

SISTEM IDENTIFIKASI POLA BUJUR SANGKAR DENGAN METODE KODE BERANTAI

Ernastuti

*Fakultas Ilmu Komputer, Universitas Gunadarma
ernas@staff.gunadarma.ac.id*

Abstrak

Citra biner adalah citra yang hanya mempunyai 2 nilai derajat keabuan yaitu hitam dan putih. Pada tulisan ini aplikasi citra biner dibutuhkan untuk keperluan merepresentasikan citra objek bujur sangkar yang akan diidentifikasi. Sistem identifikasi bujur sangkar mempunyai 2 buah fase yaitu fase pelatihan dan fase pengenalan. Pada fase pelatihan, beberapa contoh citra objek dipelajari dan diamati untuk menentukan fitur/ciri yang akan digunakan pada proses pengenalan serta prosedur klasifikasinya. Pada fase pengenalan, objek diambil fitur/cirinya dengan metode kode berantai 8-arah, kemudian ditentukan kelas kelompoknya. Objek input yang hendak dicari identitasnya, ditransformasikan ke dalam bentuk fitur berupa barisan kode berantai, yang selanjutnya dengan mencocokkan urutan kode berantainya dengan urutan kode yang ada dalam himpunan pelatihan, maka identitasnya akan dikenali. Pengamatan yang dilakukan pada 32 citra objek dalam fase pelatihan mendapatkan suatu nilai sebagai fitur bujur sangkar dengan tingkat kepercayaan 95%. Program yang digunakan untuk implementasi sistem ini adalah MATLAB ver 7.0. Hasil pengamatan menunjukkan pola bujur sangkar mempunyai ciri, bila objek diputar untuk setiap rotasi : $0^\circ, 1^\circ, 2^\circ, \dots, 360^\circ$ maka (1) untuk $i = 1, 2, \dots, 8$ $Hist[i]$ mempunyai tinggi bervariasi dengan batas toleransi ≤ 5 , dan (2) untuk $i = 1, 3, 5, 7$, $Hist[i]$ mempunyai tinggi yang sama atau hampir sama dengan batas toleransi ≤ 1 , dan untuk $i = 2, 4, 6, 8$, $Hist[i]$ juga mempunyai tinggi yang sama atau hampir sama dengan batas toleransi ≤ 1 .

Kata kunci : Citra Biner, bujur sangkar, kode berantai, fase pelatihan, fase pengenalan, deteksi tepi

PENDAHULUAN

Dalam bidang komputer ada tiga bidang studi yang berkaitan dengan data citra, yaitu Grafika komputer, Pengolahan citra dan Pengenalan pola. Grafika komputer bertujuan menghasilkan citra dengan primitif geometri seperti garis, lingkaran dan sebagainya.

Pengolahan citra bertujuan memperbaiki kualitas citra agar mudah diinterpretasi oleh manusia atau mesin (komputer). Teknik pengolahan citra mentransformasikan citra menjadi citra lain. Masukannya adalah citra dan keluarannya juga citra, namun citra keluarannya mempunyai kualitas lebih baik daripada citra masukan.

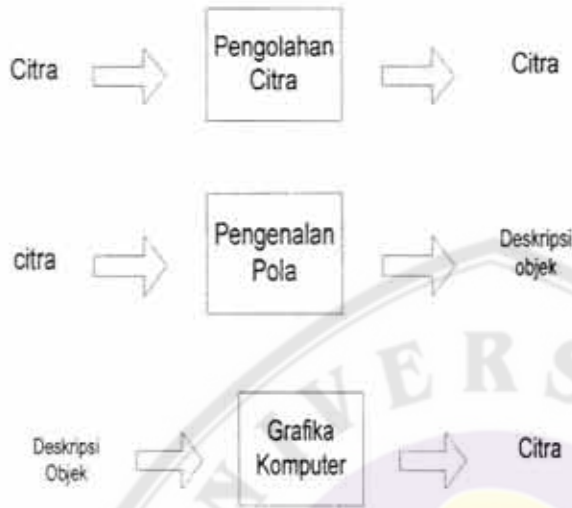
Sedangkan pengenalan pola mengelompokkan data numerik dan simbolik (termasuk citra) secara otomatis oleh mesin (dalam hal ini komputer).

Tujuan pengelompokan adalah mengenali suatu objek di dalam citra. Komputer menerima masukan berupa citra objek yang akan diidentifikasi, memproses citra tersebut, dan memberikan keluaran berupa deskripsi objek di dalam citra. Dengan menggunakan suatu algoritma pengenalan pola, diharapkan komputer dapat mengenali citra objek tersebut.

Pada tulisan ini, bidang studi yang berkaitan dengan data citra yang dibahas adalah bidang studi Pengenalan Pola.

Tulisan ini memperlihatkan bagaimana merancang dan mengimplementasi sistem

identifikasi bujur sangkar dengan menggunakan metode kode berantai, dimana citra biner digunakan untuk keperluan merepresentasikan citra objek yang akan diidentifikasi.



Gambar 1. Representasi citra objek

LANDASAN TEORI

Citra Biner. Citra biner hanya mempunyai dua nilai derajat keabuan: hitam dan putih. Piksel-piksel objek bernilai 1 dan piksel-piksel latar belakang bernilai 0. Pada waktu menampilkan gambar, 0 adalah putih dan 1 adalah hitam. Berarti, pada citra biner, latar belakang berwarna putih sedangkan objek berwarna hitam.



Gambar citra biner

Pengkodean citra biner

Gambar 2. Citra biner dan pengkodean citra biner.

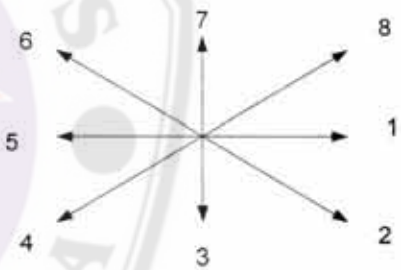
Pendeteksian Tepi. Perubahan nilai intensitas derajat keabuan yang mendadak (besar) dalam jarak yang singkat. Perbedaan intensitas inilah yang menampakkan rincian pada gambar. Tepi

biasanya terdapat pada batas antara dua daerah berbeda pada suatu citra. Tepi dapat diorientasikan dengan suatu arah, dan arah ini berbeda-beda bergabung pada perubahan intensitas.

Pendeteksian tepi merupakan langkah pertama untuk melingkupi informasi di dalam citra. Tepi mencirikan batas-batas objek dan karena itu tepi berguna untuk proses segmentasi dan identifikasi objek di dalam citra.

Tujuan operasi pendeteksian tepi adalah untuk meningkatkan penampakan garis batas suatu daerah atau objek di dalam citra.

Salah satu teknik yang digunakan untuk mendeteksi tepi adalah operator gradien pertama. Metode operator gradien adalah operator gradien selisih terpusat, operator sobel, operator canny, operator Prewit dan operator Roberts.



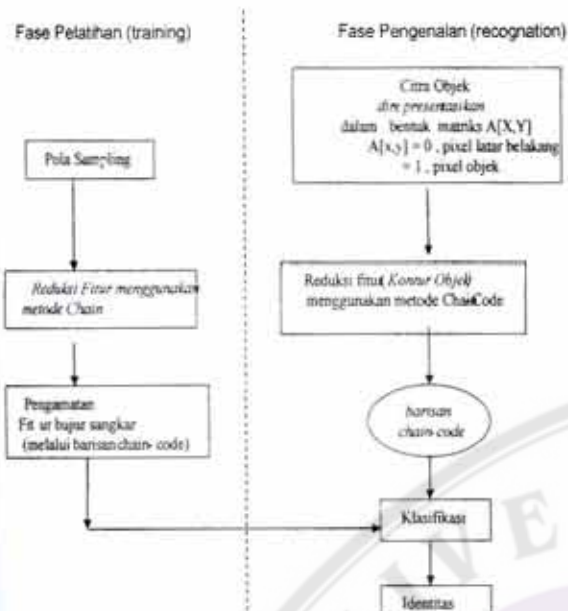
Chain code - 8 arah

Gambar 3. Kode berantai 8 arah

Representasi Kontur. Representasi kontur dapat berupa senarai tepi atau berupa kurva. Senarai tepi merupakan himpunan terurut piksel tepi. Metode Kode berantai adalah metode mengkodekan senarai tepi yang membentuk batas daerah. Kode berantai menspesifikasikan arah setiap piksel tepi di dalam senarai tepi. Arah yang digunakan Kode berantai adalah 8 arah mata angin seperti yang ditunjukkan Gambar 3.

METODE PENELITIAN

Sistem pengidentifikasi Bujur sangkar diperlihatkan pada Gambar 4.



Gambar 4. Sistem pengidentifikasian bujur sangkar

ALGORITMA DAN IMPLEMENTASI

Algoritma fase pelatihan terdiri dari beberapa tahap yang diawali dengan pola sampling, semua citra objek bujur sangkar dengan berbagai posisi pada bidang 150x150 diinput ke dalam himpunan pelatihan. Citra objek bujur sangkar dibuat dengan perangkat lunak *Paint*, disimpan dalam file BMP (monochrome bitmap). Semua Citra objek bujur sangkar tersebut direpresentasikan dalam matriks bilangan biner.

Dengan menggunakan metode kode berantai, diperoleh barisan kode berantai, dengan langkah-langkah sebagai berikut: pertama, tepi dari citra objek dideteksi. 'Metode Canny' digunakan untuk mendeteksi tepi dari citra objek. Kedua, digunakan metode kode berantai 8 arah yaitu [1,2,3,4,5,6,7,8]. Asumsikan barisan kode berantai dimulai dari titik (X,Y) ; X=1, Y=1.

Tabel 1. Kode berantai

Dari piksel (X,Y) ARAH	Posisi piksel pindah ke
1	(X,Y+1)
2	(X+1,Y+1)
3	(X+1,Y)
4	(X+1,Y-1)
5	(X-1,Y)
6	(X-1,Y-1)
7	(X-1, Y)
8	(X-1,Y+1)

Dari barisan kode berantai **Chain[]** yang terbentuk, dibuat histogram kode berantai **Hist[]**, dimana Hist(i) adalah jumlah kode berantai i di barisan Chain[] ; i : 1,2,3,4,5,6,7,8 .

Implementasi Algoritma Fase Pelatihan.

```

clc;
%Disediakan beberapa bentuk objek
sebagai berikut :
%c = imread('circle 1.bmp');
%c = imread('circle 2.bmp');
%c = imread('square 1.bmp');
%c = imread('triangle 1.bmp');
c = imread('square3.bmp');
%c = imread('segi empat 2.bmp');
%c = imread('segi_empat3.bmp');
%c = imread('trapesium1.bmp');
%c = imread('belah ketupat 1.bmp');
%pot1=c(1:150,1:150);
%m=imrotate(pot1,90,'crop');
%m=pot1
%m=imrotate(c,130,'crop');
data =double(c); %pilih citra objek
figure(1);imshow(~(data));title('Bujur
Sangkar 0');
aa = edge(data,'canny'); %deteksi tepi
obyek dengan metode canny
bb = ~(aa)
figure(2);imshow(bb);title('Deteksi Tepi
Bujur Sangkar 0');
a1 = aa;
    
```

```
[x,y] = find(a1==1); %cari tepi obyek  
dari kontur deteksi tepi
```

```
chain=[];  
hist=zeros(1,8);  
awal = [x(1) y(1)] %cari titik pertama  
dalam objek
```

```
% MULAI TENTUKAN BARISAN
```

```
CHAIN_CODE
```

```
% FUNGSI CHAIN_CODE
```

```
%Chain-code awal dari piksel 1 ke 2
```

```
dx=x(2)-x(1);dy=y(2)-y(1);  
if and(dx==0,dy==1)  
    chain(1)=1;  
elseif and(dx==1,dy==1)  
    chain(1)=2;  
elseif and(dx==1,dy==0)  
    chain(1)=3;  
elseif and(dx==1,dy==-1)  
    chain(1)=4;  
elseif and(dx==0,dy==-1)  
    chain(1)=5;  
elseif and(dx==-1,dy==-1)  
    chain(1)=6;  
elseif and(dx==-1,dy==0)  
    chain(1)=7;  
elseif and(dx==-1,dy==1)  
    chain(1)=8;  
end;  
no=2;  
xx = x(2); yy=y(2);  
while not(and(xx==x(1),yy==y(1))),  
    % menentukan kode berantai
```

```
kedua, dst..
```

```
a1(xx,yy)=0;  
if (a1(xx,yy+1))==1  
    hist(1,1)=hist(1,1)+1;  
    ch=1;  
    chain(no)=ch;no=no+1;  
    yy = yy+1;  
elseif (a1(xx+1,yy+1))==1  
    hist(1,2)=hist(1,2)+1;  
    ch=2;  
    chain(no)=2;no=no+1;  
    xx=xx+1;yy=yy+1;  
elseif (a1(xx+1,yy))==1  
    hist(1,3)=hist(1,3)+1;  
    ch=3;  
    chain(no)=3;no=no+1;  
    xx=xx+1;
```

```
elseif (a1(xx+1,yy-1))==1  
    hist(1,4)=hist(1,4)+1;  
    ch=4;  
    chain(no)=4;no=no+1;  
    xx=xx+1;yy=yy-1;  
elseif (a1(xx,yy-1))==1  
    hist(1,5)=hist(1,5)+1;  
    ch=5;  
    chain(no)=5;no=no+1;  
    yy=yy-1;  
elseif (a1(xx-1,yy-1))==1  
    hist(1,6)=hist(1,6)+1;  
    ch=6;  
    chain(no)=6;no=no+1;  
    xx=xx-1;yy=yy-1;  
elseif (a1(xx-1,yy))==1  
    hist(1,7)=hist(1,7)+1;  
    ch=7;  
    chain(no)=7;no=no+1;  
    xx=xx-1;  
elseif (a1(xx-1,yy+1))==1  
    hist(1,8)=hist(1,8)+1;  
    ch=8;  
    chain(no)=8;no=no+1;  
    xx=xx-1;yy=yy+1;  
end; %if  
end; %while  
a1(x(1),y(1))=0;  
%BATAS AKHIR FUNGSI CHAIN_CODE  
hist  
figure(4);  
bar(hist);title('Histogram Bujur Sangkar 0'  
');
```

Hasil Pengamatan. Pengamatan untuk 32 sampel citra pada program implementasi fase pelatihan adalah pola bujur sangkar mempunyai ciri untuk setiap i ganjil adalah $Hist[i]$ mempunyai tinggi yang saling sama atau hampir sama (batas toleransi ≤ 1); dan untuk setiap i genap adalah $Hist[i]$ mempunyai tinggi yang saling sama atau hampir sama (batas toleransi ≤ 1). Pola lingkaran dan pola segi-banyak-genap-sama-sisi mempunyai ciri untuk semua i adalah $Hist[i]$ mempunyai tinggi yang saling sama atau hampir sama (batas toleransi ≤ 5). Pola empat persegi panjang, jajaran-genjang dan belah-ketupat

mempunyai ciri untuk $i = 1,5$ adalah $Hist[i]$ mempunyai tinggi yang sama. Untuk $i = 2,6$ adalah $Hist[i]$ mempunyai tinggi yang sama. Untuk $i = 3,7$ adalah $Hist[i]$ mempunyai tinggi yang sama. Untuk $i = 4,8$ adalah $Hist[i]$ mempunyai tinggi yang sama. Pola segi ganjil dan genap sisi tak beraturan mempunyai ciri selain di atas.

Algoritma Fase Pelatihan. Objek direpresentasikan dalam bentuk matriks bilangan biner, dengan struktur data disimpan dalam array 2 dimensi $A[X,Y]$, dimana $A[x,y] = 0$, untuk pixel latar belakang = 1, untuk pixel Objek

Dengan menggunakan metode Kode berantai, diperoleh barisan kode berantai. Langkah-langkah berikut: pertama, tepi dari citra objek dideteksi. Metode Canny digunakan untuk mendeteksi tepi dari Citra Objek. Kedua, gunakan Metode Kode berantai 8 arah: $[1,2,3,4,5,6,7,8]$. Asumsikan barisan kode berantai dimulai dari titik (X,Y) ; $X=1, Y=1$.

Dari barisan kode berantai **Chain[]** yang terbentuk, dibuat histogram kode berantai **Hist[]**, dimana $Hist(i)$: jumlah kode berantai i di barisan **Chain[]**; i : $1,2,3,4,5,6,7,8$.

Implementasi Algoritma Fase Pelatihan

```
clc;
A=zeros(150);
A(50:100,50:100)=0;
m=A;
data =double(m); %citra objek
aa = edge(data,'canny'); %deteksi tepi
obyek dengan metode canny
bb = ~(aa)
a1 = aa;
[x,y] = find(a1==1); %cari tepi obyek
dari kontur deteksi tepi
chain=[];
hist=zeros(1,8);
awal = [x(1) y(1)] %cari titik pertama
dalam objek
% MULAI TENTUKAN BARISAN
CHAIN_CODE
```

Gunakan Fungsi Chain_Code dari program fase Pelatihan

hist

Algoritma Tahap Klasifikasi dan Identitas

Langkah 1 : Input $Hist[i]$,

$i=1,2,3,4,5,6,7,8$

Langkah 2 : Hitung Standar Deviasi dari :

1) $Hist[i]$, $i=1,3,5,7$

→ Stndr_Deviasi1

2) $Hist[i]$, $i=2,4,6,8$

→ Stndr_Deviasi2

3) $Hist[i]$,

$i=1,2,3,4,5,6,7,8$ → std_gnp_gnjl

Langkah 3 :

Dari hasil pengamatan pada fase pelatihan diperoleh ambang batas berikut :

(1) Batas toleransi yang menyatakan bahwa Objek BUKAN Bujur Sangkar adalah :

pada kondisi dimana $Stndr_Deviasi1 > 1$ atau $Stndr_Deviasi2 > 1$

atau ($Stndr_Deviasi1 > 1$ dan $Stndr_Deviasi2 > 1$)

(2) Bila Objek tidak memenuhi (1) diatas, maka ada kemungkinan

Objek adalah 'Bujur Sangkar' atau 'Segi-Genap-Sama-Sisi' lainnya.

Jika $std_gnp_gnjl > 5$ maka Objek adalah BUJUR SANGKAR

Tapi bila $std_gnp_gnjl \leq 5$ maka Rotasikan Objek sebesar

25° ,

kemudian lakukan fungsi Chain-Code

Jika ($std_gnp_gnjl > 7$) maka Objek adalah

BUJUR SANGKAR

Selain itu Objek BUKAN BUJUR SANGKAR

Jika ($Stndr_Deviasi1 > 1$ atau ($Stndr_Deviasi2 > 1$)

maka Objek BUKAN BUJUR SANGKAR

Selain itu Objek adalah BUJUR SANGKAR

Implementasi Tahap Klasifikasi dan Identitas

```

k=0;
Bentuk=['Bisa Bujur Sangkar atau Segi
Genap lainnya, lanjutkan proses
deteksi '; 'Bukan Bujur Sangkar
';
    'Nah ... Ini Baru benar Bujur Sangkar
!!          '];
Ganjil=[hist(1,1) hist(1,3) hist(1,5)
hist(1,7)];
Ganjil
Genap=[hist(1,2) hist(1,4) hist(1,6)
hist(1,8)];
Genap
Stndr_Deviasi1=std(Ganjil);
Stndr_Deviasi1
Stndr_Deviasi2=std(Genap);
Stndr_Deviasi2
if
or((or((Stndr_Deviasi1)>1,(Stndr_Deviasi2)
>1)),
and((Stndr_Deviasi1)>
1,(Stndr_Deviasi2)> 1)) k=2;
else k=1;
end;
%IDENTITAS DITENTUKAN
figure(6);bar(hist)
Bentuk(k,:);
std_gnp_gnjil=std(hist);
figure(6);bar(hist)
if and(k==1,std_gnp_gnjil>5) k=3;
Bentuk(k,:),figure(6);bar(hist);
%IDENTITAS DITENTUKAN
end;
if and(k==1,std_gnp_gnjil<5)
rot=imrotate(m,25,'crop');
%GUNAKAN FUNGSI CHAIN-CODE
data =double(rot);
figure(1);imshow(data);
aa = edge(data,'canny');
figure(2);imshow(aa);

a1 = aa; figure(3);imshow(a1);
[x,y] = find(a1==1);
z=[x,y];
chain=[];
hist=zeros(1,8);

```

```

awal = [x(1) y(1)]
dx=x(2)-x(1);dy=y(2)-y(1);
if and(dx==0,dy==1)
chain(1)=1;
elseif and(dx==1,dy==1)
chain(1)=2;
elseif and(dx==1,dy==0)
chain(1)=3;
elseif and(dx==1,dy==-1)
chain(1)=4;
elseif and(dx==0,dy==-1)
chain(1)=5;
elseif and(dx==-1,dy==-1)
chain(1)=6;
elseif and(dx==-1,dy==0)
chain(1)=7;
elseif and(dx==-1,dy==1)
chain(1)=8;
end;
no=2;
xx = x(2);yy=y(2);
while not(and(xx==x(1),yy==y(1))),
figure(4);imshow(a1);
a1(xx,yy)=0;
if (a1(xx,yy+1))==1
hist(1,1)=hist(1,1)+1;
ch=1;
chain(no)=ch;no=no+1;
yy = yy+1;
elseif (a1(xx+1,yy+1))==1
hist(1,2)=hist(1,2)+1;
ch=2;
chain(no)=2;no=no+1;
xx=xx+1;yy=yy+1;
elseif (a1(xx+1,yy))==1
hist(1,3)=hist(1,3)+1;
ch=3;
chain(no)=3;no=no+1;
xx=xx+1;
elseif (a1(xx+1,yy-1))==1
hist(1,4)=hist(1,4)+1;
ch=4;
chain(no)=4;no=no+1;
xx=xx+1;yy=yy-1;
elseif (a1(xx,yy-1))==1
hist(1,5)=hist(1,5)+1;

```



```

ch=5;
chain(no)=5;no=no+1;
yy=yy-1;
elseif (a1(xx-1,yy-1))==1
hist(1,6)=hist(1,6)+1;
ch=6;
chain(no)=6;no=no+1;
xx=xx-1;yy=yy-1;
elseif (a1(xx-1,yy))==1
hist(1,7)=hist(1,7)+1;
ch=7;
chain(no)=7;no=no+1;
xx=xx-1;
elseif (a1(xx-1,yy+1))==1
hist(1,8)=hist(1,8)+1;
ch=8;
chain(no)=8;no=no+1;
xx=xx-1;yy=yy+1;
end; %if

end; %while
a1(x(1),y(1))=0;
%AKHIR FUNGSI CHAIN-CODE
figure(4);imshow(a1);
hist

k=0;
Bentuk=['Ini Bujur Sangkar
';
        'Bukan Bujur Sangkar
'];
Ganjil=[hist(1,1) hist(1,3) hist(1,5)
hist(1,7)];
Genap=[hist(1,2) hist(1,4) hist(1,6)
hist(1,8)];
Stndr_Deviasi1=std(Ganjil);
Stndr_Deviasi1
Stndr_Deviasi2=std(Genap);
Stndr_Deviasi2

std_all=std(hist);
std_all
if (std_all > 7) k=1;
else k=2;

```

```

end;
if or((Stndr_Deviasi1)>
1,(Stndr_Deviasi2)> 1) k=2;
else k =1;
end;
%IDENTITAS DITENTUKAN
Bentuk(k,:);
figure(6);bar(hist)

```

PENUTUP

Pengamatan yang dilakukan pada 32 citra objek dalam fase pelatihan mendapatkan suatu nilai sebagai fitur Bujur Sangkar dengan tingkat kepercayaan 95 % . Hasil pengamatan dari implemenatsi dengan Program MATLAB ver 7.0. menunjukkan bahwa pola bujur sangkar mempunyai ciri : bila objek diputar , untuk setiap rotasi : 0° , 1° , 2° , ..., 360° maka (1) untuk $i = 1,2,\dots,8$ Hist[i] mempunyai tinggi bervariasi dengan batas toleransi ≤ 5 , dan (2) untuk $i = 1,3,5,7$, Hist[i] mempunyai tinggi yang sama atau hampir sama dengan batas toleransi ≤ 1 , dan untuk $i=2,4,6,8$, Hist[i] juga mempunyai tinggi yang sama atau hampir sama dengan batas toleransi ≤ 1 .

DAFTAR PUSTAKA

- Gonzales, Rafael C, '*Digital Image Processing*', Addison Wesley Publishing, 1977
- Jain, Anil K, '*Fundamentals of Digital Image Processing*', Prentice Hall International, 1989
- Munir, Rinaldi, '*Pengolahan Citra Digital dengan Pendekatan Algoritmik*', Penerbit Informatika, 2004
- Resmina, Lim, '*Face Recognition menggunakan metode Linear Discriminant Analysis (LDA)*', Proceeding Kommit , Universitas Gunadarma, 2002