



Identifikasi Huruf Kapital Tulisan Tangan Menggunakan *Linear Discriminant Analysis* dan *Euclidean Distance*

Septa Cahyani*, Rita Wiryasaputra, Rendra Gustriansyah

Jurusan Teknik Informatika, Fakultas Ilmu Komputer, Universitas Indo Global Mandiri

Naskah Diterima : 23 Oktober 2017; Diterima Publikasi : 15 Maret 2018

DOI : 10.21456/vol8iss1pp57-67

Abstract

The human ability to recognize a variety of objects, however complex the object, is the special ability that humans possess. Any normal human will have no difficulty in recognizing handwriting objects between an author and another author. With the rapid development of digital technology, the human ability to recognize handwriting objects has been applied in a program known as Computer Vision. This study aims to create identification system different types of handwriting capital letters that have different sizes, thickness, shape, and tilt (distinctive features in handwriting) using Linear Discriminant Analysis (LDA) and Euclidean Distance methods. LDA is used to obtain characteristic characteristics of the image and provide the distance between the classes becomes larger, while the distance between training data in one class becomes smaller, so that the introduction time of digital image of handwritten capital letter using Euclidean Distance becomes faster computation time (by searching closest distance between training data and data testing). The results of testing the sample data showed that the image resolution of 50x50 pixels is the exact image resolution used for data as much as 1560 handwritten capital letter data compared to image resolution 25x25 pixels and 40x40 pixels. While the test data and training data testing using the method of 10-fold cross validation where 1404 for training data and 156 for data testing showed identification of digital image handwriting capital letter has an average effectiveness of the accuracy rate of 75.39% with the average time computing of 0.4199 seconds.

Keywords: Computer vision; Euclidean distance; Linear discriminant analysis; 10-Fold Cross Validation.

Abstrak

Kemampuan manusia dalam mengenali berbagai macam objek, seberapa pun rumitnya objek tersebut, merupakan kemampuan istimewa yang dimiliki manusia. Manusia normal manapun tidak akan mengalami kesulitan dalam mengenali objek tulisan tangan antara seorang penulis dengan penulis lainnya. Permasalahannya apabila komputer melakukan pengenalan tulisan tangan yang memiliki ukuran, ketebalan, bentuk, dan kemiringan yang berbeda (ciri khas tersendiri dalam menulis dengan tulisan tangan). Dengan pesatnya perkembangan teknologi digital maka kemampuan manusia untuk mengenali objek tulisan tangan telah diterapkan dalam suatu program yang dikenal dengan nama *Computer Vision*. Penelitian ini bertujuan membuat sistem identifikasi berbagai jenis huruf kapital tulisan tangan menggunakan metode *Linear Discriminant Analysis* (LDA) dan *Euclidean Distance*. LDA digunakan untuk mendapatkan karakteristik ciri dari citra dan memberikan jarak antara kelas menjadi lebih besar, sedangkan jarak antara data *training* dalam satu kelas menjadi lebih kecil, sehingga waktu pengenalan citra digital huruf kapital tulisan tangan dengan menggunakan *Euclidean Distance* menjadi lebih cepat waktu komputasi (dengan mencari jarak terdekat antara data *training* dengan data *testing*). Hasil pengujian data sampel menunjukkan bahwa resolusi citra sebesar 50x50 piksel merupakan resolusi citra yang tepat digunakan untuk data sebanyak 1560 data huruf kapital tulisan tangan dibandingkan dengan resolusi citra 25x25 piksel dan 40x40 piksel. Sedangkan hasil pengujian data *training* dan data *testing* menggunakan metode *10-fold cross validation* dimana 1404 untuk data *training* dan 156 untuk data *testing* menunjukkan pengidentifikasian citra digital huruf kapital tulisan tangan memiliki efektifitas rata-rata tingkat akurasi sebesar 75,39 % dengan rata-rata waktu komputasi sebesar 0,4199 detik.

Kata Kunci : Computer vision; Euclidean distance; Linear discriminant analysis; 10-Fold Cross Validation

1. Pendahuluan

Pengenalan tulisan tangan merupakan pekerjaan yang sangat mudah dilakukan oleh manusia. Manusia tidak sulit dalam mengenali sebuah tulisan tangan walaupun berbeda-beda antara penulis satu dengan penulis yang lain, akan tetapi pekerjaan tersebut sulit

dilakukan oleh komputer. Permasalahan yang muncul dalam melakukan proses pengenalan tulisan tangan adalah bagaimana sebuah teknik pengenalan dapat mengenali berbagai huruf dengan ukuran, ketebalan, bentuk, dan kemiringan yang berbeda antara penulis satu dengan penulis lainnya (Wirayuda *et al.*, 2009). Dengan pesatnya perkembangan teknologi digital

*) Penulis korespondensi: chatata23@gmail.com

maka kemampuan manusia untuk mengenali objek telah diterapkan dalam bidang *Computer Vision*. *Computer vision* merupakan salah satu ruang lingkup *artificial intelligence* (AI) tentang bagaimana membuat kemampuan sebuah mesin yang dapat berpikir, menimbang tindakan yang diambil, dan mampu mengampil keputusan layaknya seorang manusia (Sutojo *et al.*, 2011).

Secara umum, ada empat tahapan dalam proses pengenalan objek yaitu akuisisi citra, *preprocessing*, ekstraksi fitur, dan klasifikasi. Dari keempat tahapan tersebut, tahapan ekstraksi ciri dan klasifikasi merupakan komponen yang sangat penting dalam proses sistem pengenalan objek (Achmad, 2006).

Meskipun metode untuk ekstraksi ciri yang paling populer adalah *Principal Component Analysis* atau sering disebut PCA. Namun demikian PCA memiliki kelemahan yaitu pemisahan antar kelas yang kurang optimal, sehingga metode LDA dibuat untuk mengatasi kekurangan PCA. Metode LDA mampu memisahkan data antar kelas menjadi lebih terpisah dengan cara memaksimalkan nilai *between-class scatter* dan meminimalkan *within-class scatter*. (Muntasa, 2015). Metode LDA digunakan dalam penelitian ini sebagai metode ekstraksi fitur untuk mengoptimalkan pemisahan antar kelas sehingga hasil pengenalannya lebih akurat. Dengan demikian, metode LDA akan diuji berdasarkan tingkat akurasi pengenalan.

Setelah melalui tahapan ekstraksi fitur, fitur-fitur penting dari citra huruf kapital akan digunakan untuk proses klasifikasi atau pengenalan yang akan digunakan dalam penelitian ini adalah metode *euclidean distance*. Metode *euclidean distance* merupakan metode pengukuran dengan menghitung kedekatan antara kasus baru dengan kasus lama, yaitu berdasarkan pada pencocokan bobot dari sejumlah fitur yang ada, dari bobot atau nilai dari *eigen distance* yang didapatkan dicari dengan nilai yang paling kecil sehingga untuk digunakan pada proses pengenalan citra (Muntasa, 2015). Penelitian ini bertujuan untuk membuat aplikasi yang dapat mengidentifikasi huruf kapital tulisan tangan menggunakan metode *Linear Discriminant Analysis* (LDA) dan *Euclidean Distance* yang diharapkan dapat melakukan pengidentifikasian citra digital huruf kapital tulisan tangan dengan tingkat akurasi pengenalan yang baik.

2. Kerangka Teori

2.1. Pengenalan Pola

Pengenalan pola telah berkembang pesat dalam penelitian komputasi terutama untuk komputasi citra. Istilah “pola” erat hubungannya dengan objek yang memiliki banyak ciri dan istilah “pengenalan” erat hubungannya dengan pengelompokan dari suatu objek berdasarkan ciri-ciri yang dimiliki. Secara umum pengenalan pola adalah proses untuk

mengenali objek secara otomatis berdasarkan ciri yang didasarkan pada data pelatihan. Suatu objek lebih mudah untuk dikenali, jika memiliki nilai pembeda yang besar dengan objek-objek lain. Untuk mengenali suatu objek, maka harus memiliki standart ciri dari masing-masing objek sehingga objek yang diisyaratkan harus sama (Muntasa, 2015). Pengenalan pola ini bersifat *conceptually driven processing* yang berarti proses dimulai dari pembentukan konsep pada objek yang akan dijumpai (Hastiana, 2010).

Pada prinsipnya untuk melakukan proses pengenalan umumnya mempunyai beberapa tahap, diantaranya adalah:

- a. Pemrosesan awal (deteksi/segmentasi)
- b. Ekstraksi fitur digunakan untuk memperoleh ciri pola
- c. Pengenalan pola atau pencocokan ciri hasil ekstraksi fitur

Ada dua pendekatan pengenalan pola yang sering digunakan yaitu metode teori *statistic* dan metode teori Bahasa formal *automata* atau sintaktik. Ada banyak aplikasi yang bisa dijadikan implementasi dari pengenalan pola, mulai dari pengenalan wajah pada manusia, pengenalan penyakit berdasarkan gejala-gejala, sampai dengan pengenalan tulisan tangan manusia (Muntasa, 2015).

2.2. Pengenalan Pola Untuk Pengenalan Tulisan Tangan

Pengenalan tulisan tangan (*Handwriting Recognition*) adalah salah satu kemampuan komputer untuk menerima dan menafsirkan *input* tulisan tangan yang dapat dimengerti dari sumber dokumen kertas, foto, layar sentuh, dan perangkat lainnya (Kasih dan Yulia, 2015).

Pengenalan tulisan tangan atau cetak merupakan aplikasi pengenalan pola yang telah banyak diimplementasikan. Mulai dari pengenalan huruf tulisan tangan sampai dengan pengenalan tulisan tangan *multi line* (Muntasa, 2015).

2.3. Akuisisi Citra

Akuisisi citra adalah tahap awal untuk mendapatkan *input* citra digital untuk data pelatihan maupun data pengenalan. Tujuan dari akuisisi citra adalah untuk menentukan data yang diperoleh dan memilih metode perekaman citra digital. Dimulai dari objek yang akan diambil gambarnya, persiapan alat-alat, sampai dengan pencitraan. Pencitraan adalah kegiatan transformasi dari citra tampak (foto, gambar, dan sebagainya) yang diubah menjadi citra digital. Beberapa alat yang dapat digunakan untuk pencitraan, seperti kamera digital dan *scanner*.

2.4. PreProcessing Citra

Proses penyalarsan citra atau *preprocessing* citra merupakan tahap yang diperlukan untuk memperoleh

lokasi objek secara tepat yang akan dikenali dengan memperbaiki kualitas citra digital sehingga akan memperlancar proses ekstraksi ciri atau fitur. Proses deteksi atau segmentasi ini terkadang juga dapat diabaikan, ketika objek yang akan dikenali sudah terdeteksi secara tepat lokasinya, misalnya pengenalan wajah yang menggunakan data wajah ORL. Data citra ORL yang disediakan dapat langsung diekstraksi dan diikuti dengan pengukuran kemiripan tanpa melalui *preprocessing* citra terlebih dahulu (Muntasa, 2015).

Pada tahap *preprocessing*, setiap kasus mempunyai *preprocessing* yang berbeda-beda. Jadi tidak ada standar yang harus dilakukan untuk melakukan *preprocessing* citra. Beberapa *preprocessing* yang digunakan dalam penelitian ini adalah sebagai berikut (Sutoyo *et al.*, 2009):

a. Merubah ukuran citra menjadi ukuran tertentu (*Resize*).

Resizing citra atau resolusi citra merupakan tahap pertama dari proses penyesuaian citra, pada tahap ini dilakukan proses pembesaran atau pengecilan dimensi citra menjadi dimensi yang telah ditentukan, sehingga tidak ada perbedaan dimensi dari matriks data citra dan memudahkan langkah selanjutnya.

b. Konversi dari citra warna ke citra keabuan (*Grayscale*).

Grayscale citra merupakan tahapan kedua dari proses penyesuaian citra, pada tahap ini dilakukan proses konversi citra *true color* yang memiliki 3 (tiga) parameter warna RGB yaitu *Red* (merah), *Green* (hijau), dan *Blue* (biru) menjadi citra keabuan yang memiliki satu atribut. *Grayscale* merupakan tahap yang penting karena jika citra warna RGB ini dimasukkan ke dalam proses ekstraksi, maka proses tersebut akan sulit dilakukan karena citra RGB terdiri dari tiga parameter, oleh karena itu diperlukan penyamaan parameter menjadi satu parameter. Citra RGB bisa diubah menjadi citra *grayscale* dengan menggunakan persamaan (1):

$$f_o(x, y) = \frac{f_i^R(x, y) + f_i^G(x, y) + f_i^B(x, y)}{3} \quad (1)$$

c. Konversi citra keabuan menjadi citra biner (*Binerisasi*).

Binerisasi citra adalah tahap terakhir dari proses penyesuaian citra, pada tahap ini dilakukan proses konversi citra warna keabuan menjadi citra warna hitam putih atau sering disebut citra biner (*binary image*). Tahap ini bertujuan untuk membedakan daerah mana yang objek dan daerah mana yang latar belakang (*background*) dari citra secara jelas, Metode yang digunakan dalam binerisasi adalah metode operasi ambang tunggal dengan menggunakan persamaan (2):

$$f_o(x, y) = \begin{cases} 0, & f_i(x, y) < 128 \\ 255, & f_i(x, y) \geq 128 \end{cases} \quad (2)$$

Hasilnya, piksel-piksel yang nilai intensitasnya di bawah 128 diubah menjadi hitam (nilai intensitas = 0), sedangkan piksel-piksel yang nilai intensitasnya di atas 128 diubah menjadi putih (nilai intensitas = 255).

2.5. Ekstraksi Fitur

Ekstraksi fitur (*feature extraction*) adalah sebuah proses untuk mengambil atau mengekstrak karakteristik atau informasi yang penting dari suatu citra untuk digunakan dalam proses klasifikasi. Tujuan dari *feature extracton* adalah:

- Menghilangkan data gambar yang berulang (*redundan*).
- Menghilangkan data gambar yang kurang atau tidak memiliki arti dalam klasifikasi dan pengenalan.
- Rekonstruksi data agar optimal dalam proses klasifikasi dan pengenalan.

Agar komputer dapat mengenali fitur-fitur karakter tersebut diperlukan sebuah metode atau algoritma. Beberapa metode yang digunakan untuk mengenali karakter, dapat dikategorikan ke dalam fitur *geometris*, fitur struktural dan fitur *space transformation*. Berikut penjelasannya:

- Fitur *geometris* merupakan metode yang digunakan untuk menangkap ciri - ciri berupa arah suatu karakter. Metode yang digunakan dalam kategori fitur geometri adalah *moments*, *histogram* dan *direction feature*.
- Fitur struktural digunakan untuk mengetahui ciri-ciri bentuk dari suatu karakter seperti titik awal dan akhirnya suatu karakter atau percabangan pada karakter serta ketebalan suatu karakter. Metode yang digunakan pada fitur struktural adalah *image registration*, *hough transform*, *line-based representation*, *fourier descriptor*, *shape approximation*, and *topologi feature*.
- Fitur *space transformation* digunakan untuk mengubah representasi dari pola untuk meningkatkan kinerja klasifikasi serta dimensi suatu gambar. Metode yang digunakan untuk *transformation feature* seperti *linear transform*, *kernels*, *principal component analysis (PCA)*, *linear discriminant analysis (LDA)*.

2.6. Vector

Vector adalah sebuah larik satu dimensi dari bilangan-bilangan yang tersusun dalam baris atau kolom (Supardi, 2010). Membangun *vector* dari sebuah citra dilakukan dengan penggabungan sederhana. Ada dua cara yang digunakan untuk melakukan transformasi dari *vector* dua dimensi menjadi satu dimensi yaitu transformasi baris atau transformasi kolom (Muntasa, 2015).

2.7. Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis (LDA) atau *Fisher's Linear Discriminant* merupakan salah satu metode yang menggunakan teori statistik yang sudah banyak digunakan secara luas baik dalam hal pembelajaran mesin, pengolahan data, maupun pengolahan citra. Metode ini pertama kali dibuat dan dipublikasikan oleh Ronald A. Fisher melalui paper *The Use of Multiple Measure in Taxonomic Problems* pada tahun 1936. LDA adalah metode ekstraksi fitur dengan perpaduan dari perhitungan operasi matematika dan statistika yang menggunakan properti statistik terpisah untuk tiap obyek (Sholahuddin *et al.*, 2010).

Meskipun metode untuk ekstraksi ciri yang paling populer adalah *Principal Component Analysis* (PCA). Namun demikian PCA memiliki kelemahan yaitu pemisahan antar kelas yang kurang optimal, sehingga metode LDA dibuat untuk mengatasi kekurangan PCA. Metode LDA mampu memisahkan data antar kelas menjadi lebih terpisah dengan cara memaksimalkan nilai *between-class scatter* dan meminimalkan *within-class scatter*. PCA dan LDA mempunyai perbedaan yang sangat jelas, karena pengklasifikasian terhadap ciri dapat dilakukan oleh PCA sedangkan LDA berfokus pada pengklasifikasian terhadap data. Pada ekstraksi fitur menggunakan LDA, data set lokasinya tetap, namun kelas yang dibentuk menjadi lebih terpisah sehingga kondisi ini menyebabkan jarak antar kelas menjadi lebih besar, sedangkan jarak antar data pelatihan dalam satu kelas menjadi lebih kecil.

Jumlah fitur yang dihasilkan oleh LDA dihitung dari sebanyak jumlah kelas dikurangi dengan satu. Dengan kata lain, jumlah fitur yang dihasilkan LDA tergantung dengan jumlah kelas dan, beberapa jumlah pose yang telah dilatih oleh LDA, dan LDA tidak mempengaruhi jumlah fitur yang dihasilkan sehingga akan membutuhkan waktu lebih sedikit saat proses ekstraksi ciri dan juga proses pengenalan citra. Berikut langkah-langkah proses ekstraksi ciri menggunakan LDA (Muntasa, 2015):

- Mengubah matriks dua dimensi menjadi satu dimensi atau ke dalam *vector* baris atau *vector* kolom.
- Mengelompokkan data *training* (pelatihan) ke dalam matriks sejumlah kelas (x_i).
- Hitung nilai rata-rata (*mean*) dari tiap kelas (μ_i). Perhitungan nilai *mean* dari tiap kelas dapat dihitung dengan persamaan (3):

$$\mu_i = \frac{1}{N_i} \sum_{x \in \omega_i} x \quad (3)$$

Jika setiap data pelatihan ditransformasikan ke dalam bentuk *vector* baris, maka perhitungan *mean* dimensi menggunakan model kolom. Jika data pelatihan ditransformasikan ke dalam bentuk kolom, maka *mean* dimensi dihitung berdasarkan

baris, sehingga jumlah dimensi *mean* yang dihasilkan sama dengan jumlah dimensi satu data pelatihan bukan dimensi jumlah dataset.

- Hitung nilai *mean* total dari semua kelas (μ). Perhitungan nilai *mean* total dari keseluruhan kelas dapat dihitung dengan persamaan (4):

$$\mu = \frac{1}{N_i + \dots + N_c} \sum_{x \in \omega_i} x \quad (4)$$

- Hitung Matriks *Between Class Scatter* (S_B) dan Matriks *Within Class Scatter* (S_W). Perhitungan matriks S_B dapat dihitung dengan persamaan (5) sedangkan matriks S_W dapat dihitung dengan menggunakan persamaan (6):

$$S_B = \sum_{i=1}^c N_i (\mu_i - \mu)^T (\mu_i - \mu) \quad (5)$$

$$S_W = \sum_{i=1}^c \sum_{j=1}^{N_i} ((x_j - \mu_i)^T (x_j - \mu_i)) \quad (6)$$

Dimana c merupakan jumlah seluruh kelas, N_i adalah jumlah sampel tiap kelas, dan i adalah mewakili jumlah kelas dari seluruh kelas.

- Hitung nilai *covariance* matriks (S). Perhitungan nilai matriks kovarian dapat dihitung dengan menggunakan persamaan (7):

$$\text{ArgMax } S = S_B * (S_W)^{-1} \quad (7)$$

Dimana *ArgMax S* adalah mencari nilai tertinggi dari matriks kovarian yang merupakan matriks hasil reduksi dari proses ekstraksi *linear discriminant analysis* (LDA) yang memiliki dimensi lebih kecil dibandingkan dengan dimensi matriks citra asli.

- Hitung *eigenvalue* (v) dan *eigenvector* (d). Setelah nilai *ArgMax S* ditentukan maka langkah selanjutnya menentukan matriks *eigenvalue* (λ) dan matriks *eigenvector* (v) dapat dihitung dengan persamaan (8):

$$|(S_B S_W^{-1})^T - \lambda I| = 0 \quad (8)$$

Setelah itu dilakukan proses faktorisasi, maka akan didapatkan nilai yang nantinya akan digunakan untuk mencari nilai *eigenvector* (v), dengan persamaan (9):

$$[v, d] = \text{eig}(S)$$

$$|S - \lambda I|v = 0 \quad (9)$$

Dimana v merupakan matriks kolom dengan elemen (x_1, x_2, \dots, x_i) di dalamnya, matriks inilah yang disebut *eigenvector*. Kedua matriks tersebut dikalikan hingga didapatkan sebuah persamaan (10):

$$(S_{11} - \lambda_1)x_1 + S_{12}x_2 + \dots + S_{cn}x_n = 0 \quad (10)$$

Hasil proyeksi *eigenvector* yang berkorelasi dengan *eigenvalue* lebih mudah dipisahkan dibandingkan menggunakan *eigenvalue* yang berkorelasi dengan *eigenvalue* yang lebih kecil.

- h. Hitung matriks proyeksi dan matriks bobot. Perhitungan matriks proyeksi dapat dihitung dengan persamaan (11). Setelah itu, menghitung matriks bobot dari proses ekstraksi LDA dapat dihitung dengan persamaan (12).

$$W_{\text{train}} = (x - \mu_i)^T * v \quad (11)$$

$$Eigen_{\text{train}} = (x - \mu_i) * v \quad (12)$$

2.8. Euclidean Distance

Euclidean Distance merupakan metode pengukuran kemiripan yang mensyaratkan agar ukuran data yang diukur mempunyai resolusi citra yang sama. Dengan mencari kasus dan menghitung kedekatan antara kasus baru dengan kasus lama, yaitu berdasarkan pada pencocokan bobot dari sejumlah fitur yang ada, dari bobot atau nilai dari *eigen distance* yang didapatkan dicari nilai yang paling kecil yang merupakan hasil akhir dari proses pengenalan citra. Berikut persamaan (13) dari metode *Euclidean Distance* yang berfungsi untuk mendapatkan nilai dari *eigen distance* (d_v) (Muntasa 2015), (Falanda *et al.*, 2016).

$$d_v = \sqrt{\sum_{i=1}^m (Eigen_{\text{train ke } i} - Eigen_{\text{uji}})^2} \quad (13)$$

Dimana $Eigen_{\text{train ke } i}$ adalah bobot citra pelatihan ke i , $Eigen_{\text{uji}}$ adalah data bobot huruf *testing*, dan m adalah jumlah data pelatihan

2.9. K-Fold Cross Validation

Cross-validation adalah bentuk sederhana dari teknik *statistic*. Jumlah *fold* standar untuk memprediksi tingkat *error* dari data adalah dengan menggunakan *k-fold cross validation* (Witten *et al.*, 2011). Metode ini tidak menghabiskan waktu cukup banyak untuk membuat variasi data uji karena proses ini diulang sebanyak k kali untuk menentukan rata-rata keberhasilan, lebih sederhana dan lebih mudah dibandingkan metode *leave-one-out cross validation* (Pratiwi, 2014). Keunggulan dari model ini dibandingkan dengan model *cross validation* lainnya adalah tidak peduli bagaimana data dibagi, setiap *subset* k akan divalidasi sebanyak satu kali dan juga akan dijadikan data *training* sebanyak satu kali, sehingga hasil evaluasi menjadi lebih baik. Untuk penggunaan jumlah *fold* terbaik untuk uji validitas, dianjurkan menggunakan *10-fold cross validation* dalam model (Han *et al.*, 2012).

Penelitian ini akan menggunakan *10-fold cross validation* untuk pembagian data *training* dan data *testing* dan hasil pengujian akurasi sistem dihitung dengan jumlah data uji yang benar dibagi dengan jumlah data uji kemudian dikalikan 100%. Secara umum dapat dilihat pada persamaan (14) (Simangunsong, 2015):

$$\text{Presentase Pengujian} = \frac{\sum \text{data uji benar}}{\sum \text{data uji}} \times 100\% \quad (14)$$

3. Metode

3.1. Identifikasi Masalah

Identifikasi masalah yang bertujuan untuk mendapatkan tujuan dalam mengatasi permasalahan tersebut. Kemampuan manusia dalam mengenali berbagai macam pola huruf kapital antara penulis satu dengan penulis lainnya, telah diterapkan dalam suatu program yang dikenal dengan nama *computer vision*. Permasalahannya adalah bagaimana sebuah komputer dapat melakukan pengenalan tulisan tangan yang memiliki ukuran, bentuk, dan kemiringan yang berbeda antara penulis satu dengan penulis lainnya. Sehingga, penelitian ini bertujuan untuk mengidentifikasi berbagai jenis huruf kapital tulisan tangan menggunakan metode *Linear Discriminant Analysis* (LDA) dan *Euclidean Distance*.

3.2. Pengumpulan Data

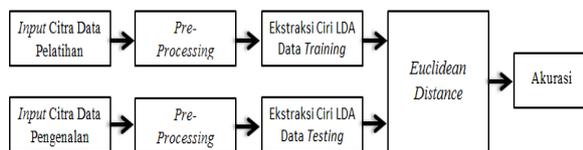
Pengumpulan data bertujuan untuk mendapatkan bahan dan alat yang dibutuhkan untuk mencapai tujuan penelitian. Bahan yang digunakan sebagai penelitian adalah data citra digital huruf kapital tulisan tangan yang dibuat pada kertas A4 berwarna putih dan dikumpulkan sebanyak 1560 pola huruf kapital tulisan tangan yang terdiri dari A-Z (26 pola huruf kapital) dari 60 orang. Setelah itu, data tersebut di *scanner* dan disimpan dalam format JPEG (*.jpg). Pembagian untuk data *training* dan data *testing* menggunakan metode *k-fold cross validation*.

Alat yang digunakan dalam penelitian ini adalah kebutuhan perangkat keras (*hardware*) dengan spesifikasi *processor* dengan Intel® Core™ i3-3217U CPU, *installed memory* (RAM) 3GB, dan *system type* 64-bit Operation System, *x64-based processor*. Sedangkan kebutuhan perangkat lunak (*software*) yang digunakan adalah sistem Operasi Windows 8.1 *Single Language* (64-bit), *Java Standar Edition Development Kit* versi 1.8, *Astah Community 6.6.3* (*Model Version* : 36), *Edraw Max* versi 8.2, dan *NeatBeans IDE* 8.1.

3.3. Analisis Sistem

Analisis sistem bertujuan untuk mendapatkan alur proses identifikasi huruf kapital tulisan tangan. Proses pengidentifikasian huruf kapital tulisan tangan pada penelitian ini dapat dilihat pada Gambar 1, yang meliputi tahap pelatihan dan tahap pengenalan. Pada

tahap pelatihan digunakan untuk melakukan proses pembelajaran citra digital huruf kapital tulisan tangan berdasarkan ciri, mulai dari proses *input* cita, *preprocessing*, ekstraksi ciri, sampai hasil akhir dari tahap pelatihan adalah sistem akan menyimpan semua hasil proses ekstraksi ciri LDA data *training*. Pada tahap pengenalan digunakan untuk melakukan proses identifikasi citra digital huruf kapital tulisan tangan, mulai dari proses *input* cita, *preprocessing*, ekstraksi ciri, sampai pencocokan ekstraksi ciri LDA data *training* dengan ekstraksi ciri LDA data *testing* menggunakan *euclidean distance*. Hasil akhir tahap pengenalan adalah informasi apakah citra huruf kapital tulisan tangan teridentifikasi benar atau teridentifikasi salah serta waktu komputasi identifikasi citra digital huruf kapital tulisan tangan, sehingga akan didapatkan hasil akurasi dan waktu komputasi dari aplikasi identifikasi huruf kapital tulisan tangan dengan menggunakan LDA dan *euclidean distance*.



Gambar 1. Alur Proses Identifikasi Huruf Kapital Tulisan Tangan

3.3.1. Input Citra

Hasil dari akuisisi data yang berupa huruf kapital tulisan tangan yang digunakan sebagai data pelatihan dan data pengujian. Selanjutnya, citra huruf kapital tulisan tangan dilakukan proses *cropping* secara manual, sehingga *input* citra dalam penelitian ini hanya berisi *content* huruf kapital tulisan tangan saja. Selanjutnya, secara otomatis sistem akan melakukan *preprocessing*

3.3.2. PreProcessing Citra

Sebelum masuk proses ekstraksi ciri LDA baik ekstraksi ciri LDA data *training* maupun ekstraksi ciri LDA data *testing*, memerlukan proses pengolahan citra yang digunakan untuk perbaikan kualitas citra. Beberapa metode untuk proses penyesuaian citra (*preprocessing*) adalah sebagai berikut:

a. Resizing Citra

Sebelum menentukan resolusi citra yang digunakan, maka dilakukan pengujian data sampel dengan empat resolusi citra yang berbeda yaitu 25x25 piksel, 40x40 piksel, 50x50 piksel, dan 60x60 piksel dengan 1404 data *training* untuk proses ekstraksi ciri dan 156 data *testing* untuk proses pengenalan huruf kapital, data *testing* dalam pengujian ini bukan bagian dari data *training*. Data latih dan data uji yang digunakan dalam pengujian resolusi citra yang berbeda adalah sama antara resolusi citra 25x25

piksel, 40x40 piksel, 50x50 piksel, dan 60x60 piksel yang dapat dilihat pada Tabel 1.

Tabel 1. Hasil pengujian data sampel resolusi citra

Resolusi Citra	Hasil Pengujian		Rerata Akurasi	Rerata Kecepatan
	Data Latih	Data Uji		
25x25	1404	156	78.85%	0.0229 second
40x40	1404	156	80.77%	0.1238 second
50x50	1404	156	82.05%	0.2450 second

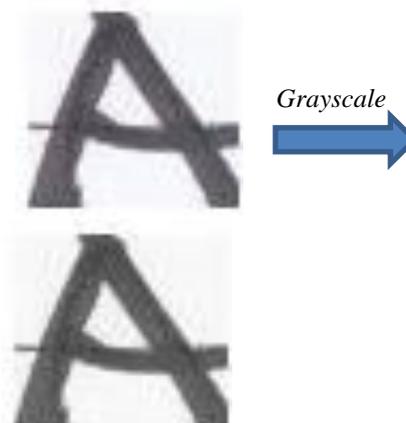
Resolusi citra 60x60 piksel tidak dapat dilakukan karena terjadi nilai singular (nilai determinannya sama dengan 0). Berdasarkan hasil pengujian yang telah dilakukan sebanyak tiga kali dengan menggunakan resolusi yang berbeda, dapat ditarik kesimpulan bahwa resolusi yang lebih baik untuk digunakan dalam proses *resizing* citra dalam penelitian ini adalah resolusi sebesar 50x50 piksel karena memiliki tingkat akurasi yang lebih besar dari dua resolusi citra yang lain, walaupun memiliki tingkat kecepatan yang lebih lama

b. Grayscale Citra

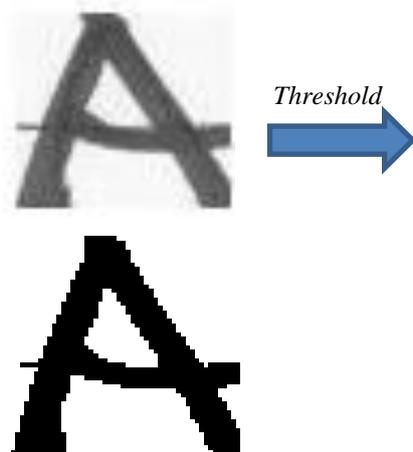
Pada Gambar 2 menampilkan contoh gambar huruf A kapital yang berubah dari citra RGB menjadi citra keabuan dengan menggunakan rumus persamaan (1), citra tersebut telah mengalami *resizing* citra 50x50 piksel sebelumnya.

c. Binerisasi Citra

Pada Gambar 3 menampilkan contoh gambar huruf A kapital yang berubah dari gambar *grayscale* menjadi citra hitam putih menggunakan rumus persamaan (2), gambar tersebut merupakan terusan dari citra *grayscale* sebelumnya.



Gambar 2. Huruf A yang mengalami proses Grayscale



Gambar 3. Huruf A yang mengalami proses binerisasi

3.3.3. Ekstraksi Ciri LDA

Proses ekstraksi ciri LDA huruf kapital tulisan tangan pada penelitian ini, terbagi menjadi dua ekstraksi ciri meliputi ekstraksi ciri LDA data pelatihan dan ekstraksi ciri LDA data pengujian atau pengenalan. Ekstraksi ciri pada proses pelatihan dilakukan dengan menggunakan metode LDA. Tahap ini bertujuan untuk mendapatkan ciri-ciri yang terpilih dari masukan data-data pelatihan. Ciri-ciri yang terpilih nantinya digunakan untuk proses pengenalan dan untuk ekstraksi ciri LDA data pengujian. Ekstraksi ciri LDA pengujian dilakukan dengan melakukan proyeksi dengan mengambil hasil ekstraksi ciri LDA data pelatihan. Hasil ekstraksi ciri LDA data pengujian ini nantinya digunakan untuk proses pengenalan ciri.

3.3.4. Pengenalan Ciri Euclidean Distance

Proses pengenalan dilakukan dengan menggunakan *euclidean distance* dengan menghitung nilai *distance* antara data *testing* dengan data *training* (*template*), kemudian dicari jarak yang terdekat (*minimum distance*) dan dijadikan sebagai hasil akhir dari proses identifikasi huruf kapital tulisan tangan, sehingga dapat diketahui akurasi dan kecepatan pengenalan dari aplikasi pengidentifikasi huruf kapital tulisan tangan dengan menggunakan teknik *statistic* yaitu *10-fold cross validation*.

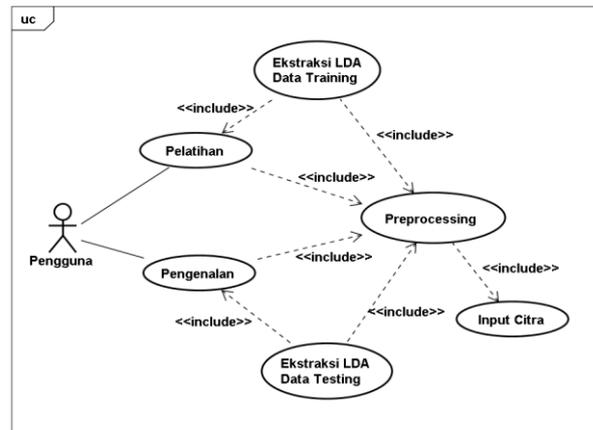
3.4. Perancangan Sistem

Perancangan sistem dengan menggunakan teknik pemrograman berorientasi objek yang bertujuan untuk mengetahui kerangka atau landasan yang akan dilakukan peneliti untuk masuk ke tahap implementasi. Perancangan aplikasi ini menggunakan teknik pemrograman berorientasi objek yaitu, *Unified Modeling Language* (UML):

3.4.1. Use Case Diagram

Diagram *use case* dalam penelitian ini, terdiri dari enam *use case* (Gambar 4). *Use case* pelatihan digunakan menyimpan hasil ekstraksi fitur LDA data

training, *use case input* citra digunakan untuk sistem membaca citra, *use case preprocessing* digunakan untuk melakukan memperbaiki kualitas citra masukan sebelum dilakukan tahap ekstraksi ciri, *use case* ekstraksi LDA data *training* digunakan untuk memperoleh nilai fitur citra, *use case* ekstraksi LDA data *testing* digunakan untuk melakukan ekstraksi ciri data *testing*, dan *use case* pengenalan digunakan untuk melakukan pencocokan citra hasil ekstraksi ciri LDA data *training* dengan hasil ekstraksi ciri LDA data *testing*.



Gambar 4. Use Case Diagram

3.4.2. Activity Diagram

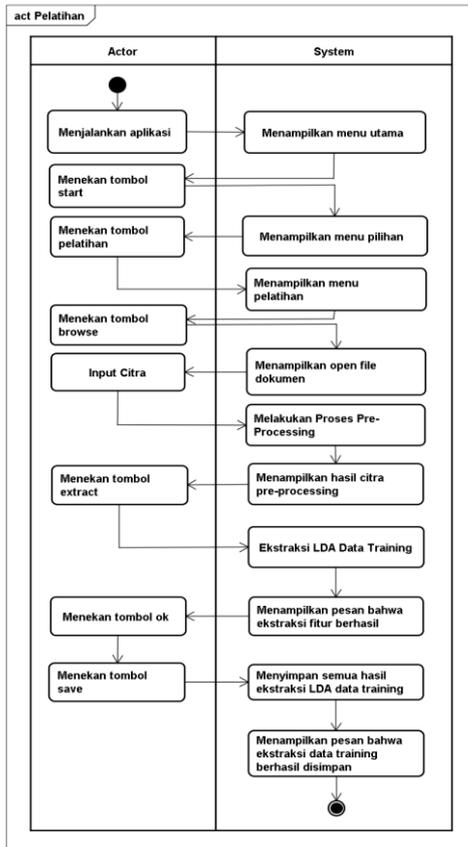
Activity diagram dalam penelitian ini, terdiri dari enam *activity* yang meliputi *activity diagram* pelatihan, *activity diagram* pengenalan, *input* citra, *preprocessing*, ekstraksi LDA data *training*, dan ekstraksi LDA data *testing*.

a. Activity Diagram Pelatihan Huruf Kapital Tulisan Tangan

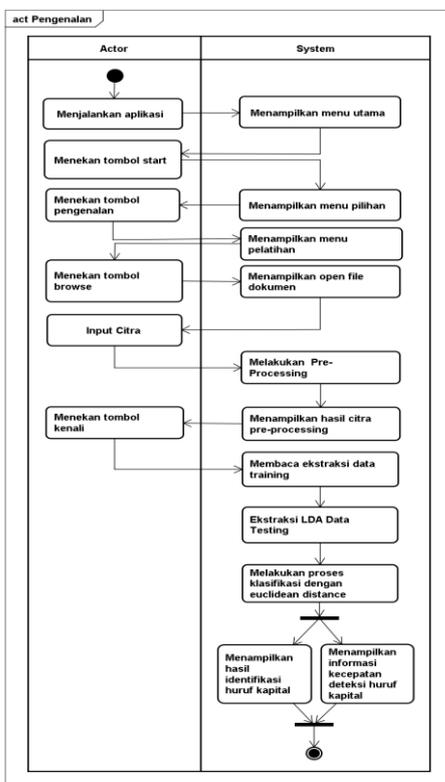
Pada Gambar 5 menggambarkan alur proses yang berjalan saat proses pelatihan citra huruf kapital tulisan tangan dilakukan. Mulai dari pengguna melakukan *input* citra dan setelah itu secara otomatis sistem melakukan *preprocessing* citra, melakukan tahap ekstraksi ciri LDA data *training*, sampai penyimpanan semua hasil ekstraksi LDA data *training* pada sistem.

b. Activity Diagram Pengenalan Huruf Kapital Tulisan Tangan

Pada Gambar 6 menggambarkan alur proses yang berjalan saat proses pengenalan citra huruf kapital tulisan tangan dilakukan. Mulai dari pengguna melakukan *input* citra dan secara otomatis sistem melakukan *preprocessing* citra, melakukan tahap ekstraksi ciri LDA data *testing*, sampai proses pencocokan hasil ekstraksi ciri LDA antara data *training* dengan data *testing* menggunakan algoritma *euclidean distance*.



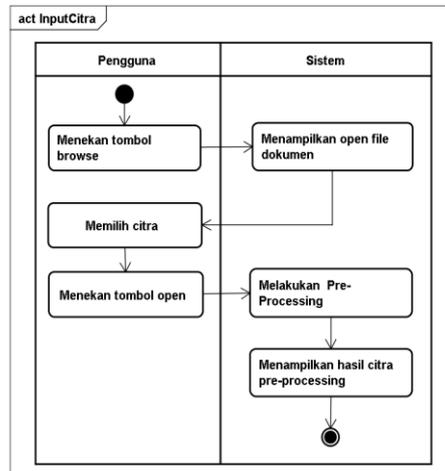
Gambar 5. Activity Diagram pelatihan



Gambar 6. Activity Diagram pengenalan

c. Activity Diagram *Input Citra*

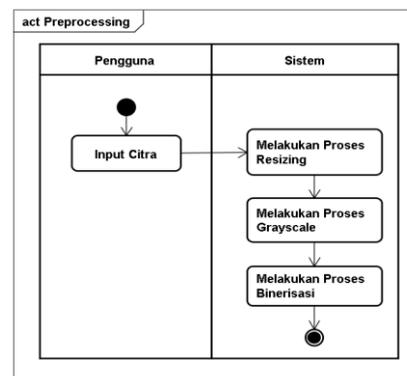
Pada gambar 7 menggambarkan proses *input* citra *training* maupun *input* citra *testing*. Mulai dari pengguna menekan tombol *browse* pada menu pelatihan ataupun menu pengenalan sampai dengan sistem menampilkan hasil *preprocessing* ke pengguna.



Gambar 7. Activity Diagram *Input* citra

d. Activity Diagram *Preprocessing*

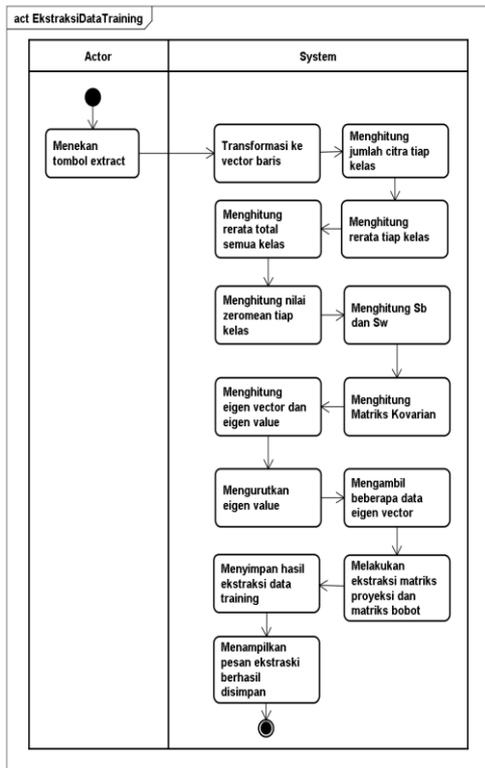
Pada Gambar 8 menggambarkan alur proses yang dilakukan oleh sistem saat proses *preprocessing* citra. Mulai dari pengguna memasukkan citra huruf kapital sampai dengan sistem menampilkan hasil *pre-processing* yang telah dilakukan kepada pengguna.



Gambar 8. Activity Diagram proses *PreProcessing*

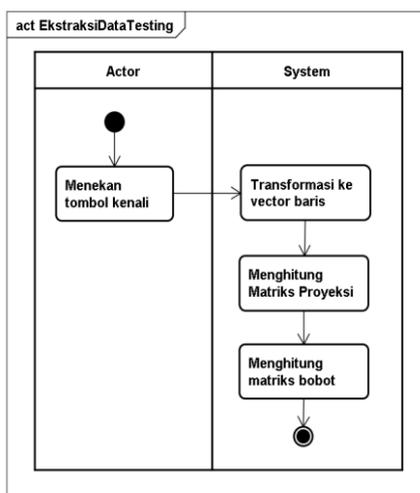
e. Activity Diagram Ekstraksi LDA Data Training

Pada gambar 9 menggambarkan alur proses yang berjalan saat proses ekstraksi LDA data *training* dilakukan. Mulai dari pengguna menekan tombol *extract* pada menu pelatihan sampai dengan sistem menampilkan pesan bahwa ekstraksi fitur berhasil dilakukan.



Gambar 9. Activity Diagram ekstraksi LDA data pelatihan

- f. Activity Diagram Ekstraksi LDA Data Testing
- Pada Gambar 10 menggambarkan alur proses yang berjalan saat proses ekstraksi LDA data testing dilakukan. Mulai dari user menekan tombol kenali pada menu pengenalan maka sistem akan melakukan proses perhitungan ekstraksi LDA data testing.



Gambar 10. Activity Diagram ekstraksi LDA data pengenalan

4. Hasil dan Pembahasan

Aplikasi pengidentifikasian huruf kapital tulisan tangan ini berhasil di implementasikan pada Bahasa

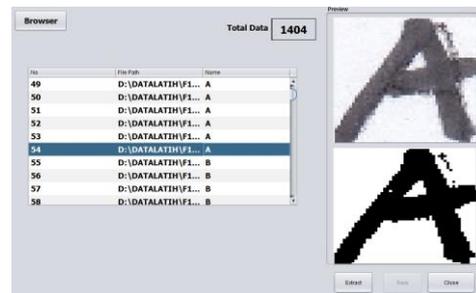
pemrograman java dengan menggunakan Neatbeans versi 8.1. Proses pengidentifikasian huruf kapital tulisan tangan memiliki dua tahap yaitu tahap pelatihan dan tahap pengenalan.

4.1. Tahap Pelatihan

Pada tahap pelatihan terdapat tiga proses diantaranya adalah *input* citra, *preprocessing* citra, dan ekstraksi ciri. Hasil akhir dari tahap pelatihan adalah sistem akan menyimpan semua hasil proses ekstraksi ciri LDA pelatihan.

4.1.1 Input Citra dan Preprocessing

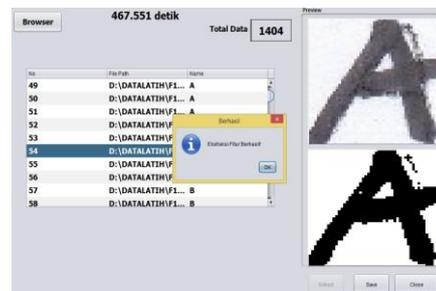
Tahap ini pengguna melakukan *input* citra dengan menekan tombol *browse* dan memilih citra digital huruf kapital tulisan tangan maka secara otomatis sistem akan menampilkan citra huruf kapital tulisan tangan asli dan hasil citra huruf kapital tulisan tangan yang telah mengalami *preprocessing* yang dapat dilihat pada Gambar 11.



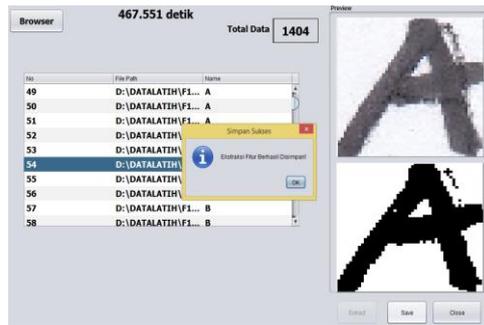
Gambar 11. Tampilan Input Citra

4.1.2. Ekstraksi LDA Data Pelatihan

Pengguna dapat langsung melakukan proses ekstraksi fitur LDA data pelatihan, dengan menekan tombol *extract* maka secara otomatis sistem akan melakukan proses ekstraksi fitur LDA dan sistem akan menampilkan pesan bahwa ekstraksi fitur berhasil dilakukan yang dapat dilihat pada Gambar 12. Selanjutnya pengguna harus menyimpan hasil ekstraksi fitur LDA dengan menekan tombol *save* dan hasil akhirnya sistem menampilkan pesan bahwa ekstraksi fitur berhasil disimpan yang dapat dilihat pada Gambar 13.



Gambar 12. Tampilan ekstraksi ciri pelatihan berhasil



Gambar 13. Tampilan simpan ekstraksi fitur pelatihan

4.2. Tahap Pengenalan

Pada tahap pengenalan citra huruf kapital tulisan tangan terdapat empat proses yang harus dilakukan yaitu *input* citra digital huruf kapital tulisan tangan, *preprocessing* citra, ekstraksi ciri LDA, dan pencocokan ekstraksi ciri LDA antara data *training* dengan data *testing* menggunakan algoritma *euclidean distance*. Hasil pencocokan fitur adalah hasil pengidentifikasian citra huruf kapital tulisan tangan apakah teridentifikasi benar atau teridentifikasi salah beserta waktu komputasi identifikasi citra digital huruf kapital tulisan tangan.

4.2.1. Input Citra dan Preprocessing

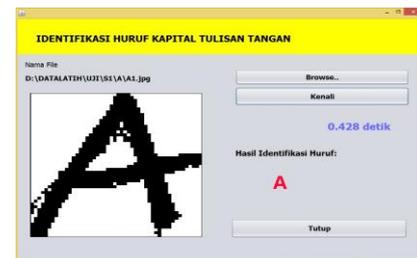
Tahap ini pengguna melakukan *input* citra dengan menekan tombol *browse* dan memilih citra digital huruf kapital tulisan tangan maka secara otomatis sistem menampilkan nama file, citra hasil *preprocessing* sebagai tanda bahwa program yang dijalankan berhasil melakukan *input* citra huruf kapital yang dapat dilihat pada Gambar 14.

4.2.2. Ekstraksi Ciri LDA dan Pengenalan Ciri Euclidean Distance

Pada tahap ini pengguna telah berhasil melakukan *input* citra huruf kapital tulisan tangan, setelah itu pengguna dapat langsung melakukan proses identifikasi huruf kapital tulisan tangan, dengan menekan tombol *kenal* maka secara otomatis sistem melakukan proses ekstraksi LDA citra *testing*, setelah itu sistem juga melakukan proses pencocokan ekstraksi ciri antara data *training* dan data *testing* menggunakan *euclidean distance* sehingga sistem menampilkan hasil identifikasi huruf kapital tulisan tangan dengan benar dan kecepatan proses pengenalan huruf kapital tulisan tangan yang dapat dilihat pada Gambar 15 atau sistem akan menampilkan hasil identifikasi huruf kapital tulisan tangan dengan salah dan kecepatan proses pengenalan huruf kapital tulisan tangan yang dapat dilihat pada Gambar 16.



Gambar 14. Tampilan *Input* citra



Gambar 15. Identifikasi huruf kapital benar



Gambar 16. Identifikasi huruf kapital salah

4.3. Pengujian Akurasi

Kinerja metode LDA dan *euclidean distance* dalam mengidentifikasi huruf kapital tulisan tangan diuji dengan menggunakan metode validasi silang (*cross validation*). Pada penelitian ini, digunakan *10-fold cross validation* dengan pengujian dilakukan sebanyak 10 kali (*10 run*) dengan membagi dataset yang awalnya berjumlah 1560 data yang terbagi menjadi 1404 data *training* dan 156 data *testing*.

Skenario uji dengan jumlah *10-fold cross validation* dan iterasi yang dilakukan sebanyak 10 kali. Dengan iterasi pertama pada *fold* pertama menggunakan 1404 data gabungan dari *subset* ke-2 sampai *subset* ke-9 sisanya yaitu 156 data yang digunakan sebagai data *testing* pada *subset* pertama, proses *training* dan proses *testing* dilakukan sampai iterasi ke-10. Detail hasil akurasi pengujian skenario pada masing-masing *fold* yang dapat dilihat pada Tabel 2.

Dari hasil percobaan yang disajikan pada tabel 2, terlihat bahwa akurasi pengidentifikasi huruf kapital tulisan tangan mengindikasikan bahwa nilai akurasi pada tiap *fold* sudah cukup stabil. Percobaan pada *fold* ke-1 dan *fold* ke-2 memperlihatkan hasil yang paling rendah diantara 8 *fold* lainnya yaitu $\approx 67.95\%$, sedangkan *fold* sisanya menghasilkan akurasi yang lebih besar dengan

akurasi paling tinggi ada pada *fold* ke-8 yang memperlihatkan hasil yang paling tinggi diantara 9 *fold* lainnya sebesar $\approx 82.05\%$. Jika dianalisis performa klasifikasi metode *linear discriminant analysis* (LDA) dan *euclidean distance* yang dihasilkan memiliki kinerja pengenalan citra digital huruf kapital tulisan tangan dengan efektifitas rata-rata akurasi sebesar $\approx 75.39\%$. Sedangkan jika dianalisis performa waktu komputasi klasifikasi dengan *linear discriminant analysis* (LDA) dan *euclidean distance* yang dihasilkan memiliki efektifitas rata-rata waktu komputasi dari setiap *fold* yaitu 0.41999 detik.

Tabel 2. Hasil Uji Aplikasi dengan 10-Fold CrossValidation

Fold	Data Training	Data Testing	Akurasi	Rerata Komputasi
Fold 1	1404	156	67.95 %	0.2738 detik
Fold 2	1404	156	67.95 %	0.2299 detik
Fold 3	1404	156	69.23 %	0.2545 detik
Fold 4	1404	156	73.08 %	0.2520 detik
Fold 5	1404	156	77.56 %	0.2494 detik
Fold 6	1404	156	78.85 %	0.2511 detik
Fold 7	1404	156	80.77 %	1.7740 detik
Fold 8	1404	156	82.05%	0.2450 detik
Fold 9	1404	156	81.41 %	0.2479 detik
Fold10	1404	156	75 %	0.4223 detik

5. Kesimpulan

Pengidentifikasi huruf kapital tulisan tangan dengan menggunakan metode *Linear Discriminant Analysis* (LDA) dan *Euclidean Distance* berhasil diterapkan untuk melakukan identifikasi citra digital huruf kapital tulisan tangan yang memiliki efektifitas rata-rata tingkat akurasi sebesar 75.39% dan waktu komputasi sebesar 0.41999 detik, dengan menggunakan resolusi citra sebesar 50x50 piksel. Penentuan data *training* dan data *testing* menggunakan metode *10-fold cross validation* dengan jumlah data sebanyak 1560 data dimana dibagi dalam 1404 data *training* dan 156 data *testing*.

Diharapkan metode ini dapat di aplikasi dalam mobile, seperti *smartphone*, *tablet computer* (Wiryasaputra *et al.*, 2013).

Daftar Pustaka

Achmad, B., 2006. Diktat Kecerdasan Buatan. Yogyakarta: Fakultas Teknik Universitas Gadjah Mada.

- Falanda, F., Rendra, G., dan Hartini, 2016. Penentuan objek wisata, objek kuliner serta akomodasi disekitar pengguna dikota palembang dengan menggunakan algoritma *Euclidean Distance*. *Ilmiah Informatika Global* 7(1):17–24.
- Han, J., Kamber, M. and Pei, J., 2012. *Data Mining: Concepts and Techniques*. 3rd ed. Massachusetts (US): Morgan Kaufmann. Retrieved ([http://eprints.uny.ac.id/41356/2/BAB II KAJIAN TEORI.pdf](http://eprints.uny.ac.id/41356/2/BAB%20II%20KAJIAN%20TEORI.pdf)).
- Hastiana, R., 2010. Segementasi Citra Digital Pembuluh Darah Mata Untuk Mendeteksi Tingkat Keperahan Diabetic Retinopathy. Malang.
- Kasih, P. dan Yuliana, M.P., 2015. Handwritten character recognition untuk evaluasi perkembangan kemampuan menulis anak PAUD. *Seminar Nasional "Inovasi dalam Desain dan Teknologi."* IDEaTech2015 pp. 273–81.
- Muntasa, A., 2015. Pengenalan Pola; Aplikasi Untuk Pengenalan Wajah, Analisis Terstruktur Objek, Pengenalan Plat Nomor Kendaraan Dan Segmentasi Pembuluh Darah. Yogyakarta: Graha Ilmu.
- Pratiwi, 2014. Metode ekstraksi ciri 2dpc pada pengenalan citra wajah dengan matlab. *Teknologi* 7:1–5.
- Sholahuddin, A., Rustam, E.S., Iping, S. dan Setiawan, H., 2010. Penerapan metode linear discriminant analysis pada pengenalan wajah, *Konferensi Nasional Matematika, Ke-15 Di UNIMA* (December 2013).
- Simangunsong, VFR. 2015. Klasifikasi Fragmen Metagenom Menggunakan Principal Component Analysis Dan K-Nearest Neighbor. Institut Pertanian Bogor.
- Sutojo, T., Edy, M. dan Suhartono, V., 2011. Kecerdasan Buatan. Yogyakarta: Andi.
- Sutoyo, T., Edy, M., Vincent, S., Oky, D.N., dan Wijanarto, 2009. *Teori Pengolahan Citra*. 1 st Publi. Yogyakarta: Andi.
- Wirayuda, Tjokorda, A.B., Sylvia, V. dan Retno, N.D., 2009. Pengenalan Huruf Komputer Menggunakan Algoritma Berbasis Chain Code Dan Algoritma Sequence Alignment, 19–24.
- Wiryasaputra, R., Rendra, G. dan Wawan, N., 2013. Pembangunan m-konseling psikologi klinis, *Seminar Nasional Teknologi Informasi* 74–78.
- Witten, H.Ian, Eibe, F. and Mark, H., 2011. *ata Mining Practical Mechine Learning Tools and Techniques Third Edition* (3rd Ed.). Elsevier Inc.