

Analisa Perbandingan Algoritma Euclidean Dan Manhattan Distance Dalam Identifikasi Wajah

Abdul Azis¹, Dinar Putra Pamungkas², Ahmad Bagus Setiawan³

^{1,2,3}Teknik Informatika, Fakultas Teknik, Universitas Nusantara PGRI Kediri

E-mail: *¹abaskrabat03@gmail.com, dinar@unpkediri.ac.id, ahmadbagus@unpkediri.ac.id

Abstrak – Wajah merupakan bagian muka kepala pada manusia dari dahi sampai dagu. Wajah berguna untuk mengenali individu dengan cara natural dan paling umum. Selain mengurangi kecurangan siswa, guru atau karyawan dan pemakaian *fingerprint* secara bergantian yang sangat beresiko menularnya covid-19, presensi dengan pengenalan wajah akan lebih aman dan tersistem. Penelitian ini menggunakan metode *Local Binary Patterns (LBP)* dalam ekstraksi fitur, sedangkan untuk klasifikasi membandingkan akurasi algoritma perhitungan jarak yaitu *Euclidean* dan *Manhattan* yang diterapkan pada proses pengujian dengan beberapa skenario jumlah data training dan data testing. Hasil akurasi tertinggi terdapat pada algoritma *euclidean distance* yang mencapai 100%, dibandingkan dengan algoritma *manhattan distance* yang hanya mencapai 85%. Semakin banyak data latih yang digunakan, semakin tinggi tingkat akurasi pengujian.

Kata Kunci — *Euclidean, Local Binary Patterns, Manhattan, Wajah*

1. PENDAHULUAN

Wajah merupakan bagian muka kepala pada manusia dari dahi sampai dagu [1]. Pengenalan wajah (*face recognition*) adalah salah satu sistem yang digunakan untuk pengakuan individu. Karena wajah merupakan faktor pembeda setiap individu dan bagian yang paling terlihat dari anatomi manusia [2]. Sehingga untuk mengenali individu dengan cara natural dan paling umum adalah menggunakan wajah. Contohnya dalam hal presensi di saat pandemi covid-19 seperti ini sangat dibutuhkan untuk *physical distancing* (jaga jarak). Selain mengurangi kecurangan siswa, guru atau karyawan dan pemakaian *fingerprint* secara bergantian yang sangat beresiko menularnya virus tersebut, presensi dengan pengenalan wajah akan lebih aman dan tersistem.

Terdapat beberapa metode yang dapat digunakan untuk pengenalan wajah oleh para peneliti sebelumnya. Maka dari itu, penelitian ini akan merujuk dari beberapa penelitian tersebut untuk membandingkan metode mana yang lebih baik dalam pengenalan wajah.

Penelitian yang dimaksud, diantaranya penelitian yang dilakukan oleh Yusuf Fadlila Rachman pada tahun 2020. Berdasarkan hasil pengujiannya didapatkan tingkat akurasi 64,81% dalam Klasifikasi Digitalretina Penderita Diabetes Retinopati Menggunakan Metode *Euclidean* [3].

Penelitian lain yang melandasi penelitian ini adalah penelitian yang dilakukan oleh Arike Septi Audianingrum dan Dinar Putra Pamungkas pada tahun 2020. Dari uji coba yang dilakukan empat kali dalam Pengenalan Jenis Bunga Dengan Metode

Learning Vector Quantization dan *Manhattan Distance*, menghasilkan akurasi rata-rata 96,75% [4].

Penelitian lain yang menjadi referensi adalah penelitian yang berjudul Pengenalan Wajah Menggunakan *Local Binary Pattern (LBP)* dan *Support Vector Machine (SVM)*. Pada penelitian tersebut, nilai akurasi untuk pengujian dengan data latih sebesar 100% dan nilai untuk pengujian dengan data uji sebesar 96,0% [5].

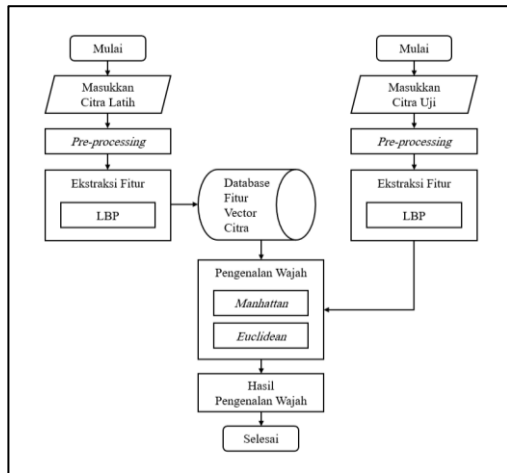
Penelitian pada tahun 2018 oleh Dinar Putra Pamungkas dengan judul Implementasi Ekstraksi Fitur dan K-Nearest Neighbor untuk Identifikasi Wajah Personal. Akurasi pengenalan wajah mencapai 80%, dengan nilai rata-rata FAR terendah 20% dan FRR 15% [6].

Selain itu ada penelitian dari Fauzi Yusa Rahman pada tahun 2017 yang berjudul Pendeteksian Citra Wajah Manusia dari Berbagai Posisi Menggunakan *Local Binary Pattern (LBP)*. Hasil penelitian tersebut sangat memuaskan karena hasil pendeteksian dari citra wajah hampir bisa dikenali [7].

Penulis mengimplementasikan sebuah sistem pengenalan wajah dengan ekstraksi fitur menggunakan metode *Local Binary Patterns*, sedangkan klasifikasinya dengan perhitungan jarak *Euclidean* dan *Manhattan* untuk membandingkan performa kedua algoritma tersebut. Identifikasi citra menggunakan bahasa pemrograman Matlab R2017a. Data citra diambil dari 10 teman kerja di Ponpes Wali Barokah Kediri.

2. METODE PENELITIAN

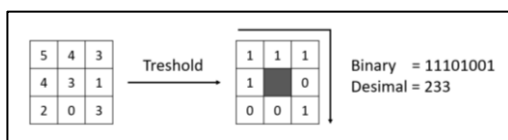
Alur sistem dalam penelitian ini, dimulai dari memasukkan citra yang dilanjutkan tahap *pre-processing* dan tahap *processing* yang meliputi ekstraksi fitur dengan metode *Local Binary Patterns* dan pengenalan melalui algoritma *Manhattan* dan *Euclidean Distance*. Selanjutnya, hasil ekstraksi fitur pada citra latih akan disimpan dalam *database*. Adapun proses citra uji, hasil ekstraksi fitur langsung diolah pada tahap pengenalan wajah yang data modelnya diperoleh dari *database*.



Gambar 1. Diagram Alur Sistem Citra Latih dan Uji

2.1 Local Binary Patterns (LBP)

Ekstraksi ciri untuk melakukan pengenalan wajah ada banyak. Salah satu metode ekstraksi ciri adalah metode *Local Binary Pattern* (LBP). Pendekatan ini diperkenalkan oleh Ojala dkk, pada tahun 1996. Tekstur dan bentuk citra digital dapat dideskripsikan menggunakan LBP. Operator LBP asli bekerja dengan delapan piksel tetangga, menggunakan nilai piksel pusatnya sebagai *threshold*.



Gambar 2. Operator LBP Asli

Jika koordinat dari piksel pusat adalah (x_c, y_c) maka koordinat dari tetangga $P(x_p, y_p)$ pada tepi lingkaran dengan jari-jari R dapat ditentukan dengan persamaan 2 dan 3.

$$x_p = x_c + R \cos\left(\frac{2\pi p}{p}\right) \dots\dots\dots (2)$$

$$y_p = y_c + R \sin\left(\frac{2\pi p}{p}\right) \dots\dots\dots (3)$$

Jika nilai keabuan dari piksel pusat adalah g_c dan nilai dari tetangganya adalah g_p dengan $p = 0, \dots, P - 1$, maka nilai LBP untuk piksel (x_c, y_c) dapat dicari menggunakan persamaan 4.

$$LBP_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} s(g_p - g_c) 2^p \dots\dots\dots (4)$$

dengan,

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \dots\dots\dots (5)$$

Keterangan :

- P = banyaknya piksel tetangga
- R = nilai jarak/ radius
- g_c = nilai dari piksel x dan y
- g_p = nilai piksel tetangga
- x_c, y_c = koordinat pusat

2.2 Klasifikasi

Menurut KBBI, klasifikasi adalah penyusunan bersistem dalam kelompok atau golongan menurut kaidah atau standar yang ditetapkan. Sedangkan pengertian secara umum klasifikasi adalah suatu kegiatan dengan mengelompokkan benda yang memiliki beberapa ciri yang sama dan memisahkan benda yang tidak sama [8].

2.3 Euclidean Distance

Euclidean distance adalah perhitungan untuk mengukur jarak dua titik yang mempelajari hubungan antara sudut dan jarak [9]. Dalam bahasa matematika *euclidean distance* seperti perhitungan *Pythagoras* yang digunakan untuk mengukur dua titik dalam satu dimensi [10].

Untuk menghitung *euclidean distance* menggunakan persamaan 6.

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \dots\dots\dots (6)$$

Keterangan :

- d = Jarak
- x_1 = Koordinat latitude 1
- x_2 = Koordinat latitude 2
- y_1 = Koordinat longitude 1
- y_2 = Koordinat longitude 2

2.4 Manhattan Distance

Manhattan distance adalah metode perhitungan pada ruang jarak dengan menerapkan konsep selisih mutlak [11].

Untuk menghitung *manhattan distance* menggunakan persamaan 7.

$$d(x, y) = \sum_{i=1}^n |x_i - y_i| \dots \dots \dots (7)$$

Keterangan :

- $d(x, y)$ = Jarak
- x = Koordinat lokasi 1
- y = Koordinat lokasi 2

2.5 Akurasi

Perhitungan akurasi digunakan untuk mengetahui tingkat keberhasilan dalam suatu pengujian. Tingkat keberhasilan dalam suatu pengujian, dapat dicari dengan membagi total data yang benar dengan jumlah seluruh data yang diuji dikalikan 100. Secara matematis, perhitungan akurasi ditunjukkan persamaan 8.

$$Akurasi = \frac{\sum Benar}{\sum Data Uji} \times 100\% \dots \dots \dots (6)$$

3. HASIL DAN PEMBAHASAN

3.1 User Interface



Gambar 3. Tampilan Awal

Di awal aplikasi, akan menampilkan tampilan seperti gambar 3 yang terdapat 3 menu utama, yaitu : menu ‘Training’ untuk melatih citra yang akan disimpan di database, menu ‘Testing’ untuk menguji citra dan menu ‘About’ untuk menampilkan detail aplikasi.

Kode tombol training (pilih folder) :

```
TrainDatabasePath =
uigetdir(' ', 'Pilih Folder Data
Training... ');
```

Kode tombol training (membuat database) :

```
T =
CreateDatabase (TrainDatabasePat
h);
filename = 'Database.mat';
save (filename, 'T', 'TrainDatabas
ePath');
```

Kode melatih data dalam folder pilihan :

```
TrainFiles =
dir (TrainDatabasePath);
Train_Number = 0;
for i = 1:size (TrainFiles,1)
    if
not (strcmp (TrainFiles (i).name, '
.') | strcmp (TrainFiles (i).name, '
..') | ...
strcmp (TrainFiles (i).name, 'Thum
bs.db'))
    Train_Number =
Train_Number + 1;
end
end
```

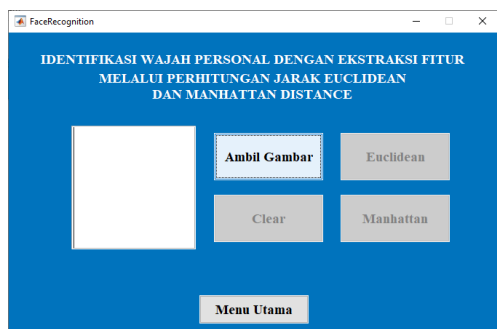
Kode ekstraksi fitur dengan LBP :

```
for i = 1:row_max
    for j = 1:col_max
        A = Input_Im (i:i+L-1,
j:j+L-1);
        A = A+1-A (C,C);
        A (A>0) = 1;
        LBP_Im (i, j) = A (C,L) +
A (L,L) *2 + A (L,C) *4 + A (L,1) *8
+ A (C,1) *...
16 + A (1,1) *32 +
A (1,C) *64 + A (1,L) *128;
    end;
end;
C1=max (max (LBP_Im));
LBP =LBP_Im./C1;
```

Kode tombol Testing :

```
set (handles.uipanel1, 'visible',
'off')
set (handles.uipanel2, 'visible',
'on')

set (handles.bAmbilGambar, 'enabl
e', 'on')
set (handles.bClear, 'enable', 'of
f')
set (handles.bEuclidean, 'enable'
, 'off')
set (handles.bManhattan, 'enable'
, 'off')
```



Gambar 4. Tampilan Ambil Gambar Data Testing

Pada gambar 4. Tampilan 'Ambil Gambar' untuk melakukan testing data akan muncul setelah pemakai memilih menu Testing. Setelah itu, pilih menu 'Ambil Gambar' dan cari gambar yang akan diidentifikasi, sehingga akan muncul tampilan seperti gambar 5.

Kode tombol Ambil Gambar :

```
[InputImage , fileCanceled]=
imgetfile;
Kode menampilkan gambar yang dipilih :
handles.inputimage =
InputImage;
guidata(hObject,handles);
imshow(InputImage, 'Parent',
handles.axes1);
```



Gambar 5. Tampilan Pemroses Data Testing

Untuk mengetahui hasil identifikasi, bisa memilih menu 'Euclidean' atau 'Mahattan'. Sehingga muncul tampilan seperti gambar 6 yang menampilkan hasil identifikasi dengan algoritma euclidean distance. Dan menampilkan hasil identifikasi dengan algoritma manhattan distance seperti gambar 7.

Kode algoritma euclidean distance :

```
Euc_dist = [];
for i = 1 : Train_Number
    q = ProjectedImages(:,i);
    temp = sum((ProjectedTestImage-
q).^2).^0.5;
    Euc_dist = [Euc_dist temp];
```

end

Kode algoritma manhattan distance :

```
Man_dist = [];
for i = 1 : Train_Number
    q = ProjectedImages(:,i);
    temp = sum(abs(
ProjectedTestImage - q));
    Man_dist = [Man_dist temp];
end
```



Gambar 6. Tampilan Hasil Identifikasi Algoritma Euclidean Distance



Gambar 7. Tampilan Hasil Identifikasi Algoritma Manhattan Distace



Gambar 8. Tampilan About

Gambar 8 merupakan tampilan about yang berisikan info tambahan tentang pembuat aplikasi. Untuk menampilkannya, dari tombol About yang ada pada gambar 4 Tampilan Awal dengan kode :

```
set(handles.uipanel1, 'visible',
'off')
```

```
set(handles.uipanel16, 'visible',  
'on')  
axes(handles.axes3);  
image(imread('about.jpg'));
```

3.2 Skenario Uji Coba

Untuk mempermudah analisa hasil penelitian, terdapat beberapa skenario untuk menguji 20 data testing dengan perbedaan jumlah data training, seperti yang terlampir di tabel 1.

Tabel 1. Skenario Uji Coba

Skenario	Jumlah Data Training
1	10
2	20
3	30
4	40
5	50
6	60
7	70
8	80
9	90

Di tabel 1 menunjukkan beberapa skenario dalam uji coba. Untuk skenario 1 terdapat data training sejumlah 10, yang mana setiap karyawan akan diambil 1 citra untuk data trainingnya.

Skenario 2 terdapat 20 data training, masing-masing karyawan ada 2 citra yang dilatih.

Skenario 3 terdapat 30 data training, masing-masing karyawan ada 3 citra yang dilatih.

Skenario 4 terdapat 40 data training, masing-masing karyawan ada 4 citra yang dilatih.

Skenario 5 terdapat 50 data training, masing-masing karyawan ada 5 citra yang dilatih.

Skenario 6 terdapat 60 data training, masing-masing karyawan ada 6 citra yang dilatih.

Skenario 7 terdapat 70 data training, masing-masing karyawan ada 7 citra yang dilatih.

Skenario 8 terdapat 80 data training, masing-masing karyawan ada 8 citra yang dilatih.

Skenario 9 terdapat 90 data training, masing-masing karyawan ada 9 citra yang dilatih.

3.3 Hasil Uji Coba

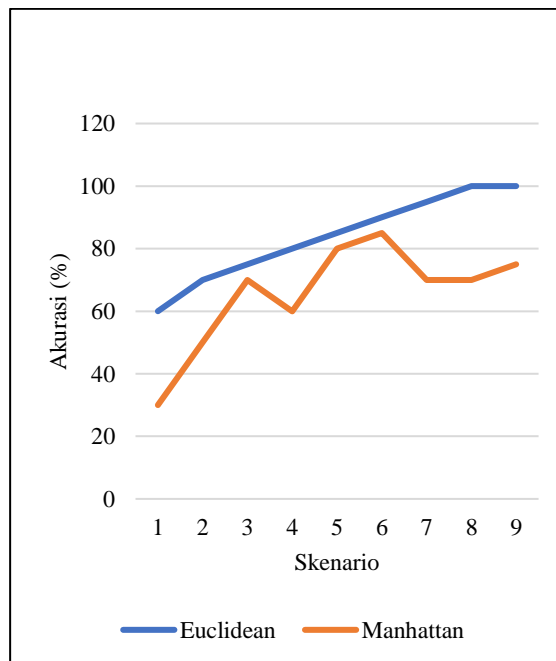
Dari sembilan skenario uji coba, setiap proses uji coba terdapat 20 data uji, yang diperoleh dari 10

petugas Kantor Pendidikan Ponpes Wali Barokah Kota Kediri dengan masing-masing 2 foto. Hasil uji coba dari Sembilan skenario, dapat dilihat pada tabel 2.

Tabel 2. Hasil uji coba terhadap data training

Skenario	Akurasi Euclidean (%)	Akurasi Manhattan (%)
1	60%	30%
2	70%	50%
3	75%	70%
4	80%	60%
5	85%	80%
6	90%	85%
7	95%	70%
8	100%	70%
9	100%	75%

Gambar 9 menunjukkan grafik perbedaan akurasi algoritma manhattan distance dan euclidean distance dari skenario 1 sampai skenario 9.



Gambar 9. Grafik Perbedaan Akurasi Setiap Skenario

4. SIMPULAN

Berdasarkan Analisa uji coba dengan beberapa skenario yang telah dilakukan, dapat diambil kesimpulan sebagai berikut :

1. Algoritma penghitungan jarak dengan presentase tertinggi adalah Euclidean

- Distance yang mana mencapai 100%, dibandingkan dengan algoritma Manhattan Distance yang hanya mencapai 85%.
2. Semakin banyak data latih yang digunakan, akan semakin akurat hasil uji cobanya untuk algoritma Euclidean Distance.
 3. Akurasi maksimal diperoleh saat data latih berjumlah minimal 8 citra setiap karyawan.

5. SARAN

Adapun saran untuk selanjutnya mengenai pengenalan wajah berbasis penghitungan jarak yaitu perlu menambahkan data latih agar semakin akurat dalam pengujian.

DAFTAR PUSTAKA

- [1] Kamus Besar Bahasa Indonesia (KBBI), 2009. Penerbit : Balai Pustaka.
- [2] Ghani, A.D., Salman, N., Intan, I. 2019. Sistem Presensi Kehadiran Menggunakan Metode Fisherface. SENSITif. Hal:915-923.
- [3] Rachman, Y.F., Kusriani, dan Al-Fatta, H. 2020. Klasifikasi Citra Digital Retina Penderita Diabetes Retinopati Menggunakan Metode Euclidean. Journal of Computer and Information Technology. Vol.3, No.2:75-82.
- [4] Audianingrum, A.S., Pamungkas, D.P. 2020. Pengenalan Jenis Bunga dengan Metode Learning Vector Quantization dan Manhattan Distance. Seminar Nasional Inovasi Teknologi. e-ISSN:2549-7952.
- [5] Mujib, K., Hidayatno, A., dan Prakoso, T. 2018. Pengenalan Wajah Menggunakan Local Binary Pattern (LBP) dan Support Vector Machine (SVM). TRANSIENT. Vol.7, No.1:123-130.
- [6] Pamungkas, D.P., Setiawan, A.B. 2018. Implementasi Ekstraksi Fitur dan K-Nearest Neighbor untuk Identifikasi Wajah Personal. JOUTICA. Vol.3, No.2:187-193.
- [7] Rahman, F.Y. 2017. Pendeteksian Citra Wajah Manusia Dari Berbagai Posisi Menggunakan Local Binary Pattern (LBP). Technologia. Vol.8, No.4:257-266.
- [8] Fadlil, A.(2012). Sistem Pengenalan Citra jenis-jenis Tekstil. Spektrum industri. Vol.10, No.1: 24.
- [9] Derisma, Firdaus, & Yusya, R. P. 2016. Perancangan Ikat Pinggang Elektronik Untuk Tunanetra Menggunakan Mikrokontroler Dan Global Positioning System (Gps) Pada Smartphone Android. Jurnal Teknik Elektro ITP, 5(2), 130-13.
- [10] Mustofa, Z., & Suasana, I. S. 2018. Algoritma Clustering K-Medoids Pada E-Government Bidang Information And Communication Technology Dalam Penentuan Status Edgi. Jurnal Teknologi Informasi dan Komunikasi, 9(1).
- [11] Buaton, R., Sundari, Y., & Maulita, Y. (2016). Klustering Tindak Kekerasan Pada Anak Menggunakan Algoritma K-Means Dengan Perbandingan Jarak Kedekatan Manhattan City Dan Euclidean. Media Informasi Analisa dan Sistem, 1(2), 47-53.