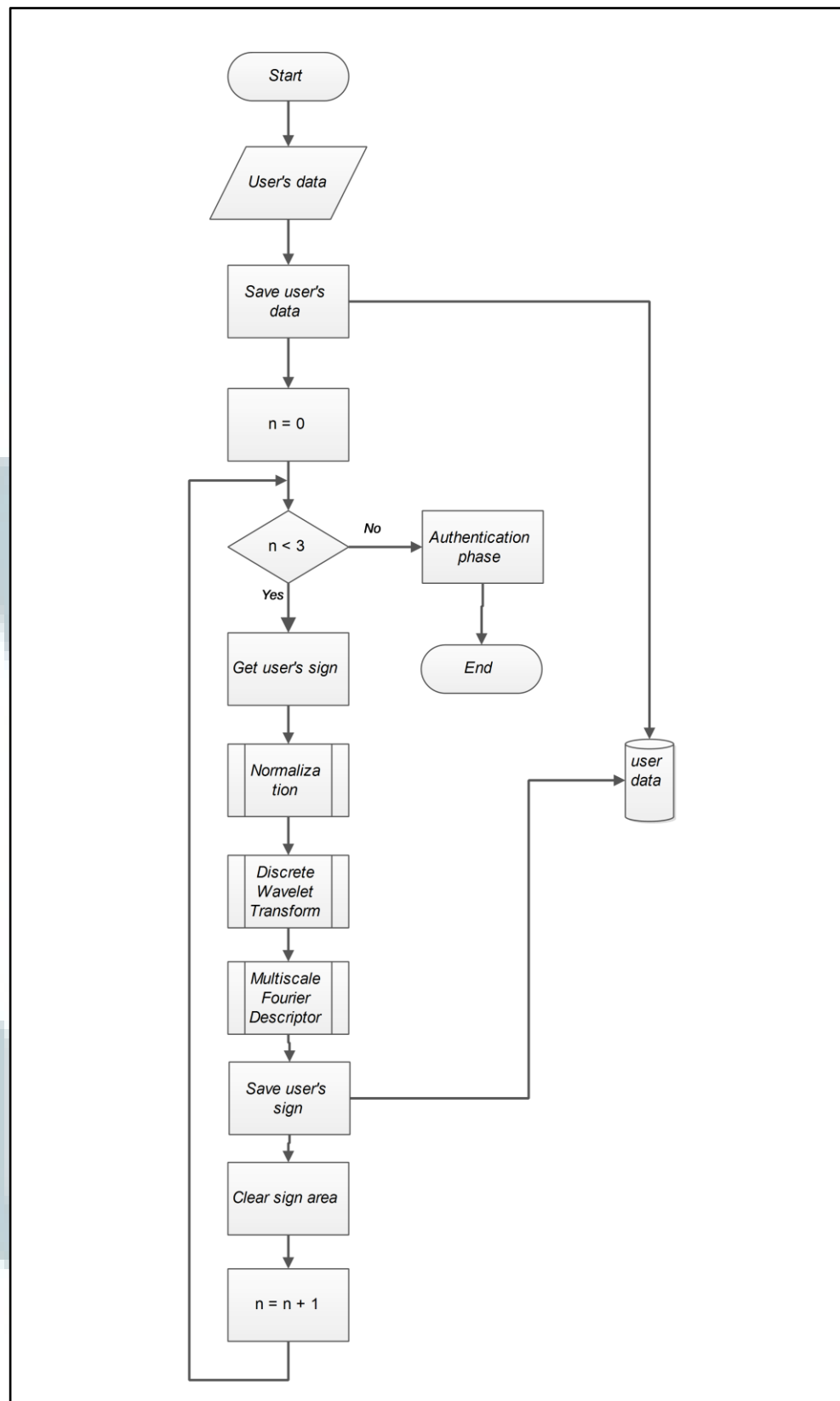
Pada fase registrasi, dilakukan tanda tangan di layar pada area yang disediakan. Setelah menekan tombol simpan, maka *subroutine* normalisasi dipanggil. Pada *subroutine* normalisasi, gambar tanda tangan melalui proses *cropping* dan *scaling*. Proses *cropping* dilakukan untuk memastikan bahwa hanya gambar tanda tangan tanpa terlalu banyak *pixel* yang akan diproses oleh *subroutine* selanjutnya. *Scaling* dilakukan untuk menyamakan resolusi gambar tanda tangan sehingga pemrosesan dapat dilakukan lebih optimal tanpa menghilangkan atau merusak gambar tanda tangan. Proses *scaling* menghasilkan gambar sebesar 1024x1024 *pixels*. Gambar inilah yang akan

diproses oleh *subroutine* selanjutnya yakni *subroutine* Discrete Wavelet Transform.

Pada *subroutine* Discrete Wavelet Transform, gambar hasil *subroutine* normalisasi diubah dari domain spasial menjadi domain frekuensi dengan koefisien wavelet yang ada. Proses ini dilakukan sebanyak tiga iterasi atau dapat juga disebut dengan 3 *level* Discrete Wavelet Transform. Pada setiap iterasi atau *level*, blok frekuensi LH diambil dan digunakan untuk diproses kembali pada iterasi selanjutnya. Setelah tiga kali iterasi, maka blok frekuensi LH kembali diambil untuk pemrosesan pada *subroutine* Multiscale Fourier Descriptor.

Subroutine Multiscale Fourier Descriptor melakukan pemrosesan piksel per piksel dari gambar hasil *subroutine* Discrete Wavelet Transform. Pemrosesan dilakukan mengacu pada rumus 2.11. Setelah proses pada *subroutine* Multiscale Fourier Descriptor selesai, maka gambar hasil *subroutine* ini disimpan di memori internal perangkat dan akan digunakan sebagai gambar pembanding pada fase otentikasi.

Berdasarkan penjelasan di atas, diagram alir fase registrasi dapat didefinisikan dengan gambar 3.2 berikut.



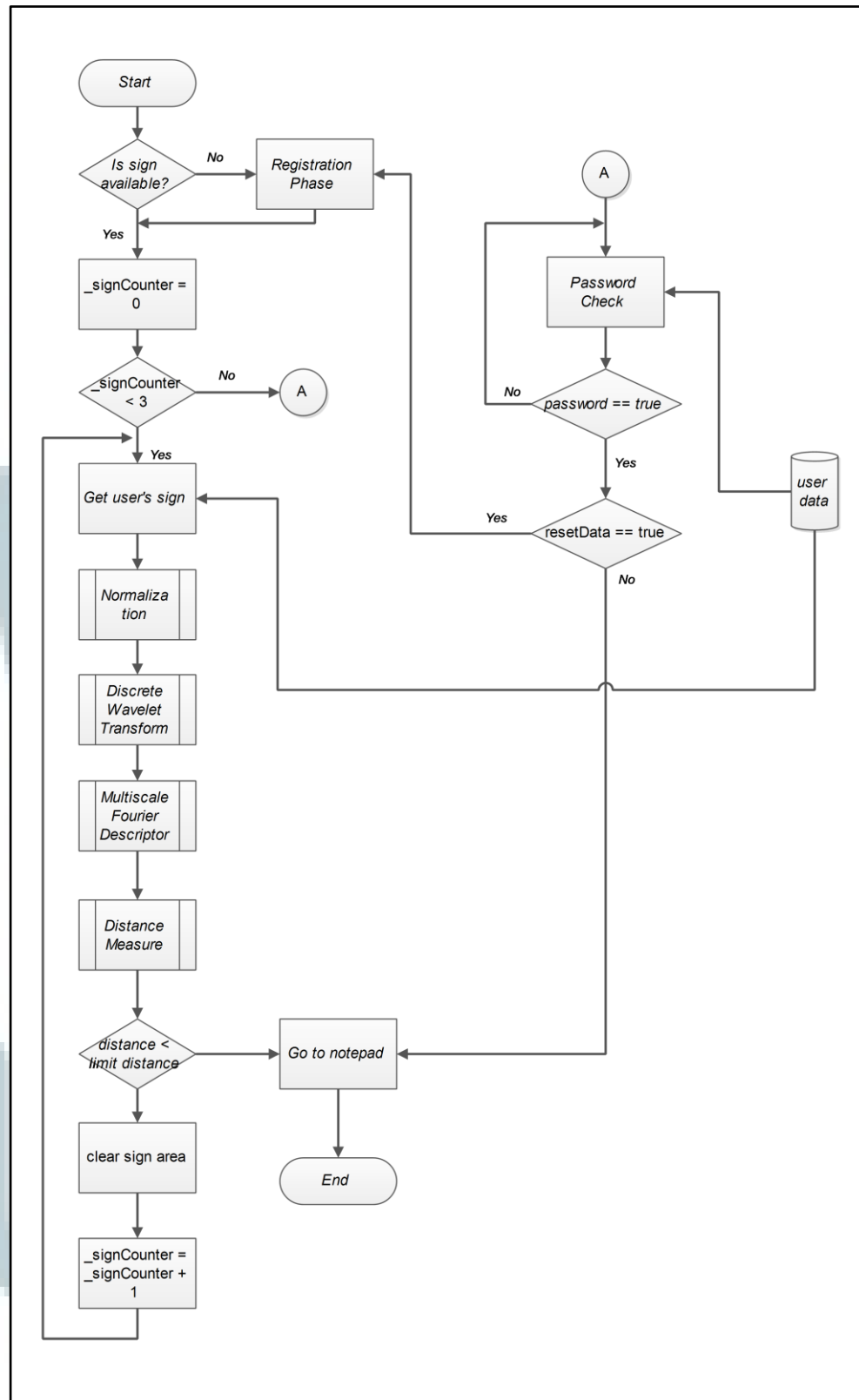
Gambar 3.2 Diagram alir fase registrasi

2. Desain fase otentikasi

Proses pertama dalam fase otentikasi adalah melakukan pengecekan apakah sudah pernah melakukan registrasi sebelumnya. Apabila tanda tangan belum tersedia atau tidak pernah melakukan registrasi sebelumnya, maka akan diarahkan ke proses fase registrasi. Jika tanda tangan sudah tersedia, maka proses pengambilan tanda tangan dilakukan untuk melakukan otentikasi. Kesempatan tanda tangan diberikan sebanyak tiga kali. Tanda tangan dilakukan hanya pada area yang telah disediakan.

Setelah selesai melakukan tanda tangan, maka tombol *sign in* akan ditekan. Tombol ini berfungsi untuk membandingkan tanda tangan yang diambil pada saat fase otentikasi dengan setiap tanda tangan yang diambil pada fase registrasi. Jika tanda tangan sesuai, maka akan diarahkan pada aplikasi penyimpanan catatan. Apabila tanda tangan tersebut tidak ada yang sesuai dengan tanda tangan yang diambil pada fase registrasi sebanyak tiga kali, maka akan ditanyakan mengenai kode sandi yang dimasukkan pada saat registrasi. Setelah kode sandi dimasukkan dan tepat, maka akan muncul pilihan untuk melakukan *reset* semua data registrasi atau menuju ke aplikasi penyimpanan catatan. Jika kode sandi salah, maka akan kembali memasukkan kode sandi hingga berhasil.

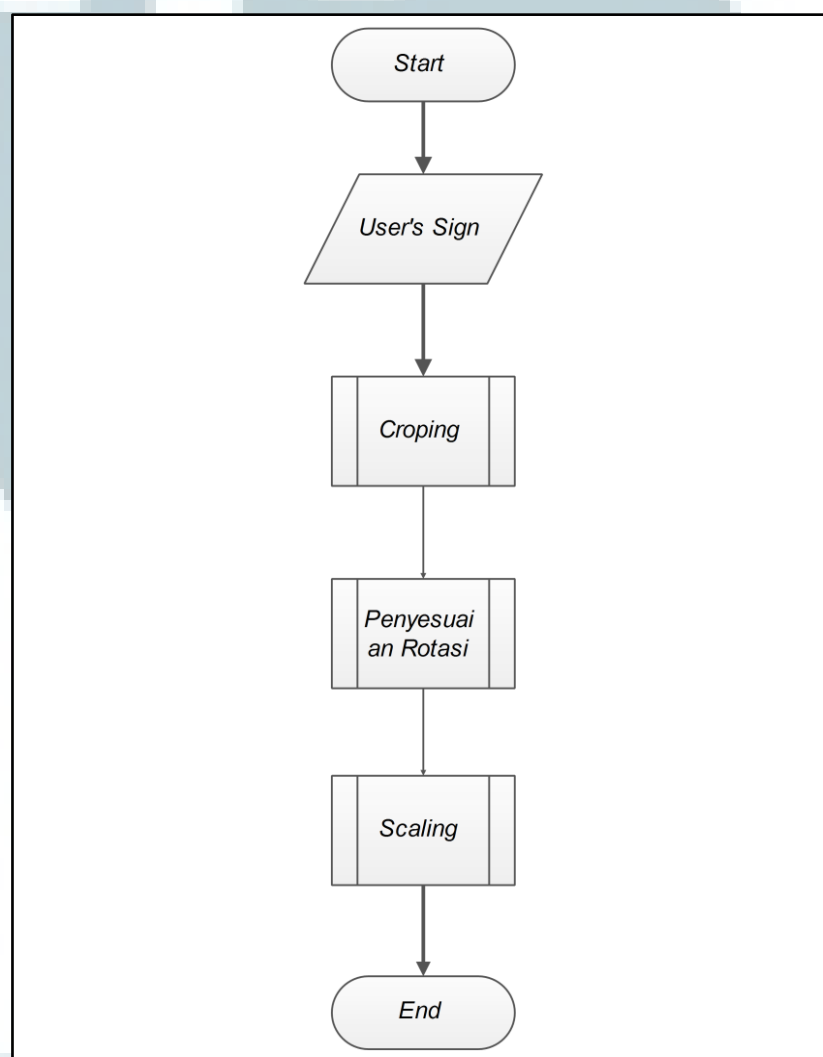
Berdasarkan penjelasan di atas, gambar 3.3 berikut berisi diagram alir yang menjelaskan fase otentikasi.



Gambar 3.3 Diagram alir fase otentikasi

3. Desain *subroutine normalization*

Modul ini berfungsi untuk melakukan normalisasi pada gambar tanda tangan. Normalisasi yang dilakukan disini hanya meliputi *cropping*, penyesuaian rotasi dan *scaling*. Gambar 3.4 berikut adalah diagram alir *normalization* secara umum.



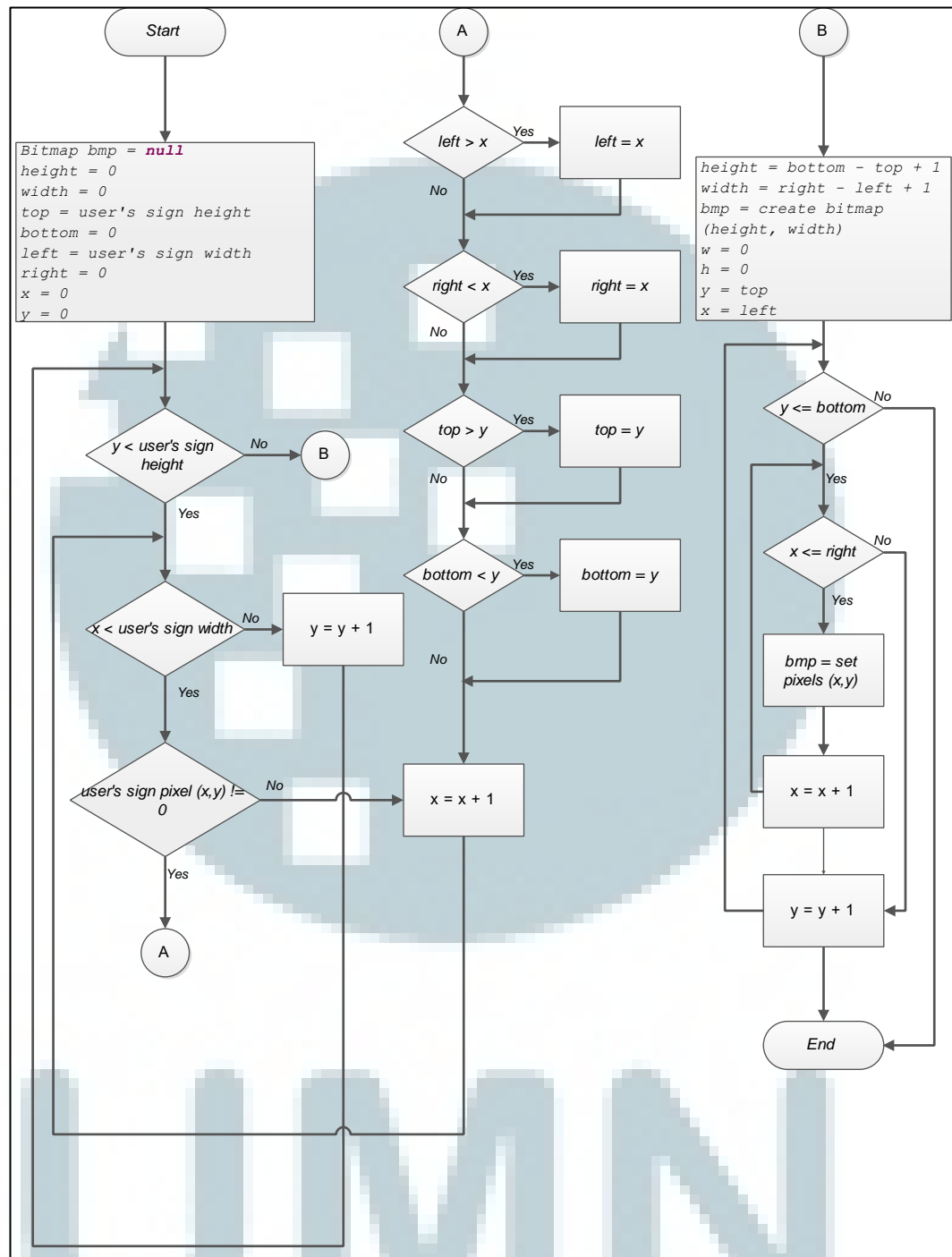
Gambar 3.4 Diagram alir *subroutine normalization*

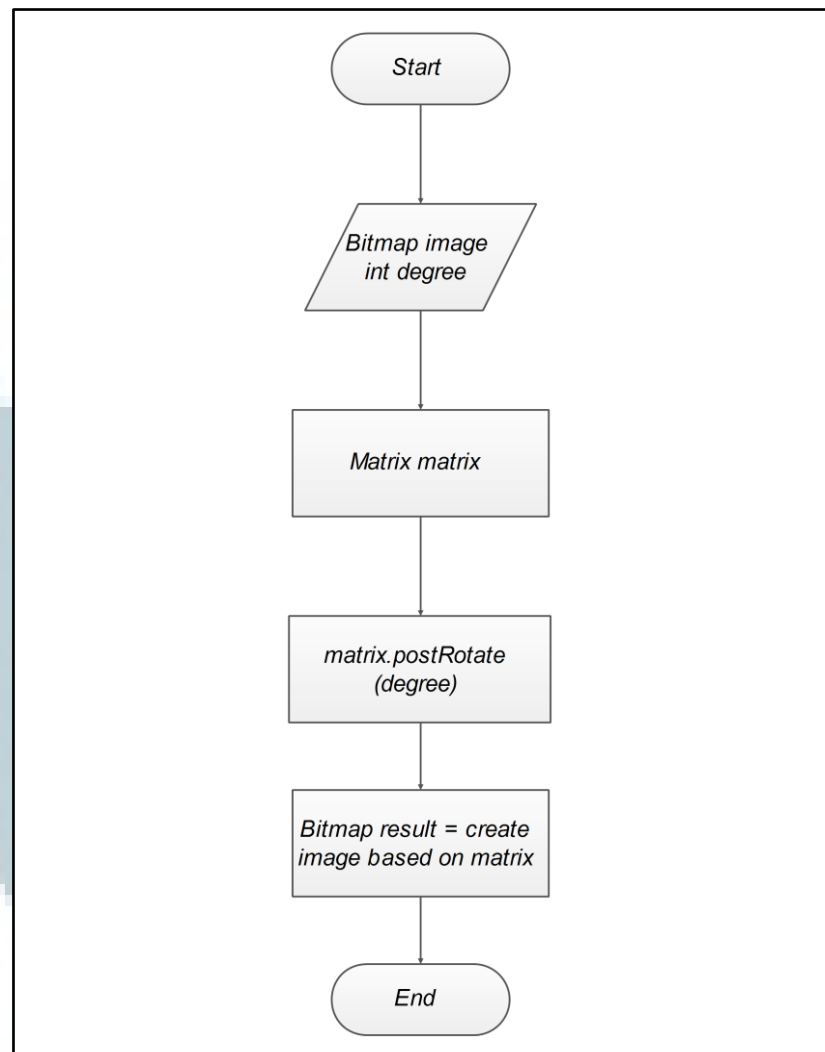
Pada *subroutine normalization* terdapat dua *subroutine* yang dieksekusi yakni *cropping*, penyesuaian rotasi dan *scaling*. *Subroutine*

cropping memastikan bahwa hanya gambar tanda tangan tanpa terlalu banyak *pixel* kosong lah yang akan diambil dan diproses sehingga tidak akan membebani kerja perangkat. Sisa latar pada gambar tanda tangan tidak akan ikut untuk diproses. Diagram alir *subroutine cropping* seperti dijabarkan pada gambar 3.5.

Pada *subroutine* penyesuaian rotasi, dilakukan proses rotasi pada gambar tanda tangan yang telah melalui fase *cropping*. Hal ini dilakukan untuk menyesuaikan kemiringan tanda tangan pada fase otentikasi dan registrasi. Proses ini perlu dilakukan karena menurut Ismail (2010) metode Multiscale Fourier Descriptor dan Wavelet Transform ini *rotation invariant* sehingga perlu ditanggulangi dengan *subroutine* penyesuaian rotasi ini. Proses pada *subroutine* ini dibantu dengan menggunakan *library* Android. Diagram alir *subroutine* penyesuaian rotasi seperti dijabarkan pada gambar 3.6.

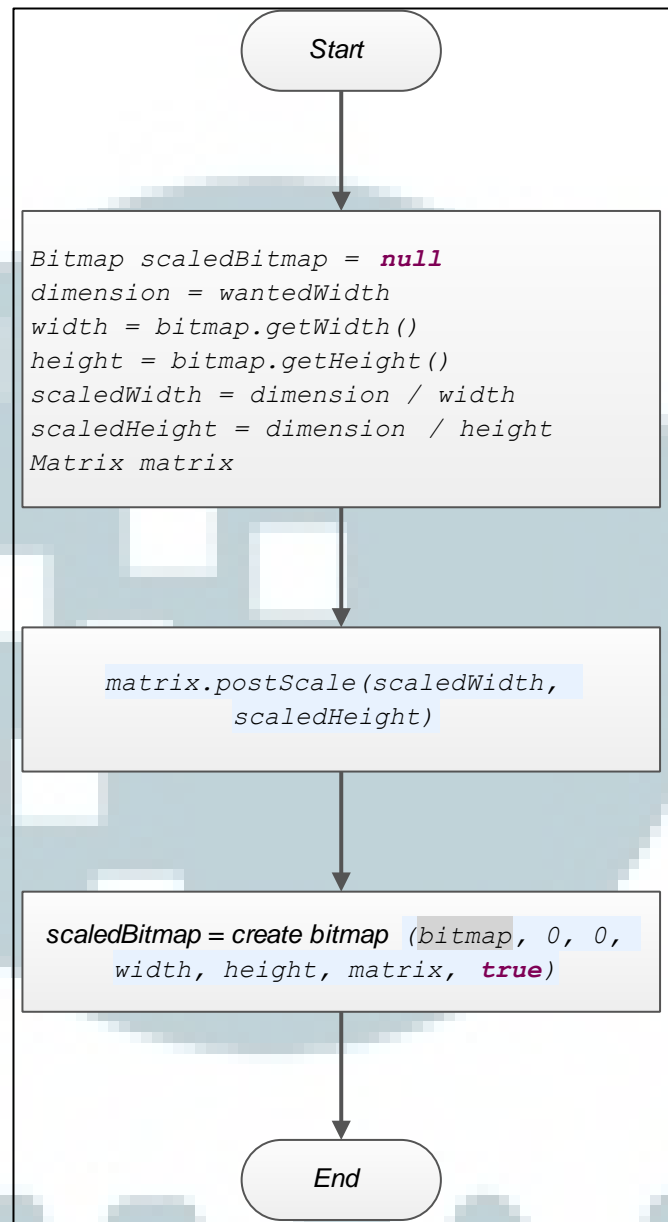
Subroutine scaling digunakan untuk melakukan penyesuaian terhadap ukuran tanda tangan menjadi ukuran yang sama, menjadi 1024x1024 *pixels*, baik gambar tanda tangan pada fase registrasi maupun fase otentikasi. Hal ini akan sangat berguna untuk membantu perhitungan *subroutine distance measure*. Proses dalam *subroutine scaling* menggunakan fungsi yang sudah disediakan oleh *library* Android. Diagram alir *subroutine scaling* dijabarkan dalam gambar 3.7.

Gambar 3.5 Diagram alir *subroutine cropping*



Gambar 3.6 Diagram alir *subroutine* penyesuaian rotasi

U M N

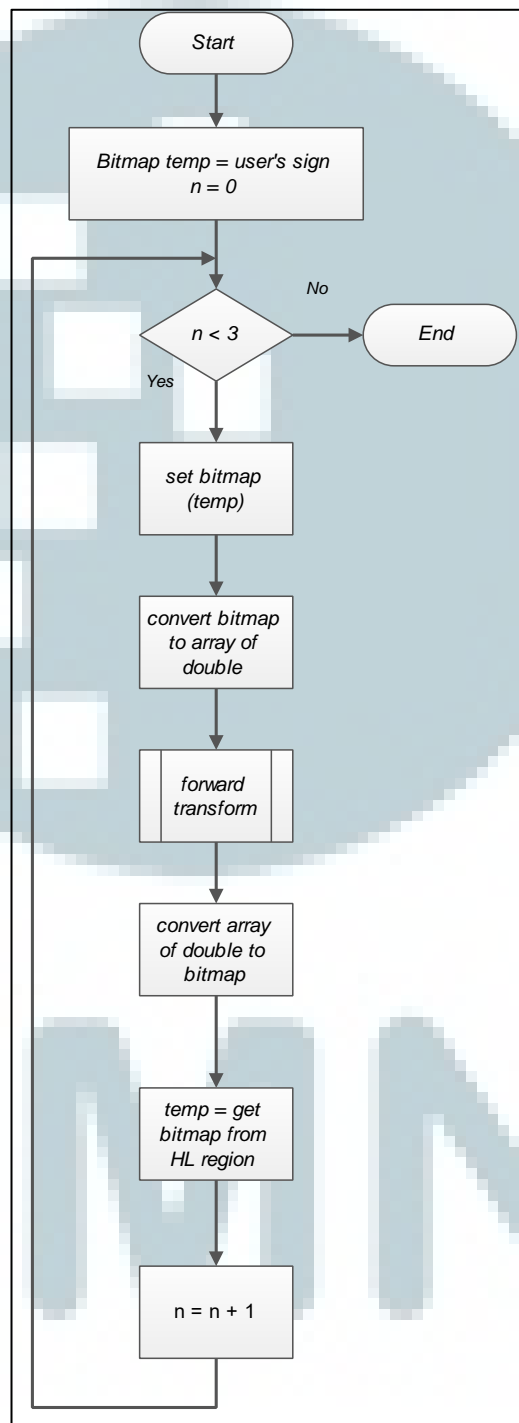


Gambar 3.7 Diagram alir *Subroutine scaling*

4. Desain *subroutine* 3-level Discrete Wavelet Transform

Pada penelitian ini, *subroutine* Discrete Wavelet Transform digunakan untuk mengubah gambar tanda tangan dari domain spasial menjadi domain frekuensi. Dengan mengubah gambar tanda tangan tersebut menjadi domain frekuensi, proses *matching* dapat langsung dilakukan.

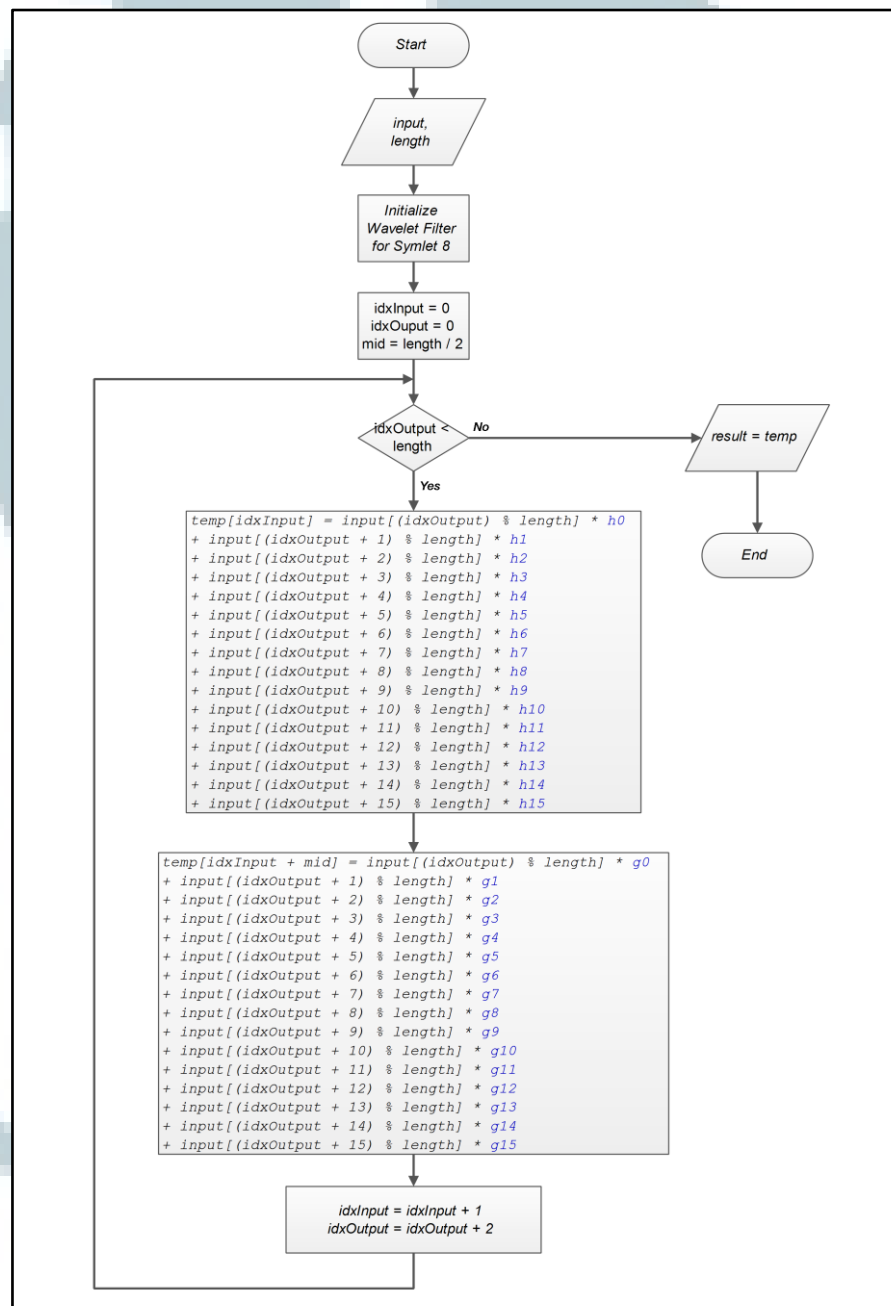
Gambar 3.8 merupakan diagram alir *subroutine* untuk melakukan Discrete Wavelet Transform.



Gambar 3.8 Diagram alir *subroutine* 3-level Discrete Wavelet Transform

5. Desain *subroutine Forward Transform*

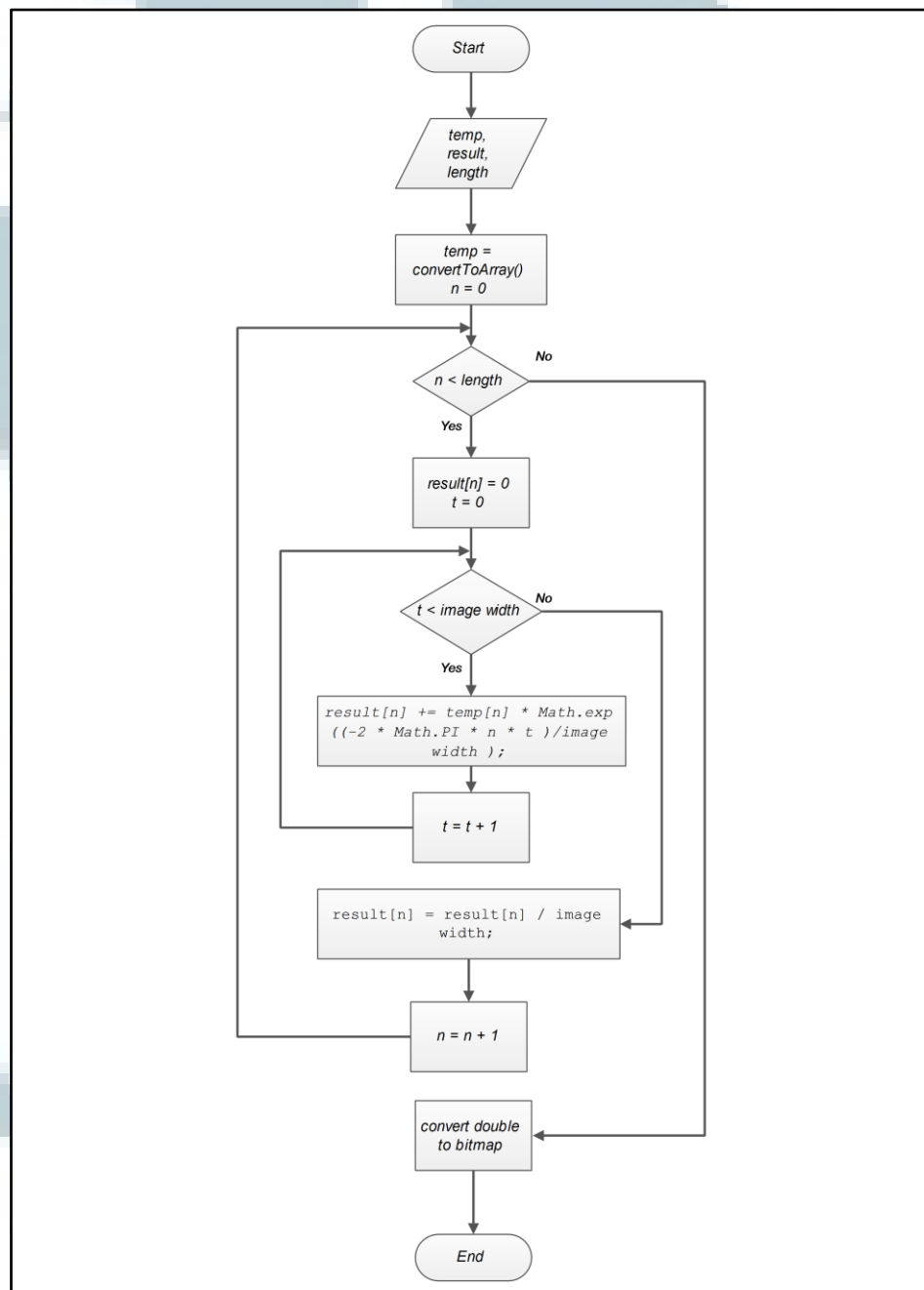
Gambar 3.9 merupakan diagram alir *subroutine* untuk melakukan *Forward Transform* terhadap *pixels* suatu gambar. *Subroutine* ini merupakan bagian dari *subroutine 3-level Discrete Wavelet Transform*.



Gambar 3.9 Diagram alir *subroutine forward transform DWT*

6. Desain *subroutine Multiscale Fourier Descriptor*

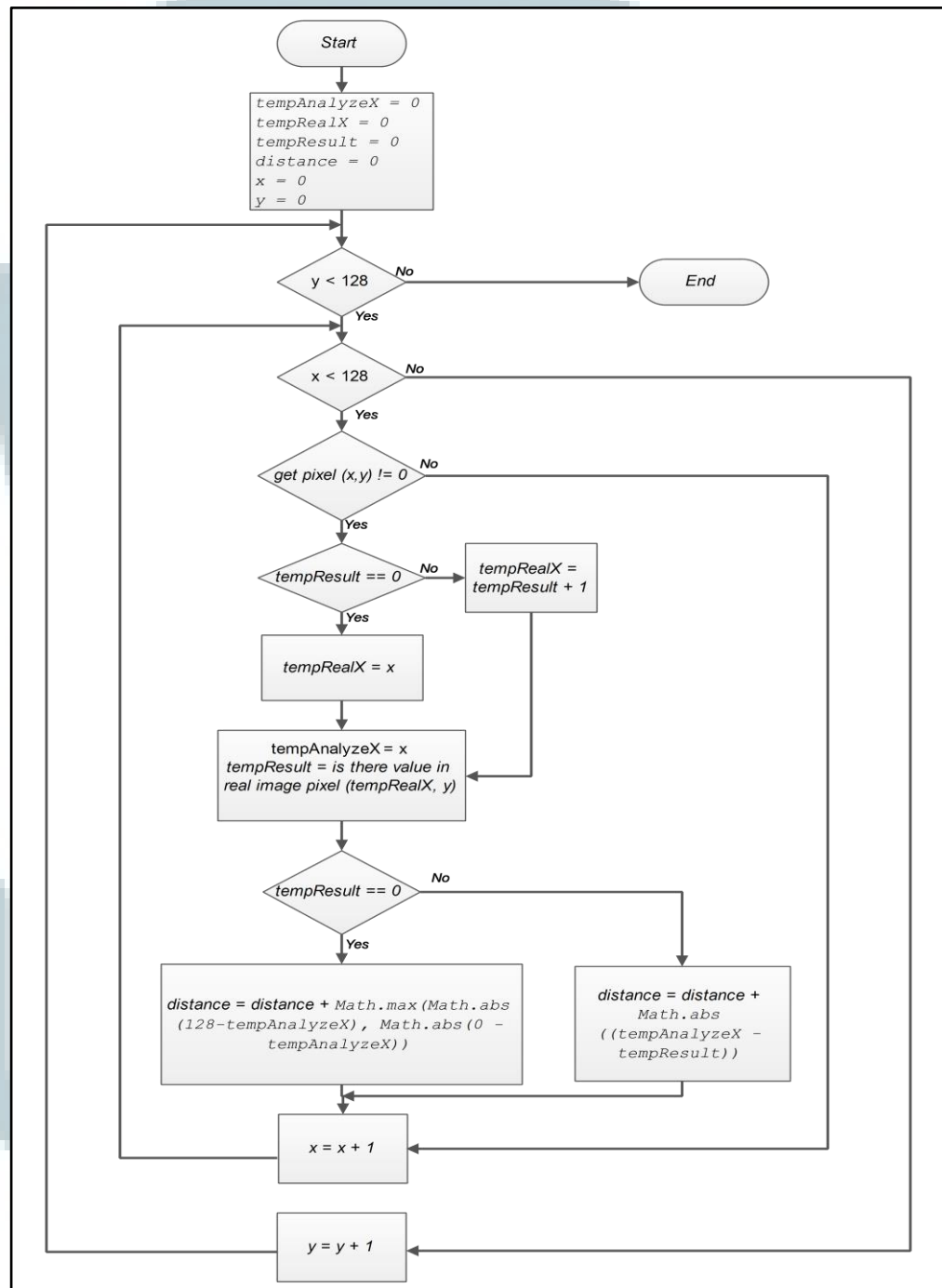
Pada gambar 3.10 merupakan diagram alir *subroutine* untuk melakukan *Multiscale Fourier Descriptor* yang berdasarkan pada rumus 2.11.



Gambar 3.10 Diagram alir *Multiscale Fourier Descriptor*

7. Desain *subroutine Distance Measure*

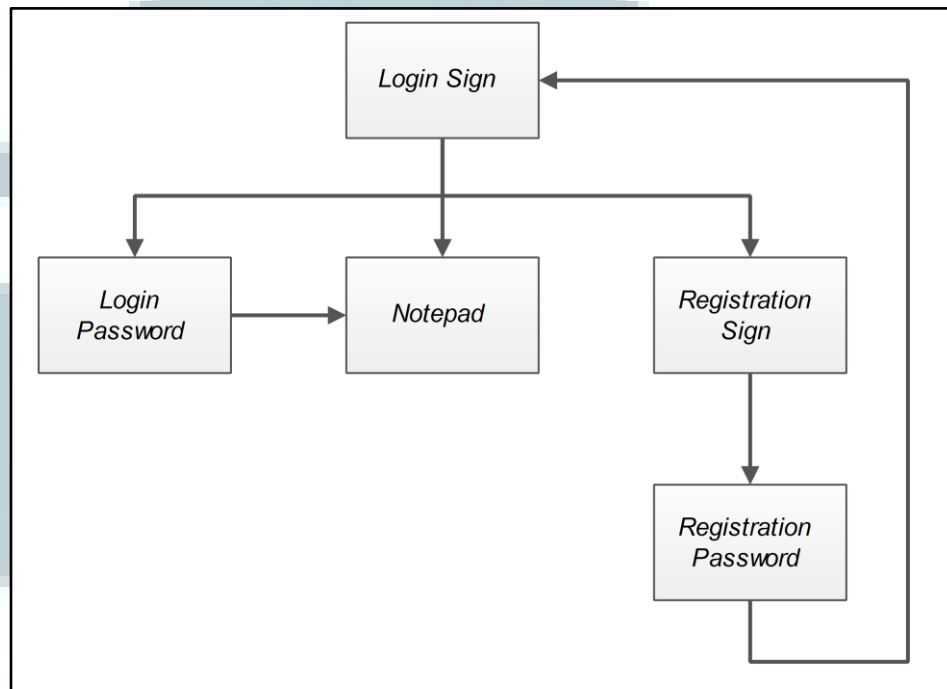
Gambar 3.11 merupakan diagram alir dari *subroutine distance measure* yang mengacu pada Manhattan Distance di rumus 2.21.



Gambar 3.11 Diagram alir *subroutine distance measure*

3.4.2 Hirarki Menu

Hirarki menu dari aplikasi yang dikembangkan dijabarkan dalam diagram pada gambar 3.12.



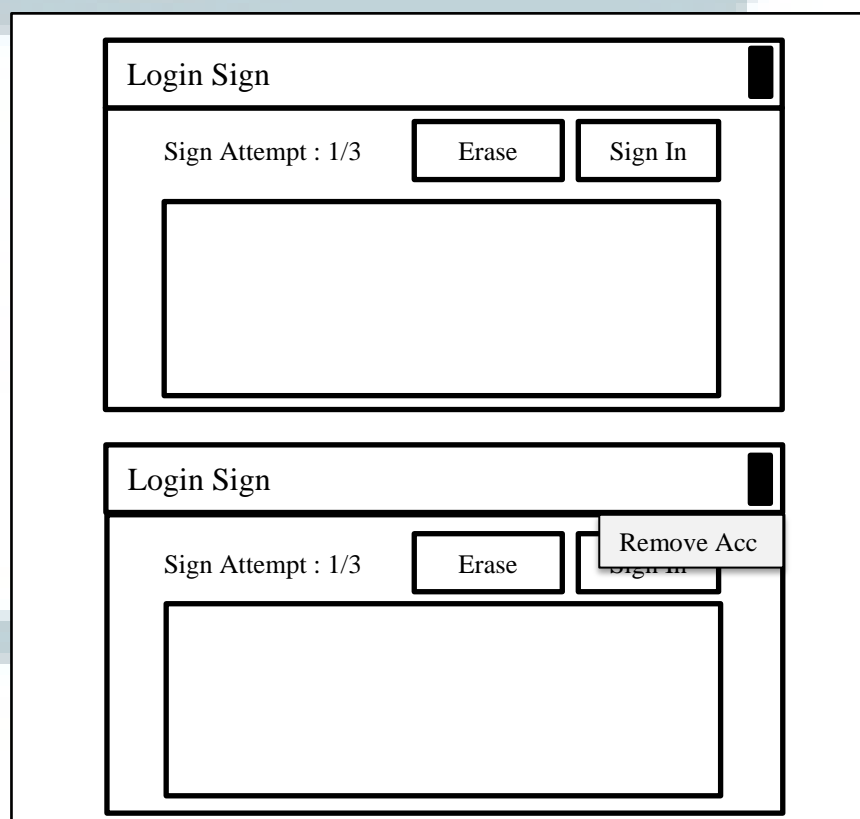
Gambar 3.12 Hirarki menu

Pengembangan aplikasi pada sistem operasi Android menggunakan bahasa pemrograman Java dengan menggunakan XML sebagai antar muka dan Java sebagai logika. Setiap halaman aplikasi yang sedang aktif pada sistem operasi android disebut dengan *activity*. Setiap komponen pada hirarki di atas akan memiliki masing-masing satu *activity*.

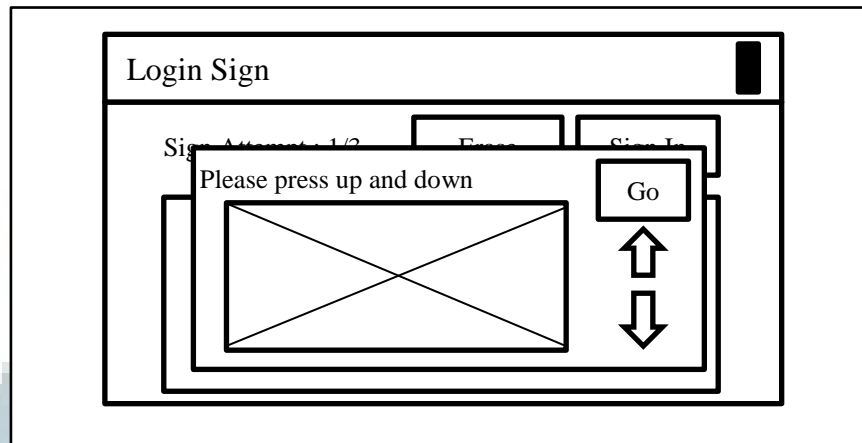
3.4.3 Tampilan Antarmuka

1. Login Sign

Activity login sign adalah *activity* yang pertama kali dijalankan ketika menjalankan aplikasi. Apabila belum melakukan registrasi sebelumnya, maka aplikasi akan melakukan *redirect* menuju *activity registration sign*. Orientasi layar pada *activity* ini adalah *landscape* dengan tujuan untuk memberikan area tanda tangan yang lebih besar. *Activity* ini terdiri dari area untuk tanda tangan, tombol untuk menghapus area tanda tangan, tombol untuk masuk ke *activity registration sign*, dan tombol *sign in* untuk melakukan otentikasi. Gambar 3.13 dan 3.14 berikut ini merupakan rancangan antarmuka untuk *activity login sign*.



Gambar 3.13 Rancangan antarmuka *activity login sign* 1



Gambar 3.14 Rancangan antarmuka *activity login sign 2*

2. *Login Password*

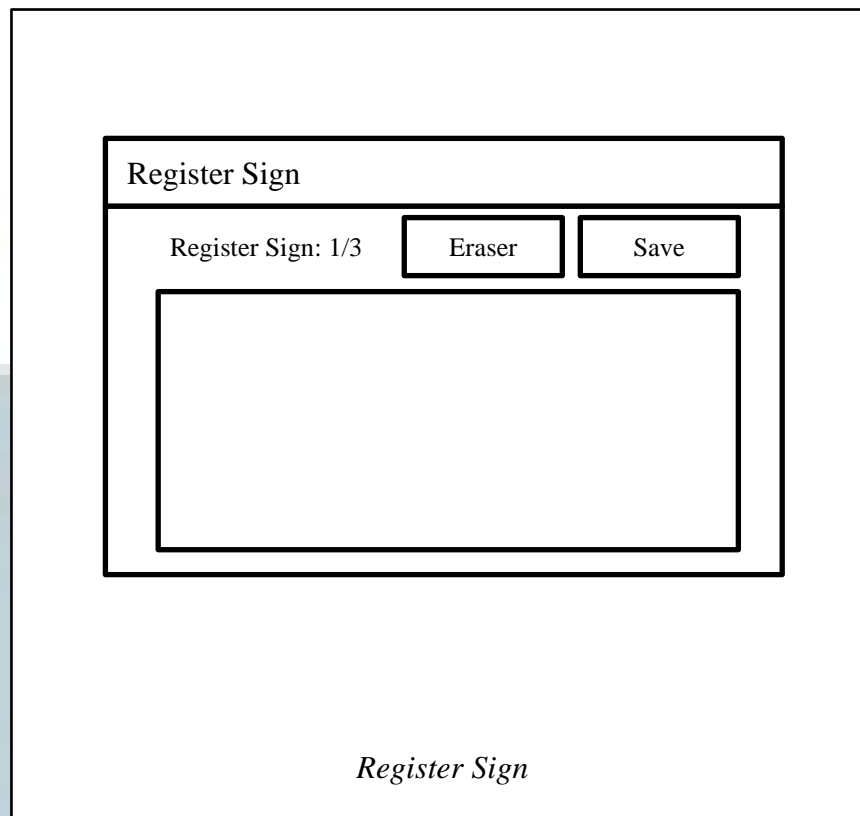
Activity login password ini hanya akan terpanggil jika melakukan *login* dengan menggunakan tanda tangan, dan tanda tangan tersebut dinilai tidak valid sebanyak tiga kali. *Activity* ini dibuat agar pengguna masih dapat membuka catatan yang dibuat meskipun tanda tangan tidak dapat terdeteksi dengan baik. *Activity login password* akan menyimpan data *username* dan *password* ke dalam *shared preference* di dalam memori internal perangkat Android. Gambar 3.15 berikut merupakan rancangan antarmuka untuk *activity login password*.

Login Password

Gambar 3.15 Rancangan antarmuka *activity Login Password*

3. *Register Sign*

Activity register sign merupakan *activity* dimana tanda tangan didaftarkan sebagai alat bantu otentikasi. Tanda tangan akan diambil sebanyak tiga kali dan disimpan pada memori internal perangkat Android. Gambar 3.16 berikut ini merupakan rancangan antarmuka untuk *activity register sign*.



Gambar 3.16 Rancangan antarmuka *activity register sign*

4. *Register Password*

Activity register password merupakan *activity* dimana *username* dan *password* diambil dan akan disimpan ke dalam memori internal dari sebuah perangkat android. Data ini akan tersimpan di dalam sebuah *shared preference*. Gambar 3.17 berikut ini merupakan rancangan antarmuka *activity register password*.

Register Password

username

password

Verify
password

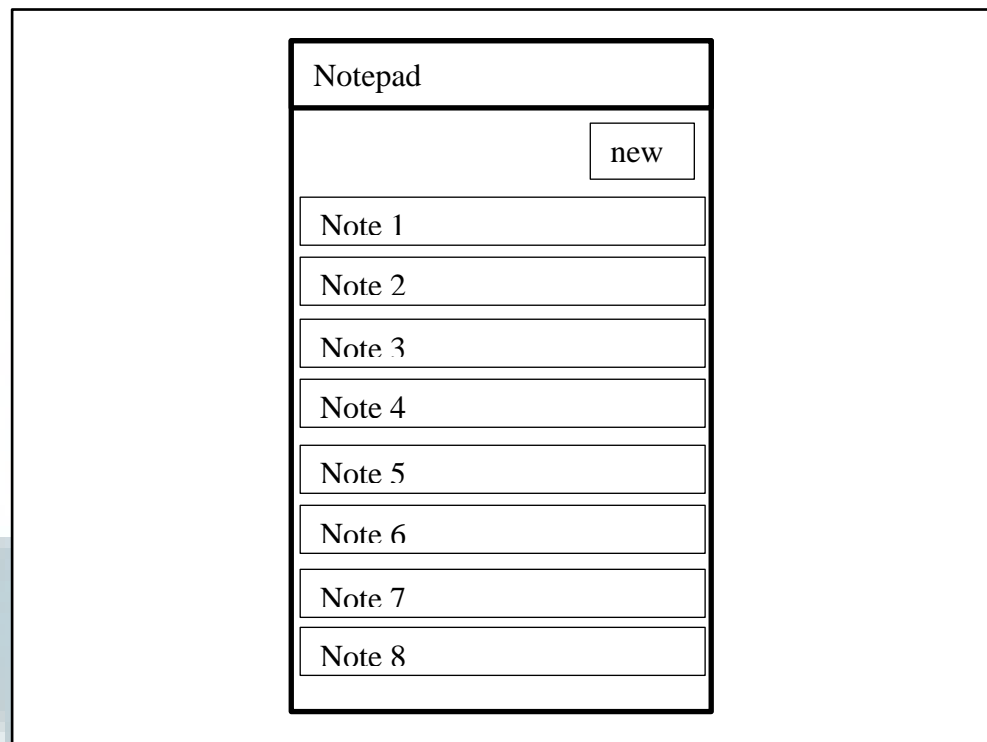
Sign Up

Register Password

Gambar 3.17 Rancangan antarmuka *activity register password*

5. *Notepad*

Activity notepad ini adalah *activity* yang akan muncul apabila tanda tangan pada *activity login sign* atau *password* yang dimasukkan pada *activity login password* tepat. *Activity notepad* ini dibuat berdasarkan contoh yang diberikan Google dalam SDK. Gambar 3.18 berikut ini merupakan rancangan antarmuka untuk *notepad*.



Gambar 3.18 Rancangan antarmuka *activity notepad*