

**VERIFIKASI TANDA TANGAN MENGGUNAKAN EKSTRAKSI
FITUR LBP DAN KLASIFIKASI LVQ**

Tugas akhir
untuk memenuhi sebagian persyaratan
mencapai derajat Sarjana S-1 Program Studi Teknik Informatika



Oleh:
Medeline Widia Andani
FID 016 053

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MATARAM
2020

TUGAS AKHIR

**VERIFIKASI TANDA TANGAN MENGGUNAKAN EKSTRAKSI FITUR LBP
DAN KLASIFIKASI LVQ**

Oleh :

MEDELINE WIDIA ANDANI

F1D016053

Telah diperiksa oleh Tim Pembimbing :

1. Pembimbing Utama



Tanggal: 15/07/2020

Prof. Dr. Eng. I Gede Pasek Suta Wijaya, ST., MT.
NIP. 197311302000031001

2. Pembimbing Pendamping



Tanggal: 14/07/2020

Gibran Satya Nugraha, S.Kom., M.Eng.
NIP. 199203232019031012

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Teknik
Universitas Mataram



Prof. Dr. Eng. I Gede Pasek Suta Wijaya, ST., MT.
NIP. 197311302000031001

TUGAS AKHIR

**VERIFIKASI TANDA TANGAN MENGGUNAKAN EKSTRAKSI FITUR LBP
DAN KLASIFIKASI LVQ**

Oleh :

MEDELINE WIDIA ANDANI

F1D016053

Telah diujikan di depan penguji

Pada tanggal 6 Juli 2020

Dan dinyatakan telah memenuhi syarat mencapai derajat Sarjana S-1
Program Studi Teknik Informatika

Susunan Tim Penguji :

1. Penguji 1



Tanggal: 14/07/2020

Fitri Bimantoro, ST., M.Kom.
NIP. 198606222015041002

2. Penguji 2



Tanggal: 13/07/2020

Ramaditia Dwiyanaputra, S.T., M.Eng.
NIP. -

3. Penguji 3



Tanggal: 13/07/2020

Ariyan Zubaidi, S.Kom., MT
NIP. 198609132015041001

Mataram, 21 Juli 2020
Dekan Fakultas Teknik
Universitas Mataram



Akmaluddin, ST., M.Sc Eng., Ph.D.
NIP. 196812311994121001

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Saya menyatakan bahwa tugas akhir ini dengan judul “Verifikasi Tanda Tangan Menggunakan Ekstraksi Fitur LBP dan Klasifikasi LVQ” sepenuhnya adalah karya sendiri. Tidak ada bagian di dalamnya yang merupakan plagiat dari karya orang lain dan saya tidak melakukan penjiplakan atau pengutipan dengan cara yang tidak sesuai dengan etika keilmuan yang berlaku. Atas pernyataan ini, saya siap menanggung resiko/sanksi yang dijatuhkan kepada saya apabila kemudian ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya ini, atau ada klaim dari pihak lain terhadap keaslian karya saya ini.

Mataram, 22 Juli 2020

Yang membuat pernyataan,

Medeline Widia Andani

PRAKATA

Assalamu'alaikum Warahmatullahi Wabarakatuh

Segala puji bagi Allah SWT yang telah memberikan rahmat dan karunianya kepada penulis, sehingga penulis dapat menyelesaikan skripsi dengan judul “Verifikasi Tanda Tangan Menggunakan Ekstraksi Fitur LBP dan Klasifikasi LVQ” tepat pada waktunya sebagai syarat untuk menyelesaikan Program Studi (S1) pada Program Sarjana Fakultas Teknik Jurusan Informatika Universitas Mataram. Shalawat dan salam senantiasa tercurahkan kepada Rasulullah SAW yang mengantarkan manusia dari zaman kegelapan ke zaman yang terang benderang.

Penulis tentu menyadari Tugas Akhir ini masih jauh dari kata sempurna dan masih terdapat kesalahan dan kekurangan di dalamnya dikarenakan terbatasnya pengalaman dan pengetahuan yang dimiliki penulis. Untuk itu, penulis mengharapkan segala bentuk kritik serta saran dari pembaca untuk Tugas Akhir ini, agar Tugas Akhir ini dapat menjadi lebih baik lagi.

Demikian yang bisa disampaikan, Terimakasih atas semua pihak yang telah membantu dalam penyelesaian Tugas Akhir ini baik moril maupun materil. Semoga Tugas Akhir ini dapat bermanfaat bagi para pembaca.

Wassalamual'aikum Warahmatullahi Wabaraktuh

Mataram, 22 Juli 2020

Penulis

UCAPAN TERIMAKASIH

Penulis menyadari bahwa selesainya Tugas Akhir ini tentunya bukan hanya dari usaha penulis saja. Tugas Akhir ini bisa selesai tepat waktu tentunya berkat dukungan dari semua pihak yang terlibat juga. Oleh karena itu, pada kesempatan ini penulis menghaturkan terimakasih yang sebesar-besarnya kepada:

1. Kedua Orang Tua dan Kakak satu-satunya yaitu Ayu Ditha Puspita yang selalu memberikan dukungan kepada penulis dalam segala lini kehidupan selama perkuliahan.
2. Bapak I Gede Pasek Suta Wijaya S.T., M.T., D.Eng selaku dosen pembimbing utama yang telah memberikan bimbingan dan arahan kepada penulis selama penyusunan Tugas Akhir sehingga dapat selesai dengan baik.
3. Bapak Gibran Satya Nugraha, S.Kom., M.Eng selaku dosen pembimbing pendamping yang telah memberikan bimbingan dan arahan kepada penulis selama penyusunan Tugas Akhir sehingga dapat selesai dengan baik.
4. Insomania 16 selaku teman diskusi dan *support system* selama pengerjaan Tugas Akhir penulis.
5. Novita selaku fasilitator dan Gungmas selaku pengingat agar penulis tidak bermalas-malasan mengerjakan Tugas Akhir.
6. Nanang, Farhan, dan Naufal yang selalu membantu menghadapi *error code* hingga penulis bisa lulus tepat waktu walau pandemi Covid-19 melanda.
7. Qeluarga Tjemara yang selalu menghibur penulis ketika menghadapi stresnya Tugas Akhir. Khususnya untuk Mia selaku pemberi nasihat.
8. HMB yang selalu setia mendengarkan setiap keluh kesah penulis dan selalu menghibur penulis selama mengerjakan Tugas Akhir.
9. Semua pihak yang tidak dapat saya sebutkan namanya satu persatu yang telah memberikan dukungan baik moril maupun materil sehingga penulis dapat menyelesaikan Tugas Akhir dengan baik.

Semoga Allah SWT selalu memberikan rahmat dan hidayah-Nya dan memberikan imbalan yang setimpal atas bantuan yang diberikan kepada penulis.

DAFTAR ISI

HALAMAN COVER.....	i
LEMBAR PENGESAHAN.....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR.....	iv
PRAKATA.....	v
UCAPAN TERIMAKASIH.....	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR.....	xi
ABSTRAK.....	xii
ABSTRACT.....	xiii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah.....	4
1.4 Tujuan.....	4
1.5 Manfaat.....	4
1.6 Sistematika Penulisan.....	5
BAB II TINJAUAN PUSTAKA DAN DASAR TEORI.....	7
2.1 Tinjauan Pustaka.....	7
2.2 Dasar Teori.....	10
2.2.1 Biometrik tanda tangan.....	10
2.2.2 Verifikasi tanda tangan.....	10
2.2.3 Citra digital.....	11
2.2.4 Pre-processing <i>resize</i>	11
2.2.5 Ekstraksi fitur LBP.....	11

2.2.6	Ekstraksi Fitur DCT	14
2.2.7	Klasifikasi dengan LVQ.....	15
2.2.8	Proses verifikasi tanda tangan.....	17
2.2.9	Evaluasi Hasil Klasifikasi	18
BAB III METODOLOGI PENELITIAN.....		19
3.1	Alat dan bahan.....	19
3.2	Rencana Penelitian	19
3.3	Perancangan sistem.....	21
3.3.1	Proses pelatihan.....	22
3.3.2	Proses pengujian	22
3.4	<i>Preprocessing</i>	23
3.5	Ekstraksi fitur	23
3.5.1	Pembentukan matriks LBP.....	23
3.5.2	Perhitungan fitur LBP	25
3.6	Tahap Klasifikasi LVQ	29
3.6	Verifikasi tanda tangan.....	34
3.8	Skenario pengujian.....	34
BAB IV HASIL DAN PEMBAHASAN		37
4.1	Pengumpulan <i>Dataset</i>	37
4.2	Mekanisme Penelitian	37
4.3	<i>Preprocessing</i>	38
4.4	Model LVQ	39
4.5	<i>Training</i>	40
4.6	<i>Testing</i>	41
4.7	Verifikasi Tanda Tangan.....	41

4.8	Pengujian dengan Ekstraksi Fitur Statistik LBP	41
4.9	Pengujian dengan Kombinasi Ekstraksi Fitur LBP dan DCT	43
4.9.1	Pengujian Koefisien DCT	44
4.10	Hasil Pengujian Terhadap Skenario Uji	45
4.10.1	Pengujian Terhadap Parameter LBP	46
4.10.2	Pengujian Terhadap Parameter LVQ	47
4.10.3	Pengujian Terhadap Ukuran Citra.....	48
4.10.4	Pengujian Terhadap Variabel <i>Threshold</i>	48
4.11	Pengecekan Kesalahan	49
BAB V KESIMPULAN DAN SARAN.....		52
5.1	Kesimpulan.....	52
5.2	Saran.....	53
DAFTAR PUSTAKA		54

DAFTAR TABEL

Tabel 2.1 <i>Confusion matrix</i>	18
Tabel 3.1 <i>Input</i> vektor	29
Tabel 3.2 Inisialisasi bobot	29
Tabel 3.3 <i>Confussion matrix</i> dengan data <i>dummy</i>	36
Tabel 4.1 Hasil pengujian dengan fitur statistik LBP.....	42
Tabel 4.2 Hasil pengujian DCT.....	43
Tabel 4.3 Hasil pengujian dengan klasifikasi LVQ.....	44
Tabel 4.4 Hasil pengujian koefisien DCT.....	45
Tabel 4.5 Hasil pengujian parameter LBP.....	46
Tabel 4.6 Hasil pengujian parameter LVQ	47
Tabel 4.7 Hasil pengujian ukuran citra	48
Tabel 4.8 Hasil pengujian terhadap variabel <i>threshold</i>	49
Tabel 4.9 Hasil verifikasi.....	50

DAFTAR GAMBAR

Gambar 2.1 Ilustrasi LBP[8].....	10
Gambar 2.2 <i>Learning Vector Quantization</i> [29].....	12
Gambar 3.1 Diagram alir proses penelitian.....	16
Gambar 3.2 Diagram perancangan sistem.....	21
Gambar 3.3 Contoh matriks citra.....	24
Gambar 3.4 Konversi nilai piksel ke biner.....	24
Gambar 3.5 Membentuk bilangan biner 8 bit.....	24
Gambar 3.6 Nilai titik pusat berubah menjadi nilai LBP.....	25
Gambar 3.7 Nilai pixel hasil perhitungan LBP.....	25
Gambar 4.1 <i>Dataset</i> tanda tangan (a) hari ke 1, (b) hari ke 2, (c) hari ke3.....	37
Gambar 4.2 <i>Dataset</i> tanda tangan (a) asli, (b) palsu.....	37
Gambar 4.3 Citra masukan.....	38
Gambar 4.4 Citra <i>training</i>	39
Gambar 4.5 Citra <i>testing</i>	39
Gambar 4.6 Arsitektur LVQ.....	40
Gambar 4.7 Persebaran nilai fitur keseluruhan <i>dataset</i>	42
Gambar 4.8 Proses ekstraksi fitur	44
Gambar 4.9 Mekanisme <i>K-fold cross validation</i>	46
Gambar 4.10 Persebaran rata-rata hasil <i>similarity</i> dataset tanda tangan yang dinyatakan asli dan palsu pada tiap kelas.....	51

ABSTRAK

Tanda tangan merupakan salah satu media yang digunakan untuk keperluan verifikasi dan legalisasi terhadap suatu informasi yaitu dokumen-dokumen yang berkaitan erat dengan legalitas secara hukum. Secara umum, verifikasi tanda tangan dilakukan dengan cara manual, yaitu dengan membandingkan secara langsung, hal ini tentunya tidak efektif terutama jika melakukan cukup banyak verifikasi. Oleh karena itu, dibutuhkan suatu sistem komputer yang otomatis dapat memverifikasi tanda tangan seseorang untuk menghemat waktu dalam pencocokan dan mengurangi kesalahan. Penelitian ini dilakukan dengan menggunakan kombinasi metode ekstraksi fitur *Local Binary Pattern* (LBP) dan *Discrete Cosine Transform* (DCT) serta klasifikasi *Learning Vector Quantization* (LVQ). *Dataset* yang akan digunakan dalam penelitian ini adalah citra tanda tangan sebanyak 600 citra dengan ukuran 500x500 pixel yang diambil dari 30 responden dimana masing - masing responden diambil 15 tanda tangan asli dan 5 tanda tangan palsu. Hasil dari penelitian ini yaitu pada proses identifikasi tanda tangan dihasilkan akurasi sebesar 93% dan pada proses verifikasi tanda tangan dihasilkan akurasi sebesar 63%, sensitivity sebesar 89%, dan specificity sebesar 42%.

Kata Kunci: Tanda Tangan, Verifikasi, LBP, LVQ, DCT

ABSTRACT

Signature is one of the media used for verification and legalization of information, such as documents that are closely related to legality. In general, signature verification is done manually by direct comparing, this is certainly not effective, especially if doing a lot verification. Therefore, we need a computer system that can automatically verify a person's signature to save time in matching and reducing errors. This research was conducted using combination feature of Local Binary Pattern (LBP) and Discrete Cosine Transform (DCT) to obtain parameters values that will be used as input to classification. Then, Learning Vector Quantization (LVQ) algorithm is used as classifier to identify and verification features of the signature. The Datasets that used in this research are 600 signature images with a size of 500x500 pixels taken from 30 respondents where each respondent taken 15 original signatures and 5 fake signatures. The results of this research are that the signature identification process resulted in 93% and the verification process resulted in an accuracy of 63%, a sensitivity of 89%, and a specificity of 42%.

Keywords: *signature, verification, LBP, DCT, LVQ*

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pada perkembangan teknologi yang cukup pesat ini keamanan menjadi kebutuhan manusia yang tidak bisa dilepaskan. Baik keamanan untuk dirinya sendiri maupun keamanan hal-hal yang berkaitan dengan dirinya, seperti dokumen-dokumen legal dan penting. Sidik jari dan PIN merupakan contoh pengaman yang dapat digunakan. Salah satu pengaman yang terkenal adalah dengan tanda tangan. Tanda tangan dianggap lebih mudah digunakan, murah, cukup efektif[1].

Tanda tangan adalah hasil dari proses menulis seseorang yang bersifat khusus sebagai substansi simbolik, tanda tangan setiap orang umumnya identik namun tidak sama[2]. Tanda tangan merupakan salah satu media yang digunakan untuk keperluan verifikasi dan legalisasi terhadap suatu informasi. Tanda tangan mempunyai peranan penting dalam memverifikasi dan melegalisasi suatu dokumen. Dokumen-dokumen yang membutuhkan tanda tangan biasanya berupa *Memorandum of Understanding* (MoU), surat pernyataan, surat pengantar secara resmi, profil pribadi dan dokumen-dokumen yang berkaitan erat dengan legalitas secara hukum[1]. Namun, masalah pemalsuan tanda tangan masih menjadi ancaman bagi penggunaan tanda tangan.

Salah satu contoh kasus pemalsuan tanda tangan terjadi pada tahun 2013 di Kota Solo. Menurut data yang dipublikasikan oleh Solo Pos tahun 2013, menyatakan bahwa di kota Solo terjadi tindak kecurangan dalam pengiriman berkas lamaran para Calon Pegawai Negeri Sipil (CPNS) dengan memalsukan tanda tangan pada legalisir ijazah sebanyak 40% dari 247 peserta dan hal ini diketahui setelah tim verifikasi memeriksa berkas lamaran[3]. Seringnya terjadi pemalsuan tanda tangan, antara lain disebabkan oleh sistem verifikasi yang kurang baik.

Verifikasi tanda tangan (*signature verification*) meliputi dua bagian yang berbeda tetapi berkaitan erat satu sama lain. Yang pertama adalah identifikasi dari pemilik tanda tangan sedangkan yang kedua adalah keputusan apakah tanda tangan tersebut asli atau dipalsukan[4]. Secara umum, verifikasi tanda tangan dilakukan

dengan cara manual, yaitu dengan membandingkan secara langsung dengan menggunakan mata manusia yang mempunyai banyak kelemahan sehingga penglihatan menjadi letih, terutama jika melakukan cukup banyak verifikasi, dapat menyebabkan kebosanan orang yang melakukannya, sehingga ketelitian dan ketepatan hasil yang diinginkan seringkali kurang memuaskan[5]. Oleh karena itu, dibutuhkan suatu sistem komputer yang otomatis dapat memverifikasi tanda tangan seseorang untuk menghemat waktu dalam pencocokan dan mengurangi kesalahan. Namun, komputer tidak dapat langsung melakukan verifikasi tanda tangan, sehingga dibutuhkan proses pengenalan pola terlebih dahulu dengan melakukan ekstraksi fitur pada citra tanda tangan dimana fitur tersebut akan diklasifikasi sehingga citra tanda tangan dapat dikenali.

Penelitian untuk pengenalan tanda tangan telah banyak dilakukan dengan metode yang sangat beragam, baik metode ekstraksi fitur maupun metode klasifikasi. Terdapat beberapa metode yang digunakan untuk melakukan ekstraksi fitur. *Local Binary Pattern* (LBP) merupakan suatu metode yang digunakan untuk mencari pola-pola tekstur lokal pada citra. Berdasarkan penelitian terdahulu, metode LBP teruji ampuh untuk mendeskripsikan tekstur karena memiliki daya pembeda yang akurat, mudah diimplementasikan dan merupakan metode ekstraksi fitur yang cukup cepat dengan proses komputasi yang rendah[6], metode LBP juga telah banyak digunakan untuk mengekstraksi fitur pada penelitian yang terkait citra tulisan tangan seperti pengenalan tulisan aksara jawa dengan akurasi 82,5%[7]. Selain itu, pada penelitian dalam mengidentifikasi citra tanda tangan yang memiliki akurasi sebesar 94%[8]. Kemudian, terdapat metode ekstraksi fitur lainnya seperti *Discrete Cosine Transform* (DCT).

Metode DCT merupakan sebuah teknik yang mengubah sinyal ke dalam bentuk frekuensi dasar. DCT dipilih karena memiliki kemampuan yang baik untuk mengumpulkan informasi fitur penting pada frekuensi rendahnya dan fitur pada gambar tersebut dapat memberikan informasi mengenai fitur dengan kolerasi yang tinggi. DCT tidak berkorelasi dengan indeks frekuensi lawannya sehingga jika sebagian komponen non-dominan dihapus maka tidak akan secara signifikan

mengurangi kualitas informasinya[9]. Sehingga DCT merupakan metode yang cocok digunakan dilihat dari *dataset* tanda tangan yang memiliki lebih banyak jumlah *background* dibandingkan dengan tulisan tanda tangan yang dibutuhkan.

Salah satu metode klasifikasi tanda tangan yang telah digunakan yaitu metode *Learning Vektor Quantization* (LVQ). LVQ adalah suatu metode untuk melakukan pembelajaran pada lapisan kompetitif yang terawasi. Suatu lapisan kompetitif akan secara otomatis belajar untuk mengklasifikasikan vektor-vektor *input*. Jika dua vektor *input* mendekati sama, maka lapisan kompetitif akan meletakkan kedua vektor *input* tersebut ke dalam kelas yang sama. Berdasarkan penelitian terdahulu, keunggulan metode LVQ adalah kemampuannya untuk memberikan pelatihan terhadap lapisan-lapisan kompetitif sehingga secara otomatis dapat mengklasifikasikan vektor *input* yang diberikan[10], menghasilkan error yang lebih kecil dan waktu yang lebih cepat dibandingkan dengan *backpropogation*[11], metode LVQ juga terbukti merupakan metode yang baik untuk mengidentifikasi tanda tangan serta menghasilkan akurasi yang tinggi yaitu sebesar 94,16%[12].

Berdasarkan uraian di atas, penulis mencoba melakukan penelitian untuk perancangan sebuah model verifikasi tanda tangan menggunakan ekstraksi fitur dengan metode LBP dan DCT serta klasifikasi dengan metode LVQ. Data diperoleh dari citra tanda tangan yang ditulis pada *smartphone* dalam bentuk digital dari 30 responden dimana tiap responden diambil 15 tanda tangan asli yang diambil pada 3 hari yang berbeda dimana setiap 1 hari diambil 5 tanda tangan asli dan untuk tanda tangan palsu diambil 5 tanda tangan untuk setiap responden, dimana tanda tangan palsu dikerjakan oleh 5 orang yang dipercaya peneliti dapat meniru tanda tangan. Keluaran dari penelitian ini diharapkan sistem dapat melakukan verifikasi tanda tangan.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan sebelumnya, rumusan masalah yang ingin dijawab pada penelitian ini adalah sebagai berikut:

1. Bagaimana cara memverifikasi tanda tangan seseorang menggunakan metode ekstraksi fitur LBP dengan klasifikasi LVQ.

2. Bagaimana performa dari *classifier* LVQ dan metode ekstraksi fitur LBP untuk memverifikasi tanda tangan seseorang.
3. Berapa nilai variabel *threshold* yang tepat digunakan untuk memverifikasi tanda tangan sebagai tolak ukur untuk menentukan keaslian tanda tangan.

1.3 Batasan Masalah

Penelitian ini memiliki batasan-batasan masalah untuk memberikan lingkup penelitian agar lebih terfokus dalam pengerjaan. Adapun batasan masalah yang diberikan adalah sebagai berikut.

1. Citra tanda tangan yang digunakan sebagai data adalah citra yang diambil secara acak dari mahasiswa Teknik Informatika Universitas Mataram sebanyak 30 orang.
2. Citra tanda tangan yang akan diproses tidak memiliki gangguan (coretan atau gangguan lainnya).

1.4 Tujuan

Tujuan yang diharapkan dari penelitian ini adalah sebagai berikut:

1. Mengimplementasi metode ekstraksi fitur LBP dengan klasifikasi LVQ untuk memverifikasi tanda tangan seseorang.
2. Mengetahui performa dari *classifier* LVQ dengan ekstraksi fitur LBP untuk memverifikasi tanda tangan seseorang.
3. Mengetahui nilai variabel *threshold* yang tepat digunakan untuk memverifikasi tanda tangan sebagai tolak ukur untuk menentukan keaslian tanda tangan.

1.5 Manfaat

Manfaat dari penelitian ini secara umum dapat diperoleh oleh dua subjek antara lain.

1. Bagi penulis
 - a. Dapat menerapkan pengetahuan selama proses perkuliahan di Program Studi Informatika Fakultas Teknik Universitas Mataram terutama pengetahuan tentang pengolahan citra digital.
 - b. Dapat menambah pengetahuan di bidang pengenalan pola.
 - c. Melatih dalam menyelesaikan suatu permasalahan dengan menerapkan konsep pengenalan pola.

2. Bagi pembaca
 - a. Dapat mengetahui bagaimana cara untuk mengekstraksi fitur dengan menggunakan kombinasi metode ekstraksi fitur LBP dengan klasifikasi menggunakan LVQ.
 - b. Model yang dihasilkan dari penelitian ini dapat digunakan sebagai referensi untuk pembuatan aplikasi verifikasi tanda tangan untuk digunakan langsung oleh konsumen.
 - c. Mendapat wawasan mengenai pemanfaatan pola dalam menyelesaikan suatu permasalahan.

1.6 Sistematika Penulisan

Sistematika penulisan dari penelitian ini disajikan dalam beberapa bab antara lain sebagai berikut.

1. Bab I Pendahuluan

Bab ini menjelaskan dasar-dasar dari penulisan laporan tugas akhir, yang terdiri dari latar belakang, rumusan masalah, batasan masalah, tujuan, serta sistematika penulisan laporan tugas akhir.

2. Bab II Tinjauan Pustaka dan Teori

Bab ini membahas tentang penelitian terdahulu yang mengimplementasikan kombinasi metode ekstraksi fitur LBP dengan klasifikasi LVQ serta teori-teori sebagai referensi penulis ketika melakukan penelitian.

3. Bab III Metodologi Penelitian

Bab ini membahas tentang metodologi yang digunakan untuk verifikasi tanda tangan dengan kombinasi metode ekstraksi fitur LBP dengan klasifikasi LVQ.

4. Bab IV Analisis dan Perancangan

Bab ini merupakan pembahasan tentang analisis perangkat lunak, meliputi analisis masalah, analisis metode, analisis kebutuhan sistem, serta perancangan sistem yang terdiri dari perancangan diagram alir (*flowchart*).

5. Bab V Implementasi dan Pengujian Metode

Bab ini membahas implementasi yang dilakukan terhadap pengolahan citra digital pada citra di desktop menggunakan Jupyter Notebook dengan kombinasi metode ekstraksi fitur LBP dengan klasifikasi LVQ.

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Penelitian mengenai verifikasi tanda tangan, ekstraksi fitur LBP dan klasifikasi LVQ sudah pernah dilakukan oleh beberapa peneliti. Perbedaan paling menonjol dari setiap penelitian terletak pada *dataset* dan fitur-fitur yang digunakan. Penelitian-penelitian sebelumnya akan dijadikan sebagai rujukan ketika pelaksanaan penelitian ini.

Penelitian – penelitian yang dimaksud antara lain verifikasi tanda tangan menggunakan ekstraksi fitur *Edge Detection* dengan metode *prewitt* dan klasifikasi dengan metode LVQ dimana *dataset* berjumlah 400 yaitu 300 tanda tangan asli dan 100 tanda tangan palsu dengan melakukan pemindaian terhadap kertas tanda tangan sehingga dilakukan proses *preprocessing* yaitu yaitu *resize*, *grayscale*, binerisasi, *thinning*, dan *cropping*. pada penelitian ini digunakan data latih berjumlah 200 dan data uji berjumlah masing-masing 100 dengan akurasi sebesar 70%. Pada penelitian ini, citra uji dianggap tanda tangan asli jika bobot citra uji mendekati bobot target sedangkan jika bobotnya terlalu jauh dengan bobot target maka dinyatakan palsu atau tidak cocok[5]. Pada penelitian lainnya yaitu tentang verifikasi tanda tangan berdasarkan sifat keacakan (entropi), terdapat 30 responden dimana setiap responden diambil 30 tanda tangan asli dan 5 tanda tangan palsu dengan melakukan pemindaian terhadap kertas tanda tangan sehingga dilakukan proses *preprocessing* yaitu binerisasi. Penelitian ini menghasilkan akurasi sebesar 96,67%[13]. Serta penelitian lainnya yaitu tentang verifikasi tanda tangan berbasis perceptron dimana data berjumlah 42 citra dari 2 responden dengan melakukan pemindaian terhadap kertas tanda tangan sehingga dilakukan proses *preprocessing* yaitu *cropping*, penskalaan, *grey-scaling*, dan binerisasi. Proses verifikasi dilakukan dengan memberi nilai 1 untuk menunjukkan bahwa citra tersebut merupakan citra tanda tangan asli dan nilai -1 menunjukkan bahwa citra tersebut merupakan citra dengan tanda tangan palsu, akurasi yang diperoleh sebesar 92,85%[14].

Ekstraksi ciri menggunakan metode LBP telah dilakukan antara lain untuk identifikasi ciri telapak tangan dimana metode KNN digunakan untuk klasifikasi. Data yang digunakan berjumlah 78 dan hasil terbaik dari metode LBP terdapat pada parameter jarak tetangga(R) = 2, jumlah tetangga(P) = 8 yaitu terdiri dari 16 *region* dimana pembagian nilai *height* = 4 dan *width* = 4 sehingga akurasi mencapai 92,31%[15]. Pada penelitian lainnya tentang analisis citra daun digunakan 10 fitur LBP yaitu *Mean*, *Standar Deviasi*, *Skewness*, *Entropy*, *Kurtosis*, *Variance*, *Contrast*, *Correlation*, *Energy* dan *Homogeneity*, penelitian ini menggunakan metode KNN untuk klasifikasi, akurasi tertinggi diperoleh dari ekstraksi fitur LBP yaitu 94.28%[16], pada penelitian untuk mengidentifikasi tulisan karakter jawa dimana pada penelitian ini digunakan metode klasifikasi KNN. Akurasi tertinggi diperoleh dengan menggunakan parameter [64 64] dengan $k=3$ [7]. Dan metode LBP juga digunakan untuk mengidentifikasi cacat kayu dimana akurasi tertinggi mencapai 89% dengan menggunakan pola ketetanggaan LBP yaitu [4,1][17].

Ekstraksi fitur dengan metode DCT juga telah dilakukan antara lain untuk mengidentifikasi daun tembakau dimana *dataset* yang digunakan berjumlah 40 citra dengan 30 citra sebagai data latih dan 10 citra sebagai data uji. Metode klasifikasi yang digunakan yaitu *backpropogation*. Penelitian ini menghasilkan akurasi sebesar 90%[18]. selain itu, metode DCT juga digunakan untuk pengenalan iris dimana Total data yang digunakan adalah 420 citra iris yang kemudian dibagi menjadi 360 citra latih dan 60 citra uji yang terdiri dari 30 kelas. Pada penelitian ini, tahap pengujian dilakukan terhadap 60 citra iris kiri dan kanan. Nilai K yang digunakan adalah 3, 5 dan 9, sedangkan disctance yang digunakan adalah Euclidean Distance. Metode klasifikasi yang digunakan yaitu KNN. pada penelitian ini, akurasi tertinggi yaitu 71,67% pada $K = 9$ untuk kelas iris 20[19].

Teknik klasifikasi menggunakan metode LVQ juga telah dilakukan antara lain untuk identifikasi tingkat autisme dimana hasil akurasi terbaik dengan menggunakan algoritme *pattern matching* yaitu algoritme *city block* untuk mengidentifikasi orang yang tidak autis yaitu 100% dan orang yang autis ringan yaitu 96%, algoritme *mahattan distance* menghasilkan akurasi terbaik untuk

identifikasi autisme berat yaitu 100% dan sedang yaitu 96%[20]. Penelitian lainnya yaitu tentang identifikasi kesehatan susu sapi dengan ekstraksi fitur metode *watershed*. Pada penelitian ini data latih terdiri dari 20 citra untuk setiap kelasnya yaitu susu sapi murni, susu sapi campuran air 25%, 50% dan 75%, dan dengan total keseluruhan 120 data. Sedangkan data uji terdiri dari 10 citra susu pada setiap kelasnya, dengan total keseluruhan 30 data sehingga dihasilkan akurasi yaitu 92,5%[21]. Serta penelitian mengenai klasifikasi citra daging sapi dan daging babi. Hasil analisis parameter untuk pelatihan LVQ yaitu akurasi tertinggi dengan menggunakan *MaxEpoch* 100, α sebesar 0.01 sebesar 76,25%[22].

Penelitian tentang pengenalan ciri menggunakan kombinasi metode LBP dan klasifikasi LVQ sebelumnya sudah pernah dilakukan beberapa kali. Penelitian-penelitian yang dimaksud antara lain identifikasi sub-fosil gigi geraham pada manusia Digunakan 270 citra gigi geraham yang terbagi menjadi 140 citra latih dan 130 citra uji yang digunakan untuk data pada rentang umur kematian dari sub-fosil manusia. Lalu terdapat 140 citra latih dan 110 citra uji untuk jenis gigi graham. Dalam mengidentifikasi jenis gigi geraham pada manusia didapatkan akurasi sebesar 81,81% dan akurasi sebesar 77,69% dalam mengidentifikasi rentang umur kematian dari sub-fosil manusia[23]. penelitian lainnya yaitu tentang deteksi penyakit diabetes retinotopi pada retina mata dengan menggunakan *pre-processing* yaitu *resize* dan *grayscale* dan akurasi yang diperoleh yaitu sebesar 85%[24]. Selain itu, terdapat penelitian tentang pengenalan iris. Pada penelitian ini dilakukan studi banding yaitu dengan kombinasi metode LBP dan LVQ, kombinasi metode LBP, histogram dan LVQ, serta kombinasi metode LBP, histogram dan LVQ dengan $n=3$. Akurasi tertinggi diperoleh dari kombinasi metode LBP dan LVQ [25].

Penelitian tentang pengenalan citra menggunakan metode ekstraksi fitur DCT dan klasifikasi LVQ juga sudah pernah dilakukan yaitu penelitian mengenai deteksi kanker serviks dimana *dataset* yang digunakan yaitu Citra sel serviks berjumlah 73 buah yang terdiri atas lima puluh buah citra sel normal dan 23 buah citra sel kanker. Proses pelatihan LVQ menggunakan 35 buah citra sel normal dan

empat belas buah citra sel kanker. Proses pengujian LVQ menggunakan 15 buah citra sel normal dan sembilan buah citra sel. Penelitian ini menggunakan Preprocessing yaitu *cropping* dan *grayscale*. Dari hasil pengujian, didapatkan nilai akurasi, *sensitivity* dan *specificity*, dan akurasi sebesar 88,89 %, 100 %, dan 95,83 %[26].

Berdasarkan penelitian-penelitian tersebut, dapat diketahui bahwa metode ekstraksi fitur LBP dan klasifikasi menggunakan metode LVQ dapat bekerja dengan baik untuk pengklasifikasian citra. Oleh karena itu, penulis bermaksud untuk menggunakan metode ini untuk melakukan verifikasi citra tanda tangan.

2.2 Dasar Teori

2.2.1 Biometrik tanda tangan

Secara umum, karakteristik manusia dapat dikelompokkan menjadi dua, yaitu karakteristik fisiologis atau fisik (*physiological/physical characteristic*) dan karakteristik perilaku (*behavioral characteristic*). Biometrika berdasarkan karakteristik fisiologis atau fisik menggunakan bagian-bagian fisik dari tubuh sebagai kode unik untuk pengenalan, seperti DNA, telinga, jejak panas pada wajah, geometri tangan, retina, gigi, telinga, dan bau (komposisi kimia) dari keringat tubuh. Untuk biometrika berdasarkan karakteristik perilaku menggunakan perilaku seseorang sebagai kode unik untuk melakukan pengenalan, seperti gaya berjalan, hentakan tombol, dan tanda tangan. Secara umum, terdapat dua model sistem biometrika, yaitu Sistem Verifikasi (*Verification System*) dan Sistem Identifikasi (*Identification System*). Sistem identifikasi bertujuan untuk memecahkan identitas seseorang, sedangkan sistem verifikasi bertujuan untuk menerima atau menolak identitas yang diklaim oleh seseorang[27].

2.2.2 Verifikasi tanda tangan

Verifikasi tanda tangan adalah suatu proses yang digunakan untuk mengenal tanda tangan seseorang. Sistem verifikasi tanda tangan dapat dibagi menjadi dua bagian yaitu verifikasi tanda tangan secara *off-line* dan verifikasi tanda tangan secara *on-line*[28]. Verifikasi tanda tangan secara *offline* mengambil sebuah *image* tanda tangan sebagai *input* yang akan digunakan dalam proses selanjutnya.

Sedangkan *input* untuk verifikasi tanda tangan secara *on-line* diambil dari tanda tangan yang didapatkan langsung dari *digitizer* yang dapat menghasilkan nilai – nilai dinamik, seperti nilai koordinat, lama tanda tangan, dan kecepatan tanda tangan[29].

2.2.3 Citra digital

Citra digital merupakan representasi dari citra analog *continue* yang diubah ke dalam bentuk diskret. Suatu citra didefinisikan sebagai fungsi dua dimensi $f(x, y)$, dimana x dan y adalah koordinat spasial, dan amplitudo dari f pada setiap sembarang pasangan koordinat (x, y) disebut sebagai intensitas atau gray level di level tertentu. Ketika (x, y) dan nilai intensitas dari semua f terbatas dan nilainya diskrit maka disebut sebagai citra digital[30].

2.2.4 Pre-processing *resize*

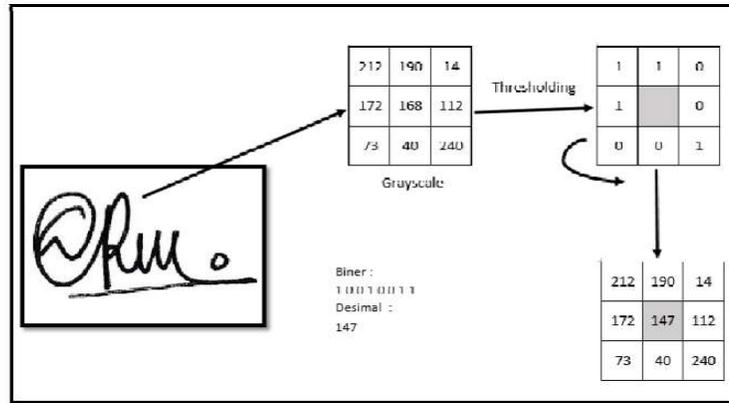
Ukuran dimensi citra pada data yang digunakan cukup besar sehingga pada penelitian ini hanya dilakukan satu tahap *preprocessing* yaitu *resize*. *Resize* bertujuan untuk menyeragamkan ukuran seluruh citra yang ada pada *database*[31].

2.2.5 Ekstraksi fitur LBP

Algoritma LBP adalah suatu metode yang digunakan untuk mendeskripsikan ciri tekstur pada objek citra *grayscale*. LBP digunakan untuk mencari pola-pola tekstur lokal pada citra (*texture in local neighborhood*)[16]. Operator LBP merupakan deskriptor tekstur yang menggunakan perbandingan nilai keabuan dari piksel-piksel ketetanggaan. Operator dasar LBP berukuran 3×3 menggunakan 8 piksel ketetanggaan dari nilai tengah. Piksel ketetanggaan ke- n tersebut di-*threshold* menggunakan nilai keabuan dari piksel dan fungsi *thresholding*. Kode *binary* hasil operator LBP piksel ketetanggaan akan digunakan untuk merepresentasikan fitur dari piksel tengah[8].

Gambar 2.1 merupakan ilustrasi dari proses LBP. Proses pertama adalah piksel ketetanggaan di-*threshold* menggunakan nilai keabuan dari piksel tengah sebagai fungsi *thresholding* jika hasilnya lebih besar atau sama dengan 0 maka diberi nilai 1 dan jika hasilnya kurang dari 0 maka diberi 0. Setelah itu, nilai biner piksel ketetanggaan akan disusun berlawanan arah jarum jam dan 8 bit biner

tersebut dikonversi ke dalam nilai desimal untuk menggantikan nilai piksel tengah[8].



Gambar 2.1 Ilustrasi LBP[8].

Pada penelitian sebelumnya tentang identifikasi sub fosil gigi manusia, digunakan 3 fitur LBP yaitu *mean*, *variance*, dan *entropy* dimana akurasi yaitu 77,69%[23]. Penelitian lain mengenai identifikasi telapak tangan manusia, seluruh pixel LBP dijadikan fitur dimana akurasi diperoleh sebesar 92,31%[15]. Selain itu penelitian mengenai analisis daun menggunakan 10 fitur ciri dalam ekstraksi fitur LBP antara lain akan menghitung nilai *Mean*, *Standar Deviasi*, *Skewness*, *Entropy*, *Kurtosis*, *Variance*, *Contrast*, *Correlation*, *Energy* dan *Homogeneity* dimana akurasi yang diperoleh sebesar 94%[16]. Sehingga pada penelitian ini, peneliti menggunakan 10 fitur LBP yaitu *Mean*, *Standar Deviasi*, *Skewness*, *Entropy*, *Kurtosis*, *Variance*, *Contrast*, *Correlation*, *Energy* dan *Homogeneity* karena akurasi penelitian sebelumnya yang tinggi dan untuk mempercepat proses pembelajaran dan klasifikasi yang menggunakan ANN yaitu LVQ. Berikut adalah penjelasan dari 10 ciri fitur yang digunakan pada fitur LBP[16].

1. *Mean* adalah rata-rata dari suatu sebaran nilai intensitas citra keabuan

$$\mu = \sum_n f_n P(f_n) \quad (2-1)$$

Dimana f_n merupakan suatu nilai intensitas keabuan, sementara $P(f_n)$ menunjukkan nilai histogram (probabilitas kemunculan intensitas tersebut pada citra)

2. Standar Deviasi: Menunjukkan sebaran nilai piksel pada bidang citra *grayscale*. Standar Deviasi merupakan cerminan dari rata-rata penyimpangan data dari *mean*[16]. Dimana L merupakan panjang maximum citra *greyscale*, I_k merupakan nilai intensitas pixel, $E(Y)$ merupakan nilai *mean*. $pdf(I_k)$ merupakan nilai probabilitas kemunculan intensitas citra [32].

$$\text{std} = \sqrt{\sum_{k=0}^{L-1} (I_k - E(Y))^2 \times pdf(I_k)} \quad (2-2)$$

3. *Skewness*: Menunjukkan tingkat kemiringan relatif kurva histogram dari suatu citra. σ adalah standar deviasi.

$$\alpha_3 = \frac{1}{\sigma^3} \sum_n (f_n - \mu)^3 P(f_n) \quad (2-3)$$

4. *Entropy*: Menunjukkan ukuran ketidakteraturan bentuk dari suatu citra.

$$H = - \sum_n P(f_n)^2 \log P(f_n) \quad (2-4)$$

5. *Kurtosis*: Menunjukkan tingkat keruncingan relatif kurva histogram dari suatu citra.

$$\alpha_4 = \frac{1}{\sigma^4} \sum_n (f_n - \mu)^4 P(f_n) - 3 \quad (2-5)$$

6. *Variance*: Untuk mengetahui variasi elemen pada histogram dari citra.

$$\sigma^2 = \sum_n (f_n - \mu)^2 P(f_n) \quad (2-6)$$

7. *Contrast*: Menghitung ukuran penyebaran elemen-elemen matriks citra. Jika posisi letaknya jauh dari diagonal utama, maka nilai kekontrasan yang dihasilkan besar. Secara visual, nilai kekontrasan merupakan ukuran variasi antar derajat keabuan suatu daerah citra.

$$CON = \sum_k K^2 [\sum_i \sum_j P(i, j)] \quad (2-7)$$

$$|i - j| = k$$

8. *Correlation*: Menunjukkan ukuran ketergantungan linear derajat keabuan citra sehingga dapat memberikan petunjuk adanya struktur linear dalam citra.

$$COR = \frac{\sum_i \sum_j (i, j) \cdot p(i, j) - \mu_x \mu_y}{\sigma_x \sigma_y} \quad (2-8)$$

9. *Energy* menyatakan ukuran konsentrasi pasangan dengan intensitas keabuan tertentu pada matriks, dimana (i, j) menyatakan nilai pada baris i dan kolom j pada matriks ko-okurensi.

$$Energy = \sum_{i,j=0}^{n-1} P(i,j)^2 \quad (2-9)$$

i dan j adalah sifat keabuan dari resolusi 2 piksel yang berdekatan $P(i,j)$ adalah *Co-occurrence Matrix* Simetris Ternormalisasi.

10. *Homogeneity*: homogenitas yang menunjukkan kehomogenan citra berderajat keabuan sejenis. Citra homogen akan memiliki nilai homogenitas yang besar.

$$H = \sum_{i,j=0}^{n-1} P(i,j)/[1 + (i,j)^2] \quad (2-10)$$

i dan j adalah sifat keabuan dari resolusi 2 piksel yang berdekatan $P(i,j)$ adalah probabilitas kolom (i,j) .

2.2.6 Ekstraksi Fitur DCT

DCT telah banyak digunakan dalam literatur untuk efisiensi pemilihan fitur tekstur. DCT menggunakan cosinus dari berbagai frekuensi spasial sebagai fungsi dasar dan umumnya dikenal karena penerapannya dalam standar kompresi JPEG seperti pada penelitian [33] berhasil dilakukan kompresi citra secara maksimal dengan metode DCT dan algoritma Huffman pada proses kuantisasi, kemudian DCT juga digunakan sebagai *preprocessing* seperti pada penelitian mengenai identifikasi fitur wajah, dimana DCT memperbaiki kekurangan pada metode LDA dan PCA yang digunakan sebagai ekstraksi fitur[9], dan metode DCT digunakan sebagai ekstraksi fitur dan menghasilkan akurasi yang tinggi seperti pada penelitian mengenai identifikasi tembakau[18]. Pada penelitian DCT yang menjadi acuan yaitu DCT yang digunakan sebagai ekstraksi fitur tekstur. Untuk ekstraksi fitur, Sebagian besar energi sinyal terletak pada komponen frekuensi rendah, yang mana muncul di sudut kiri atas DCT. DCT mengubah informasi spasial ke dalam domain frekuensi, fitur tekstur dapat didefinisikan sebagai spektrum energi di berbagai lokasi blok lokal. Karena koefisien DC mewakili nilai rata-rata skala abu-abu dari setiap blok $N \times N$, maka informasi tekstur yang didapat kurang spesifik detail tekstur yang diekstraksi. DCT merupakan metode berbasis penampilan lokal yang memiliki sifat-sifat seperti de-korelasi, Keterpisahan, pemadatan energi, simetris, Orthogonal. Komponen frekuensi rendah tersimpan di sudut kiri atas. Komponen-komponen ini dapat diekstraksi untuk pengklasifikasian citra.

Penelitian ini mengekstraksi fitur tekstur tanda tangan dengan mengaplikasikan DCT dimana derajat keabuan setiap piksel dalam citra hasil ekstraksi LBP dinormalisasi menjadi 128 melalui operasi *subtraction*. Kemudian, citra dibagi menjadi 8 x 8 blok, dan setiap blok diproses dengan operasi 2D-DCT seperti pada persamaan (2-11).

$$C(u, v) = a(u)a(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cos \left[\frac{(2x+1)\pi u}{2N} \right] \cos \left[\frac{(2y+1)\pi v}{2N} \right] \quad (2-11)$$

Selama proses DCT berlangsung, dihitung pula invers dari 2D-DCT dengan menggunakan persamaan (2-12)[18].

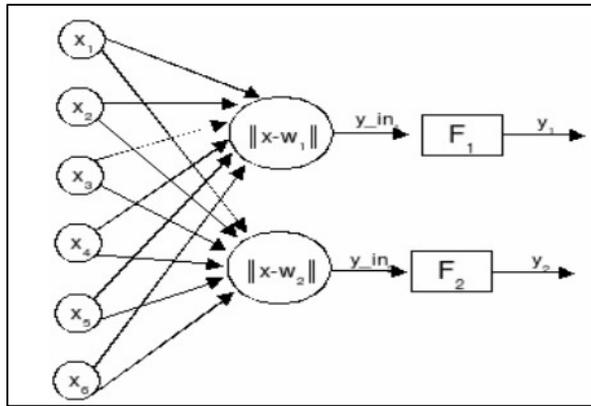
$$f(x, y) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cos \left[\frac{(2x+1)\pi u}{2N} \right] \cos \left[\frac{(2y+1)\pi v}{2N} \right] \quad (2-12)$$

Matriks hasil transformasi DCT yang telah terisi dengan koefisien DCT selanjutnya diproses pada tahap kuantisasi, dimana data yang terletak pada kiri atas merupakan korelasi dari frekuensi-frekuensi rendah dari data asli. Sedangkan yang terletak pada kanan bawah merupakan korelasi dari frekuensi-frekuensi tinggi dari data asli.

Pada tahap kuantisasi, digunakan algoritma *Huffman* yaitu dengan Menyusun bilangan menggunakan fungsi zigzag *scanning*[33]. Proses ini bertujuan untuk menentukan fitur terbaik dari citra yang selanjutnya akan diproses pada tahap klasifikasi, dimana jumlah fitur yang diambil sejumlah nilai koefisien yang digunakan.

2.2.7 Klasifikasi dengan LVQ

LVQ merupakan jaringan syaraf dengan tipe arsitektur jaringan lapis-tunggal umpan-maju (*Single Layer Feedforward*) yang terdiri atas unit masukan dan unit keluaran. Suatu lapisan kompetitif akan secara otomatis belajar untuk mengklasifikasikan vektor-vektor masukan. Kelas-kelas yang didapatkan sebagai hasil dari lapisan kompetitif ini hanya tergantung pada jarak antara vektor - vektor masukan. Jika 2 vektor masukan mendekati sama, maka lapisan kompetitif akan meletakkan kedua vektor masukan tersebut ke dalam kelas yang sama. Arsitektur jaringan LVQ seperti terlihat pada Gambar 2.2.



Gambar 2.2 *Learning Vector Quantization*[34].

Gambar 2.2 memperlihatkan bahwa yang bertindak sebagai dendrit atau data masukan adalah $X_1 - X_6$, yang bertindak sebagai sinapsis atau bobot adalah W , sedangkan soma atau badan sel dari jaringan ini adalah perhitungan $X - W$. Dan yang bertindak sebagai akson atau data keluaran adalah Y [34]. Algoritma LVQ adalah sebagai berikut[35]:

1. Tentukan maksimum Epoch (banyaknya proses pelatihan yang akan diulangi), eps (error minimum yang diharapkan) dan nilai alpha.
2. Hasil ekstraksi ciri pertama dari masing-masing pola digunakan sebagai data awal (inisialisasi). Data inisialisasi ini akan diisi sebagai nilai bobot awal (w).
3. Epoch = 0
4. Selama (Epoch < *MaxEpoch*) atau (alpha > eps), maka lakukan hal berikut:
 - a. Epoch = Epoch+1
 - b. Untuk setiap data hasil ekstraksi ciri, lakukan hal berikut:
 - 1) Set x = hasil ekstraksi ciri dari pola.
 - 2) Set T = nomor urut dari setiap kelas
 - 3) Hitung jarak hasil ekstraksi ciri pola saat ini dengan masing-masing bobot. Misalkan dihitung jarak hasil ekstraksi ciri pola pertama dengan setiap bobot dapat menggunakan persamaan (2-11).

$$J = \sqrt{(x_{11} - w_{11})^2 + \dots + (x_{1m} - w_{1m})^2} \quad (2-13)$$

Dengan:

x_{1m} = bit ekstraksi ciri dari pola-1 yang ke-m.

w_{lm} = bobot $W_{(l,m)}$

m = banyak bit ekstraksi ciri

- 4) Bila nomor kelas pada bobot yang memiliki jarak terkecil sama dengan nilai nomor urut (T) pola, maka hitung:

$$W_j(\text{baru}) = W_j(\text{lama}) + \alpha(x - W_j(\text{lama})) \quad (2-14)$$

- 5) Bila tidak, maka hitung:

$$W_j(\text{baru}) = W_j(\text{lama}) - \alpha(x - W_j(\text{lama})) \quad (2-15)$$

- c. kurangi nilai α .

$$\alpha = \alpha - (0.1 * \alpha) \quad (2-16)$$

2.2.8 Proses verifikasi tanda tangan

Pada proses pengujian, verifikasi tanda tangan dilakukan setelah sistem melakukan pengenalan tanda tangan. Verifikasi digunakan untuk membedakan tanda tangan asli dan tanda tangan palsu. Pada penelitian ini, digunakan rumus *Euclidean Distance* untuk menghitung kemiripan atau *similarity* data uji dengan nilai rata-rata fitur kelasnya. Pada penelitian sebelumnya, teknik *Euclidean Distance* menghasilkan akurasi lebih baik dibandingkan teknik *Canberra distance*[36], *Minkowski Distance* dan *Manhattan Distance*[37] dalam menghitung kemiripan dua vektor. Rumus *Euclidean Distance* setiap index vektor dapat dihitung dengan persamaan (2-17) dimana pada penelitian ini, x merupakan index vektor data yang diuji y merupakan vektor nilai rata-rata fitur kelasnya yang telah diprediksi pada proses klasifikasi.

$$d(x, y) = \sqrt{(x_i - y_i)^2} \quad (2-17)$$

Setelah itu, dilakukan normalisasi terhadap setiap index *array*. Persamaan (2-18) digunakan untuk melakukan normalisasi terhadap setiap index *array*, dimana x merupakan nilai index *array*, $nilai_{min}$ merupakan selisih terkecil dari semua index pada *array*, dan $nilai_{max}$ merupakan selisih terbesar dari semua index pada *array*.

$$x' = \frac{x - nilai_{min}}{nilai_{max} - nilai_{min}} \quad (2-18)$$

Setelah dilakukan normalisasi, maka dicari nilai rata-rata *array* dengan menggunakan persamaan (2-19) dimana x_i merupakan. Tanda tangan asli dan palsu ditentukan berdasarkan batas rentang nilai *variabel threshold*.

$$x = \frac{\sum_0^n x_i}{n} \quad (2-19)$$

2.2.9 Evaluasi Hasil Klasifikasi

Evaluasi hasil klasifikasi dapat dilakukan dengan menghitung nilai *True Positive*, *True Negative*, *False Positive* dan *False Negative*. Keempat nilai ini dapat dihitung berdasarkan *confusion matrix* seperti yang tertera Tabel 2.1[38]. Nilai-nilai yang didapatkan dari *confusion matrix* kemudian akan digunakan untuk menghitung parameter-parameter evaluasi hasil klasifikasi.

Tabel 2.1 *Confusion matrix*.

Hasil Klasifikasi \ Kelas Sebenarnya	Positif	Negatif
Positif	a	B
Negatif	c	D

Accuracy merupakan proporsi dari kedua kelas (positif dan negatif) dari total kelas yang diuji. Persamaan (2-20) menunjukkan perhitungan untuk nilai akurasi.

$$Accuracy = \frac{a+d}{a+b+c+d} \quad (2-20)$$

True Positive Rate (recall) merupakan proporsi dari kelas positif yang diklasifikasikan dengan benar. Persamaan (2-21) menunjukkan perhitungan untuk nilai *True Positive Rate*.

$$True Positive Rate (sensitivity) = \frac{a}{a+b} \quad (2-21)$$

True Negative Rate (specificity) merupakan proporsi dari kelas negatif yang diklasifikasikan dengan benar. Persamaan (2-22) menunjukkan perhitungan untuk nilai *True Negative Rate*.

$$True Negative Rate (specificity) = \frac{d}{c+d} \quad (2-22)$$

BAB III

METODOLOGI PENELITIAN

3.1 Alat dan bahan

Alat dan bahan pada penelitian yang dilakukan berupa *software* dan *hardware* serta data-data yang dibutuhkan selama kegiatan penelitian.

1. Alat Penelitian

Alat – alat yang digunakan dalam melakukan penelitian ini adalah sebagai berikut.

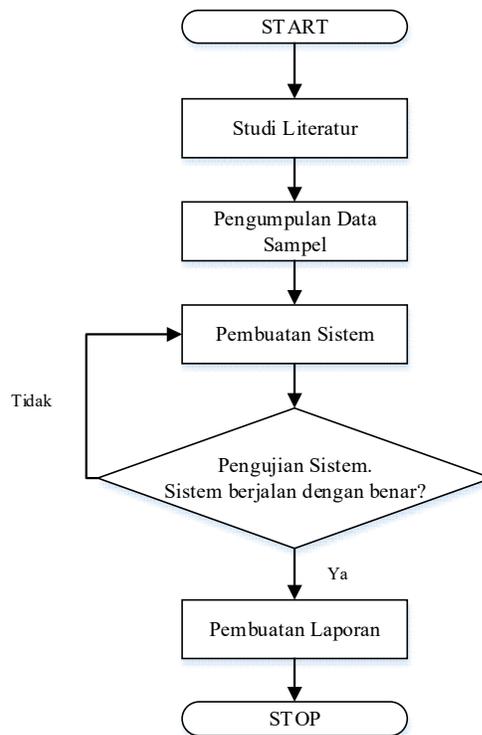
- Processor Laptop: Intel Core i3-5005U, 2.0GHz, RAM 4GB
- Operating System: Windows 10 pro 64-bit.
- *Software* Jupyter Notebook (Anaconda 3), python 3.7 diperlukan untuk membuat program
- Software iBisPaint X diperlukan sebagai media untuk mengumpulkan *dataset* tanda tangan
- Samsung Galaxy Tab A

2. Bahan Penelitian

Bahan – bahan yang akan digunakan dalam penelitian ini adalah citra tanda tangan sebanyak 600 citra dengan format PNG dan ukuran 500x500 pixel yang diambil dari 30 responden dimana tiap responden diambil 15 tanda tangan asli yang diambil pada 3 hari yang berbeda dimana setiap 1 hari diambil 5 tanda tangan asli dan untuk tanda tangan palsu diambil 5 tanda tangan untuk setiap responden, dimana tanda tangan palsu dikerjakan oleh 5 orang yang dipercaya peneliti dapat meniru tanda tangan

3.2 Rencana Penelitian

Diagram alir pembuatan sistem dari mulai pengumpulan data hingga pembuatan laporan tertuang dalam Gambar 3.1.

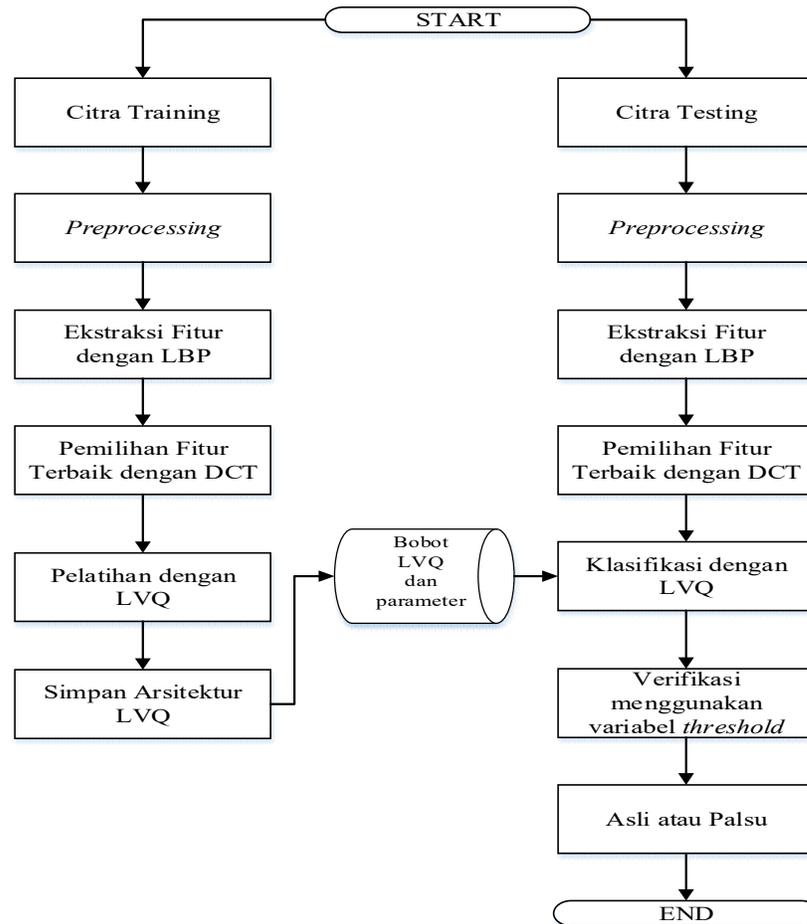


Gambar 3.1 Diagram alir proses penelitian.

Berdasarkan Gambar 3.1, ada beberapa tahapan yang dilakukan dalam penelitian ini, tahap pertama dilakukan studi literatur untuk menambah wawasan penulis mengenai apa yang akan dilakukan. Studi literatur ini dilakukan melalui referensi-referensi yang berasal dari jurnal, buku, dan penelitian-penelitian yang pernah dilakukan sebelumnya. Selanjutnya, dilakukan pengumpulan *dataset* yang akan digunakan sebagai data latih dan data uji. Setelah *dataset* berhasil dikumpulkan maka penelitian akan diteruskan dengan membangun model menggunakan metode ekstraksi fitur LBP dan klasifikasi LVQ. Hasil dari model ini akan menjadi bahan analisis untuk mengetahui performa dari model yang dibangun. Jika model masih memiliki akurasi yang kurang memuaskan maka penelitian akan diulang kembali dari tahap studi literatur. Jika sudah mendapat hasil sesuai maka akan diambil kesimpulan mengenai performa dari model. Terakhir, dilakukan penyelesaian dokumentasi penelitian dalam bentuk laporan akhir

3.3 Perancangan sistem

Di dalam subbab ini dijelaskan bagaimana sistem dirancang mulai dari tahapan pelatihan citra hingga sistem dapat mengetahui akurasi optimal dari citra latih dan citra uji. Block diagram tertuang dalam Gambar 3.2.



Gambar 3.2 Diagram perancangan sistem.

Seperti terlihat pada Gambar 3.2, terdapat dua proses utama dalam penelitian ini, yaitu proses pelatihan dan proses pengujian. *Preprocessing* dan ekstraksi fitur yang dilakukan pada proses pelatihan dan proses pengujian sama. Proses pelatihan dan proses pengujian tersebut dijelaskan sebagai berikut.

3.3.1 Proses pelatihan

Proses pelatihan sistem meliputi tahap-tahap berikut ini:

- a. *Input* citra tanda tangan untuk pelatihan, citra yang dimasukkan ke dalam sistem merupakan citra hitam putih dan berukuran 500×500 .
- b. Tahap *preprocessing* merupakan proses manipulasi gambar sesuai dengan yang diinginkan dengan melakukan *resize*, untuk proses *resize* ukuran gambar diubah menjadi ukuran 64×64 , 128×128 dan 256×256 piksel. selanjutnya citra diubah menjadi bentuk *grayscale* yang digunakan dalam proses ekstraksi.
- c. Ekstraksi fitur pada penelitian ini menggunakan metode LBP yang akan digunakan untuk mendapatkan nilai 10 fitur LBP yaitu *Mean*, *Standar Deviasi*, *Skewness*, *Entropy*, *Kurtosis*, *Variance*, *Contrast*, *Correlation*, *Energy* dan *Homogeneity*.
- d. Pelatihan data citra dengan menggunakan klasifikasi LVQ dengan nilai *maximal Epoch*, dan *learning rate*.

3.3.2 Proses pengujian

Proses pengujian sistem meliputi tahap-tahap berikut ini:

- a. *Input* citra tanda tangan untuk pengujian, citra yang dimasukkan ke dalam sistem merupakan citra hitam putih dan berukuran 500×500 .
- b. Tahap *preprocessing* merupakan proses manipulasi gambar sesuai dengan yang diinginkan dengan melakukan *resize*, untuk proses *resize* ukuran gambar diubah menjadi ukuran 64×64 , 128×128 , dan 256×256 piksel. selanjutnya citra diubah menjadi bentuk *grayscale* yang digunakan dalam proses ekstraksi.
- c. Ekstraksi fitur pada penelitian ini menggunakan metode LBP yang akan digunakan untuk mendapatkan nilai 10 fitur LBP yaitu *Mean*, *Standar Deviasi*, *Skewness*, *Entropy*, *Kurtosis*, *Variance*, *Contrast*, *Correlation*, *Energy* dan *Homogeneity*.
- e. Klasifikasi citra pelatihan menggunakan metode LVQ menggunakan bobot akhir dari hasil proses pelatihan.
- f. Dilakukan pengecekan terhadap hasil akurasi, apabila hasil akurasi lebih rendah dari 50% maka hasil ekstraksi fitur LBP akan diolah dengan metode DCT

dimana metode DCT digunakan untuk mengambil fitur terbaik dari citra hasil ekstraksi fitur LBP. Kemudian, hasil DCT inilah yang akan diklasifikasikan dengan metode LVQ.

- d. Tahap verifikasi dilakukan dengan menghitung *similarity* antara fitur-fitur citra latih dengan kelasnya. Citra tanda tangan asli atau palsu ditentukan berdasarkan nilai variabel *threshold* yang ditentukan pada saat penelitian dilakukan.
- e. Output berupa akurasi optimal.

3.4 Preprocessing

Pengolahan awal (*preprocessing*) bertujuan untuk mengelola citra agar dapat diperoleh informasi dari suatu citra secara optimal, tahap pengolahan awal citra dalam penelitian ini adalah melakukan *resize*. Pada tahap ini citra sudah tersimpan di dalam komputer, citra hasil *resize* ini yang nantinya akan dijadikan sebagai data latih dan data uji.

Pada tahap *resize*, dilakukan perubahan ukuran citra agar setiap *input* citra memiliki resolusi yang sama. Resolusi pada citra awal adalah 500×500 . Pada saat *preprocessing* citra di *resize* menjadi 64×64 , 128×128 , dan 256×256 piksel. Semakin tinggi resolusinya, yang berarti semakin kecil ukuran pixel atau semakin banyak jumlah pixelnya, semakin halus citra yang diperoleh dan juga bertujuan untuk mempercepat proses komputasi.

3.5 Ekstraksi fitur

Proses ekstraksi fitur dengan metode LBP dibagi menjadi dua tahap, yaitu pembentukan matriks LBP dan perhitungan nilai fitur LBP sampai mencari nilai 10 fitur.

3.5.1 Pembentukan matriks LBP

Algoritma LBP adalah suatu metode yang digunakan untuk mendeskripsikan ciri tekstur pada objek citra *grayscale*. Pada *dataset* citra tanda tangan ini, citra awal sudah berupa *grayscale*. LBP digunakan untuk mencari pola-pola tekstur lokal pada citra. Jenis LBP yang digunakan dalam penelitian ini yaitu LBP dengan 8 titik tetangga dengan jari-jari atau jarak titik dengan tetangga yaitu 1. Misalkan terdapat citra dengan matriks 5×5 pada Gambar 3.3.

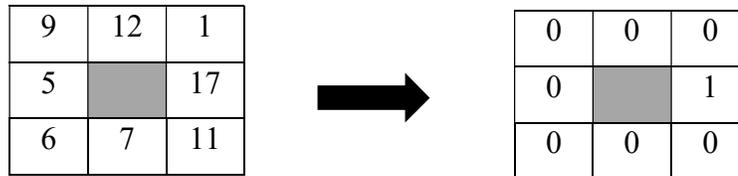
A =

9	12	1	4	2
5	15	17	10	6
6	7	11	22	21
20	8	6	5	14
17	22	10	4	4

Gambar 3.3 Contoh matriks

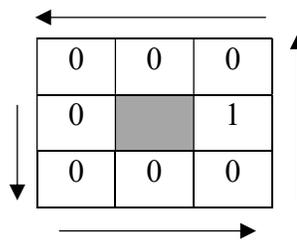
Adapun langkah-langkah pembentukan matriks LBP sebagai berikut:

- Membagi matriks citra menjadi bagian-bagian yang lebih kecil atau disebut kernel yaitu matriks 3×3 . Bandingkan nilai piksel pada titik pusat dengan nilai piksel pada 8 titik tetangga. Jika nilai piksel tetangga lebih besar atau sama dengan dari titik pusat maka diberi nilai 1, jika nilai piksel tetangga lebih kecil dari titik pusat maka diberi nilai 0 seperti pada Gambar 3.4.



Gambar 3.4 Konversi nilai piksel ke biner.

- Nilai disusun searah dengan jarum jam sehingga menjadi bilangan biner 8 bit seperti pada Gambar 3.5.



Gambar 3.5 Membentuk bilangan biner 8 bit.

- Konversikan bilangan biner ke desimal dengan mengalikan nilai biner dengan bilangan eksponensial dari 2. Nilai desimal yang diperoleh akan menjadi nilai LBP pada satu kernel.

$$00001000 = 0 \times 128 + 0 \times 64 + 0 \times 32 + 0 \times 16 + 1 \times 8 + 0 \times 4 + 0 \times 2 + 0 \times 1 = 8$$

Sehingga, nilai LBP nya yaitu 8 dan diletakkan pada kernel seperti pada Gambar 3.6.

	8	

Gambar 3.6 Nilai titik pusat berubah menjadi nilai LBP.

- d. Langkah a-c diulang kembali untuk menghitung nilai LBP pixel lainnya. Dan diperoleh hasil akhir seperti pada Gambar 3.7.

A =

24	48	248	112	224
62	8	16	240	224
60	110	11	0	128
16	244	231	207	3
10	1	128	146	131

Gambar 3.7 Nilai pixel hasil perhitungan LBP.

3.5.2 Perhitungan fitur LBP

Pada penelitian ini digunakan 10 fitur ciri dalam ekstraksi fitur LBP, antara lain akan menghitung nilai *Mean*, *Standar Deviasi*, *Skewness*, *Entropy*, *Kurtosis*, *Variance*, *Contrast*, *Correlation*, *Energy* dan *Homogeneity*.

- a. *Mean*

Untuk menghitung nilai *Mean*, digunakan Persamaan (2-1).

$$\begin{aligned} \mu &= (24 \times \frac{1}{25}) + (48 \times \frac{1}{25}) + (248 \times \frac{1}{25}) + (112 \times \frac{1}{25}) + (224 \times \frac{2}{25}) + (62 \times \frac{1}{25}) + (8 \times \frac{1}{25}) \\ &\quad + (16 \times \frac{2}{25}) + (240 \times \frac{1}{25}) + (60 \times \frac{1}{25}) + (110 \times \frac{1}{25}) + (11 \times \frac{1}{25}) + (0 \times \frac{1}{25}) + \\ &\quad (128 \times \frac{2}{25}) + (244 \times \frac{1}{25}) + (231 \times \frac{1}{25}) + (207 \times \frac{1}{25}) + (3 \times \frac{1}{25}) + (10 \times \frac{1}{25}) + (1 \times \frac{1}{25}) \\ &\quad + (146 \times \frac{1}{25}) + (131 \times \frac{1}{25}) \\ \mu &= 0,96 + 0,48 + 1,24 + 0,56 + 4,96 + 1,28 + 0,6 + 2,4 + 4,72 + 8,32 + 0 + 0,08 \\ &\quad + 0,32 + 1,88 + 9,72 + 7,68 + 3,2 + 9,64 + 7,72 \\ \mu &= 7,56 \end{aligned}$$

b. Standar Deviasi

Untuk menghitung nilai Standar Deviasi, digunakan Persamaan (2-2).

$$\begin{aligned} std^2 = & (24-75,56)^2 \times \frac{1}{25} + (48-75,56)^2 \times \frac{1}{25} + (248-75,56)^2 \times \frac{1}{25} + (112-75,56)^2 \times \frac{1}{25} + \\ & (224-75,56)^2 \times \frac{2}{25} + (62-75,56)^2 \times \frac{1}{25} + (8-75,56)^2 \times \frac{2}{25} + (16-75,56)^2 \times \frac{2}{25} + \\ & (240-75,56)^2 \times \frac{1}{25} + (224-75,56)^2 \times \frac{2}{25} + (60-75,56)^2 \times \frac{1}{25} + (110- \\ & 75,56)^2 \times \frac{1}{25} + (11-75,56)^2 \times \frac{2}{25} + (0-75,56)^2 \times \frac{2}{25} + (128-75,56)^2 \times \frac{1}{25} + \\ & (244-75,56)^2 \times \frac{1}{25} + (231-75,56)^2 \times \frac{1}{25} + (207-75,56)^2 \times \frac{1}{25} + (3-7,56)^2 \times \frac{1}{25} + \\ & (10-75,56)^2 \times \frac{1}{25} + (1-75,56)^2 \times \frac{1}{25} + (146-75,56)^2 \times \frac{1}{25} + (131-75,56)^2 \times \frac{1}{25} \end{aligned}$$

$$std = \sqrt{736,176}$$

$$std = 85,7915$$

c. *Skewness*

Untuk menghitung nilai *Skewness*, digunakan Persamaan (2-3).

$$\begin{aligned} \alpha_3 = & \frac{1}{85,7915^3} (24-75,56)^3 \times \frac{1}{25} + (48-75,56)^3 \times \frac{1}{25} + (248-75,56)^3 \times \frac{1}{25} + (112- \\ & 75,56)^3 \times \frac{1}{25} + (224-75,56)^3 \times \frac{2}{25} + (62-75,56)^3 \times \frac{1}{25} + (8-75,56)^2 \times \frac{2}{25} + (16- \\ & 75,56)^3 \times \frac{2}{25} + (240-75,56)^3 \times \frac{1}{25} + (224-75,56)^3 \times \frac{2}{25} + (60-75,56)^3 \times \frac{1}{25} + \\ & (110-75,56)^3 \times \frac{1}{25} + (11-75,56)^3 \times \frac{2}{25} + (0-75,56)^3 \times \frac{2}{25} + (128-75,56)^3 \times \frac{2}{25} + \\ & (244-75,56)^3 \times \frac{1}{25} + (231-75,56)^3 \times \frac{1}{25} + (207-75,56)^3 \times \frac{1}{25} + (3-7,56)^3 \times \frac{1}{25} + \\ & (10-75,56)^3 \times \frac{1}{25} + (1-75,56)^2 \times \frac{1}{25} + (146-75,56)^3 \times \frac{1}{25} + (131-75,56)^3 \times \frac{1}{25} \end{aligned}$$

$$\alpha_3 = 4,322$$

d. *Entropy*

Untuk menghitung nilai *Entropy*, digunakan Persamaan (2-4).

$$\begin{aligned} H = & - \left(\left(\frac{1}{25} \right)^2 \times \log \left(\frac{1}{25} \right) \right) + \left(\left(\frac{1}{25} \right)^2 \times \log \left(\frac{1}{25} \right) \right) + \left(\left(\frac{1}{25} \right)^2 \times \log \left(\frac{1}{25} \right) \right) + \left(\left(\frac{1}{25} \right)^2 \times \log \left(\frac{1}{25} \right) \right) \\ & + \left(\left(\frac{2}{25} \right)^2 \times \log \left(\frac{2}{25} \right) \right) + \left(\left(\frac{1}{25} \right)^2 \times \log \left(\frac{1}{25} \right) \right) + \left(\left(\frac{1}{25} \right)^2 \times \log \left(\frac{1}{25} \right) \right) + \left(\left(\frac{2}{25} \right)^2 \times \log \left(\frac{2}{25} \right) \right) \\ & + \left(\left(\frac{1}{25} \right)^2 \times \log \left(\frac{1}{25} \right) \right) + \left(\left(\frac{2}{25} \right)^2 \times \log \left(\frac{2}{25} \right) \right) + \left(\left(\frac{1}{25} \right)^2 \times \log \left(\frac{1}{25} \right) \right) + \left(\left(\frac{1}{25} \right)^2 \times \log \left(\frac{1}{25} \right) \right) + \\ & \left(\left(\frac{1}{25} \right)^2 \times \log \left(\frac{1}{25} \right) \right) + \left(\left(\frac{1}{25} \right)^2 \times \log \left(\frac{1}{25} \right) \right) + \left(\left(\frac{2}{25} \right)^2 \times \log \left(\frac{2}{25} \right) \right) + \left(\left(\frac{1}{25} \right)^2 \times \log \left(\frac{1}{25} \right) \right) + \end{aligned}$$

$$\left(\left(\frac{1}{25}\right)^2 \times \log\left(\frac{1}{25}\right)\right) + \left(\left(\frac{1}{25}\right)^2 \times \log\left(\frac{1}{25}\right)\right) + \left(\left(\frac{1}{25}\right)^2 \times \log\left(\frac{1}{25}\right)\right) + \left(\left(\frac{1}{25}\right)^2 \times \log\left(\frac{1}{25}\right)\right) +$$

$$\left(\left(\frac{1}{25}\right)^2 \times \log\left(\frac{1}{25}\right)\right) + \left(\left(\frac{1}{25}\right)^2 \times \log\left(\frac{1}{25}\right)\right) + \left(\left(\frac{1}{25}\right)^2 \times \log\left(\frac{1}{25}\right)\right)$$

$$H = -(-0,2196)$$

$$H = 0,2196$$

e. Kurtosis

Untuk menghitung nilai kurtosis, digunakan Persamaan (2-5).

$$\alpha_4 = \frac{1}{87,0453^4} \left(\left(\frac{1}{25} \times (24-75,56)^4 - 3\right) + \left(\frac{1}{25} \times (48-75,56)^4 - 3\right) + \left(\frac{1}{25} \times (248-75,56)^4 - 3\right) + \right.$$

$$\left. \left(\frac{1}{25} \times (112-75,56)^4 - 3\right) + \left(\frac{2}{25} \times (224-75,56)^4 - 3\right) + \left(\frac{1}{25} \times (62-75,56)^4 - 3\right) + \left(\frac{2}{25} \times (8-75,56)^4 - 3\right) + \right.$$

$$\left. \left(\frac{2}{25} \times (16-75,56)^4 - 3\right) + \left(\frac{1}{25} \times (240-75,56)^4 - 3\right) + \left(\frac{2}{25} \times (224-75,56)^4 - 3\right) + \right.$$

$$\left. \left(\frac{1}{25} \times (60-75,56)^4 - 3\right) + \left(\frac{1}{25} \times (110-75,56)^4 - 3\right) + \left(\frac{1}{25} \times (11-75,56)^4 - 3\right) + \left(\frac{1}{25} \times (0-75,56)^4 - 3\right) + \right.$$

$$\left. \left(\frac{2}{25} \times (128-75,56)^4 - 3\right) + \left(\frac{1}{25} \times (244-75,56)^4 - 3\right) + \left(\frac{1}{25} \times (231-75,56)^4 - 3\right) + \right.$$

$$\left. \left(\frac{1}{25} \times (207-75,56)^4 - 3\right) + \left(\frac{1}{25} \times (3-75,56)^4 - 3\right) + \left(\frac{1}{25} \times (10-75,56)^4 - 3\right) + \left(\frac{1}{25} \times (1-75,56)^4 - 3\right) + \right.$$

$$\left. \left(\frac{1}{25} \times (146-75,56)^4 - 3\right) + \left(\frac{1}{25} \times (131-75,56)^4 - 3\right) \right)$$

$$\alpha_4 = \frac{1}{57409208} (118637894)$$

$$\alpha_4 = 2,0665$$

f. Variance

Untuk menghitung nilai Variance, digunakan Persamaan (2-6).

$$\sigma^2 = \left(\frac{1}{25} (24-75,56)^2\right) + \left(\frac{1}{25} (48-75,56)^2\right) + \left(\frac{1}{25} (248-75,56)^2\right) + \left(\frac{1}{25} (112-75,56)^2\right)$$

$$+ \left(\frac{2}{25} (224-75,56)^2\right) + \left(\frac{1}{25} (62-75,56)^2\right) + \left(\frac{2}{25} (8-75,56)^2\right) + \left(\frac{2}{25} (16-75,56)^2\right) +$$

$$\left(\frac{1}{25} (240-75,56)^2\right) + \left(\frac{2}{25} (224-75,56)^2\right) + \left(\frac{1}{25} (60-75,56)^2\right) + \left(\frac{1}{25} (110-75,56)^2\right) +$$

$$\left(\frac{1}{25} (11-75,56)^2\right) + \left(\frac{1}{25} (0-75,56)^2\right) + \left(\frac{2}{25} (128-75,56)^2\right) + \left(\frac{1}{25} (244-75,56)^2\right) +$$

$$\left(\frac{1}{25} (231-75,56)^2\right) + \left(\frac{1}{25} (207-75,56)^2\right) + \left(\frac{1}{25} (3-75,56)^2\right) + \left(\frac{1}{25} (10-75,56)^2\right) +$$

$$\left(\frac{1}{25} (1-75,56)^2\right) + \left(\frac{1}{25} (146-75,56)^2\right) + \left(\frac{1}{25} (131-75,56)^2\right)$$

$$\sigma^2 = 7360,176$$

g. Contrast

Untuk menghitung nilai Contrast, digunakan Persamaan (2-7).

$$\begin{aligned}
CON = & ((0-0)^2 \times \frac{1}{25}) + ((0-1)^2 \times \frac{1}{25}) + ((0-2)^2 \times \frac{1}{25}) + ((0-3)^2 \times \frac{1}{25}) + ((0-4)^2 \times \frac{1}{25}) + ((0-5)^2 \times \frac{1}{25}) \\
& + ((1-0)^2 \times \frac{1}{25}) + ((1-1)^2 \times \frac{1}{25}) + ((1-2)^2 \times \frac{1}{25}) + ((1-3)^2 \times \frac{1}{25}) + ((1-4)^2 \times \frac{1}{25}) + ((1-5)^2 \times \frac{1}{25}) \\
& + ((2-0)^2 \times \frac{1}{25}) + ((2-1)^2 \times \frac{1}{25}) + ((2-2)^2 \times \frac{1}{25}) + ((2-3)^2 \times \frac{1}{25}) + ((2-4)^2 \times \frac{1}{25}) + ((2-5)^2 \times \frac{1}{25}) \\
& + ((3-0)^2 \times \frac{1}{25}) + ((3-1)^2 \times \frac{1}{25}) + ((3-2)^2 \times \frac{1}{25}) + ((3-3)^2 \times \frac{1}{25}) + ((3-4)^2 \times \frac{1}{25}) + ((3-5)^2 \times \frac{1}{25}) \\
& + ((4-0)^2 \times \frac{1}{25}) + ((4-1)^2 \times \frac{1}{25}) + ((4-2)^2 \times \frac{1}{25}) + ((4-3)^2 \times \frac{1}{25}) + ((4-4)^2 \times \frac{1}{25}) + ((4-5)^2 \times \frac{1}{25}) \\
& + ((5-0)^2 \times \frac{1}{25}) + ((5-1)^2 \times \frac{1}{25}) + ((5-2)^2 \times \frac{1}{25}) + ((5-3)^2 \times \frac{1}{25}) + ((5-4)^2 \times \frac{1}{25}) + ((5-5)^2 \times \frac{1}{25})
\end{aligned}$$

$$CON = 11$$

h. *Correlation*

Untuk menghitung nilai *Correlation*, digunakan Persamaan (2-8).

$$\begin{aligned}
COR = & \frac{(0-204,84) \times (0-171,8) \times 24 \times \frac{1}{25}}{3056976,281 \times 2150757} + \frac{(0-204,84) \times (1-171,8) \times 12 \times \frac{1}{25}}{3056976,281 \times 2150757} + \\
& \frac{(0-204,84) \times (2-171,8) \times 31 \times \frac{1}{25}}{3056976,281 \times 2150757} + \frac{(0-204,84) \times (3-171,8) \times 14 \times \frac{1}{25}}{3056976,281 \times 2150757} + \\
& \frac{(0-204,84) \times (4-171,8) \times 7 \times \frac{1}{25}}{3056976,281 \times 2150757} + \dots + \frac{(4-204,84) \times (0-171,8) \times 80 \times \frac{1}{25}}{3056976,281 \times 2150757} + \\
& \frac{(4-204,84) \times (1-171,8) \times 0 \times \frac{1}{25}}{3056976,281 \times 2150757} + \frac{(4-204,84) \times (2-171,8) \times 1 \times \frac{1}{25}}{3056976,281 \times 2150757} + \\
& \frac{(4-204,84) \times (3-171,8) \times 241 \times \frac{1}{25}}{3056976,281 \times 2150757} + \frac{(4-204,84) \times (4-171,8) \times 193 \times \frac{1}{25}}{3056976,281 \times 2150757} \\
COR = & \frac{-61786084}{3056976,281 \times 2150757}
\end{aligned}$$

$$COR = -0,0001$$

i. *Energy*

Untuk menghitung nilai *Energy*, digunakan Persamaan (2-9).

$$\begin{aligned}
Energy = & (24)^2 + (48)^2 + (248)^2 + (112)^2 + (224)^2 + (62)^2 + (8)^2 + (16)^2 + \\
& (240)^2 + (62)^2 + (8)^2 + (16)^2 + (240)^2 + (224)^2 + (60)^2 + (110)^2 + \\
& (11)^2 + (0)^2 + (128)^2 + (16)^2 + (244)^2 + (231)^2 + (207)^2 + (3)^2 + \\
& (10)^2 + (1)^2 + (146)^2 + (131)^2
\end{aligned}$$

$$Energy = 332155$$

j. *Homogeneity*

Untuk menghitung nilai *Energy*, digunakan Persamaan (2-10).

$$Homogeneity = \left(\frac{1}{1+2} + \frac{1}{1+48} + \frac{1}{1+248} + \frac{1}{1+112} + \frac{2}{1+224} + \frac{1}{1+62} + \frac{1}{1+8} + \frac{2}{1+16} + \frac{1}{1+240} + \frac{2}{1+224} + \frac{1}{1+60} + \frac{1}{1+110} + \frac{2}{1+11} + \frac{1}{1+0} + \frac{2}{1+128} + \frac{1}{1+244} + \frac{1}{1+231} + \frac{1}{1+207} + \frac{1}{1+3} + \frac{1}{1+10} + \frac{1}{1+1} + \frac{1}{1+146} + \frac{1}{1+131} \right)$$

$$Homogeneity = 0,1540$$

3.6 Tahap Klasifikasi LVQ

Proses klasifikasi yang dilakukan dalam penelitian ini menggunakan metode LVQ. Misalkan terdapat 6 *input* vektor dari hasil ekstraksi LBP sebelumnya ke dalam 3 kelas seperti pada tabel 3.1.

Tabel 3.1 *Input* vektor.

Mean	Standar Deviasi	Skewness	Entropi	Kurtosis	Variance	Contrast	Correlation	Energy	Homogeneity	Kelas
75,56	85,7915	4,3222	0,2196	2,0665	7360,176	11	-0,0001	332155	0,1540	1
70	80	5	0,4	3	7360	15	-0,0002	302115	0,2	1
1	0	0	0	1	0	0	0	1	0	2
0	0	1	0	1	0	0	1	0	0	2
1	0	0	1	0	1	0	1	0	1	3
0	0	1	1	0	0	0	1	0	1	3

Langkah-langkah yang dilakukan dalam melakukan klasifikasi LVQ adalah sebagai berikut,

- Menginisialisasi bobot yang diambil dari *dataset* pada Tabel 3.1. Jumlah bobot yang diambil sejumlah kelas yang akan digunakan, dimana pada contoh ini diambil 3 *input* untuk dijadikan inisialisasi bobot yang mewakili setiap kelas, Berikut merupakan inisialisasi bobot.

Tabel 3.2 Inisialisasi bobot.

bobot	Vektor	Kelas
1	(75,56, 85,7915, 4,3222, 0,2196, 2,0665, 7360,176, 11, -0,0001, 332155, 0,1540)	1
2	(1, 0, 0, 0, 1, 0, 0, 0, 1, 0)	2
3	(1, 0, 0, 1, 0, 1, 0, 1, 0, 1)	3

- b. Melakukan pembelajaran dimana Sebagai nilai awal dipilih *learning rate* ($\alpha=0,05$), dengan pengurangan sebesar $0,1*\alpha$; dan maksimum Epoch ($MaxEpoch=10$), Contoh berikut merupakan Perhitungan yang dilakukan pada Epoch ke-1:

Data ke-1: (70, 80, 5, 0,4, 3, 7360, 15, -0,0002, 302115, 0,2)

Menghitung jarak pada bobot 1.

$$J_1 =$$

$$\sqrt{(70 - 75,56)^2 + (80 - 85,791)^2 + (5 - 4,322)^2 + (0,4 - 0,22)^2 + (3 - 2,067)^2 + (7360 - 7360,176)^2 + (15 - 11)^2 + (-0,0002 - 0,0001)^2 + (302115 - 332155)^2 + (0,2 - 0,154)^2}$$

$$J_1 = \sqrt{902401682}$$

$$J_1 = 30040,001$$

Menghitung jarak pada bobot 2.

$$J_2 = \sqrt{(70 - 1)^2 + (80 - 0)^2 + (5 - 0)^2 + (0,4 - 0)^2 + (3 - 1)^2 + (7360 - 0)^2 + (15 - 0)^2 + (-0,0002 - 0)^2 + (302115 - 1)^2 + (0,2 - 0)^2}$$

$$J_2 = \sqrt{91327050011}$$

$$J_2 = 302203,665$$

Menghitung jarak pada bobot 3.

$$J_3 = \sqrt{(70 - 1)^2 + (80 - 0)^2 + (5 - 0)^2 + (0,4 - 1)^2 + (3 - 0)^2 + (7360 - 1)^2 + (15 - 0)^2 + (-0,0002 - 1)^2 + (302115 - 0)^2 + (0,2 - 1)^2}$$

$$J_3 = \sqrt{91327639528}$$

$$J_3 = 302204,632$$

Setelah melakukan perhitungan jarak, diketahui bahwa target data ke-1 sama dengan bobot minimum yaitu bobot ke-1, Sehingga bobot ke-1 diperbaharui sebagai berikut.

$$w_{10} = w_{10} + \alpha (x_{10} - w_{10}) = 75,56 + 0,05*(70-75,56) = 75,282$$

$$w_{11} = w_{11} + \alpha (x_{10} - w_{11}) = 85,7915 + 0,05*(80-85,7915) = 85,5019$$

$$w_{12} = w_{12} + \alpha (x_{10} - w_{12}) = 4,3222 + 0,05*(5-4,3222) = 4,3561$$

$$w_{13} = w_{13} + \alpha (x_{10} - w_{13}) = 0,2196 + 0,05*(0,4-0,2196) = 0,2286$$

$$w_{14} = w_{14} + \alpha (x_{10} - w_{14}) = 2,0665 + 0,05*(3-2,0665) = 2,1122$$

$$w_{15} = w_{15} + \alpha (x_{10} - w_{15}) = 75,56 + 0,05*(7360-75,56) = 7360,17$$

$$w_{16} = w_{16} + \alpha (x_{10} - w_{16}) = 11 + 0,05*(15-11) = 11,2$$

$$w_{17} = w_{17} + \alpha (x_{10} - w_{17}) = -0,0001 + 0,05*(-0,0002-(-0,0001)) = -0,0001$$

$$w_{18} = w_{18} + \alpha (x_{10} - w_{18}) = 332155 + 0,05*(302115-332155) = 330653$$

$$w_{19} = w_{19} + \alpha (x_{10} - w_{19}) = 0,1540 + 0,05*(0,2-0,1540) = 0,1563$$

Sehingga, bobot 1 nilainya telah berubah menjadi (75,282, 85,5019, 4,3561, 0,2286, 2,1122, 7360,17, 11,2, -0,0001, 330653, 0,1563),

Data ke-2: (0, 0, 1, 0, 1, 0, 0, 1, 0, 0)

Menghitung jarak pada bobot 1.

$$J_1 = \sqrt{(0 - 75,282)^2 + (0 - 85,502)^2 + (1 - 4,356)^2 + (0 - 0,229)^2 + (1 - 2,112)^2 + (0 - 7360,17)^2 + (0 - 11,2)^2 + (1 - (-0,0001))^2 + (0 - 330653)^2 + (0 - 0,156)^2}$$

$$J_1 = \sqrt{2256004,2}$$

$$J_1 = 1502,0001$$

Menghitung jarak pada bobot 2.

$$J_2 = \sqrt{(0 - 1)^2 + (0 - 0)^2 + (1 - 0)^2 + (0 - 0)^2 + (1 - 1)^2 + (0 - 0)^2 + (0 - 0)^2 + (1 - 0)^2 + (0 - 1)^2 + (0 - 0)^2}$$

$$J_2 = \sqrt{4}$$

$$J_2 = 2$$

Menghitung jarak pada bobot 3.

$$J_3 = \sqrt{(0 - 1)^2 + (0 - 0)^2 + (1 - 0)^2 + (0 - 1)^2 + (1 - 0)^2 + (0 - 1)^2 + (0 - 0)^2 + (1 - 1)^2 + (0 - 0)^2 + (0 - 1)^2}$$

$$J_3 = \sqrt{6}$$

$$J_3 = 2,4493$$

Setelah melakukan perhitungan jarak, diketahui bahwa target data ke-2 sama dengan bobot minimum yaitu bobot ke-2, Sehingga bobot ke-2 diperbaharui sebagai berikut.

$$w_{20} = w_{20} + \alpha (x_{10} - w_{20}) = 1 + 0,05*(0-1) = 0,95$$

$$w_{21} = w_{21} + \alpha (x_{10} - w_{21}) = 0 + 0,05*(0-0) = 0$$

$$w_{22} = w_{22} + \alpha (x_{10} - w_{22}) = 0 + 0,05*(1-0) = 0,05$$

$$w_{23} = w_{23} + \alpha (x_{10} - w_{23}) = 0 + 0,05*(0-0) = 0$$

$$w_{24} = w_{24} + \alpha (x_{10} - w_{24}) = 1 + 0,05*(1-1) = 1$$

$$w_{25} = w_{25} + \alpha (x_{10} - w_{25}) = 0 + 0,05*(0-0) = 0$$

$$w_{26} = w_{26} + \alpha (x_{10} - w_{26}) = 0 + 0,05*(0-0) = 0$$

$$w_{27} = w_{27} + \alpha (x_{10} - w_{27}) = 0 + 0,05*(1-0) = 0,05$$

$$w_{28} = w_{28} + \alpha (x_{10} - w_{28}) = 1 + 0,05*(0-1) = 0,95$$

$$w_{29} = w_{29} + \alpha (x_{10} - w_{29}) = 0 + 0,05*(0-0) = 0$$

Sehingga, bobot 2 nilainya telah berubah menjadi (0,95, 0, 0,05, 0, 1, 0, 0, 0,05, 0,95, 0).

Data ke-3: (0, 0, 1, 1, 0, 0, 0, 1, 0, 1)

Menghitung jarak pada bobot 1.

$$J_1 = \sqrt{\frac{(0 - 75,282)^2 + (0 - 85,502)^2 + (1 - 4,356)^2 + (1 - 0,229)^2 + (0 - 2,112)^2 + (0 - 7360,17)^2 + (0 - 11,2)^2 + (1 - (-0,0001))^2 + (0 - 330653)^2 + (1 - 0,156)^2}{10}}$$

$$J_1 = \sqrt{109385591592}$$

$$J_1 = 330374,9265$$

Menghitung jarak pada bobot 2.

$$J_2 = \sqrt{\frac{(0 - 0,95)^2 + (0 - 0)^2 + (1 - 0,05)^2 + (1 - 0)^2 + (0 - 1)^2 + (0 - 0)^2 + (0 - 0)^2 + (1 - 0,05)^2 + (0 - 0,95)^2 + (1 - 0)^2}{10}}$$

$$J_2 = \sqrt{6,61}$$

$$J_2 = 2,571$$

Menghitung jarak pada bobot 3.

$$J_3 = \sqrt{\frac{(0 - 1)^2 + (0 - 0)^2 + (1 - 0)^2 + (1 - 1)^2 + (0 - 0)^2 + (0 - 1)^2 + (0 - 0)^2 + (1 - 1)^2 + (0 - 0)^2 + (1 - 1)^2}{10}}$$

$$J_3 = \sqrt{3}$$

$$J_3 = 1,7321$$

Setelah melakukan perhitungan jarak, diketahui bahwa target data ke-3 sama dengan bobot minimum yaitu bobot ke-3, Sehingga bobot ke-3 diperbaharui sebagai berikut.

$$w_{20} = w_{20} + \alpha (x_{10} - w_{20}) = 1 + 0,05*(1-0) = 0,95$$

$$w_{21} = w_{21} + \alpha (x_{10} - w_{21}) = 0 + 0,05*(0-0) = 0$$

$$w_{22} = w_{22} + \alpha (x_{10} - w_{22}) = 0 + 0,05*(0-1) = 0,05$$

$$w_{23} = w_{23} + \alpha (x_{10} - w_{23}) = 1 + 0,05*(1-1) = 1$$

$$w_{24} = w_{24} + \alpha (x_{10} - w_{24}) = 0 + 0,05*(0-0) = 0$$

$$w_{25} = w_{25} + \alpha (x_{10} - w_{25}) = 1 + 0,05*(1-0) = 0,95$$

$$w_{26} = w_{26} + \alpha (x_{10} - w_{26}) = 0 + 0,05*(0-0) = 0$$

$$w_{27} = w_{27} + \alpha (x_{10} - w_{27}) = 1 + 0,05*(1-1) = 1$$

$$w_{28} = w_{28} + \alpha (x_{10} - w_{28}) = 0 + 0,05*(0-0) = 0$$

$$w_{29} = w_{29} + \alpha (x_{10} - w_{29}) = 1 + 0,05*(0-1) = 1$$

Sehingga, bobot 3 nilainya telah berubah menjadi (0,95, 0, 0,05, 1, 0, 0,95, 0, 1, 0, 1),

$$\alpha = \alpha - 0,01*\alpha = 0,05 - 0,01*0,05 = 0,045$$

proses ini diteruskan untuk Epoch ke-2 sampai Epoch ke-10, untuk setiap data dengan menggunakan cara yang sama, Setelah mencapai Epoch yang ke-10 diperoleh bobot akhir sebagai berikut.

$$w_1 = (75,2, 85, 4, 0,22, 2,1, 7362, 12, -0,0001, 330653, 0,15)$$

$$w_2 = (0,3727, 0,2161, 0,6347, -0,2164, 0,7981, 0,4254, 0,05, 0,95, 0, 1)$$

$$w_3 = (0,0000, 0,7696, 0,7900, 1, 0,5869, 0,2171, 0, 1, 0, 0)$$

Apabila kita ingin mensimulasikan *input*: (0, 1, 0, 1, 1, 0, 1, 0, 0, 1), maka kita cari terlebih dahulu jarak *input* tersebut dengan kedua bobot, Nomor dari bobot dengan jarak yang terpendek akan menjadi kelasnya,

Menghitung jarak pada bobot 1.

$$J_1 = \sqrt{\begin{matrix} (0 - 75,2)^2 + (1 - 85)^2 + (0 - 4)^2 + (1 - 0,22)^2 + (1 - 2,1)^2 + \\ (0 - 7362)^2 + (1 - 12)^2 + (0 - (-0,0001))^2 + (0 - 330653)^2 + \\ (1 - 0,15)^2 \end{matrix}}$$

$$J_1 = \sqrt{109312886744}$$

$$J_1 = 330624,9941$$

Menghitung jarak pada bobot 2.

$$J_2 = \sqrt{\begin{matrix} (0 - 0,373)^2 + (1 - 0,216)^2 + (0 - 0,635)^2 + (1 - (-0,216))^2 + \\ (1 - 0,798)^2 + (0 - 0,425)^2 + (1 - 0,05)^2 + (0 - 0,95)^2 + (0 - 0)^2 \\ + (1 - 1)^2 \end{matrix}}$$

$$J_2 = \sqrt{4,662}$$

$$J_2 = 2,159$$

Menghitung jarak pada bobot 3.

$$J_3 = \sqrt{(0-0)^2 + (1-0,769)^2 + (0-0,79)^2 + (1-1)^2 + (1-0,587)^2 + (1-0)^2 + (0-1)^2 + (0-0)^2 + (1-0)^2}$$

$$J_3 = \sqrt{3,895}$$

$$J_3 = 1,973$$

Jarak terkecil ada pada bobot ke-3, sehingga *input* termasuk dalam kelas 3.

3.6 Verifikasi tanda tangan

Pada proses pengujian, setelah dilakukan klasifikasi tanda tangan maka dilakukan verifikasi tanda tangan. Pada *input* sebelumnya, diketahui bahwa *input* telah diklasifikasi ke kelas 3. Untuk verifikasi tanda tangan vektor *input* dan vektor bobot kelas 3 akan dihitung nilai *Euclidean Distance* dari setiap index vektor dengan menggunakan persamaan (2-17) seperti berikut.

$$\text{Index}[0] = \sqrt{(0-0,373)^2} = 0,373$$

Perhitungan dilakukan sampai index ke 9, dan hasilnya dimasukkan ke dalam *array*. Sehingga nilai pada *array* yaitu (0,373, 0,784, 0,635, 1,216, 0,202, 0,425, 0,95, 0,95, 0, 0). Kemudian dilakukan normalisasi pada setiap *index array* menggunakan persamaan (2-18).

$$x' = \frac{0,373-0}{1,261-0}$$

$$x' = 0,476$$

Perhitungan dilakukan sampai index ke 9 sehingga nilai pada *array* menjadi (0,295, 0,622, 0,503, 1, 0,16, 0,337, 0,753, 0,753, 0, 0). Setelah itu, dihitung nilai *mean* pada *array* dengan menggunakan persamaan (2-19).

$$x = \frac{0,295+0,622+0,503+1+0,16+0,337+0,753+0,753+0+0}{10}$$

$$x = 0,4423$$

3.8 Skenario pengujian

Pengujian pada penelitian ini maksudnya adalah untuk menguji model yang dihasilkan dari ekstraksi fitur LBP dan pelatihan LVQ sebelumnya. Model yang

dibangun akan dilihat keakuratannya dengan memperhatikan parameter percobaan. Pada metode LBP parameter yang diperhatikan yaitu radius, dan jumlah tetangga. Pada metode LVQ parameter yang diperhatikan yaitu Epoch dan *learning rate*. Pada pengujian ini, penentuan jenis data citra yaitu citra latih dan citra uji digunakan *k-fold cross validation* dimana $k=5$. Berikut merupakan jenis skenario pengujian.

1. Pengujian akurasi berdasarkan parameter pada metode yang digunakan. Pada metode LBP, parameter yang diuji menggunakan nilai $R = 1$, $P = 4$ dan $R=2$, $P=8$, dimana R merupakan jarak tetangga dan P merupakan jumlah tetangga dari titik pusat. Nilai ini diambil dari penelitian sebelumnya dengan akurasi tertinggi yaitu mengenai identifikasi cacat kayu, akurasi tertinggi pada ekstraksi fitur LBP terdapat pada parameter $R=1$ dan $P=4$ [17]. Dan penelitian mengenai identifikasi tanda tangan, akurasi tertinggi pada ekstraksi fitur LBP terdapat pada parameter $R=2$ dan $P=8$ [15].
2. Pengujian akurasi berdasarkan parameter LVQ yang diuji yaitu Epoch dan *learning rate* dimana Epoch yang digunakan yaitu pada 50,100,150,200 serta nilai *learning rate* yaitu dari 0,1, 0,01, dan 0,001.
3. Pengujian akurasi berdasarkan jenis ukuran citra yang digunakan dalam melakukan verifikasi tanda tangan. Adapun ukuran citra yang digunakan adalah 64×64 , 128×128 dan 256×256 piksel. Parameter LBP dan LVQ menggunakan parameter yang menghasilkan akurasi tertinggi pada skenario sebelumnya.
4. Pengujian akurasi dengan variabel threshold sebagai batas penentuan keaslian tanda tangan. pada penelitian ini akan diuji batas variabel threshold yaitu pada rentang nilai 0,1-0,2.

Dalam proses pengujian akan dihitung nilai akurasi. *True Positif Rate* dan *True Negative Rate*. Misalkan terdapat data *dummy* seperti pada Tabel 3.3.

Tabel 3.3 *Confusion matrix* dengan data *dummy*.

Kelas Sebenarnya \ Hasil Klasifikasi	Positif	Negatif
Positif	19	1
Negatif	2	18

Berdasarkan *confussion matrix* di atas. dapat dihitung nilai akurasi. *True Positive Rate* dan *True Negative Rate*.

- Nilai akurasi dihitung dengan menggunakan persamaan (2-20).

$$\text{Akurasi} = \frac{19+18}{19+118+1} = 0,925$$

- Nilai *True Postive Rate (sensitivity)* dihitung dengan persamaan (2-21).

$$\text{True Postive Rate} = \frac{19}{19+1} = 0,95$$

- Nilai *True Negative Rate (specificity)* dihitung dengan persamaan (2-22).

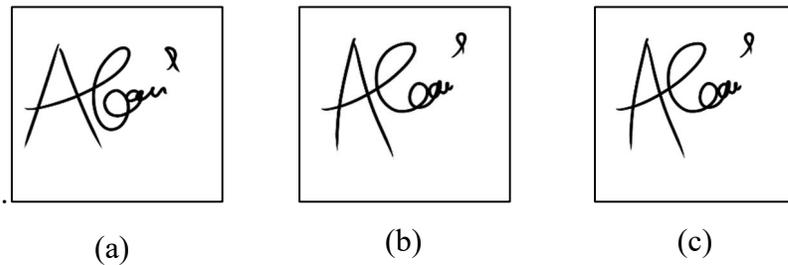
$$\text{True Negative Rate} = \frac{18}{2+18} = 0,9$$

BAB IV

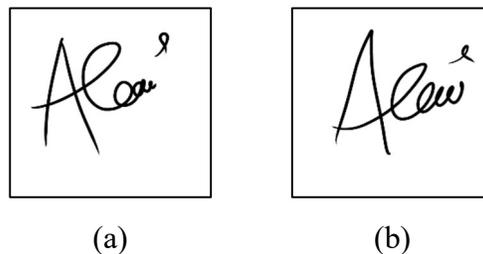
HASIL DAN PEMBAHASAN

4.1 Pengumpulan *Dataset*

Pada penelitian ini, pembuatan model *machine learning* menggunakan *dataset* keseluruhan berjumlah 600 citra tanda tangan yang diambil dari 30 mahasiswa Teknik Informatika Universitas Mataram secara acak sehingga *dataset* terdiri dari 30 kelas. *Dataset* terdiri dari 450 tanda tangan asli dan 150 tanda tangan palsu. Tanda tangan asli diambil pada tiga hari yang berbeda dengan tujuan untuk menambah variasi tanda tangan, setiap 1 hari diambil 5 tanda tangan asli sehingga setiap kelas terdiri dari 15 citra tanda tangan. Tanda tangan palsu setiap kelas terdiri dari 5 tanda tangan yang dikerjakan oleh 5 orang yang dipercaya peneliti dapat meniru tanda tangan. *Dataset* tersebut diambil menggunakan *smartphone* Samsung Galaxy Tab A dengan menggunakan aplikasi IBisPaintX.



Gambar 4.1 *Dataset* tanda tangan (a) hari ke 1, (b) hari ke 2, (c) hari ke3



Gambar 4.2 *Dataset* tanda tangan (a) asli, (b) palsu

4.2 Mekanisme Penelitian

Mekanisme penelitian diperlukan untuk mendapatkan model terbaik dari *machine learning* yang dibentuk pada penelitian ini. *Dataset* yang digunakan untuk pencarian model ini adalah *dataset* tanda tangan asli untuk proses *training* dan *testing*

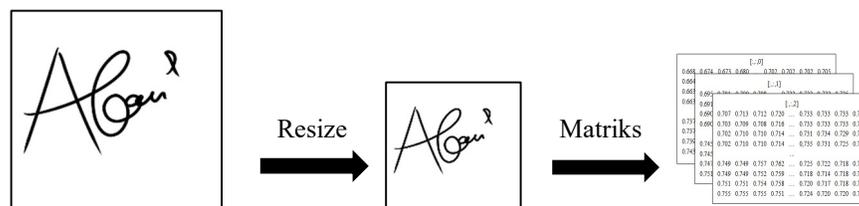
pada proses identifikasi tanda tangan, sedangkan tanda tangan palsu hanya digunakan untuk pengujian verifikasi tanda tangan.

Selanjutnya, dilakukan pengujian terhadap model *machine learning* yang dibangun menggunakan *dataset* yang sudah disebutkan sebelumnya. Pengujian ini dilakukan dengan berbagai parameter dengan urutan pengujian sebagai berikut.

1. Parameter LBP (Radius (R)=2, jumlah tetangga(P)=8 dan R=1, P=4)
2. Jumlah Epoch LVQ (50,100,150,200)
3. Jumlah *Learning Rate* (LR) LVQ (0,001, 0,01, 0,1)
4. Ukuran citra masukan (64x64, 128x128 dan 256x256)
5. Variabel *Threshold* (0,1, 0,11, 0,12, 0,13, 0,14, 0,15, 0,16, 0,17, 0,18, 0,19, 0,2)

4.3 Preprocessing

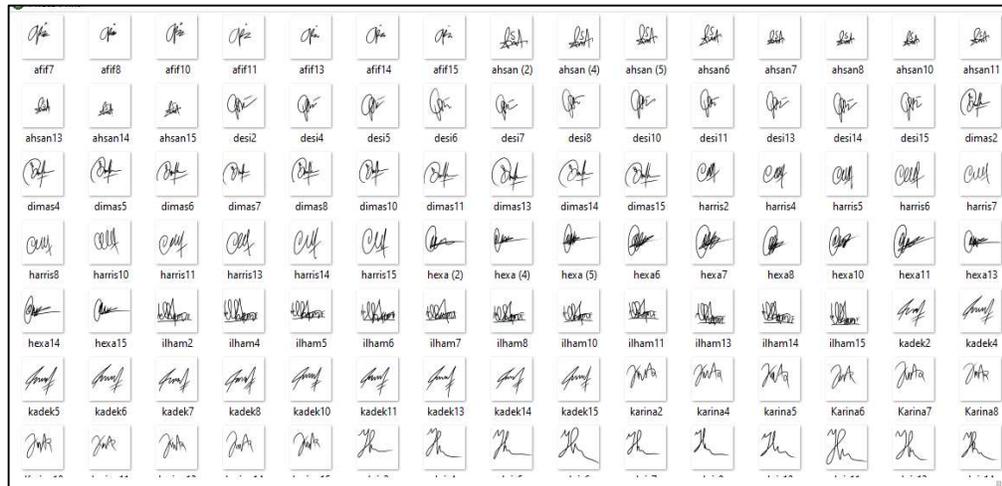
Tahap paling awal dari proses pengujian model *machine learning* adalah melakukan *load dataset*. Pengambilan *dataset* pada penelitian ini menggunakan *method* “*imread*” dari *library* “*opencv*”. Selanjutnya, citra masuk ke proses *resize* untuk mendapatkan ukuran citra yang diinginkan. Adapun variasi ukuran citra yang digunakan dalam pengujian yaitu 64×64 , 128×128 dan 256×256 pixel *grayscale*. Gambar 4.3 menunjukkan proses *resize* pada citra.



Gambar 4.3 Citra Masukan

Selanjutnya citra yang sudah di-*resize* kemudian dibagi menjadi citra untuk *training* dan citra untuk *testing*. Skenario *training* dan *testing* yang digunakan untuk melakukan pembagian pada penelitian ini adalah 70:30 dengan kombinasi citra *training* dan citra *testing* dipisahkan terlebih dahulu dalam *folder* berbeda. Pemisahan ini dilakukan agar ketika dilakukan penelitian untuk menguji setiap parameter, citra-citra yang digunakan ketika *training* dan *testing* merupakan citra yang sama. Gambar

4.4 menunjukkan *dataset* yang digunakan sebagai citra *training* sedangkan Gambar 4.5 menunjukkan *dataset* yang digunakan sebagai data *testing*.



Gambar 4.4 Citra *Training*

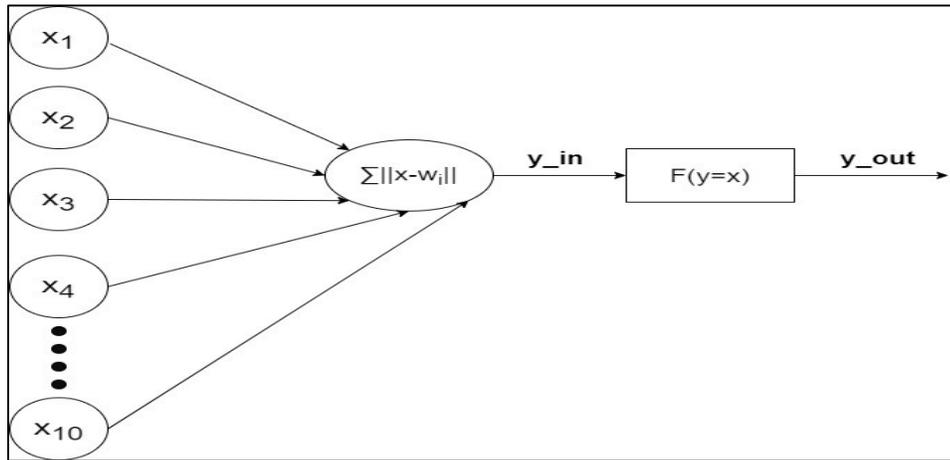


Gambar 4.5 Citra *Testing*

4.4 Model LVQ

LVQ yang digunakan pada penelitian ini adalah *single layer* LVQ sehingga tidak menggunakan *hidden layer* dan hanya terdiri atas unit masukan dan unit keluaran. Unit masukan berjumlah 10 *node* karena setiap citra memiliki 10 fitur, sedangkan unit keluaran berjumlah satu *node* yang merupakan kelas citra. Aktivasi fungsi yang digunakan yaitu fungsi linier sehingga jika 2 vektor masukan mendekati sama, maka

lapisan kompetitif akan meletakkan kedua vektor masukan tersebut ke dalam kelas yang sama. Gambar 4.6 merupakan arsitektur jaringan LVQ yang digunakan.



Gambar 4.6 Arsitektur LVQ

Gambar 4.6 menunjukkan bahwa yang bertindak sebagai dendrit atau data masukan adalah $X_1 \sim X_{10}$ yang merupakan fitur-fitur dari proses ekstraksi sebelumnya. Kemudian, yang bertindak sebagai sinapsis atau bobot adalah W , sedangkan soma atau badan sel dari jaringan ini adalah perhitungan $||X - W||$. *Input* akan diproses dengan mengurangi nilai masukan dengan masing-masing bobot dimana perhitungannya menggunakan rumus *Euclidean Distance* sesuai dengan algoritme LVQ. Hasil dari proses tersebut adalah y_{in} yang selanjutnya diproses pada F yang merupakan aktivasi fungsi, dan yang bertindak sebagai akson atau data keluaran adalah y_{out} yang menampilkan kelas citra.

4.5 Training

Proses *training* dilakukan dengan mengekstraksi citra menggunakan metode LBP dengan 10 fitur yaitu *mean, SD, skewness, entropy, kurtosis, variance, contrast, correlation, energy, dan homogeneity*. Nilai dari masing-masing fitur akan disimpan pada *file* tipe *csv* untuk dikelompokkan berdasarkan nama pemilik citra sehingga nanti dapat dijadikan acuan dalam proses klasifikasi. Kemudian pada metode LVQ dilakukan *training* dengan cara melakukan *update weight* pada model. Sehingga pada tiap Epoch, model mengalami perbaikan dari sisi keberhasilannya mengklasifikasikan citra.

4.6 Testing

Proses *testing* merupakan proses untuk menguji model yang didapatkan pada proses *training*. *Testing* dilakukan pada tiap Epoch setelah semua citra selesai di *training*. *Testing* model menggunakan citra yang masuk ke dalam kelompok data *testing* yang mana citra tersebut belum pernah masuk ke dalam proses selama *training*.

4.7 Verifikasi Tanda Tangan

Verifikasi tanda tangan dilakukan setelah proses klasifikasi. Data *testing* akan diuji asli atau palsu dengan menghitung nilai *Euclidean distance* antara fitur-fitur citra latih dengan nilai rata-rata setiap fitur citra pada kelasnya dan disimpan dalam array. Setelah itu, setiap index array akan dinormalisasi agar nilai keseluruhan bernilai positif dan berada direntang 0-1. Kemudian, semua nilai *index* pada *array* tersebut dirata-ratakan dan dijadikan nilai untuk dibandingkan dengan batas variabel *threshold* untuk menentukan citra tanda tangan merupakan tanda tangan asli atau palsu.

4.8 Pengujian dengan Ekstraksi Fitur Statistik LBP

Sebelum masuk ke pengujian terhadap skenario uji, dilakukan penelitian terhadap metode yang dipilih. Penelitian ini dilakukan dengan menggunakan metode ekstraksi fitur LBP dengan 10 fitur statistik yaitu *mean*, *standar deviasi*, *skewness*, *entropy*, *kurtosis*, *variance*, *contrast*, *correlation*, *energy*, dan *homogeneity*. Uji coba ini dilakukan dengan menggunakan *dataset* tanda tangan asli karena tujuannya hanya untuk melihat keberhasilan metode dalam mengidentifikasi tanda tangan dengan pembagian *dataset* yaitu 70% untuk *training* dan 30% untuk *testing*.

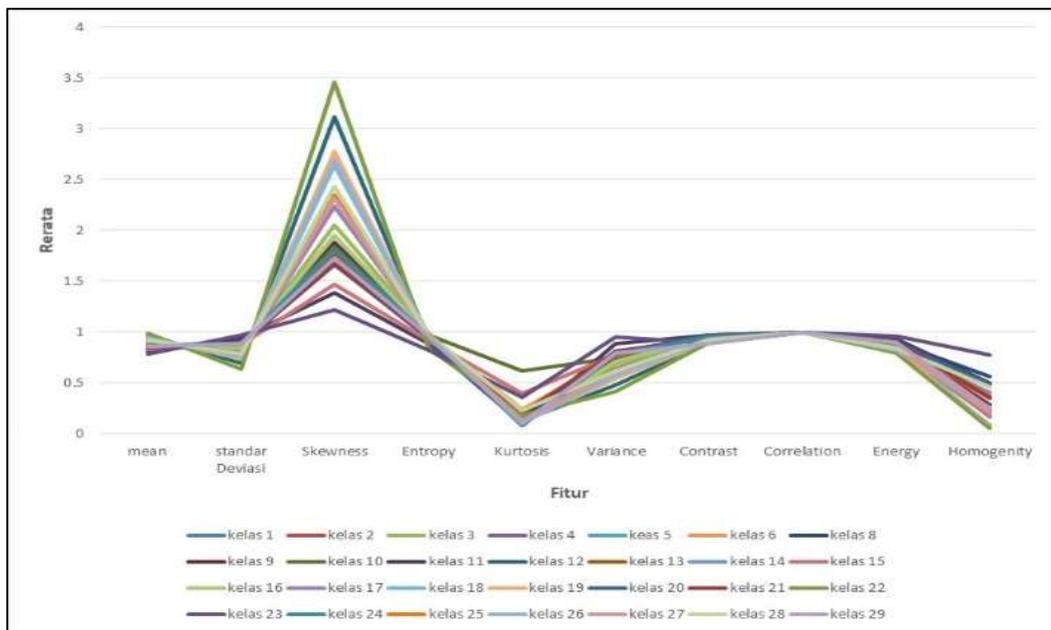
Uji coba ini dilakukan untuk mengetahui apakah fitur statistik pada metode LBP yang dipilih cocok digunakan untuk reorientasi tanda tangan. Uji coba ini menggunakan klasifikasi KNN karena KNN umumnya digunakan untuk mengecek keberhasilan ekstraksi fitur, KNN juga merupakan metode yang sederhana dan menghasilkan akurasi yang tinggi. Dalam penelitian ini, digunakan KNN dengan menggunakan $K=1$ karena hanya $K=1$ yang menghasilkan akurasi *training* 100% dan dengan *size image* yang digunakan yaitu 64×64 . Parameter LBP yang digunakan yaitu $R=2$ dan $P=8$ karena pada penelitian [15] parameter LBP tersebut menghasilkan akurasi yang tinggi. Pengujian pertama menggunakan 10 fitur LBP yaitu fitur LBP *mean*, *SD*,

skewness, entropy, kurtosis, variance, contrast, correlation, energy, homogeneity. Kemudian dicoba juga menggunakan fitur statistik LBP dengan menggunakan 3 fitur yaitu *mean, entropy, dan variance* seperti pada penelitian [23] tujuannya untuk mengetahui kombinasi fitur statistik terbaik yang dapat digunakan untuk mengidentifikasi tanda tangan. Hasil dari pengujian dapat dilihat pada Tabel 4.1

Tabel 4.1 Hasil pengujian dengan fitur statistik LBP

Fitur LBP	Akurasi Training	Akurasi Testing
10 fitur statistik	100%	16%
3 fitur statistik	100%	16%

Hasil pengujian pada Tabel 4.1 menunjukkan bahwa ekstraksi fitur metode LBP dengan menggunakan 10 fitur statistik untuk citra tanda tangan yang telah dirancang menghasilkan akurasi yang sangat rendah yaitu 16%. Pada pengujian dengan 3 fitur statistik LBP diharapkan menghasilkan akurasi yang tinggi, namun hasil akurasi tetap hanya 16%, sehingga dilakukan pengecekan terhadap persebaran nilai fitur pada keseluruhan *dataset*. Gambar 4.7 menunjukkan plot persebaran nilai fitur keseluruhan *dataset*.



Gambar 4.7 Persebaran nilai fitur keseluruhan *dataset*

Berdasarkan Gambar 4.7 dapat diketahui bahwa nilai 10 fitur LBP pada setiap kelas terdapat banyak nilai yang tumpang tindih terutama pada fitur *correlation*, dan pada fitur lainnya terlihat garis banyak berdempet sehingga kita dapat mengetahui bahwa nilai fitur pada tiap kelas hampir sama, sehingga fitur statistik LBP yaitu *mean*, *standar deviasi*, *skewness*, *entropy*, *kurtosis*, *variance*, *contrast*, *correlation*, *energy*, dan *homogeneity* tidak cocok digunakan untuk reorientasi citra tanda tangan.

4.9 Pengujian dengan Kombinasi Ekstraksi Fitur LBP dan DCT

Pengujian dengan metode DCT dilakukan sebagai solusi untuk memperbaiki permasalahan pada ekstraksi fitur sebelumnya. Dilakukan pengujian dengan memanfaatkan metode DCT dimana pengujian pertama dilakukan dengan kombinasi metode DCT dan klasifikasi KNN. Klasifikasi KNN digunakan untuk mengecek keberhasilan ekstraksi fitur. pengujian kedua dilakukan dengan melakukan ekstraksi fitur terhadap citra masukan menggunakan metode LBP kemudian hasil LBP tersebut diolah dengan metode DCT untuk mengambil komponen-komponen dominan yang diperlukan pada citra sebagai fitur yang akan diklasifikasikan oleh metode KNN dengan nilai $K=1$. Parameter LBP yang digunakan yaitu $R=2$ dan $P=8$. Hasil dari pengujian dapat dilihat pada Tabel 4.2.

Tabel 4.2 Hasil pengujian dengan DCT

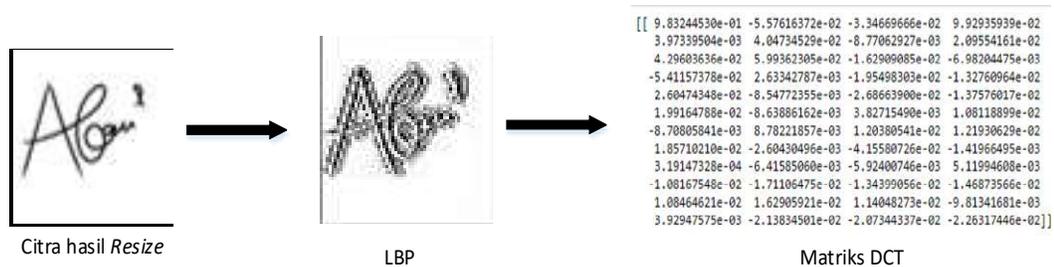
Metode	Akurasi <i>Training</i>	Akurasi <i>Testing</i>
DCT+KNN	100%	98%
LBP+DCT+KNN	100%	97%

Pada Tabel 4.2 menunjukkan bahwa kombinasi metode DCT dan KNN menghasilkan akurasi yang tinggi yaitu 98%. Kombinasi metode LBP dan DCT dengan klasifikasi KNN juga menghasilkan akurasi yang tinggi yaitu 97%, hal ini menunjukkan bahwa pemanfaatan metode DCT setelah ekstraksi fitur dengan metode LBP untuk citra tanda tangan cocok digunakan. Kemudian, dicoba pengujian dengan menggunakan metode klasifikasi LVQ dengan *learning rate* yaitu 0,01 dan Epoch yaitu 100, parameter tersebut dipilih karena pada penelitian [22] nilai parameter ini menghasilkan akurasi tertinggi. Hasil dari pengujian dapat dilihat pada Tabel 4.3.

Tabel 4.3 Hasil pengujian dengan klasifikasi LVQ

Metode	Akurasi <i>Training</i>	Akurasi <i>Testing</i>
DCT+LVQ	95%	91%
LBP+DCT+LVQ	95%	93,3%

Pada Tabel 4.3 diketahui bahwa kombinasi metode LBP, DCT dan LVQ menghasilkan akurasi yang lebih tinggi dari metode DCT dan LVQ yaitu 95% untuk *training* dan 93,3% untuk *testing*, Sehingga penelitian ini akan dilanjutkan dengan menggunakan metode ekstraksi fitur LBP dengan memanfaatkan metode DCT seperti pada Gambar 4.8.



Gambar 4.8 Proses ekstraksi fitur

Gambar 4.8 menunjukkan proses ekstraksi fitur yang akan digunakan pada penelitian ini, dimana pada citra hasil *preprocessing* yaitu *resize* dilakukan ekstraksi fitur dengan menggunakan metode LBP. Hasil dari proses LBP tersebut kemudian diolah dengan metode DCT. Metode DCT digunakan untuk mengambil fitur penting atau fitur terbaik dengan memilih komponen-komponen dominan pada citra hasil ekstraksi fitur LBP, sehingga hasil dari proses DCT yang akan diklasifikasikan. Untuk menentukan model terbaik pada DCT dilakukan pengujian terhadap nilai koefisien DCT.

4.9.1 Pengujian Koefisien DCT

Metode DCT digunakan untuk mengambil komponen-komponen yang dominan pada citra yang digunakan sebagai fitur untuk diklasifikasikan. Pengujian koefisien DCT digunakan untuk menentukan koefisien terbaik untuk mengidentifikasi tanda tangan. Pemilihan koefisien pada penelitian ini yaitu dengan rentang 32-128. Tabel 4.4 menunjukkan hasil pengujian koefisien DCT.

Tabel 4.4 Hasil pengujian koefisien DCT

Koefisien	Akurasi <i>Training</i>	Akurasi <i>Testing</i>
32	93%	91%
48	97%	93,3%
64	96%	92,6%
80	95%	92,6%
96	96%	92%
112	94%	90%
128	94%	90%

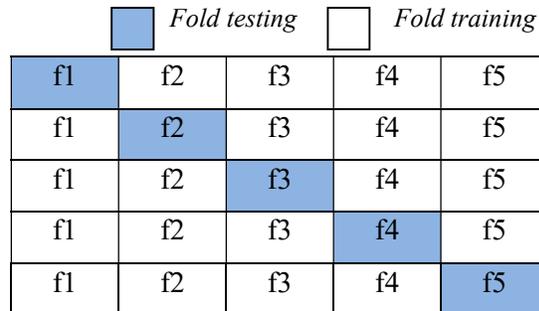
Berdasarkan Tabel 4.4 didapatkan bahwa akurasi terbaik diperoleh pada koefisien DCT 48 dengan akurasi *training* yaitu 97% dan akurasi *testing* dengan jumlah tertinggi yaitu 93,3%, sehingga untuk pengujian selanjutnya digunakan ekstraksi fitur LBP dengan memanfaatkan metode DCT dengan koefisien 48.

4.10 Hasil Pengujian Terhadap Skenario Uji

Pengamatan terhadap hasil pengujian dilakukan secara berurutan sesuai dengan urutan yang telah dijabarkan pada subjudul skenario pengujian. Adapun arsitektur awal untuk pengujian pertama yaitu pada parameter LVQ digunakan jumlah Epoch yaitu 100 dan jumlah *learning rate* yaitu 0,01. Parameter tersebut dipilih karena pada penelitian [22] nilai parameter ini menghasilkan akurasi tertinggi, variabel *threshold* bernilai 0,15, dan koefisien DCT yang digunakan yaitu 48 sesuai dengan hasil pengujian sebelumnya (Tabel 4.4). Teknik pengujian yang digunakan untuk mengetahui performa model adalah *k-fold cross validation*. *Cross validation* merupakan metode statistik untuk mengevaluasi serta membandingkan algoritme learning dengan membagi data menjadi dua segmen, satu segmen digunakan untuk melatih model dan yang lain digunakan untuk memvalidasi model. *Cross validation* bertujuan untuk menghindari tumpang tindih pada data *testing*[19].

Pada penelitian ini, digunakan *5-fold cross validation* dimana teknik ini dilakukan dengan cara membagi seluruh citra pada *dataset* ke dalam 5 *folder* yang berbeda. Pada proses verifikasi, 150 tanda tangan palsu ditambahkan sehingga jumlah data *testing* pada setiap *fold*-nya yaitu 270 tanda tangan yang terdiri dari 120 tanda

tangan asli dan 150 tanda tangan palsu. Visualisasi *5-fold cross validation* dapat dilihat pada Gambar 4.9. Selanjutnya untuk menentukan model yang terbaik akan diambil rata-rata performa dari iterasi *5-fold cross validation*.



Gambar 4.9 Mekanisme *k-fold cross validation*

Pada pengujian ditampilkan evaluasi hasil pada proses klasifikasi dan proses verifikasi. Pada proses klasifikasi ditampilkan akurasi *training* dan akurasi *testing*. Pada tahap verifikasi ditampilkan *accuracy*, *sensitivity* dan *specificity*. *accuracy* didapat dengan menghitung nilai keseluruhan dari kelas yang diuji. Nilai *sensitivity* didapat dari kelas positif yang diklasifikasikan dengan benar dimana pada penelitian ini yaitu tanda tangan asli, dan nilai *specificity* didapat dari kelas negatif yang diklasifikasikan dengan benar dimana pada penelitian ini yaitu tanda tangan palsu.

4.10.1 Pengujian Terhadap Parameter LBP

Pengujian ini dilakukan untuk mencari parameter terbaik pada metode ekstraksi fitur LBP dalam mengidentifikasi dan memverifikasi tanda tangan. Tabel 4.5 menunjukkan hasil pengujian parameter LBP.

Tabel 4.5 Hasil pengujian parameter LBP

Parameter		Klasifikasi		Verifikasi			Time
P	R	Akurasi Training	Akurasi Testing	Accuracy	Sensitivity	Specificity	
4	1	95%	93%	57%	71%	46%	23 s
8	2	95%	93%	52%	46%	57%	24 s

Berdasarkan Tabel 4.5 dapat diketahui bahwa akurasi *training* dan akurasi *testing* kedua uji menghasilkan akurasi yang sama. Pada proses verifikasi, akurasi paling optimal yaitu pada R =1 dan P=4 dengan akurasi sebesar 57%, *sensitivity* sebesar 71% dan *specificity* sebesar 46% begitu pula dengan waktu komputasinya lebih cepat

dibandingkan menggunakan R=1 dan P=8. sehingga parameter LBP dengan R=1 dan P=4 akan digunakan untuk skenario selanjutnya.

4.10.2 Pengujian Terhadap Parameter LVQ

Pengujian ini dilakukan untuk mencari parameter terbaik pada *classifier* LVQ dalam mengidentifikasi dan memverifikasi tanda tangan. Tabel 4.6 menunjukkan hasil pengujian parameter LVQ.

Tabel 4.6 Hasil pengujian parameter LVQ

Parameter		Klasifikasi		Verifikasi			Time
LR	Epoch	Akurasi <i>Training</i>	Akurasi <i>Testing</i>	<i>Accuracy</i>	<i>Sensitivity</i>	<i>Specificity</i>	
0,001	50	86%	84%	57%	69%	48%	22 s
	100	89%	88%	57%	70%	46%	24 s
	150	93%	91%	58%	70%	48%	24 s
	200	92%	90%	56%	70%	45%	25 s
0,01	50	96%	93%	57%	70%	46%	22 s
	100	94%	92%	57%	70%	47%	23 s
	150	96%	93%	58%	70%	47%	25 s
	200	95%	93%	57%	70%	47%	26 s
0,1	50	95%	92,3%	58%	71%	47%	22 s
	100	95%	92,5%	58%	70%	47%	23 s
	150	94%	91,83%	57%	71%	46%	24 s
	200	95%	93%	58%	72%	47%	25 s

Pada Tabel 4.6 dapat diketahui bahwa pada proses klasifikasi hasil akurasi *testing* tertinggi yaitu 93% pada LR=0,01 dengan Epoch 150 dan 200, serta pad LR=0,1 dengan Epoch 200. Pada proses verifikasi, ketiga jenis evaluasi hasil di rata-ratakan nilainya untuk mendapatkan hasil evaluasi paling optimal. Sehingga, hasil evaluasi paling optimal untuk proses verifikasi yaitu pada LR = 0,1 dengan Epoch=200 dimana hasil akurasi sebesar 58%, *sensitivity* sebesar 72% dan *specificity* sebesar 47%. Waktu komputasi pada setiap parameter uji tidak menghasilkan perbedaan yang signifikan sehingga tidak digunakan sebagai patokan untuk menentukan parameter LVQ terbaik.

Sehingga parameter terbaik untuk *classifier* LVQ yaitu LR=0,1 dan Epoch = 200 dan parameter ini akan digunakan untuk skenario selanjutnya.

4.10.3 Pengujian Terhadap Ukuran Citra

Ukuran citra merupakan bagian yang diatur di dalam persiapan sebelum melakukan proses *training* atau bisa disebut sebagai bagian yang ditentukan pada tahap *preprocessing*. Semakin kecil ukuran citra maka detail dari citra itu sendiri semakin tidak terlihat. pada penelitian ini akan diuji 3 ukuran citra yaitu 64×64 , 128×128 dan 256×256 . Hasil dari pengujian dapat dilihat pada Tabel 4.7.

Tabel 4.7 Hasil pengujian ukuran citra

Ukuran citra	Klasifikasi		Verifikasi			Time
	Akurasi Training	Akurasi Testing	Accuracy	Sensitivity	Specificity	
64x64	95%	93%	58%	71%	47%	26 s
128x128	96%	93%	63%	89%	42%	30 s
256x256	96%	91%	62%	91%	40%	35 s

Pada Tabel 4.7 dapat diketahui bahwa dapat diketahui bahwa akurasi *testing* tertinggi yaitu 93% pada ukuran citra 64x64 dan 128x128, kemudian dihitung rata-rata ketiga jenis hasil evaluasi pada proses verifikasi untuk menentukan hasil evaluasi yang paling optimal, maka hasil evaluasi yang paling optimal yaitu pada ukuran citra 128x128 dengan akurasi *testing* yaitu 93%, hasil *accuracy* sebesar 63%, *sensitivity* sebesar 89% dan *specificity* sebesar 42%. Dari segi waktu komputasi, ukuran citra 128x128 membutuhkan waktu yang tidak terlalu signifikan perbedaannya dengan ukuran citra 64x64 yaitu hanya lebih lama 4 detik, sehingga ukuran citra terbaik yaitu pada ukuran citra 128x128.

4.10.4 Pengujian Terhadap Variabel *Threshold*

Pengujian terhadap variabel *threshold* dilakukan untuk mengetahui *threshold* terbaik sebagai tolak ukur untuk menentukan keaslian tanda tangan. Pada pengujian ini, nilai variabel *threshold* berada pada rentang 0,1-0,2 karena melihat dari hasil *similarity* baik data tanda tangan palsu maupun tanda tangan asli sangat mirip. Tabel 4.8 menunjukkan hasil pengujian terhadap variabel *threshold*.

Tabel 4.8 Hasil pengujian terhadap variabel *threshold*

Variabel <i>Threshold</i>	<i>Accuracy</i>	<i>Sensitivity</i>	<i>Specificity</i>
0,1	38%	10%	62%
0,11	48%	26%	66%
0,12	54%	54%	55%
0,13	61%	76%	50%
0,14	62%	85%	45%
0,15	63%	89%	42%
0,16	63%	91%	40%
0,17	62%	93%	37%
0,18	61%	92%	37%
0,19	62%	92%	38%
0,2	62%	93%	37%

Pada Tabel 4.8, ketiga jenis evaluasi hasil yaitu *accuracy*, *sensitivity* dan *specificity* di rata-ratakan untuk mendapatkan evaluasi hasil paling optimal, sehingga dapat diketahui bahwa nilai variabel *threshold* 0,15 dan 0,16 menghasilkan rata-rata akurasi yang tertinggi. Namun, pada variabel *threshold* dengan nilai 0,16 pembagian akurasi tidak rata karena nilai *sensitivity* yang sangat tinggi namun nilai *specificity* sangat rendah, sehingga variabel *threshold* yang paling optimal dipilih yaitu 0,15 dengan menghasilkan hasil evaluasi yaitu *accuracy* sebesar 63%, *sensitivity* yaitu sebesar 89% dan *specificity* sebesar 42%. Karena hasil yang diperoleh pada *accuracy* hanya 63% dan *specificity* yang sangat rendah yaitu 42% yang berarti tanda tangan palsu banyak ditandai sebagai tanda tangan asli hal ini tentunya tidak diinginkan, sehingga perlu dicari tahu penyebab dari banyaknya kesalahan dalam menentukan tanda tangan palsu.

4.11 Pengecekan Kesalahan

Dalam mengetahui kekurangan dalam penelitian ini dilakukan beberapa pengujian. Pertama, dilakukan uji dengan menggunakan metode yang telah dipakai pada penelitian ini yaitu LBP+DCT+LVQ dimana nilai keseluruhan evaluasi hasil akan dibandingkan dengan kombinasi metode lainnya. Kedua, pengujian dengan kombinasi

metode DCT+KNN yang bertujuan untuk mengetahui apakah kombinasi metode LBP dan LVQ tidak cocok digunakan untuk melakukan verifikasi tanda tangan. Ketiga, dicoba menggunakan DCT+LVQ yang bertujuan untuk mengetahui apakah ekstraksi fitur LBP yang tidak cocok digunakan untuk melakukan verifikasi tanda tangan. Keempat, dicoba menggunakan kombinasi LBP+KNN dengan tujuan untuk mengetahui apakah metode LVQ yang tidak cocok digunakan sebagai *classifier*. Dan terakhir, dicoba menggunakan perbandingan *dataset* dari 70:30 menjadi 80:20 untuk mengetahui apakah pembagian *dataset* yang menyebabkan hasil menjadi kurang optimal. Semua pengujian menggunakan parameter terbaik yang telah diperoleh pada skenario uji. Tabel 4.9 menunjukkan hasil verifikasi dengan metode-metode yang telah disebutkan.

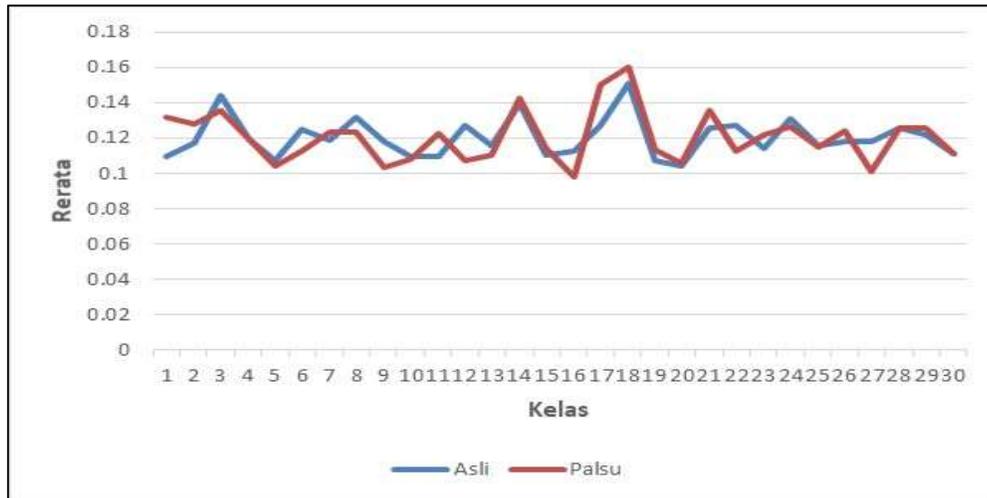
Tabel 4.9 Hasil verifikasi

Uji	<i>Accuracy</i>	<i>sensitivity</i>	<i>specificity</i>	Rata-rata
LBP+DCT+LVQ	63%	89%	42%	64,67%
DCT+LVQ	64%	87%	44%	65%
DCT+KNN	59%	94%	31%	61,33%
LBP+DCT+KNN	60%	92%	34%	62%
<i>Dataset</i> 80:20	52%	73%	34%	53%

Berdasarkan Tabel 4.9 dapat diketahui rata-rata hasil evaluasi verifikasi tanda tangan dengan metode yang digunakan pada penelitian ini yaitu LBP+DCT+LVQ menghasilkan akurasi yang lebih tinggi daripada kombinasi metode DCT+KNN dan kombinasi metode LBP+DCT+KNN yaitu sebesar 64,67%, Namun, dapat kita ketahui bahwa metode DCT+LVQ menghasilkan akurasi 0,33% lebih tinggi dengan rata-rata mencapai 65%. Perbedaan ini tidak terlalu signifikan, sehingga dapat disimpulkan bahwa kesalahan bukan terletak pada metodenya.

Dalam pembagian *dataset*, perbandingan 70:30 yang sebelumnya digunakan juga sudah tepat karena bila dicoba menggunakan pembagian *dataset* 80:20 maka akurasi menurun hal ini disebabkan karena jumlah tanda tangan palsu pada data *testing* jauh lebih banyak yaitu 150 data sedangkan *dataset* tanda tangan asli hanya 90 data sehingga lebih banyak kesalahan yang dapat terjadi. Kemudian, diperhatikan nilai hasil

rata-rata perhitungan *similarity* antara nilai fitur citra uji dengan jumlah rata-rata setiap fitur citra pada kelasnya yang dinyatakan sebagai tanda tangan asli dan palsu.



Gambar 4.10 Persebaran rata-rata nilai *similarity* tanda tangan yang dinyatakan asli dan palsu pada tiap kelas

Berdasarkan Gambar 4.10 dapat diketahui bahwa tanda tangan asli dan tangan palsu pada banyak kelas sangat mirip bahkan ada nilai yang tumpang tindih. Hal ini disebabkan karena aplikasi IBisPaintX yang digunakan untuk mengambil *dataset* merekam setiap proses pada tanda tangan yang dibuat sehingga si peniru mendapat lebih banyak informasi untuk dipelajari terutama dalam menarik garis-garis awal yang dilakukan oleh pemilik tanda tangan asli, sehingga peniru dapat membuat tanda tangan yang sangat mirip dengan aslinya.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan penelitian yang sudah dilakukan, terdapat beberapa hal yang bisa penulis simpulkan antara lain sebagai berikut.

1. Metode LBP dengan menggunakan 10 fitur statistik yaitu *mean*, *standar deviasi*, *skewness*, *entropy*, *kurtosis*, *variance*, *contrast*, *correlation*, *energy*, dan *homogeneity* tidak cocok untuk reorientasi citra tanda tangan karena fitur tumpang tindih dan hanya menghasilkan akurasi 16%
2. Penggunaan metode DCT sebagai pemilihan fitur terbaik setelah ekstraksi fitur LBP meningkatkan akurasi dari 16% menjadi 93,3% dengan koefisien terbaik yaitu 48.
3. Hasil pengujian parameter LBP terbaik yaitu pada R=1 dan P=4 dimana pada proses klasifikasi menghasilkan akurasi *training* yaitu 95% dan akurasi *testing* yaitu 93% dan pada proses verifikasi menghasilkan akurasi sebesar 57%, *sensitivity* sebesar 71%, dan *specificity* sebesar 46%.
4. Hasil pengujian parameter LVQ terbaik yaitu pada LR=0,1 dan Epoch=200 dimana pada proses klasifikasi menghasilkan akurasi *training* yaitu 95% dan akurasi *testing* yaitu 93% dan pada proses verifikasi menghasilkan akurasi sebesar 58%, *sensitivity* sebesar 72%, dan *specificity* sebesar 47%.
5. Waktu komputasi untuk ukuran citra 64x64, 128x128, dan 256x256 tidak memiliki perbedaan waktu yang signifikan. Akurasi paling optimal terdapat pada ukuran citra 128x128.
6. Variabel *threshold* terbaik yaitu pada variabel dengan nilai 0,15 dimana hasilnya yaitu akurasi sebesar 63%, *sensitivity* sebesar 89%, dan *specificity* sebesar 42%.
7. Hasil verifikasi tanda tangan yang kurang optimal disebabkan oleh *dataset* tanda tangan asli dan palsu yang sangat mirip.

5.2 Saran

Ada beberapa saran yang dapat penulis berikan apabila penelitian ini akan dikembangkan kembali antara lain sebagai berikut.

1. Persiapan sebelum penelitian seperti pengambilan citra harus diperhatikan untuk mendapatkan data penelitian yang bagus.
2. Penambahan jumlah *dataset* untuk setiap kelas diperlukan agar *classifier* dapat mempelajari lebih banyak tanda tangan, sehingga akurasi menjadi lebih baik.
3. Penambahan proses segmentasi pada citra sehingga kesalahan pengenalan akibat perbedaan posisi bisa dihindari.
4. Menggunakan ekstraksi fitur lain seperti zoning untuk meningkatkan akurasi verifikasi tanda tangan.
5. Menggunakan algoritma lain dalam melakukan verifikasi tanda tangan sehingga sistem dapat membedakan tanda tangan asli dan tanda tangan palsu lebih baik.

DAFTAR PUSTAKA

- [1] A. Kartono and D. Alamsyah, “Verifikasi Tanda Tangan Dengan Algoritma K-Nearest Neighbor dan Ekstraksi Ciri Harris Corner,” *STMIK GI MDP*, vol. 10, pp. 1–10, 2012.
- [2] R. Abbas, “A Prototype System For O -Line Signature Verification Using Multilayered Feedforward Neural Networks,” *REMIT of Computer Science*, 1995.
- [3] Khamdi M., “LOWONGAN CPNS : 40 Persen Pelamar CPNS Pemkot Solo Palsukan Legalisir Ijazah,” 2013. [Online]. Available: www.solopos.com.
- [4] Y. D. Qur’ani and S. Rosmalinda, “Jarangan Syaraf Tiruan Learning Vector Quantization untuk Aplikasi Pengenalan Tanda Tangan,” in *Seminar Nasional Aplikasi Teknologi Informasi*, 2010, pp. 1–5.
- [5] I. Amelia, A. N. Hermana, and A. Pramana, “Verifikasi Tanda Tangan Dengan Edge Detection dan Metode Learning Vector Quantization,” *MIND J.*, vol. 1, no. 1, pp. 49–56, 2016.
- [6] M. D. Turiyanto, D. Purwanto, and R. Dikairono, “Penerapan Teknik Pengenalan Wajah Berbasis Fitur Local Binary Pattern pada Robot Pengantar Makanan,” Institut Teknologi Sepuluh November (ITS), 2014.
- [7] A. Susanto, D. Sinaga, C. A. Sari, and E. H. Rachmawanto, “A High Performace of Local Binary Pattern on Classify Javanese Character Classification,” *Sci. J. Informatics*, vol. 5, no. 1, pp. 1–8, 2018.
- [8] A. S. Syair, M. I. Sarita, and I. P. Ningrum, “Identifikasi citra tanda tangan menggunakan LBP pada smartphone berbasis android,” *semanTIK*, vol. 5, no. 1, pp. 17–24, 2019.
- [9] I. G. P. S. Wijaya, K. Uchimura, and Z. Hu, “Face Recognition Based on Dominant Frequency Features and Multiresolution Metric,” *Int. J. Immovative Comput. Inf. Control*, vol. 5, no. 1349–4198, pp. 641–651, 2009.
- [10] D. A. Dermawan, S. Mardi, and M. Hariadi, “Simulasi Multi Atribut di Dasarkan pada Agen untuk Keandalan Distribusi Energi Listrik Menggunakan Metode

LVQ,” 2011, pp. 55–63.

- [11] D. M. Wuryandari and I. Afrianto, “Perbandingan Metode Jaringan Syaraf Tiruan Backpropogation dan Learning Vector Quantization Pada Pengenalan Wajah,” *J. Komput. dan Inform.*, vol. 1, no. 1, pp. 1–7, 2012.
- [12] Y. R. Sipayung and S. I. Novichasari, “Identifikasi Pola Tanda Tangan Berbasis Jaringan Syaraf Tiruan dengan Metode Learning Vector Quantization,” *Multimatrix*, vol. I, no. 1, pp. 13–17, 2018.
- [13] J. Arifin and M. Zidny, “Verifikasi Tanda Tangan Asli Atau Palsu Berdasarkan Sifat Keacakan (Entropi),” *J. Inform.*, vol. 9, no. 1, pp. 130–135, 2017.
- [14] L. Listyalina and I. Buyung, “Verifikasi citra tanda tangan berbasis perceptron,” *Teknoin*, vol. 24, no. 2, pp. 135–146, 2018.
- [15] D. Retnoningrum, A. W. Widodo, and M. A. Rahman, “Ekstraksi Ciri Pada Telapak Tangan dengan Metode Local Binary Pattern (LBP),” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 3, pp. 2611–2618, 2019.
- [16] A. Hidayat and Makshun, “Analisis Citra Daun Berdasarkan Fitur Local Binary Pattern dan Fitur Canny Edge Detection Menggunakan Metode Klasifikasi K-Nearest Neighbor (K-NN),” STMIK Eresha, 2019.
- [17] F. N. Achsani, R. D. Atmaja, and R. Purnamasari, “Deteksi Adanya Cacat Pada Kayu Menggunakan Metode Local Binary Pattern,” *e-Proceeding Eng.*, vol. 2, no. 1, pp. 298–305, 2015.
- [18] R. V. Nahari, A. S. Editya, and R. Alfita, “Ekstraksi Fitur Daun Tembakau Berbasis Discrete Cosine Transform (DCT),” *J. Appl. Informatics Comput.*, vol. 4, no. 1, pp. 8–12, 2020.
- [19] F. E. Alfian, I. G. P. S. Wijaya, and F. Bimantoro, “Identifikasi Iris Mata Menggunakan Metode Wavelet Daubechies dan K-Nearest Neighbor,” *JTIKA*, vol. 2, no. 1, pp. 1–10, 2020.
- [20] C. S. Kanimozhiselvi and D. Jayaprakash, “Machine Learning based Autism Grading for Clinical Decision Making,” *Int. J. Recet Technol. Eng.*, vol. 8, no. 4, pp. 7443–7446, 2019.
- [21] M. F. Permana, B. Hidayat, and S. Darana, “Identifikasi Kualitas Kesegaran

- Susu Sapi Melalui Pengolahan Citra Digital Menggunakan Metode Watershad dan Klasifikasi Learning Vector Quantization (LVQ),” *e-Proceeding Eng.*, vol. 5, no. 3, pp. 4851–4858, 2018.
- [22] U. Sudiby, E. H. Kusumaningrum, Purwanti DesiRachmawanto, and C. A. Sari, “Optimasi Algoritma Larning Vector Quantization (LVQ) dalam Pengklasifikasikan Citra Daging Sapi dan Daging Babi,” *J. SIMETRIS*, vol. 9, no. 1, pp. 1–10, 2018.
- [23] A. A. P. B. Dwi, B. Hidayat, and J. Arif, “Identifikasi Sub-Fosil Gigi Graham pada Manusia Berbasis Pengolahan Citra Digital Menggunakan Metode Local Binary Pattern (LBP) dan Klasifikasi Learning Vector Quantization (LVQ),” in *e-proceeding of Engineering*, 2018, vol. 5, no. 2, pp. 2090–2098.
- [24] A. P. Putra, Y. I. Nurhasanah, and A. Zulkarnain, “Deteksi Penyakit Diabetes Retinopati Pada Retina Mata Berdasarkan Pengolahan Citra,” in *Jurnal Teknik Informatika dan Sistem Informasi*, 2017, vol. 3, no. 2, pp. 376–390.
- [25] M. Shams, M. Rashad, and O. Nomir, “Iris Recognition Based on LBP and Combined LVQ Classifier,” *Intrnational J. Comput. Sci. Inf. Technol.*, vol. 3, no. 5, pp. 67–78, 2011.
- [26] D. A. Dharmawan, “Deteksi Kanker Serviks Otomatis Berbasis Jaringan Saraf Tiruan LVQ dan DCT,” *JNTETI*, vol. 3, no. 4, pp. 3–6, 2014.
- [27] D. Putra, *Sistem Biometrika*. Yogyakarta: Andi, 2009.
- [28] R. Munir, *Pengolahan Citra Digital dengan Pendekatan Algoritmatik*. Bandung: Informatika, 2004.
- [29] A. Hayatunnufus, Andrizal, and D. Yendri, “Pendeteksi dan verifikasi tanda tangan menggunakan metode image domain spasial,” pp. 1–8, 2014.
- [30] R. C. Gonzalez and R. E. Woods, *Digital Image Processing Third Edition*, 3rd ed. London: Pearson Prentice HAll, 2008.
- [31] J. Y. Sari and R. A. Saputra, “Pengenalan Finger Vein Menggunakan Local Line Binary Pattern dan Learning Vector Quantization,” *Ultim. Comput.*, vol. 9, no. 2, pp. 52–57, 2017.
- [32] J. Mun, Y. Jang, Y. Nam, and J. Kim, “Edge-enhancing bi-histogram

- equalisation using guided image filter,” *J. Vis. Commun. Image Represent.*, vol. 58, no. 13, pp. 688–700, 2019.
- [33] R. Kasmala, A. B. Purba, and U. T. Lenggana, “Kompresi Citra Dengan Menggabungkan Metode Discrete Cosine Transform (DCT) dan Algoritma Huffman,” *JOIN*, vol. 2, no. 1, pp. 1–9, 2017.
- [34] D. M. Wuryandari and I. Afrianto, “Perbandingan Metode Jaringan Syaraf Tiruan Backpropogation dan Learning Vector Quantization Pada Pengenalan wajah,” *J. Komput. dan Inform.*, vol. 1, no. 1, pp. 45–51, 2012.
- [35] L. Fausett, *Fundamental of Neural Networks, Architecture, Algorithms, and Application*, 1st ed. Englewood: Pearson, 1994.
- [36] S. R. Wurdianarto, S. Novianto, and U. Rosyidah, “Perbandingan Euclidean Distance dengan Canberra Distance pada Face Recognition,” *Techno.COM*, vol. 13, no. 1, pp. 31–37, 2014.
- [37] M. Nishom, “Perbandingan Akurasi Euclidean Distance , Minkowski Distance , dan Manhattan Distance pada Algoritma K- Means Clustering berbasis Chi-Square,” *J. Inform. J. Pengerbangan IT(JPIT)*, vol. 04, no. 01, pp. 20–24, 2019.
- [38] M. Sokolova and G. Lapalme, “A systematic analysis of performance measures for classification tasks,” *Inf. Process. Manag.*, vol. 45, no. 4, pp. 427–437, 2019.