

PENGELOMPOKAN CITRA RAMBU LALU LINTAS DENGAN *HIERARCHICAL AGGLOMERATIVE CLUSTERING* BERBASIS *SCALE INVARIANT FEATURE TRANSFORM*

Galla Zulhy Hidayat, Catur Supriyanto
Fakultas Ilmu Komputer, Universitas Dian Nuswantoro Semarang

Abstract In the field of transport, has been widely developed ideas for improving security in transit. To support this system, called Advanced Driver Assistance systems (ADAS). FENNEL has the purpose of helping system and direct the rider on while driving so it can improve traffic safety, traffic efficiency and improved environmental conditions. Traffic signs recognition systems is one important part of the operational system. Traffic signs are the signs erected at the side or at the top of the ramp to provide information to road users. Many countries use pictorial signs or writings which are simplified. In addition to standardized international traffic signs to facilitate international travel and in General to help improve traffic safety. Such pictorial signs use the symbol, using words and usually based on international protocol. In this case it takes a method that can recognize traffic signs image by way of extracting image feature called Scale Invariant Feature Transform (SIFT). Then the image feature will be grouped using Hierarchical Agglomerative Clustering algorithms. The results of this study, the performance of Hierarchical Agglomerative Clustering algorithms with techniques having performance linkage ward 52.5% better than single linkage techniques with a 35% performance, complete linkage with performance of 37.5% and average linkage with performance of 37.5%.

Keywords: hierarchical agglomerative clustering, scale invariant feature transform, SIFT, HAC, clustering, data mining, image vision

I. PENDAHULUAN

Dalam bidang transportasi, telah banyak dikembangkan ide-ide untuk meningkatkan keamanan dalam perjalanan. Sistem untuk mendukung hal ini disebut Advanced Driver Assistance System (ADAS) [1] [2], yang memiliki tujuan dasar untuk membantu dan mengarahkan pengemudi pada saat mengendarai sehingga dapat meningkatkan keselamatan lalu lintas, efisiensi lalu lintas dan perbaikan kondisi lingkungan. Rambu Lalu lintas adalah tanda yang didirikan di samping atau di atas jalan untuk memberikan informasi kepada pengguna jalan [1] [3]. Dengan volume lalu lintas meningkat sejak tahun 1930-an, banyak negara mengadopsi tanda-tanda bergambar atau tulisan yang disederhanakan [4]. Selain itu dibuat standar rambu lalu lintas internasional untuk memfasilitasi perjalanan internasional di mana perbedaan bahasa menciptakan hambatan, dan secara umum untuk membantu meningkatkan keselamatan lalu lintas. Tanda-tanda bergambar tersebut menggunakan simbol, menggunakan kata-kata dan biasanya didasarkan pada

protokol internasional.

Pada tahun 1999, Lowe [5] pertama kali memperkenalkan suatu metode ekstraksi fitur yang disebut Scale Invariant Feature Transform (SIFT). Pada Lowe [5] dan Lowe [6] dikatakan bahwa metode ini dapat memperoleh fitur yang tidak rentan oleh perubahan skala objek, terdapat translasi atau rotasi objek, serta sedikit terpengaruh oleh perbedaan intensitas cahaya, serta perbedaan proyeksi tiga dimensi pada objek yang dikenali. Terdapat pula algoritma Speeded Up Robust Features (SURF) [7] yang menurut Luo Juan dan Oubong Gwun [8] memiliki kelebihan pada performa namun kurang stabil pada rotasi dan perubahan cahaya, sedangkan SIFT memiliki kelebihan pada perubahan cahaya tergantung pada perbedaan rotasi, perubahan skala dan transformasi [9].

Hierarchical Agglomerative Clustering merupakan salah satu algoritma pengelompokan yang fungsinya dapat digunakan untuk mengelompokkan objek [10]. Algoritma ini dapat menghasilkan suatu kumpulan partisi berurutan, dimana dalam kumpulan tersebut terdapat kelompok yang mempunyai poin individu. Kelompok ini disebut single cluster yang terletak di level paling atas. HAC dimulai dengan setiap contoh sebagai kelompok yang terpisah [11]. Kelompok-kelompok ini digabungkan sampai hanya ada

satu kelompok yang tersisa. Metode clustering dibatasi dengan menggabungkan tiap data untuk meningkatkan hasil clustering. Ada pula algoritma klastering lain bernama K-Mean. Algoritma klaster ini juga memiliki fungsi yang sama seperti Hierarchical Agglomerative Clustering namun K-Means hanya menugaskan setiap objek dalam satu klaster [12].

II. METODE YANG DIUSULKAN

A. Rambu Lalu Lintas

Rambu lalu lintas merupakan perangkat yang paling sering digunakan dalam *Traffic Control Device* (TCD) [3]. Tanda-tanda ini menyampaikan pesan dalam kata-kata atau simbol dan digunakan untuk mengatur, memperingatkan, atau memandu pengguna jalan (pengendara, pejalan kaki, dll.). Rambu lalu lintas yang umum digunakan sebagai alat keselamatan lalu lintas, terutama dikembangkan untuk memberikan informasi penting dalam waktu singkat untuk mendukung pengguna jalan dengan aman [4].

B. Scale Invariant Feature Transform

Sebagai metode ekstraksi fitur pada pengenalan objek, SIFT memiliki kelebihan-kelebihan sebagai berikut [17]:

- Hasil ekstraksi fitur bersifat invarian terhadap ukuran, translasi dan rotasi dua dimensi.
- Hasil ekstraksi fitur bersifat invarian sebagian terhadap perubahan iluminasi dan perubahan sudut pandang tiga dimensi.
- Mampu meng-ekstrak banyak *keypoint* dari citra yang tipikal.
- Hasil ekstraksi fitur mencirikan secara khusus (*distinctive*).

Dengan kelebihan-kelebihan tersebut, penggunaan metode SIFT banyak dikembangkan untuk aplikasi pengenalan objek.

Secara garis besar, algoritma yang digunakan pada metode SIFT terdiri dari empat tahap [6], yaitu:

- Mencari Nilai Ekstrim Pada Skala Ruang (Scale-Space Extrema Detection)
- Menentukan *Keypoint* (*Keypoint Localization*)
- Penentuan Orientasi (*Orientation Assignment*)
- Deskriptor *Keypoint* (*Keypoint Descriptor*)

B.1 Scale-Space Extrema Detection

Pencarian nilai ekstrim pada skala ruang merupakan tahap awal dalam penentuan *keypoint* dari suatu citra. Dengan menggunakan fungsi Gaussian, citra pada skala ruang dapat didefinisikan sebagai fungsi $L(x,y,\sigma)$, yang diperoleh dari hasil konvolusi skala-variabel Gaussian, $G(x,y,\sigma)$ [6] [9], dengan citra masukan $I(x,y)$, sehingga diperoleh:

$$L(x,y,\sigma)=G(x,y,\sigma)*I(x,y)$$

dimana * adalah operasi konvolusi antara x dan y dan

$G(x,y,\sigma)$ adalah fungsi Gaussian, dengan persamaan sebagai berikut [6]:

$$G(x,y,\sigma)=\frac{1}{(2\pi\sigma^2)^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

Citra hasil Difference-of-Gaussian, (x, y, σ) , diperoleh dengan melakukan operasi konvolusi pada citra masukan dengan filter Difference of Gaussian, maka [6]:

$$\begin{aligned} D(x,y,\sigma) &= (G(x,y,k\sigma)-G(x,y,\sigma))*I(x,y) \\ &= L(x,y,k\sigma)-L(x,y,\sigma) \end{aligned}$$

terlihat bahwa citra hasil Difference-of-Gaussian sebenarnya merupakan selisih antara citra hasil pengkaburan Gaussian dengan nilai skala k yang berbeda.

C. Keypoint Localization

Setelah kandidat *keypoint* ditemukan melalui tahapan pada penjelasan sebelumnya, maka langkah selanjutnya ialah untuk mengambil detail dari kandidat *keypoint* tersebut [9]. Detail yang diambil merupakan lokasi, skala dan rasio kelengkungan inti dari kandidat *keypoint*. Pada tahap ini akan terjadi pengurangan jumlah kandidat *keypoint*. Dimana setiap kandidat *keypoint* yang dianggap sangat rentan terhadap gangguan (*noise*) akan dihilangkan, yaitu kandidat *keypoint* yang memiliki nilai kontras yang rendah dan kandidat *keypoint* yang kurang jelas dan terletak di sepanjang tepi.

Dari penggabungan Deret Taylor dan posisi *extrema*, dihasilkan persamaan :

$$D(\hat{x}) = D + \frac{1}{2} \frac{\partial D^T}{\partial X} \hat{x}$$

Pada SIFT ini, semua nilai ekstrim, $|D(\hat{x})|$ yang bernilai kurang dari 0,03 akan dihilangkan. Untuk menghilangkan *keypoint* yang kurang jelas pada tepi maka digunakan persamaan berikut [6]:

$$\frac{Tr(H)^2}{Det(H)} < \frac{(r+1)^2}{r}$$

dimana H merupakan matrix Hessian 2x2 dan r merupakan ambang batas dari kecekungan inti yang diperbolehkan. Pada SIFT ini, semua *keypoint* yang memiliki nilai r lebih besar dari 10 akan dihilangkan.

D. Orientation Assignment

Pada tahap ini, masing-masing *keypoint* yang diperoleh akan diberikan suatu orientasi yang tetap berdasarkan sifat-sifat lokal pada citra. Dengan adanya proses ini maka *keypoint* yang diperoleh dapat direpresentasikan relatif terhadap orientasi ini sehingga *keypoint* yang dihasilkan tidak terpengaruh terhadap adanya rotasi pada citra. Untuk menentukan orientasi dari masing-masing *keypoint* maka dilakukan perhitungan terhadap besarnya gradien dan sudut

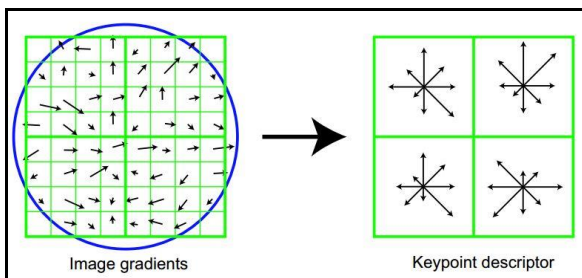
arah orientasi [9]. Adapun perhitungan terhadap besar nilai gradien, $m(x,y)$, dan arah orientasi, $\theta(x,y)$, dilakukan menggunakan persamaan berikut [6]:

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2}$$

$$\theta(x,y) = \tan^{-1}((L(x,y-1) - L(x,y+1))/(L(x+1,y) - L(x-1,y)))$$

E. Keypoint Descriptor

Pada proses ini, masing-masing *keypoint* yang telah diorientasikan akan diberikan pencirian khusus (deskriptor) [9]. Proses ini bertujuan untuk mendapatkan *keypoint* yang invarian terhadap perubahan intensitas cahaya atau perubahan sudut pandang tiga dimensi. Deskriptor akan diukur sebagai suatu histogram orientasi pada wilayah piksel dengan ukuran 4x4. Nilai orientasi diperoleh dari citra Gaussian yang memiliki skala terdekat dengan skala *keypoint* yang akan dihitung. Agar *keypoint* yang diperoleh invarian terhadap orientasi, maka koordinat dari deskriptor dan gradien orientasi akan di rotasi relatif terhadap orientasi dari *keypoint*. Kemudian fungsi pembebanan Gaussian, dengan besar nilai σ satu setengah kali dari besar jendela deskriptor, akan digunakan sebagai pembebanan pada setiap besaran nilai dari titik sampel. Proses ini ditunjukkan pada lingkaran yang terdapat pada **Gambar 1** sebelah kiri



Gambar 1 Deskriptor dari perhitungan besar gradien dan orientasi

Deskriptor *keypoint* pada Gambar 2.4 menunjukkan adanya 8 arah pada masing-masing histogram orientasi dengan panjang masing-masing anak panah sesuai dengan besar nilai dari histogram asal. Selanjutnya deskriptor *keypoint* yang telah diperoleh akan dinormalisasi untuk mengatasi pengaruh perubahan cahaya.

Setelah melalui tahapan-tahapan tersebut, maka pada hasil akhir akan diperoleh suatu citra dengan *keypoint* yang invarian terhadap berbagai macam perubahan. *Keypoint* ini yang kemudian menjadi fitur-fitur lokal pada suatu citra dan akan di cocokkan dengan *keypoint-keypoint* yang terdapat pada citra lain untuk menyesuaikan dengan objek yang tersedia pada data set training

F. Clustering Analysis

Clustering analysis merupakan metode pengelompokan setiap objek ke dalam satu atau lebih dari satu kelompok, sehingga tiap objek yang berada dalam satu kelompok akan memiliki nilai interaksi yang sama [10]. *Clustering analysis* bertujuan untuk membentuk kelompok dengan karakteristik yang sama. Pada algoritma *clustering*, data akan

dikelompokkan berdasarkan kemiripan satu data dengan data yang lain. Data yang dikelompokkan dalam satu *cluster* memiliki kemiripan yang tinggi, sedangkan antara data pada satu *cluster* dengan data pada *cluster* lainnya memiliki kemiripan yang rendah. Prinsip dari *clustering* adalah memaksimalkan kesamaan antar anggota satu kelas dan meminimumkan kesamaan antar kelas atau *cluster*. Banyak algoritma *clustering* memerlukan fungsi jarak untuk mengukur kemiripan antar data. Algoritma *cluster* yang digunakan dalam penilitan ini adalah *hierarchical clustering*, karena algoritma tersebut dapat menghasilkan jarak antar titik yang kemudian dikelompokkan berdasarkan karakteristik yang sama sehingga terbentuk satu cluster.

G. Hierarchical Agglomerative Clustering

Hierarchical Agglomerative Clustering merupakan prosedur yang membentuk perbedaan matriks menjadi urutan partisi bersarang [13]. Perbedaan matriks dituliskan **D** yang berbentuk persegi dan matriks simetris yang berisi perbedaan matriks yang berpasangan disetiap sampelnya. Objek yang akan di-*cluster* didefinisikan dengan O :

$$O = \{o_1, o_2, o_3, \dots, o_n\}$$

Tiap elemen **D** didefinisikan sebagai $D_{ij} = \text{dissimilarity}(o_i, o_j)$, dengan $i, j = 1 \dots n$.

Hierarchical Clustering menghasilkan output yang selalu sama. Sebuah partisi yang di inialisasi P dari objek n membagi O menjadi himpunan bagian $\{S_1, S_2, \dots, S_m\}$ yang memenuhi aturan berikut :

$$S_i \cap S_j = \emptyset, \quad \text{for } i, j \in [1, m], i \neq j$$

$$S_1 \cup S_2 \cup \dots \cup S_m = O$$

Partisi P_1 bersarang dengan P_2 jika setiap komponen P_1 merupakan bagian dari P_2 . Dengan cara ini, partisi dapat dibentuk dengan menggabungkan partisi bersarangnya. Penggunaan *agglomerative* algoritma *hierarchical clustering* menghasilkan jarak tiap objek n [18] yang kemudian digunakan teknik penggabungan *single linkage*. *Single linkage* sendiri merupakan teknik pengelompokan objek berdasarkan jarak antara objek dalam dua kelompok [9]. Rata-rata jarak D_{RQ} diantara dua kelompok, didefinisikan sebagai nilai dari semua jarak setiap objek dalam kelompok R dan setiap objek dalam kelompok Q [13]:

$$D_{RQ} = \frac{\sum_{i \in R} \sum_{j \in Q} D_{ij}}{|R| \cdot |Q|}$$

Dimana $|\cdot|$ menunjukkan kardinalitas *cluster*. Untuk membangun perbedaan matriks D berukuran $N \times N$, di mana N adalah jumlah total gambar yang akan dikelompokkan. Prosedur berikut ini digunakan untuk mengevaluasi perbedaan antara gambar A_i, A_j yaitu elemen D_{ij} dari

matrik. Fitur vektor diekstraksi dari gambar A_i dan A_j , kemudian pencocokan dilakukan dengan mencari kandidat keypoint berdasarkan jarak *Euclidean* dari fitur vektor, seperti yang diusulkan pada [6]. Perbandingan antara dua keypoint di A_i dan A_j diterima jika jarak vektor fitur kurang dari *Threshold* *distRatio* (didefinisikan pada [6]) dengan jarak perbandingan terdekat kedua. Hasilnya adalah ditemukan sejumlah kecocokan keypoint pada gambar. Karena pencocokan A_i terhadap A_j tidak menghasilkan kecocokan yang sama dengan pencocokan A_i terhadap A_j , Sedangkan ketidaksamaan matrik harus simetris. Sehingga perlu dilakukan pencocokan dua kali, satu untuk pasangan (A_i, A_j) , dan satu lagi untuk pasangan (A_j, A_i) . Jumlah maksimum kecocokan keypoint ditemukan dalam dua pencocokan merupakan hasil akhir pencocokan untuk pasangan gambar tertentu. Jumlah kecocokan keypoint diatas ditransformasikan ke rasio perbedaan (DR_{ij}) antara dua gambar yang dibandingkan dengan menggunakan rumus :

$$DR_{ji} = DR_{ij} = 100 \left(1 - \frac{M_{ij}}{\min(K_i, K_j)} \right)$$

Dimana M_{ij} merupakan jumlah maksimum kecocokan keypoint yang ditemukan diantara pasangan (A_i, A_j) , (A_j, A_i) dan K_i, K_j adalah keypoint yang ditemukan pada A_i, A_j berturut-turut. $DR_{ij} \in [0, 100]$ dan nilai tertinggi DR_{ij} menunjukkan perbedaan besar antar citra. DR_{ij} dianggap sebagai elemen ketidaksamaan matrik D_{ij} , yang dibangun untuk citra N .

H. Single Linkage

Single Linkage merupakan teknik pengelompokan dengan mencari jarak terdekat antara dua objek yang telah terkuster [20]. Dengan menggunakan persamaan berikut single linkage bekerja [21] :

$$d(r, s) = \min(\text{dist}(x_{ri}, x_{sj})), i \in (1, \dots, n_r), j \in (1, \dots, n_s)$$

Dimana :

x = objek dalam kluster

r, s = kluster

I. Complete Linkage

Complete Linkage merupakan teknik pengelompokan objek dengan mencari jarak maksimum antara objek dari dua kluster [HYPERLINK \l "Pai14" 22], complete linkage juga merupakan teknik kebalikan dari single linkage. Dengan persamaan berikut dapat diperoleh hasil complete linkage:

$$d(r, s) = \max(\text{dist}(x_{ri}, x_{sj})), i \in (1, \dots, n_r), j \in (1, \dots, n_s) \quad (2)$$

Dimana :

x = objek dalam kluster

r, s = kluster

J. Ward Linkage

Ward Linkage menggunakan *sum of squares* yaitu untuk peningkatan total dalam jumlah kuadrat kluster sebagai hasil dari gabungan dua kelompok. Dalam kluster *sum of squares* didefinisikan sebagai jumlah kuadrat dari jarak antara semua objek dalam kluster dan pusat kluster [23]. Berikut persamaannya :

$$d(r, s) = \sqrt{\frac{2n_r n_s}{(n_r + n_s)} \|\bar{X}_r - \bar{X}_s\|_2} \quad (2.14)$$

Dimana :

$\|\ \|_2$ = Euclidian distance

\bar{X}_r dan \bar{X}_s = centroid dari kluster r dan s

n_r dan n_s = jumlah elemen kluster r dan s

K. Average Linkage

Average linkage juga dikenal sebagai rata-rata dari kelompok yang menggunakan semua jarak rata-rata semua pasangan obyek dalam dua kelompok/kluster [13][22].

Berikut persamaan average linkage:

$$d(r, s) = \frac{1}{n_r n_s} \sum_{i=1}^{n_r} \sum_{j=1}^{n_s} \text{dist}(x_{ri}, x_{sj}) \quad (2.15)$$

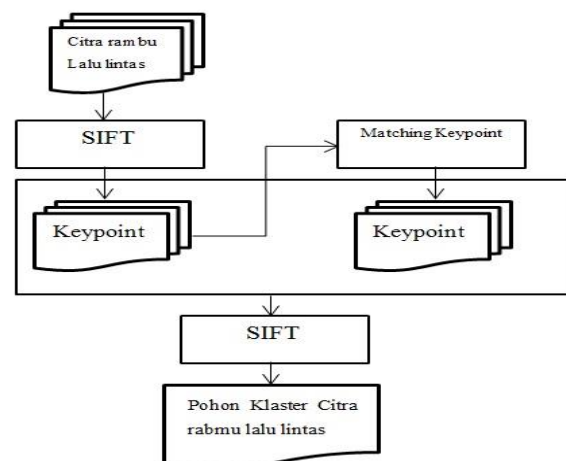
Dimana :

x = objek dalam kluster

r, s = kluster

III. IMPLEMENTASI

Metode yang diusulkan oleh peneliti pada penelitian ini menggunakan Algoritma Hierarchical Agglomerative Clustering dan Scale Invariant Feature Transform. Berikut ini adalah gambaran metode yang diusulkan oleh peneliti:



Gambar 2 Metode Yang Diusulkan

IV. HASIL & PEMBAHASAN

A. Pre-Processing Citra

Langkah *pre-processing* citra rambu lalu lintas sebagai data uji. Berikut merupakan langkah-langkah dalam proses *pre-processing* :

a. Menyiapkan citra rambu lalu lintas

Siapkan 40 citra rambu lalu lintas berekstensi *.jpg terdiri dari 4 jenis rambu lalu lintas (rambu perintah, rambu larangan, rambu peringatan, rambu petunjuk) yang setiap jenisnya terdapat 10 citra rambu lalu lintas. Baris kode untuk membaca citra

```
image=imread(imageFile);
```

b. Konversi citra RGB (*Red Green Blue*) ke *Grayscale*

Proses *grayscale* adalah mengubah citra RGB menjadi citra keabuan dengan menggunakan fungsi *rgb2gray* pada matlab. Diperlukan juga kondisi untuk mengecek apakah citra yang diinput merupakan citra RGB atau tidak. Koding dalam matlab :

```
If size(image,3)==3
    image=rgb2gray(image);
end
```

B. Ekstraksi Citra Menggunakan *Scale Invariant Feature Transform*

Ekstraksi fitur dilakukan agar memperoleh keypoint yang merupakan suatu titik yang invarian terhadap ukuran, translasi, dan rotasi. Hasil ekstraksi fitur mencirikan secara khusus. Tahap selanjutnya membuat sebuah file temporary berkecstensi *.pgm untuk menyimpan keypoint. Koding dalam matlab,

```
[rows, cols] = size(image);
f = fopen('tmp.pgm','w');
fwrite(f,image,'uint8');
fclose(f);
```

Kemudian menjalankan file *executable* 'siftWin32.exe'. Di dalam file tersebut sudah terdapat proses pencarian nilai ekstrim pada skala ruang, mengalokasi keypoint, penentuan orientasi, deskripsi keypoint. Koding pada matlab,

```
command = '!siftWin32';
command = [command ' <tmp.pgm >tmp.key'];
eval(command);
```

kemudian hasil citra yang telah diekstraksi disimpan pada file 'tmp.key'. untuk mendapatkan keypoint yang tersimpan pada file 'tmp.key' dengan menggunakan koding,

```
g = fopen('tmp.key','r');
[header, count] = fscanf(g,'%d %d', [1 2]);
keypoint = header(1);
```

Tabel 1 Keypoint Citra Rambu Lalu Lintas

Rambu	Keypoint	Rambu	Keypoint
RL1.jpg	86	RPt1.jpg	102
RL2.jpg	172	RPt2.jpg	149
RL3.jpg	65	RPt3.jpg	96
RL4.jpg	48	RPt4.jpg	85

RL5.jpg	95	RPt5.jpg	67
RL6.jpg	70	RPt6.jpg	46
RL7.jpg	65	RPt7.jpg	118
RL8.jpg	215	RPt8.jpg	54
RL9.jpg	89	RPt9.jpg	123
RL10.jpg	270	RPt10.jpg	42
RPg1.jpg	182	RPj1.jpg	326
RPg2.jpg	40	RPj2.jpg	280
RPg3.jpg	77	RPj3.jpg	675
RPg4.jpg	54	RPj4.jpg	442
RPg5.jpg	103	RPj5.jpg	151
RPg6.jpg	157	RPj6.jpg	565
RPg7.jpg	166	RPj7.jpg	262
RPg8.jpg	120	RPj8.jpg	394
RPg9.jpg	90	RPj9.jpg	186
RPg10.jpg	241	RPj10.jpg	321

Keterangan :

RL = Rambu Larangan
 RPg = Rambu Peringatan
 RPt = Rambu Perintah
 RPj = Rambu Petunjuk

C. Proses Pencocokan Citra

Pencocokan citra rambu lalu lintas dilakukan setelah memperoleh keypoint pada masing-masing citra yang hendak dicocokkan. Sebelum mencocokkan 2 citra perlu menentukan nilai *threshold*, pada penelitian ini *threshold* diberi nilai sebesar 0,9. Koding pada matlab menggunakan fungsi *match()*,

```
m = match(image1, image2);
```

D. Menghitung *Distance Ratio* menggunakan *Hierarchical Agglomerative Clustering*.

Untuk menghitung *Distance Ratio* antar citra rambu lalu lintas perlu membentuk matrik $N \times N$, dimana N merupakan jumlah citra rambu lalu lintas yang akan diklaster. Pada penelitian ini menggunakan 40 citra sehingga akan terbentuk matrik sebanyak 40×40 . Dengan menggunakan persamaan 2.11 yang ditulis kedalam koding matlab,

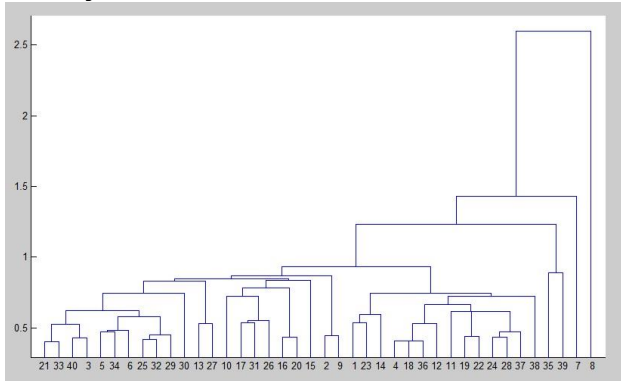
```
DR = 100 * (1 - (m/min(k1, k2)));
```

Setelah mendapatkan *Distance Ratio* tahap berikutnya *Distance Ratio* harus dikelompokkan ke dalam klaster dengan menggunakan fungsi *linkage*. Sedangkan fungsi *linkage* sendiri memiliki beberapa metode untuk mengukur

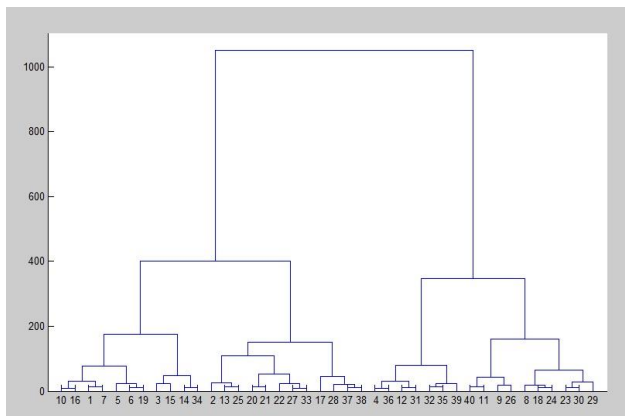
jarak tiap kluster. Pada penelitian ini peneliti menggunakan metode *ward, single, complete, average*. Berikut koding dalam matlab,

```
method = 'single'; %single,ward,average,complete
Z = linkage(matrikij,method);
```

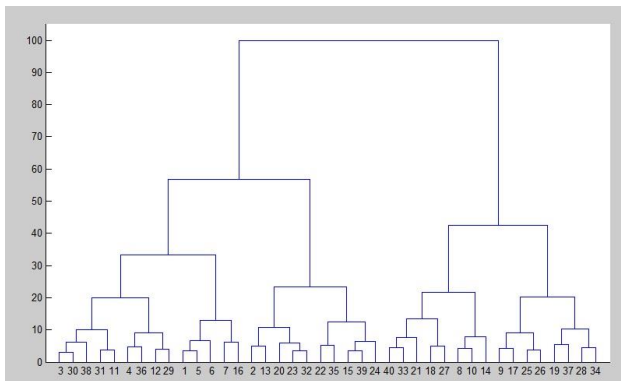
Dendrogram akan menampilkan diagram pohon kluster berdasarkan hasil dari *linkage*. Cara menggunakan *toolbox dendrogram* matlab dengan menggunakan fungsi berikut, `dendrogram(Z,40)`;



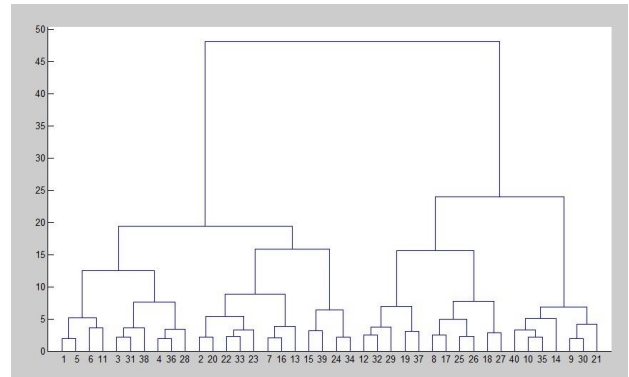
Gambar 3 1 Dendrogram Single Linkage



Gambar 4 Dendrogram Ward Linkage



Gambar 5 Dendrogram Complete Linkage



Gambar 6 Dendrogram Average Linkage

Label tersebut kemudian ditulis kedalam table kluster sebagai berikut:

Tabel 2 Label Kluster Average Linkage

Kluster 1	Kluster 2	Kluster 3	Kluster 4
1,5,6,11,3, 31,38 4,36,28	2,20,22,33 ,23,7,16 13,15,39,2 4,34	12,32,29,1 9,37,8 17,25,26,1 8,27	40,10,35,1 4,9 30,21S

Tabel 3 Label Kluster Complete Linkage

Kluster 1	Kluster 2	Kluster 3	Kluster 4
3,30,38,31 ,11,4 36,12,29,1 ,5,6 7,16	2,13,20,23 ,32,22 35,15,39,2 4	40,33,21,1 8,27 8,10,14	9,17,25,26, 19 37,28,34

Tabel 4 Label Kluster Ward Linkage

Kluster 1	Kluster 2	Kluster 3	Kluster 4
10,16,1,7, 5,6 19,3,15,14 ,34	2,13,25,20, 21,22 27,33,17,2 8,37,38	4,36,12,31 ,32 35,39	40,11,9,26, 8,18 24,23,30,2 9

Tabel 5 Label Kluster Single Linkage

Kluster 1	Kluster 2	Kluster 3	Kluster 4
21,33,40,3,5,34 6,25,32,29,30 13,27,10,17,31,26 16,20,15,2,9 1,23,14,4,18,36 12,11,19,22,24,28 37,38	35,39	7	8

E. Performa Kluster Menggunakan Purity

Purity diperoleh dari hasil kluster dengan average linkage, complete linkage, single linkage, dan ward linkage. Dengan menjumlahkan kelas yang paling dominan tiap kluster [24].

$$purity(\Omega, C) = \frac{1}{N} \sum_k \max_j |\omega_k \cap c_j|$$

Tabel 6 Hasil Purity Tiap Linkage

	single	ward	complete	Average
Purity	0,35	0,525	0,375	0,375

menunjukkan bahwa *ward linkage* memiliki performa klustering lebih baik daripada *single linkage*, *complete linkage*, *average linkage*

V. PENUTUP

Berdasarkan penelitian yang telah dilakukan, maka dapat disimpulkan berdasarkan uji coba yang telah dilakukan menggunakan data citra rambu lalu lintas menunjukkan bahwa terdapat banyak titik yang tidak cocok ketika mencocokkan citra rambu lalu lintas dan pada pengujian performa klustering untuk 40 citra rambu lalu lintas, teknik *ward linkage* memiliki performa 52.5% yang lebih baik dari teknik *single linkage* dengan performa 35%, *complete linkage* dengan performa 37,5%, dan *average linkage* dengan performa 37,5%.

REFERENCES

- [1] G. Yannis and C. Antonious, "The role of Advanced Driver Assistance Systems on traffic safety and efficiency," *National University of Athens.*, vol. 4-18.
- [2] Meng Lu, Kees Wevers, and Rob Van Der Heijden, "Technical feasibility of Advanced Driver Assistance Systems (ADAS) for road traffic safety," *Transportation Planning and Technology*, vol. 28, no. 3, pp. 167-187, 2005.
- [3] Olyemisi O. and David A., "Understanding of Traffic Signs by Drivers – A Case of Akure City , Ondo State, Nigeria," *ARNP Journal of Science and Technology*, vol. 2, no. 7, pp. 608-612, August 2012.
- [4] Larsson F. and Felsberg M., "Using Fourier Descriptors and Spatial Models for Traffic Sign Recognition," in *SCIA konferens*, 2011, pp. 238-249.
- [5] D. G. Lowe, "Object recognition from local scale-invariant features," in *International Conference on*, Corfu, 1999.
- [6] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, 2004.
- [7] Bay H., Tuytelaars T., and Van Gool L., "SURF : Speeded Up Robust Features," in *9th European Conference on Computer Vision*, 2006.
- [8] Juan Luo and Gwun Oubong, "A Comparison of SIFT, PCA-SIFT and SURF," *International Journal of Image Processing (IJIP)*, vol. 3, no. 4.
- [9] P. N. Andono, E. M. Yuniarno, M. Hariadi, and V. Venus, "3D reconstruction of under water coral reef images using low cost multi-view cameras," in *Multimedia Computing and Systems (ICMCS)*, 2012, pp. 803 - 808.
- [10] Anil K. Jain and Richard C. Dubes, *Algorithms for Clustering*] *Data*. Englewood Cliffs, N.J: Prentice-Hall, 1988.
- [11] Zhao Haifeng and Qi Zijie, "Hierarchical Agglomerative] Clustering with Ordering Constraints," in *Department of Computer Science*, 2010, pp. 195-199.
- [12] Sueli A. M. and Joab O. L., "Comparing SOM neural network with] Fuzzy c-means, K-means and traditional hierarchical clustering algorithms," *European Journal of Operational Research*, vol. 174, pp. 1742–1759, 2006.
- [13] Panagiotis Antonopoulos, Nikos Nikolaidis, and Ioannis Pitas,] "Hierarchical Face Clustering Using SIFT Image Features," in *Dept. of Informatics*, Thessaloniki, 2007, pp. 325 - 329.
- [14] Alexander A. S. Gunawan, Pascal Gerardus A., and Wikaria] Gazali, "Pendeteksian Rambu Lalu Lintas Dengan Algoritma Speeded Up Robust Features (SURF)," *Mathematics & Statistics Department*, vol. 13, pp. 91-96, Juli 2013.
- [15] Momeni Hajar, T Sadeghi Mohammad, and R Abutalebi Hamid,] "Fast Face Recognition Using a Combination of Image Pyramid and Hierarchical Clustering Algorithms," in *Wireless Communications & Signal Processing, 2009. WCSP 2009*, 2009, pp. 1-5.
- [16] Zachariasova Martina, Hudec Robert, Benco Miroslav, and] Kamencay Patrik, "The Object Recognition based on Scale-Invariant Feature Transform and Hybrid Segmentation," in *ELEKTRO*, 2012, pp. 109 - 113.
- [17] R. R. Arief, "ANALISIS PENGGUNAAN SCALE INVARIANT] FEATURE TRANSFORM SEBAGAI METODE EKSTRAKSI FITUR PADA PENGENALAN JENIS KENDARAAN," Universitas Indonesia, Depok, thesis 2010.
- [18] Fida Maisa Hana, "Identification System Finger Knuckle Print] Biometrics Using Histogram Equalization and Principal Component Analysis (PCA)," Universitas Dian Nuswantoro, Semarang, PhD Thesis 2014.
- [19] Ginanjar I., Hierarchical Clustering Untuk Otomatisasi] Pengelompokan Objek Pada Peta Dua Dimensi Hasil Analisis Multidimensional Scaling Satu Arah, 2008.
- [20] Havens T., Bezdek J., and Palaniswami M., "Scalable single] linkage hierarchical clustering for big data," *2013 IEEE Eighth International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pp. 396-401, 2013.
- [21] Sibson R., "SLINK : an optimally efficient algorithm for the] single-link cluster method," *King's College Research Centre*, vol. 16, p. 1.
- [22] Naidu Annan Paidi and Prabha Arna Jena, "Analysis of Complete-] Link Clustering for Identifying and Visualizing Multi-attribute Transactional Data using MATLAB," in *International Journal of Computer Applications*, 2014, pp. 44-50.
- [23] Fionn M. and Pierre L., "Ward's Hierarchical Clustering Method:] Clustering Criterion and Agglomerative Algorithm," *Science Foundation Ireland*, vol. II, pp. 62-85, December 2011.
- [24] Kumar Singh Vivek, Tiwari Nisha, and Garg Shekhar, "Document] Clustering using K-means, Heuristic K-means and Fuzzy C-means," *International Conference on Computational Intelligence and Communication Systems*, 2011.