

**NASKAH PUBLIKASI**

**PENGENALAN CITRA SIDIK JARI MENGGUNAKAN METODE  
JARINGAN SARAF TIRUAN BACKPROPAGATION BERBASIS  
ANDROID**



Disusun oleh:

**MUH. JAMAALUDDIN IRSYAAD**

**5150411368**

**PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI DAN ELEKTRO  
UNIVERSITAS TEKNOLOGI YOGYAKARTA  
2020**

NASKAH PUBLIKASI

**PENGENALAN CITRA SIDIK JARI MENGGUNAKAN METODE  
JARINGAN SARAF TIRUAN BACKPROPAGATION BERBASIS  
ANDROID**

Disusun oleh:

**MUH. JAMAALUDDIN IRSYAAD**

**51504111368**



Telah disetujui oleh pembimbing

Pembimbing



M.S. Hendriyawan Ahmad, S.T., M.Eng.

Tanggal: 2 / 3 / 2020

# PENGENALAN CITRA SIDIK JARI MENGGUNAKAN METODE JARINGAN SARAF TIRUAN BACKPROPAGATION BERBASIS ANDROID

**MUH. JAMAALUDDIN IRSYAAD, M.S. HENDRIYAWAN AHMAD, S.T., M.Eng.**

*Program Studi Informatika, Fakultas Teknologi Informasi dan Elektro*

*Universitas Teknologi Yogyakarta*

*Jl. Ringroad Utara Jombor Sleman Yogyakarta*

*E-mail : [mjirsyaad14@gmail.com](mailto:mjirsyaad14@gmail.com) / [jamaaluddinirsyaad@gmail.com](mailto:jamaaluddinirsyaad@gmail.com)*

## ABSTRAK

*Identifikasi sidik jari manusia merupakan proses untuk mengenali dan menentukan sidik jari dari seseorang. Teknologi identifikasi sidik jari termasuk di dalam biometrik yang menggunakan karakteristik perilaku alami manusia. Dalam identifikasi sidik jari sangat tidak dimungkinkan menggunakan cara konvensional karena bentuknya yang rumit dan sulit untuk dibedakan dengan kasat mata. Oleh sebab itu perlu dilakukan penelitian untuk membuat sebuah sistem pengenalan yang terkomputerisasi. Pada penelitian ini digunakan metode jaringan saraf tiruan perambatan balik (backpropagation) untuk mengidentifikasi citra sidik jari. Langkah-langkah pengembangan aplikasinya meliputi: grayscalling, tresholding, morfologi opening, median filtering, resizing dan pelatihan dan pengujian. Proses pelatihan yang dilakukan adalah menentukan bobot awal sistem, menentukan target keluaran sebagai patokan pelatihan menyesuaikan keluaran, melakukan perbaikan bobot sampai kesalahan yang terhitung lebih kecil daripada kesalahan toleransi. Perbaikan bobot dilakukan dengan melakukan umpan balik sinyal keluaran ke lapis tersembunyi dan lapis masukan. Bobot terakhir yang diperoleh disimpan pada basisdata, yang kemudian akan digunakan pada proses pengujian. Selain itu, pengujian dilakukan untuk mengetahui tingkat keberhasilan pengenalan data yang dimasukkan setelah melewati tahap pelatihan. Perangkat lunak yang digunakan dalam membangun aplikasi ini adalah Java & XML sebagai bahasa pemrograman, SQLite sebagai database, Android Studio sebagai penunjang. Pengenalan Citra Sidik Jari Menggunakan Metode Jaringan Saraf Tiruan Galat Mundur ini merupakan aplikasi yang berfungsi untuk memudahkan dalam identifikasi sidik jari seseorang melalui sistem yang terkomputerisasi dengan hasil akurasi data pelatihan sebesar 90,42% dan akurasi data pengujian sebesar 71,67%.*

**Kata kunci :** Identifikasi Citra, Backpropagation, Sidik Jari.

## 1. PENDAHULUAN

Sejalan dengan berkembangnya zaman saat ini, segala sesuatunya dimudahkan dengan pemanfaatan teknologi baik itu komputer maupun smartphone yang memudahkan suatu pekerjaan dapat dilakukan dengan mudah dan cepat. Data yang dihasilkan komputer akan lebih mudan diperbaiki, keamanannya pun lebih terjamin dan tidak mudah hilang. Salah satu pemanfaatan teknologi yaitu pada bidang pendidikan, kesehatan maupun keamanan, bidang yang dapat diterapkan teknologi saat ini yaitu pada bidang biometrik, bidang yang membahas tentang metode otomatis untuk mengenali manusia berdasarkan satu atau lebih bagian tubuh manusia atau perilaku dari manusia itu sendiri, contoh dari bidang ini adalah sidik jari yang digunakan sebagai identifikasi seseorang.

Pengertian dari sidik jari sendiri adalah sidik jari merupakan salah satu identitas manusia yang tidak dapat diganti atau dirubah. Selain itu juga dari sidik jari pula lah seseorang dapat dikenali. "Tidak ada manusia di dunia ini yang mempunyai sidik jari yang sama". Ungkapan ini mengungkapkan bahwa setiap manusia mempunyai sidik jari yang berbeda-beda. Sidik jari menjadi kekhasan setiap manusia, dalam proses dalam proses identifikasi itu sendiri sangat tidak dimungkinkan menggunakan cara konvensional karena bentuk citra sidik jari itu sendiri sangat sulit dicermati dan untuk membedakan satu citra sidik jari dengan citra sidik jari lainnya sangat tidak mungkin hanya menggunakan indra penglihatan, hal ini menyebabkan sidik jari sangat memungkinkan sekali untuk dapat diduplikasi karena bentuknya yang susah untuk diidentifikasi secara kasat mata, hal ini menyebabkan kerawanan pemalsuan.

Dari permasalahan yang terjadi penulis mencoba membuat suatu sistem aplikasi pengenalan citra sidik jari menggunakan kecerdasan buatan dengan tujuan memudahkan seseorang dalam mengenali seseorang secara cepat dan akurat melalui citra sidik jari menggunakan piranti *smartphone*. Untuk itulah penulis membuat laporan proyek tugas akhir ini dengan judul Pengenalan Citra Sidik Jari menggunakan Metode Jaringan Saraf Tiruan Galat Mundur.

## 2. LANDASAN TEORI

### 2.1. Sidik Jari

Menurut G. Sach dalam Soemirat dan Elvinaro Ardianto (2007:171) citra adalah pengetahuan mengenai kita dan sikap-sikap terhadap kita yang mempunyai kelompok-kelompok yang berbeda.

Pengertian citra ini kemudian disitir oleh Effendi dalam Soemirat dan Elvinaro Ardianto (2007:171) bahwa citra adalah dunia sekeliling kita yang memandang kita.

Sedangkan menurut Katz dalam Soemirat dan Elvinaro Ardianto (2007: 113), citra adalah cara bagaimana pihak lain memandang sebuah perusahaan, seseorang, suatu komite, atau suatu aktivitas.

Menurut Frank Jefkins dalam Soemirat dan Elvinaro Ardianto (2007:114), citra diartikan sebagai kesan seseorang atau individu tentang sesuatu yang muncul sebagai hasil dari pengetahuan dan pengalamannya.

Menurut Komarinski (2005:3), *Fingerprint* atau sidik jari adalah sebuah biometric yang telah digunakan secara sistematis untuk identifikasi selama 100 tahun yang telah diukur, diduplikasi dan diperiksa secara ekstensif, sebuah biometric yang tidak berubah dan relatif mudah untuk diambil. Pada jari atau ibu jari, *ridge-ridge* membentuk tiga buah pola, yakni *loops*, *whorls* dan *arches*.



**Gambar 1** Pola pada *ridge*. (A) *Loop*, (B) *Arch*, (C) *Whorl*.

Pola-pola pada *ridge* :

a) Gambar 1 (A) adalah gambar pola *Loop*, *Loop* dapat menaik ke arah ujung jari, atau menjatuh ke arah pergelangan tangan. *Common Loop* bergerak

ke arah ibu jari, sementara *Radial Loop* (*Loop* terbalik) bergerak mengarahkan ujung pemukunya ke sisi lengan.

b) Gambar 1 (B) adalah gambar pola *Arch*, *Arch* bisa terlihat sebagai sebuah *Flat Arch*, atau *Tented Arch*. Perhatikan setiap pola *Arch* menaik sangat tinggi. Pola *Arche* menandakan nilai-nilai tradisional dan akhlak yang tinggi. Orang-orang dengan pola ini mengalami kesulitan untuk melihat sifat-sifat negatif mereka sendiri, dan untuk memahami bahwa “masa lampau” yang mereka kunci di atas noda atau perasaan malu hanyalah sebuah pengalaman yang diperlukan untuk perkembangan kepribadian secara penuh. Orang dengan *Flat Arch* mengikuti tradisi dengan sedikit pemikiran mandiri, sedangkan orang dengan pola *Tented Arch* mengungkapkan suatu kedalaman intelektual.

c) Gambar 1 (C) adalah gambar pola *Whorl*, *Whorl* bisa berbentuk sebuah *Spiral*, *Bulls-eye*, atau *Double Loop*. *Whorl* adalah titik-titik menonjol dan kontras, dan bisa dilihat dengan mudah. Cetakan *Spiral* dan *Bulls-eye* adalah persis sebangun dalam interpretasinya, namun yang kedua memberikan sedikit lebih banyak fokus. Di mana pun di bagian tangan, *Whorl* menyoroti dan menekankan kepada daerah tertentu, menjadikannya sebuah wilayah fokus di dalam kehidupan subyek.

### 2.2. Pengolahan Citra Digital

Menurut T. Sutoyo, Edy Mulyanto, Vincent Suhartono, Oky Dwi Nurhayati Wijanarto (2009), Pengolahan citra digital merupakan disiplin ilmu yang mempelajari hal-hal yang berkaitan dengan perbaikan kualitas gambar untuk tujuan analisis.

Beberapa alasan dilakukan pengolahan citra digital adalah sebagai berikut:

- a) Untuk mendapatkan citra asli dari citra yang sudah rusak karena pengaruh noise yang bercampur dengan citra asli dalam suatu proses tertentu proses pengolahan citra bertujuan untuk mendapatkan citra yang mendekati citra asli.
- b) Untuk mendapatkan citra dengan karakteristik tertentu dan cocok secara visual yang dibutuhkan dalam proses lanjut dalam proses lanjut dalam pemrosesan analisis citra operasi pengolahan citra dapat diklasifikasikan dalam beberapa jenis sebagai berikut:
  - *Image Enhancemet* (Perbaikan kualitas citra)
  - *Image Restoration* (Pemugaran Citra)
  - *Image Compression* (Pemampatan Citra)
  - *Image Segmentation* (Segmentasi Citra)
  - *Image Analysis* (Analisa Citra)
  - *Image Recontruction* (Rekonstruksi Citra)

Operasi-Operasi tersebut bertujuan untuk membentuk objek dari beberapa citra hasil proyeksi. Pada citra digital, dengan tipe bitmap tipe warna pada titik-titik pixel dibentuk dari sebuah data numerik. Tinggi dan rendahnya keabuan pixel dinyatakan dalam bentuk intensitas atau derajat keabuan. Satuan lebar intensitas merupakan lebar memori (bit) citra yang disebut dengan format pixel.

### 2.3. Jaringan Saraf Tiruan

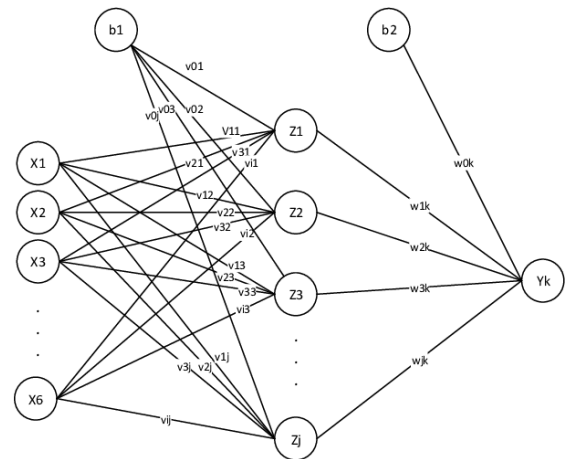
Menurut Hermawan (2006), Jaringan saraf tiruan didefinisikan sebagai suatu sistem pemrosesan informasi yang mempunyai karakteristik menyerupai jaringan saraf manusia. Jaringan saraf tiruan juga dikenal sebagai kotak hitam (*black box technology*) atau tidak transparan (*opaque*) karena tidak dapat menerangkan bagaimana suatu hasil didapatkan. Hal inilah yang membuat jaringan saraf tiruan mampu digunakan untuk menyelesaikan persoalan yang tidak terstruktur dan sulit didefinisikan. Kenyataan inilah yang menyebabkan jaringan saraf tiruan telah meluas dipakai sebagai alat bantu memecahkan masalah pada berbagai bidang dan disiplin ilmu.

Menurut Hermawan (2006), Jaringan perambatan galat mundur (*backpropagation*) memiliki algoritma pelatihan dengan menggunakan metode belajar terbimbing pada jaringan diberikan sepasang pola yang terdiri atas pola masukan dan pola yang diinginkan ketika suatu pola diberikan kepada jaringan, bobot-bobot diubah untuk memperkecil perbedaan pola keluaran dan pola yang diinginkan latihan ini dilakukan secara berulang-ulang agar semua pola yang dikeluarkan jaringan dapat memenuhi pola yang diinginkan.

#### a) Arsitektur Backpropagation

Backpropagation memiliki beberapa unit yang ada dalam satu atau lebih layer tersembunyi. Gambar 2 adalah arsitektur backpropagation dengan n buah masukan (ditambah sebuah bias), sebuah layer tersembunyi yang terdiri dari p unit (ditambah sebuah bias), serta m buah unit keluaran.

$V_{ji}$  merupakan bobot garis dari unit masukan  $X_i$  ke unit layar tersembunyi  $Z_j$  ( $V_{j0}$  merupakan bobot garis yang menghubungkan bias di unit masukan ke unit layar tersembunyi  $Z_j$ ).  $W_{kj}$  merupakan bobot dari unit layar tersembunyi  $Z_j$  ke unit keluaran  $Y_k$  ( $W_{k0}$  merupakan bobot dari bias di layar tersembunyi ke unit keluaran  $Z_k$ ).



Gambar 2 Arsitektur Backpropagation

#### b) Fungsi Aktivasi

Dalam backpropagation, fungsi aktivasi yang dipakai harus memenuhi beberapa syarat yaitu : kontinu, terdiferensial dengan mudah dan merupakan fungsi yang tidak turun. Salah satu fungsi yang memenuhi ketiga syarat tersebut sehingga sering dipakai adalah fungsi signoid yang memiliki range (0, 1), dapat dilihat pada persamaan 2.1.

$$f(x) = \frac{1}{1 + e^{-x}} \text{ dengan turunan } f'(x) = f(x)(1 - f(x)) \quad (2.1)$$

#### c) Pelatihan Standar Backpropagation

Pelatihan Backpropagation meliputi 3 fase. Fase pertama adalah fase maju. Pola masukan dihitung maju mulai dari layar masukan hingga layar keluaran menggunakan fungsi aktivasi yang ditentukan. Fase kedua adalah fase mundur. Selisih antara keluaran jaringan dengan target yang diinginkan merupakan kesalahan yang terjadi. Kesalahan tersebut dipropagasikan mundur, dimulai dari garis yang berhubungan langsung dengan unit-unit di layar keluaran. Fase ketiga adalah modifikasi bobot untuk menurunkan kesalahan yang terjadi.

#### d) Algoritma Pelatihan Backpropagation

Algoritma pelatihan untuk jaringan dengan satu layar tersembunyi (dengan fungsi aktivasi signoid biner) adalah sebagai berikut :

Langkah 0 : Inisialisasi semua bobot dengan bilangan acak kecil antara (-0,5 – 0,5)

Langkah 1 : Jika kondisi penghentian belum terpenuhi, lakukan langkah 2 – 9

Langkah 2 : Untuk setiap pasang data pelatihan, lakukan langkah 3 – 8

Fase I : Propagasi maju

Langkah 3 : Tiap unit masukan menerima sinyal dan meneruskannya ke unit tersembunyi di atasnya

Langkah 4 : Hitung semua keluaran di unit tersembunyi  $z_j$  ( $j = 1, 2, \dots, P$ ), dengan menggunakan persamaan 2.2  $z_{net_j}$  dan persamaan 2.3 untuk mencari  $z_j$ .

$$z_{net_j} = v_{jo} + \sum_{i=1}^n x_i v_{ji} \quad (2.2)$$

$$z_j = f(z_{net_j}) = \frac{1}{1 + e^{-z_{net_j}}} \quad (2.3)$$

Langkah 5 : Hitung semua keluaran jaringan di unit  $y_k$  ( $k = 1, 2, \dots, m$ ), dengan menggunakan persamaan 2.4  $y_{net_k}$  dan persamaan 2.5 untuk mencari  $y_k$ .

$$y_{net_k} = w_{ko} + \sum_{j=1}^p z_j w_{kj} \quad (2.4)$$

$$y_k = f(y_{net_k}) = \frac{1}{1 + e^{-y_{net_k}}} \quad (2.5)$$

Fase II : Propagasi mundur

Langkah 6 : Hitung faktor  $\delta$  unit keluaran berdasarkan kesalahan di setiap unit keluaran  $y_k$  ( $k = 1, 2, \dots, m$ ), dapat dilihat pada persamaan 2.6.

$$\begin{aligned} \delta_k &= (t_k - y_k) f'(y_{net_k}) \\ &= (t_k - y_k) y_k (1 - y_k) \end{aligned} \quad (2.6)$$

$\delta_k$  merupakan unit kesalahan yang akan dipakai dalam perubahan bobot layar di bawahnya (langkah 7)

Dengan menggunakan persamaan 2.7, hitung suku perubahan bobot  $w_{kj}$  (yang akan dipakai nanti untuk merubah bobot  $w_{kj}$ ) dengan laju percepatan  $\alpha$

$$\begin{aligned} \Delta w_{kj} &= \alpha \delta_k z_j \quad ; \quad k \\ &= 1, 2, \dots, m \quad ; \quad j \\ &= 0, 1, \dots, p \end{aligned} \quad (2.7)$$

Langkah 7 : Hitung faktor  $\delta$  unit tersembunyi berdasarkan kesalahan di setiap unit tersembunyi  $z_j$  ( $j = 1, 2, \dots, p$ ) dengan menggunakan persamaan 2.8.

$$\delta_{net_j} = \sum_{k=1}^m \delta_k w_{kj} \quad (2.8)$$

Faktor  $\delta$  unit tersembunyi menggunakan persamaan 2.9 :

$$\begin{aligned} \delta_j &= \delta_{net_j} f'(z_{net_j}) \\ &= \delta_{net_j} z_j (1 - z_j) \end{aligned} \quad (2.9)$$

Hitung suku perubahan bobot  $v_{ji}$  (yang akan dipakai nanti untuk merubah bobot  $v_{ji}$ ) menggunakan persamaan 2.10

$$\begin{aligned} \Delta v_{ji} &= \alpha \delta_j x_i \quad ; \quad j \\ &= 1, 2, \dots, p \quad ; \quad i \\ &= 0, 1, \dots, n \end{aligned} \quad (2.10)$$

Fase III : Perubahan Bobot

Langkah 8 : Hitung semua perubahan bobot

Dengan menggunakan persamaan 2.11 untuk melakukan perubahan bobot garis yang menuju ke unit keluaran :

$$\begin{aligned} w_{kj}(\text{baru}) &= w_{kj}(\text{lama}) + \Delta w_{kj} \\ (k &= 1, 2, \dots, m \quad ; \quad j = 0, 1, \dots, p) \end{aligned} \quad (2.11)$$

Dengan menggunakan persamaan 2.12 untuk melakukan perubahan bobot garis yang menuju ke unit tersembunyi :

$$\begin{aligned} v_{ji}(\text{baru}) &= v_{ji}(\text{lama}) + \Delta v_{ji} \\ (j &= 1, 2, \dots, p \quad ; \quad i = 0, 1, \dots, n) \end{aligned} \quad (2.12)$$

Setelah pelatihan selesai dilakukan, jaringan dapat dipakai untuk pengenalan pola. Dalam hal ini, hanya propagasi maju (langkah 4 dan 5) saja yang dipakai untuk menentukan keluaran jaringan. Apabila fungsi aktivasi yang dipakai bukan signois biner, maka langkah 4 dan 5 harus disesuaikan. Demikian juga turunannya pada langkah 6 dan 7.

e) Momentum

Pada standar backpropagation, perubahan bobot didasarkan atas gradien yang terjadi untuk pola yang dimasukkan saat itu. Modifikasi yang dapat dilakukan adalah melakukan perubahan bobot yang didasarkan atas arah gradien pola terakhir dan pola sebelumnya (disebut momentum) yang dimasukkan. Jadi tidak hanya pola masukan terakhir saja yang diperhitungkan.

Penambahan momentum dimaksudkan untuk menghindari perubahan bobot yang mencolok akibat adanya data yang sangat berbeda dengan yang lain (outlier). Apabila beberapa data terakhir yang diberikan jaringan memiliki pola serupa (berarti arah gradien sudah benar), maka perubahan

bobot dilakukan secara cepat. Namun apabila data terakhir yang dimasukan memiliki pola yang berbeda dengan pola sebelumnya, maka perubahan dilakukan secara lambat.

Dengan penambahan momentum, bobot baru pada waktu ke  $(t + 1)$  didasarkan atas bobot pada waktu  $t$  dan  $(t - 1)$ . Disini harus ditambahkan 2 variabel baru yang mencatat besarnya momentum untuk 2 iterasi terakhir. Jika  $\mu$  adalah konstanta ( $0 \leq \mu \leq 1$ ) yang menyatakan parameter momentum maka bobot baru dihitung berdasarkan persamaan 2.13 :

$$w_{kj}(t + 1) = w_{kj}(t) + \alpha \delta_k z_j + \mu(w_{kj}(t) - w_{kj}(t - 1)) \quad (2.13)$$

Dan persamaan 2.14

$$v_{ji}(t + 1) = v_{ji}(t) + \alpha \delta_j x_i + \mu(v_{ji}(t) - v_{ji}(t - 1)) \quad (2.14)$$

### 3. METODOLOGI PENELITIAN

#### 3.1 Obyek Penelitian

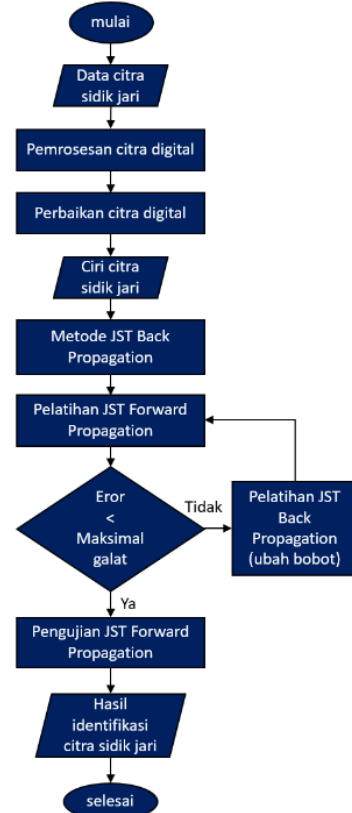
Pada penelitian ini penulis mengambil data citra sidik jari dari pengambilan citra melalui kamera *smartphone* dalam bentuk format .jpg, setelah itu dilakukan proses pengolahan citra untuk mendapatkan ciri dari tiap-tiap citra sidik jari. Dalam melakukan pengolahan citra terdapat beberapa tahap antara lain yaitu (*resizing*) pada tahap ini dilakukan segmentasi citra sidik jari ke dalam blok-blok berukuran  $300 \times 300$  piksel, (*image enhancement*) pada tahap ini menggunakan proses *grayscale* untuk merubah citra RGB menjadi citra berskala keabuan, (*thresholding*) ini adalah proses terakhir dari pengolahan citra dimana proses ini mengkonversi citra berskala keabuan menjadi citra biner yang nantinya akan dijadikan ciri dari citra sidik jari setelah itu dihitung arah dominan dari setiap blok sehingga dihasilkan vektor berdimensi  $1 \times 90.000$  yang merepresentasikan ciri berupa sudut yang dibentuk dalam blok dan satu vektor mewakili satu citra sidik jari, selanjutnya adalah proses perbaikan citra yaitu morfologi citra untuk menggabungkan garis sidik jari yang terputus akibat proses cap sidik jari dan median *filtering* untuk menghaluskan tepi garis sidik jari.

Setelah didapatkan ciri dari citra sidik jari kemudian ciri tersebut dijadikan objek masukan jaringan saraf tiruan *backpropagation* yang digunakan dalam proses pelatihan dan pengujian untuk mendapatkan objek keluaran berupa hasil yang akan menentukan identifikasi seseorang.

### 3.2 Desain dan Pembuatan Program

#### a) Flowchart Program

Algoritma proses kerja dari sistem pengenalan citra sidik jari yang akan dibangun terlihat pada Gambar 3.



Gambar 3 Flowchart Algoritma Program

#### b) Desain

Desain sistem yang dibangun berupa menu input daftar pemilik sidik jari untuk memasukan data pemilik sidik jari, menu input Citra sidik jari melalui kamera/galeri, menu pelatihan dan pengujian data sidik jari, dan menu identifikasi sidik jari.

#### c) Pembuatan Program

Sistem ini akan dibangun dalam bentuk aplikasi mobile. Bahasa pemrograman yang digunakan berupa Bahasa Pemrograman Java, kemudian menggunakan XML untuk mengatur tampilan antarmuka sitem. Sistem akan melakukan penyimpanan data pada Database menggunakan SQLite.

### 3.3 Implementasi Sistem

Tahap awal Implementasi dilakukan menggunakan Android Studio sebagai editor teks, Bahasa Pemrograman Java untuk mengembangkan

sistem *back-end* dan XML untuk membangun *front-end*. Selanjutnya implementasi yang sebenarnya akan dilakukan dengan menginstall aplikasi pada smartphone dengan OS Android.

### 3.4 Pengujian Sistem

Tahap ini meliputi proses pengujian menggunakan metode *confusion matrix*. Menurut Han dan Kamber (2011:365), *confusion matrix* adalah alat yang berguna untuk menganalisis seberapa baik *classifier* mengenali *tuple* dari kelas yang berbeda. TP dan TN memberikan informasi ketika *classifier* benar, sedangkan FP dan FN memberitahu ketika *classifier* salah. Contoh gambar *Confusion matrix* ditunjukkan pada Gambar 4.

		Predicted Class	
		T (1)	F (0)
Actual Class	T (1)	TP	FN
	F (0)	FP	TN

Gambar 4 Confusion Matrix

Pada penelitian ini penulis melakukan pengujian dengan menggunakan citra sidik jari sebanyak 120 citra uji dari 3 kelas yang berbeda, maka untuk setiap kelas memiliki 40 citra sidik jari. Dalam pengujiannya menggunakan metode *confusion matrix* salah satu citra digunakan sebagai acuan utama untuk perhitungan akurasi dan presisi dengan citra lainnya sebagai perbandingan, misal 40 citra sidik jari dari kelas A sebagai citra utama maka 40 citra lainnya berasal dari citra bukan kelas A, dari hasil perhitungan tersebut maka akan didapatkan hasil nilai akurasi dan presisi dari perbandingan antara citra kelas A dengan citra bukan kelas A.

Untuk mendapatkan nilai akurasi dan presisi didapatkan persamaan 3.1 dan 3.2 sebagai berikut :

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (3.1)$$

$$Presisi = \frac{TP}{FP + TP} \times 100\% \quad (3.2)$$

Sistem yang dibuat pada penelitian ini menggunakan bahasa java sebagai bahasa pemrograman *back-end* dan bahasa XML sebagai bahasa pemrograman *front-end*, tools media perancangannya menggunakan aplikasi android studio.

### 3.5 Perangkat Pendukung Penelitian

Adapun perangkat pendukung berupa perangkat lunak (*Software*) dan perangkat keras (*Hardware*) dengan spesifikasi sebagai berikut:

- a) Kebutuhan Perangkat Lunak (*Software*)
  - Sistem Operasi : Windows 10 Pro
  - Pengelola Kata : Microsoft Office Word 2010
  - Editor Text : Android Studio
  - Bahasa Pemrograman : Java
  - Desain : XML, Microsoft Office, Visio 2010, Microsoft Powerpoint 2010
  - Database : SQLite management
- b) Kebutuhan Perangkat Keras (*Hardware*)
  - Merk Komputer : Lenovo IdeaPad 110 14AST
  - Penyimpanan : 500GB SSHD
  - RAM : 4 GB
  - Processor : AMD APU A9-9400 up to 3.2 Ghz

## 4. HASIL DAN PEMBAHASAN

### 4.1. HASIL

Berikut adalah proses-proses yang ada untuk mendapatkan hasil identifikasi dalam sistem pengenalan citra sidik jari menggunakan jaringan saraf tiruan backpropagation:

#### 4.1.1 Akuisisi Citra Sidik Jari

Akuisisi citra sidik jari pada sistem ini meliputi penyiapan keseluruhan Alat maupun citra itu sendiri untuk menghasilkan citra digital, alat yang digunakan untuk mendapatkan citra sidik jari yaitu:

- a) Kertas HVS sebagai media untuk meletakkan cap sidik jari.
- b) Cap tinta *fingerprint* merk Trodat sebagai media untuk menghasil citra sidik jari.
- c) Kamera *smartphone* sebagai media untuk mengambil cap sidik jari yang telah ada pada kertas HVS menjadi citra digital yang akan disimpan pada memori telepon.

Bentuk cap sidik jari yang telah diletakan pada kertas HVS dapat dilihat pada Gambar 5.



Gambar 5 Data cap sidik Jari



Proses akuisisi citra pada sistem dapat dilihat pada Gambar 6.



**Gambar 6** Halaman Input citra sidik jari melalui Kamera

#### 4.1.2 Proses Pengolahan Citra

Setelah proses akuisisi selesai maka citra yang telah difoto kemudian dilakukan proses pengolahan Citra untuk mendapatkan ciri citra untuk digunakan dalam proses pelatihan jaringan backpropagation, Gambar 7 menunjukkan proses pengolahan Citra pada sistem ini.



**Gambar 7** Menu tampilan Citra biner

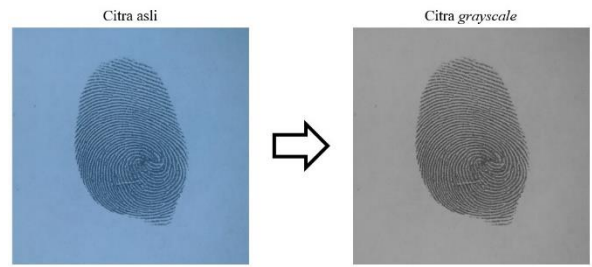
Dalam proses pengolahan Citra terdapat beberapa tahap antara lain yaitu:

##### a) Proses *grayscale*

Grayscale citra adalah proses perubahan nilai pixel dari warna (RGB) menjadi gray-level. Proses ini dilakukan dengan meratakan nilai pixel dari tiga nilai Red, Green, dan Blue (RGB) menjadi satu nilai, untuk mendapatkan nilai tersebut dapat menggunakan persamaan 5.1 sebagai berikut:

$$\begin{aligned} \text{Grayscale} = & (R * 0,2989) \\ & + (G * 0,5870) \\ & + (B * 0,1140) \end{aligned} \quad (4.1)$$

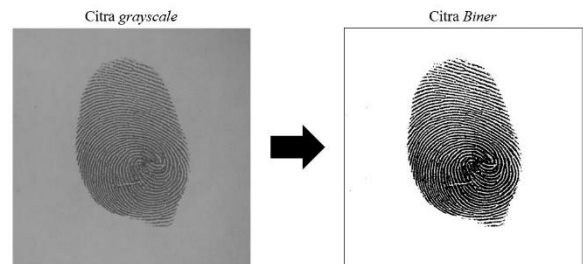
Hasil proses thresholding dapat dilihat pada Gambar 8.



**Gambar 8** Hasil Proses Grayscale Image

##### b) Proses thresholding

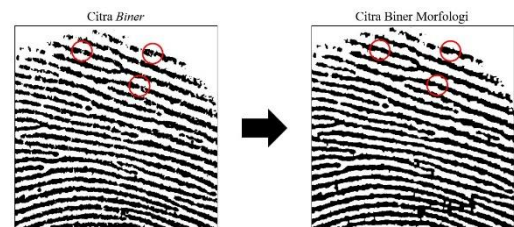
Thresholding merupakan tahapan yang dilakukan untuk mendapatkan citra biner, proses threshold tidak lepas dari proses grayscale karena untuk mendapatkan citra biner, citra awal harus sudah berbentuk menjadi citra grayscale karena untuk mengimplementasikan citra biner membutuhkan nilai rata-rata dari nilai RGB. Hasil proses thresholding dapat dilihat pada Gambar 9.



**Gambar 9** Hasil Proses thresholding Image

##### c) Proses morfologi *closing*

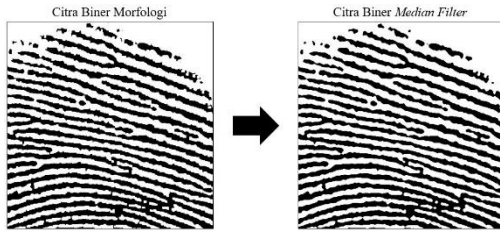
Operasi morfologi citra dilakukan untuk memperjelas tampilan citra sidik jari hasil thresholding, memperjelas tepi dari citra sidik jari, dan menghilangkan bagian-bagian yang bukan merupakan citra sidik jari. proses ini sangat berguna pada penelitian ini karena kadang pada saat melakukan cap sidik jari alur sidik jari tidak terbentuk sempurna, oleh sebab itu proses ini bertujuan untuk memperbaiki alur sidik jari yang terputus, contoh operasi morfologi *Closing* pada citra dapat dilihat pada Gambar 10.



**Gambar 10** Citra Biner Morfologi

d) Proses median filtering

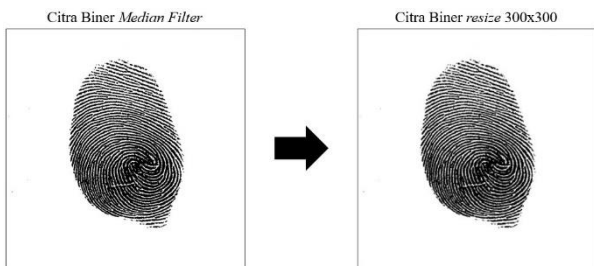
Median filtering merupakan teknik untuk melakukan peningkatan citra dan penghalusan citra, pada aplikasi ini setelah citra dilakukan operasi morfologi untuk dilakukan perbaikan citra maka selanjutnya adalah operasi median filtering untuk menghaluskan citra dan meratakan tepian objek dari citra yang tidak rata. *median filtering* atau *penghalusan citra biner* ini dapat dilihat pada Gambar 11.



Gambar 11 Citra Biner Median Filter

e) Proses resizing

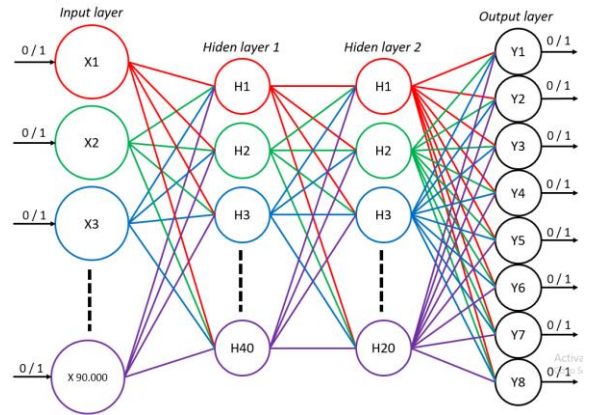
Resize atau perubahan ukuran citra dilakukan untuk memperkecil ukuran citra agar komputer mampu memproses citra dengan cepat dan tidak menghabiskan banyak memori penyimpanan. Penelitian ini menggunakan citra yang berukuran 300 x 300 pixel, ukuran tersebut dipilih agar pada saat penghitungan jumlah pixel sidik jari tidak menghabiskan waktu lama karena harus melakukan proses pelatihan pada metode JST. proses *resize image* ini dapat dilihat pada Gambar 12.



Gambar 12 Citra Biner resize

4.1.3 Proses Pelatihan Backpropagation

Proses pelatihan JST Backpropagation melibatkan 4 layer sebagai jaringannya untuk inputnya sebanyak 90.000, 20 untuk *hidden layer* pertama, 10 untuk *hidden layer* kedua dan 8 untuk *output layer* dengan maksimal epoch yaitu sebanyak 10.000 dan minimum eror sebanyak 0.00001 untuk lebih jelasnya dapat dilihat pada arsitektur pelatihan JST backpropagation pada Gambar 13.



Gambar 13 Arsitektur pelatihan JST backpropagation

Untuk data pelatihan dapat dilihat pada Tabel 1 sebagai berikut.

Tabel 1 Tabel Data Pelatihan Backpropagation

No.	Kelas / label	Citra sidik jari	Target
1.	Jamal	Jempol 1-32	00000001
2.		Telunjuk 33-64	
3.		Tengah 65-96	
4.		Manis 97-128	
5.		kelingking 129-160	
6.	hakim	Jempol 161-192	00000010
7.		Telunjuk 193-224	
8.		Tengah 225-256	
9.		Manis 257-288	
10.		kelingking 289-320	
11.	Ibnu Jarir	Jempol 321-352	00000011
12.		Telunjuk 353-384	
13.		Tengah 385-416	
14.		Manis 417-448	
15.		kelingking 449-480	

4.1.4 Proses Pengujian Forward Propagation

Proses pengujian JST backpropagation ini hampir sama dengan proses pelatihan forward propagation namun yang membedakan adalah dari langkahnya, untuk proses pengujian tidak memerlukan lagi proses forward propagation karena tidak perlu lagi merubah bobot, pada proses ini nantinya akan menghasilkan seberapa baik pelatihan jaringan tersebut. Untuk arsitektur JST nya sama dengan proses pelatihan, kemudian untuk melihat data pengujian JST forward propagation dapat dilihat pada Tabel 2.

**Tabel 2** Tabel data pengujian forward propagation

No.	Kelas / label	Citra sidik jari	Target
1.	jamal	Jempol 1-8	00000001
2.		Telunjuk 9-16	
3.		Tengah 17-24	
4.		Manis 25-32	
5.		kelingking 33-40	
6.	hakim	Jempol 41-48	00000010
7.		Telunjuk 49-56	
8.		Tengah 57-64	
9.		Manis 65-72	
10.		kelingking 73-80	
11.	Ibnu Jarir	Jempol 81-88	00000011
12.		Telunjuk 89-96	
13.		Tengah 97-104	
14.		Manis 105-112	
15.		kelingking 113-120	

## 4.2. PEMBAHASAN

Berikut pembahasan dari pengujian sistem pengenalan citra sidik jari. Proses pengujian akan dilakukan dengan menggunakan metode *confusion matrix*.

### 4.2.1 Pengujian Confusion Matrix

Berdasarkan proses *training* dan *testing* yang telah dilakukan, maka dapat dibuat *confusion matrix* dari masing-masing proses tersebut, proses ini menggunakan 2 kelas sebagai perbandingan, tujuan dari *confusion matrix* itu sendiri adalah menguji sistem untuk menghasilkan nilai akurasi dan presisi.

a) *confusion matrix* pelatihan JST backpropagation

Sebelum dilakukan proses *confusion matrix* berikut akan ditampilkan tabel hasil *output* dari proses pelatihan, hasil *output* pelatihan sistem ditampilkan pada Tabel 3 sebagai berikut.

**Tabel 3** Tabel Output Proses Pelatihan

No.	Nama	Citra	Target	Output	T/F
1	Jamal	Jempol 1	1	1	T
2		Jempol 2	1	1	T
3		Jempol 3	1	1	T
4		Jempol 4	1	1	T
5		Jempol 5	1	1	T
.		.	.	.	.
.		.	.	.	.
.		.	.	.	.
161	jarir	Jempol 1	2	2	T
162		Jempol 2	2	2	T
163		Jempol 3	2	2	T
164		Jempol 4	2	2	T

165		Jempol 5	2	2	T
.		.	.	.	.
.		.	.	.	.
.		.	.	.	.

Dari hasil *output* diatas, bisa digunakan untuk mendapatkan nilai TP, TN, FP, FN.

- **True Positive (TP)** : dari data diatas dimana *output* citra (*Positive*) 00000001, dan target (*True*) 00000001. Dari data diatas ada 156 citra. Jadi nilai TP nya adalah 156.
- **True Negative (TN)** : dari data diatas dimana *output* citra (*Negative*) 00000010, dan target (*True*) 00000010. Dari data diatas ada 160 citra dari nomor 161-320. Jadi nilai TP nya adalah 160.
- **False Positive (FP)** : dari data diatas dimana *output* citra (*Positive*) 00000001, dan target (*False*) 00000010, prediksinya salah. Dari data diatas tidak ada citra yang bernilai *false*. Jadi nilai FP nya adalah 0.
- **False Negative (FN)** : dari data diatas dimana *output* citra (*Negative*) 00000010, dan target (*True*) 00000001. Dari data diatas ada 4 citra dari nomor 36, 99, 100 dan 104. Jadi nilai TP nya adalah 4.

Dari hasil nilai yang didapatkan bentuk *confusion matrix* data pelatihan dapat dilihat pada Gambar 14.

		Predicted Class	
		T (1)	F (0)
Actual Class	T (1)	156	4
	F (0)	0	160

**Gambar 14** Confusion Matrix Pelatihan

Selanjutnya adalah perhitungan nilai akurasi dan presisi, perhitungannya adalah sebagai berikut:

$$\begin{aligned} \text{Akurasi} &= ((TP + TN) / (TP + TN + FP + FN)) \times 100\% \\ \text{Akurasi} &= ((156 + 160) / (156 + 160 + 0 + 4)) \times 100\% \\ \text{Akurasi} &= (316 / 320) \times 100\% \\ \text{Akurasi} &= 0,9875 \times 100\% = 98,75\% \end{aligned}$$

$$\begin{aligned} \text{Presisi} &= (TP / (FP + TP)) \times 100\% \\ \text{Presisi} &= (156 / (0 + 156)) \times 100\% \\ \text{Presisi} &= (156 / 156) \times 100\% \\ \text{Presisi} &= 1 \times 100\% = 100\% \end{aligned}$$

Dapat disimpulkan bahwa nilai akurasi dan presisi pada proses pelatihan JST backpropagation masing-masing bernilai 98,75% dan 100%.

b) *confusion matrix* pengujian JST backpropagation

Sebelum dilakukan proses *confusion matrix* berikut akan ditampilkan tabel hasil *output* dari proses pengujian, hasil *output* pengujian sistem ditampilkan pada Tabel 4 sebagai berikut.

**Tabel 4** Tabel Output Proses Pengujian

No.	Nama	Citra	Target	Output	T / F
1.	Jamal	Jempol1	1	1	T
2.		Jempol2	1	1	T
3.		Jempol3	1	1	T
4.		Jempol4	1	1	T
5.		Jempol5	1	1	T
6.		Jempol6	1	1	T
7.		Jempol7	1	1	T
8.		Jempol8	1	1	T
.		.	.	.	.
.		.	.	.	.
.		.	.	.	.
41.	jarir	Jempol1	2	2	T
42.		Jempol2	2	1	F
43.		Jempol3	2	2	T
44.		Jempol4	2	2	T
45.		Jempol5	2	2	T
46.		Jempol6	2	2	T
47.		Jempol7	2	2	T
48.		Jempol8	2	2	T
.		.	.	.	.
.		.	.	.	.
.		.	.	.	.

Dari hasil *output* diatas, bisa digunakan untuk mendapatkan nilai TP, TN, FP, FN.

- *True Positive* (TP) : dari data diatas dimana *output* citra (*Positive*) 00000001, dan target (*True*) 00000001. Dari data diatas ada 37 citra, Jadi nilai TP nya adalah 37.
- *True Negative* (TN) : dari data diatas dimana *output* citra (*Negative*) 00000010, dan target (*True*) 00000010. Dari data diatas ada 39 citra. Jadi nilai TP nya adalah 39.
- *False Positive* (FP) : dari data diatas dimana *output* citra (*Positive*) 00000001, dan target (*False*) 00000010, prediksinya salah. Dari data diatas ada 1 citra dari nomor 42. Jadi nilai FP nya adalah 1.
- *False Negative* (FN) : dari data diatas dimana

*output* citra (*Negative*) 00000010, dan target (*True*) 00000001. Dari data diatas ada 3 citra dari nomor 28, 34 dan 37. Jadi nilai FN nya adalah 3.

Dari hasil nilai yang didapatkan bentuk *confusion matrix* data pengujian dapat dilihat pada Gambar 15.

		Predicted Class	
		T (1)	F (0)
Actual Class	T (1)	37	3
	F (0)	1	39

**Gambar 15** Confusion Matrix Pengujian

Selanjutnya adalah perhitungan nilai akurasi dan presisi, perhitungannya adalah sebagai berikut:

$$\begin{aligned} \text{Akurasi} &= (TP + TN) / (TP + TN + FP + FN) \times 100\% \\ \text{Akurasi} &= (37 + 39) / (37 + 39 + 1 + 3) \times 100\% \\ \text{Akurasi} &= (76 / 80) \times 100\% \\ \text{Akurasi} &= 0,95 \times 100\% = 95\% \end{aligned}$$

$$\begin{aligned} \text{Presisi} &= (TP / (FP + TP)) \times 100\% \\ \text{Presisi} &= (37 / (1 + 37)) \times 100\% \\ \text{Presisi} &= (37 / 38) \times 100\% \\ \text{Presisi} &= 0,9737 \times 100\% = 97,37\% \end{aligned}$$

Dapat disimpulkan bahwa nilai akurasi dan presisi pada proses pengujian JST forward propagation adalah 95% dan 97,37%.

## 5. PENUTUP

### 5.1. Kesimpulan

Berdasarkan dari penelitian yang sudah dilakukan serta uraian rumusan masalah yang telah dijelaskan pada bab sebelumnya maka dapat disimpulkan bahwa dalam merancang dan membangun aplikasi pengenalan citra sidik jari peneliti menggunakan metode jaringan saraf tiruan galat mundur sebagai metode pengenalan citra, kemudian untuk proses pengolahan citra menggunakan metode *grayscale*, *thresholding*, morfologi *closing*, median *filter* dan *resize*, library OpenCV untuk menjalankan proses pengolahan citra dan jaringan saraf tiruan, kemudian peneliti menggunakan bahasa pemrograman java sebagai bahasa pemrograman *back-end* dan XML sebagai bahasa pemrograman *front-end* lalu IDE yang digunakan menggunakan Android studio dan SQLite sebagai basis datanya, hasil yang diperoleh nilai akurasi pelatihan citra sidik sebesar 90,42% dan nilai

akurasi pengujian sebesar 71,67% dengan maksimal epoch sebanyak 10.000 dan galat sebanyak 0,001 dengan durasi waktu pelatihan selama 15 menit dan durasi pengujian selama 15 detik. Hasil dari pengujian menggunakan *confusion matrix* untuk data latih sebesar 98,75% untuk nilai akurasi dan 100% untuk nilai presisinya, sedangkan untuk data uji sebesar 95% untuk nilai akurasi dan 97,37% untuk nilai presisinya.

## 5.2. Saran

Adapun saran yang dapat penulis sampaikan untuk pengembangan sistem selanjutnya adalah sebagai berikut:

- a. Dapat dikembangkan dari segi basis data yang pada saat ini masih menggunakan basis data lokal menjadi basis data client server, hal tersebut berguna agar pengguna aplikasi dapat menyimpan data dan menggunakan data sidik jari secara global untuk proses identifikasi.
- b. Dapat dikembangkan dengan basis maupun *platform* yang lain.

## DAFTAR PUSTAKA

- [1] Arifianto, Teguh. 2011. Membuat Interface Aplikasi Android Lebih Keren dengan LWIT. Yogyakarta : Andi Offset.
- [2] developers (2019), Mengenal Android Studio, (<https://developer.android.com/studio/intro?hl=id>), akses 15 Oktober 2019.
- [3] Devtrik (2017), Operasi Morfologi Pada PengolahanCitra, (<https://devtrik.com/opencv/operasi-morfologi-pada-pengolahan-citra/>), akses 15 Oktober 2019.
- [4] DuniaIlkom (2019), Tutorial Belajar Java Part 1: Pengertian Bahasa Pemrograman Java, (<https://www.duniailkom.com/tutorial-belajar-java-pengertian-bahasa-pemrograman-java/>), akses 16 Oktober 2019.
- [5] heartbeat.fritz.ai (2019), Running Artificial Neural Networks in Android using OpenCV, (<https://heartbeat.fritz.ai/running-artificial-neural-networks-in-android-using-opencv-3c5640778578>), akses 3 Desember 2019.
- [6] Hermawan, Arief. 2006. Jaringan Saraf Tiruan. Yogyakarta : Andi Offset.
- [7] Kristanto, Andi. 2008. Perancangan Sistem Informasi dan Aplikasinya. Yogyakarta : Gava Media.
- [8] Medium.com (2019), Mengenal Accuracy, Precision, Recall dan Specificity serta yang diprioritaskan dalam Machine Learning, (<https://medium.com/@rey1024/mengenal-accuracy-precision-recall-dan-specificity-seerta-yang-diprioritaskan-b79ff4d77de8>), akses 3 Januari 2020.
- [9] Mulyanto, Edy dkk. 2009. Teori Pengolahan Citra Digital. Yogyakarta : Andi Offset.
- [10] Naf'an M. Z. dan Arifin J. 2017, Identifikasi Tanda Tangan Berdasarkan Grid Entropy Menggunakan Multi Layer Perceptron, JURNAL INFOTEL, 9(2), 172-176.
- [11] Siallagan, Sariadin. 2009. Pemrograman Java Dasar-dasar Pengenalan dan Pemahaman. Yogyakarta : Andi Offset.
- [12] Soemirat, Soleh & Ardianto, Elvinaro. 2007. Dasar-Dasar Public Relations. Bandung : Rosdakarya.
- [13] Sukamto, & Shalahuddin. 2013. Analisa dan Desain Sistem Informasi. Yogyakarta : Andi Offset.
- [14] Sukamto, Rosa A. dan M. Salahuddin. 2014. Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek. Bandung : Informatika.
- [15] Sutrisno A. Dan Fahruzi I. 2016, Identifikasi Tandatangan Digital Menggunakan Jaringan Syaraf Tiruan Backpropagation, JURNAL INOVTEK POLBENG - SERI INFORMATIKA, 1(2), 161-168.
- [16] Wahyudi R., Soesanto O. Dan Muliadi 2015, Rancang Bangun Aplikasi Pengenalan Pola Sidik Jari, Kumpulan jurnaL Ilmu Komputer (KLIK), 2(1), 74-83.