

PENGELOMPOKKAN CITRA KENDARAAN (MOTOR DAN MOBIL) BERDASARKAN BENTUK MENGGUNAKAN ALGORITMA K-MEANS

Margi Cahyanti¹⁾, M Ridwan Dwi Septian²⁾, Ericks Rachmat Swedia³⁾, dan Ravi A Salim⁴⁾

^{1,4)}Jurusan Sistem Informasi, Fakultas Ilmu Komputer, Universitas Gunadarma

^{2,3)}Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Universitas Gunadarma

^{1,2,3,4)}Jalan Margonda Raya No.100, Pondok Cina, Depok

E-mail : margi@staff.gunadarma.ac.id¹⁾, ridwandwiseptian@staff.gunadarma.ac.id²⁾, ericks_rs@staff.gunadarma.ac.id³⁾
ravi@staff.gunadarma.ac.id⁴⁾

ABSTRAK

Penelitian ini mengimplementasikan algoritma K-Means untuk mengelompokkan data citra digital kendaraan mobil dan motor. Proses pengelompokkan terdiri dari tahapan citra *input* yang ditransformasi ke *grayscale*, hasil dari citra *grayscale* tersebut diambil nilai garis tepi menggunakan operator Prewitt. Hasil nilai citra Prewitt disimpan ke dalam basis data. Tahapan selanjutnya untuk mengelompokkan data citra tersebut menggunakan algoritma K-Means. Citra yang dikelompokkan adalah kendaraan motor dan mobil dan pengelompokkan dibagi menjadi dua jenis citra. Tahapan awal adalah menentukan centroid yang diambil dari dua data nilai citra yang terdapat dalam basis data secara acak. Tahapan akhir adalah menghitung jarak antar nilai menggunakan jarak *Euclidean* untuk mencari perpindahan kelompok. Jika tidak ada perpindahan data maka kelompok dianggap stabil dan proses pengelompokkan selesai. Uji-coba aplikasi ini menggunakan 12 data citra (enam data citra mobil dan enam data citra motor) dengan latar belakang berwarna putih. Citra ini diambil berukuran 150 x 100 piksel. Dengan adanya aplikasi ini dapat mempermudah pengguna dalam mengelompokkan data citra kendaraan.

Kata Kunci: *K-Means, Pengelompokkan, Prewitt*

1. PENDAHULUAN

Dalam ilmu komputer, pemrosesan citra digital adalah penggunaan algoritma komputer untuk melakukan pemrosesan citra digital. Pemrosesan citra digital memiliki banyak keunggulan dibandingkan pemrosesan citra analog. Hal ini memungkinkan berbagai algoritma yang lebih luas diterapkan pada data input dan dapat menghindari masalah seperti penumpukan distorsi sinyal selama pemrosesan (Chakravorty, 2018).

Dalam citra digital, RGB merupakan model warna dasar yang digunakan. Pada model warna RGB, setiap warna memperlihatkan komponen *spectral primary red*, *green*, dan *blue*. Model warna RGB adalah model warna tambahan di mana cahaya merah, hijau dan biru ditambahkan bersama dalam berbagai cara untuk mereproduksi berbagai warna yang luas. Nama model berasal dari inisial dari tiga warna primer tambahan, merah, hijau, dan biru (Artfacts, 2013).

Tujuan operasi pendeteksi tepi adalah untuk meningkatkan penampakan garis batas suatu daerah atau objek di dalam citra. Prewitt merupakan salah satu operator klasik dalam deteksi tepi yang berbasis pada perhitungan besarnya gradien. Kelebihan utama dari deteksi tepi klasik adalah selain mudah untuk digunakan, kompleksitasnya juga rendah. Namun demikian, operator klasik memiliki kelemahan, salah satunya adalah sensitif terhadap *noise* yang ada pada

citra (Kaur, 2016). Operator Prewitt didasarkan pada penggabungan citra dengan filter bernilai kecil, terpisah, dan *integer* dalam arah horizontal dan vertikal dan oleh karena itu relatif mudah dalam hal perhitungan dibandingkan dengan operator lain seperti sobel dan kannyali. Operator Prewitt dikembangkan oleh Judith M. S. Prewitt (Dim, 2013).

Pengelompokkan adalah metode untuk membagi sekumpulan data ke dalam sejumlah kelompok tertentu. Salah satu metode populer adalah algoritma K-Means. Dalam pengelompokkan dengan menggunakan K-Means, terjadi partisi kumpulan data ke dalam kelompok nomor *k* dari data. Algoritma K-means terdiri dari dua fase terpisah. Pada tahap pertama menghitung *k-centroid* dan pada fase kedua dibutuhkan perhitungan jarak dari setiap titik ke *cluster* yang memiliki titik pusat terdekat. Ada berbagai metode untuk menentukan jarak *centroid* terdekat dan salah satu metode yang paling banyak digunakan adalah jarak *euclidean*. Setelah pengelompokkan dihitung dengan mengambil jarak terdekat, dilakukan perhitungan ulang *centroid* baru. *Centroid* untuk setiap kelompok adalah titik di mana jumlah jarak dari semua objek dalam kelompok tersebut diminimalkan (Dhanachandra, 2015).

Algoritma K-Means telah banyak digunakan dalam penelitian untuk pengelompokkan ataupun segmentasi data citra. Penelitian yang dilakukan oleh Uldanay

Bairam dan Philip John Green mendeteksi keretakan pada buah Melon dengan menggunakan algoritma K-Means dengan hasil mampu mensegmentasikan bagian retak pada buah Melon (Bairam, 2018).

Pengelompokkan tanaman bunga iris menggunakan K-Means dilakukan oleh Febrianti dkk tahun 2015. Pada penelitian tersebut 150 bunga iris berhasil dikelompokkan berdasarkan panjang mahkota, lebar mahkota, panjang kelopak dan lebar kelopak. (Febrianti, 2015).

Mengelompokkan data citra digital secara otomatis membutuhkan beberapa proses pengolahan citra dan algoritma untuk pengelompokkan data. Berdasarkan metode pendeteksian tepi diharapkan dapat dilakukan pengelompokkan terhadap citra digital berdasarkan bentuk citra dengan menggunakan operator Prewitt, sedangkan pengelompokkan dapat menggunakan algoritma K-Means. Penelitian ini menjelaskan langkah-langkah pengolahan citra yang dilakukan agar citra kendaraan motor dan mobil dapat dikelompokkan dengan baik menggunakan algoritma K-Means.

Penelitian ini juga membangun sebuah aplikasi untuk mengelompokkan data citra kendaraan bermotor (mobil dan motor) secara otomatis dari basis data yang telah dimasukan oleh pengguna. Aplikasi yang dibangun diimplementasikan dengan menggunakan bahasa pemrograman C#. Basis data citra kendaraan terdiri dari citra motor dan mobil berukuran 150 x 100 piksel.

2. RUANG LINGKUP

Permasalahan pada penelitian ini dapat dilihat sebagai berikut :

1. Bagaimana mengelompokkan data citra digital secara otomatis, berdasarkan bentuk citra. Diharapkan dapat dilakukan klusterisasi terhadap citra digital dengan dikonversikan menggunakan operator Prewitt dan algoritma K-Means. Citra yang digunakan berupa citra motor dan mobil berukuran 150 x 100 piksel dengan latar belakang berwarna putih.
2. Menerapkan algoritma K-Means untuk pengelompokkan data citra.
3. Menggunakan 12 data citra kendaraan motor dan mobil dari bagian samping kendaraan. Uji coba ini diterapkan dengan mengelompokkan jenis kendaraan motor dan mobil. Format citra yang digunakan berbentuk JPG.

3. BAHAN DAN METODE

Pada bagian ini akan dibahas langkah-langkah pengolahan citra yang digunakan dalam penelitian ini dan algoritma *clustering* K-Means untuk mengelompokkan data citra yang telah diolah.

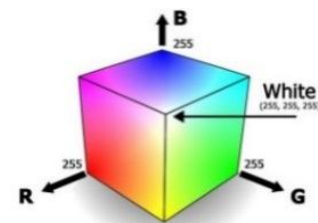
3.1 Model Warna RGB

Model warna RGB (*Red, Green, dan Blue*) adalah model warna tambahan di mana cahaya merah, hijau dan biru ditambahkan bersama dalam berbagai cara untuk mereproduksi berbagai warna yang luas. Nama model

berasal dari inisial dari tiga warna primer tambahan, merah, hijau, dan biru (Artfacts, 2013).

Tujuan utama dari model warna RGB adalah untuk penginderaan, representasi dan tampilan gambar dalam sistem elektronik, seperti televisi dan komputer, meskipun juga telah digunakan dalam fotografi konvensional. Sebelum era elektronik, model warna RGB sudah memiliki teori yang kuat di belakangnya, berdasarkan persepsi manusia tentang warna (Artfacts, 2013).

Terlihat pada gambar 1, setiap warna memperlihatkan komponen *spectral primary* yaitu, *red, green, dan blue*.



Gambar 1. Model warna RGB

RGB sering digunakan pada aplikasi komputer, karena dengan ruang warna ini tidak diperlukan transformasi untuk menyampaikan informasi pada layar monitor. Alasan di atas juga menyebabkan RGB banyak dimanfaatkan sebagai ruang warna dasar bagi sebagian besar aplikasi (Swedia, 2015).

3.2 Deteksi Tepi Prewitt

Operator Prewitt digunakan pada pemrosesan citra, khususnya dalam algoritma deteksi tepi. Secara teknis, ini adalah operator diferensiasi diskrit, menghitung aproksimasi gradien fungsi intensitas pada citra. Pada setiap piksel citra, hasil dari operator Prewitt adalah vektor gradien yang sesuai atau vektor. Operator Prewitt didasarkan pada penggabungan citra dengan filter bernilai kecil, terpisah, dan integer dalam arah horizontal dan vertikal dan oleh karena itu relatif murah dalam hal perhitungan seperti operator Sobel dan Kayyali. Tujuan operasi pendeteksi tepi untuk meningkatkan penampakan garis batas suatu daerah atau objek di dalam citra (Dim, 2013).

Metode Prewitt merupakan pengembangan metode Robert dengan menggunakan filter HPF (*High Pass Filter*) yang diberi satu angka nol penyangga. Metode ini mengambil prinsip dari fungsi Laplacian yang dikenal sebagai fungsi untuk membangkitkan HPF (*High Pass Filter*). Kernel *filter* yang digunakan dalam metode Prewitt ini adalah :

$$H = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \text{ dan } H = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad (1)$$

3.3 Algoritma Clustering

Clustering atau Pengelompokan adalah sebuah proses untuk mengelompokkan data ke dalam beberapa kelompok sehingga data dalam satu kelompok memiliki tingkat kemiripan yang maksimum dan data antar kelompok memiliki kemiripan yang minimum (Irwansyah, 2015).

Pengelompokan merupakan proses partisi satu set objek data ke dalam himpunan bagian yang disebut dengan kelompok (*cluster*). Objek yang di dalam kelompok memiliki kemiripan karakteristik antar satu sama lainnya dan berbeda dengan kelompok yang lain. Partisi tidak dilakukan secara manual melainkan dengan suatu algoritma pengelompokan. Oleh karena itu, pengelompokan sangat berguna dan dapat menemukan grup atau kelompok yang tidak dikenal dalam data.

Pengelompokan banyak digunakan dalam berbagai aplikasi seperti misalnya pada *business intelligence*, pengenalan pola citra, *web search*, bidang ilmu biologi, dan untuk keamanan (*security*). Pengelompokan juga dikenal sebagai data segmentasi karena pengelompokan mempartisi banyak data ke dalam banyak kelompok berdasarkan kesamaannya. Selain itu pengelompokan juga bisa digunakan untuk *outlier detection*, melihat data yang terasing dengan kelompok datanya (Irwansyah, 2015).

3.4 K-Means

K-Means merupakan algoritma pengelompokan yang paling umum dalam data mining dan secara luas digunakan untuk mengelompokkan data berukuran besar. Pada tahun 1967 MacQueen pertama kali mengenalkan algoritma K-Means, yang dapat mengelompokkan data dengan cepat dan mudah. Metode ini mengklasifikasikan setiap objek ke dalam k -kelompok yang berbeda melalui teknik iteratif, yang berulang hingga setiap objek dalam kelompok stabil atau tidak ada perpindahan objek dalam kelompok (Na, 2010).

K-Means merupakan salah satu metode data pengelompokan non hirarki yang berusaha mempartisi data yang ada ke dalam bentuk satu atau lebih kelompok. Metode ini mempartisi ke dalam kelompok sehingga data yang memiliki karakteristik yang sama (*High intra class similarity*) dikelompokkan ke dalam satu kelompok yang sama dan yang memiliki karakteristik yang berbeda (*Low inter class similarity*) dikelompokkan pada kelompok yang lain (Basuki, 2005). Proses pengelompokan dimulai dengan mengidentifikasi data yang akan dikelompokkan, X_{ij} ($i=1, \dots, n$; $j=1, \dots, m$) dengan n adalah jumlah data yang akan dikelompokkan dan m adalah jumlah variabel. Pada awal iterasi, pusat setiap kelompok ditetapkan secara bebas atau sembarang, C_{kj} ($k=1, \dots, k$; $j=1, \dots, m$). Kemudian dihitung jarak antara setiap data dengan setiap pusat kelompok. Untuk melakukan penghitungan jarak data ke- i (x_i) pada pusat kluster ke- k (c_k), diberi nama (d_{ik}), dapat digunakan formula *Euclidean* seperti pada persamaan (2) (Giyanto, 2008) :

$$d_{ik} = \sqrt{\sum_{j=1}^m k_{ij} - c_{kj}} \quad (2)$$

Suatu data akan menjadi anggota dari kelompok ke- k apabila jarak data tersebut ke pusat kelompok ke- k bernilai paling kecil jika dibandingkan dengan jarak ke pusat kelompok lainnya. Hal ini dapat dihitung dengan menggunakan persamaan (3). Selanjutnya, kelompokkan data-data yang menjadi anggota pada setiap kelompok (Giyanto, 2008).

$$\text{Min } \sum_{k=1}^k d_{ik} = \sqrt{\sum_{j=1}^m k_{ij} - c_{kj}} \quad (3)$$

Nilai pusat kelompok yang baru dapat dihitung dengan cara mencari nilai rata-rata dari data-data yang menjadi anggota pada kelompok tersebut, dengan menggunakan rumus pada persamaan 4 (Giyanto, 2008):

$$c_{kj} = \frac{\sum_{i=1}^p x_{ij}}{p} \quad (4)$$

Dimana $x_{ij} \in$ kelompok ke - k dan p adalah banyaknya anggota kelompok ke k . Algoritma dasar dalam K-Means adalah (Giyanto, 2008):

1. Tentukan jumlah kelompok (k), tetapkan pusat kelompok sembarang.
2. Hitung jarak setiap data ke pusat kelompok menggunakan persamaan (3).
3. Kelompokkan data ke dalam kelompok dengan jarak yang paling pendek menggunakan persamaan (3).
4. Hitung pusat kelompok yang baru menggunakan persamaan (4)

Ulangi langkah 2 sampai dengan 4 hingga sudah tidak ada lagi data yang berpindah ke kelompok yang lain (Giyanto, 2008).

3.5 Euclidean Distance

Jarak *Euclidean* merupakan jarak yang diukur lurus dari pusat titik yang satu ke titik yang lain. *Euclidean* sering digunakan karena mudah dimengerti dan mudah untuk dilakukan perhitungan. Aplikasi dari jarak *Euclidean* pada umumnya bisa dijumpai pada beberapa model konveyor, sistem transportasi dan distribusi (Gentle, 2007). Formulasi dari jarak *Euclidean* sebagai berikut:

$$\text{Euclidean} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (5)$$

keterangan :

$x_{i,j}$ = koordinat x untuk titik i, j

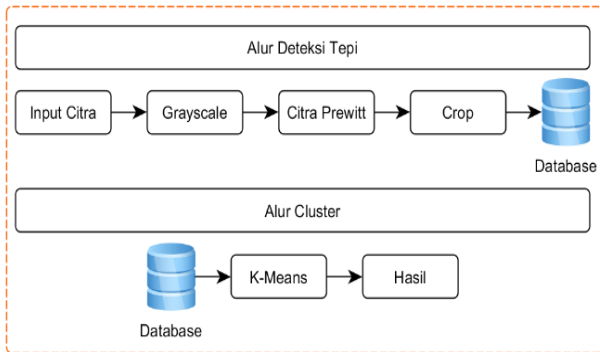
$y_{i,j}$ = koordinat y untuk titik i, j

Semakin kecil nilai $d(x, y)$, maka semakin mirip kedua vektor yang dicocokkan/dibandingkan. Sebaliknya semakin besar nilai $d(x, y)$ maka semakin berbeda kedua vektor yang dibandingkan (Santosa, 2007).

4. PERANCANGAN APLIKASI

Berikut ini adalah perancangan aplikasi yang menjelaskan proses dari awal aplikasi berjalan hingga aplikasi selesai. Terdapat dua alur diantaranya adalah

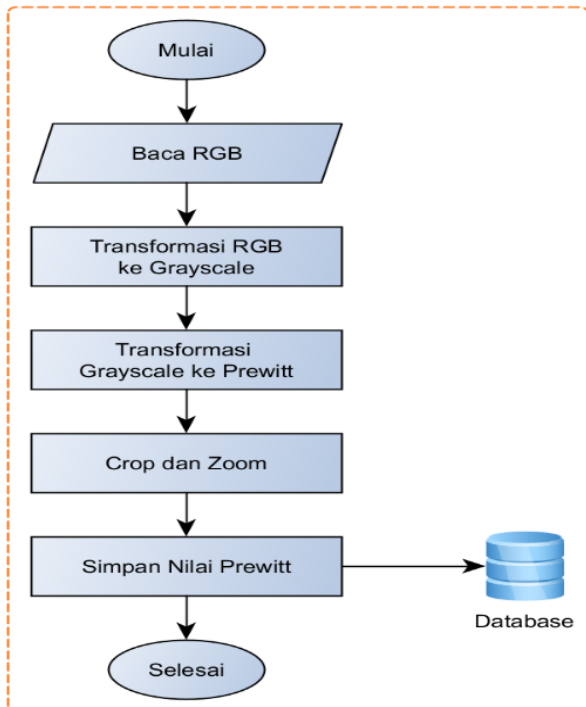
alur pra-proses citra dan alur pengelompokkan citra. Secara umum alur aplikasi yang dirancang pada penelitian ini dapat terlihat pada gambar 3.



Gambar 3. Alur Perancangan Aplikasi

4.1 Perancangan Pra-Proses Citra

Berikut ini adalah perancangan yang akan digunakan sebagai tahapan untuk menjelaskan pra-proses citra menggunakan operator Prewitt. Pada gambar 4, menjelaskan tentang alur yang digunakan untuk mendapatkan bentuk citra dengan menggunakan Prewitt. Pertama tahap membaca citra dimana citra yang dimasukan sebelumnya sudah berukuran 150 x 100 piksel. Selanjutnya citra RGB dikonversikan menjadi citra *greyscale* sehingga nilai citra pada R, G dan B memiliki nilai intensitas yang sama. Setelah dikonversikan ke *greyscale*, citra *greyscale* ini kemudian dicari nilai batas tepi citra dengan menggunakan operator Prewitt.

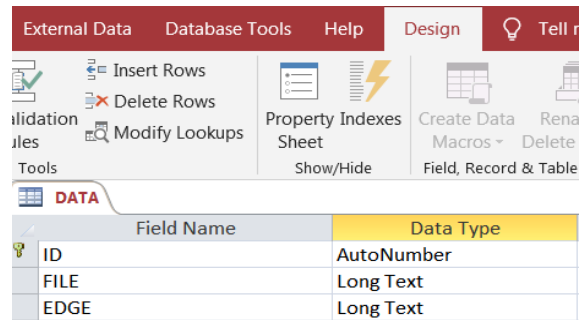


Gambar 4. Alur Pra-Proses Citra

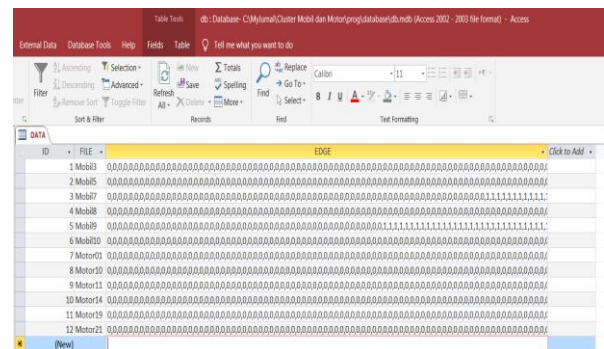
Selanjutnya citra yang sudah didapatkan batas tepinya oleh operator Prewitt di *crop dan zoom* agar citra

hasil perhitungan Prewitt berukuran sama dengan piksel awal sebesar 150 x 100. Kemudian nilai citra ini disimpan ke dalam basis data.

Basis data citra dibuat dengan menggunakan Microsoft Access berformat mdb, dengan desain dapat dilihat pada gambar 5a. Nilai deteksi tepi dari citra kendaraan berukuran 150 x 100 piksel disimpan dalam *field* EDGE, dengan cara mengubah data citra 2 dimensi (baris dan kolom) menjadi satu baris seperti terlihat dalam gambar 5b (Swedia, 2017).



Gambar 5a. Desain Basis Data



Gambar 5b. Nilai Deteksi Tepi dalam Basis Data

Proses mengubah citra hasil deteksi tepi menjadi vektor satu baris yang disimpan dalam basis data dilakukan dengan *pseudocode* berikut (Swedia, 2017),

```

int n = jumlah baris matriks citra
int m = jumlah kolom matriks citra
for (int i = 0; i < n; i++) {
    for (int j = 0; j < m; j++) {
        edge += matriks[i, j] + ',';
    }
}
    
```

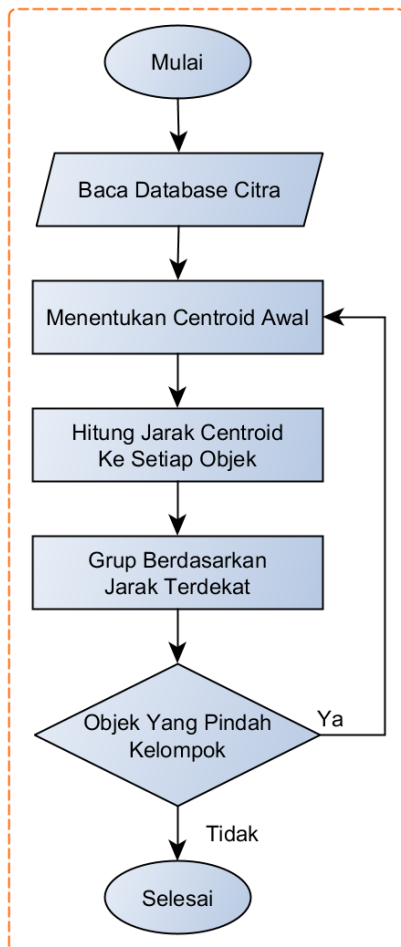
Vektor-vektor citra dalam *field* EDGE tersebut yang selanjutnya akan dilakukan proses pengelompokkan dengan algoritma K-Means..

4.2 Perancangan Pengelompokkan Citra

Pada gambar 6 diperlihatkan alur yang digunakan untuk mengelompokkan data citra yang sudah dilakukan pra-proses dengan menggunakan algoritma K-Means. Pertama data citra yang berada di basis data dibaca terlebih dahulu. Lalu *file* citra yang akan dikelompokkan

adalah kendaraan motor dan mobil maka pengelompokkan dibagi dua jenis citra, setelah itu perhitungan *centroid* awal akan diambil dua data nilai citra secara acak. Setelah itu dilakukan proses penentuan jarak dengan menggunakan formula *Euclidean distance*.

Setelah perhitungan *Euclidean distance* selesai maka data dikelompokkan dengan jarak yang terdekat. Proses selanjutnya mencari perpindahan kelompok. Bila tidak ada perpindahan kelompok maka proses pengelompokkan dianggap selesai dan mendapatkan hasil akhir dari pengelompokkan. Bila masih ada data yang mengalami perpindahan kelompok, maka dilakukan kembali perhitungan *centroid* dan proses kembali ke awal.

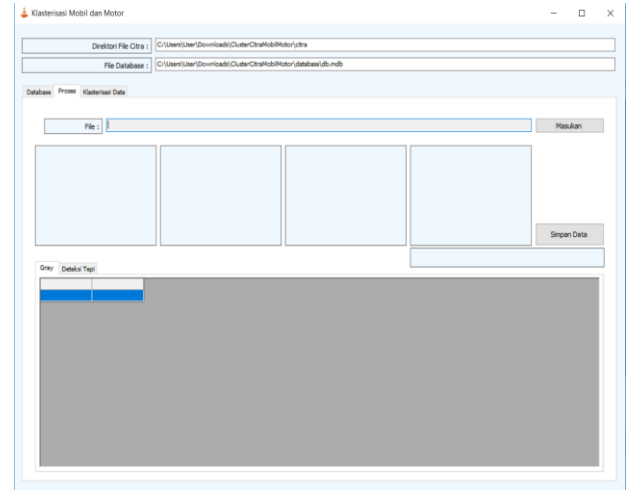


Gambar 6. Alur Pengelompokkan Citra

5. IMPLEMENTASI

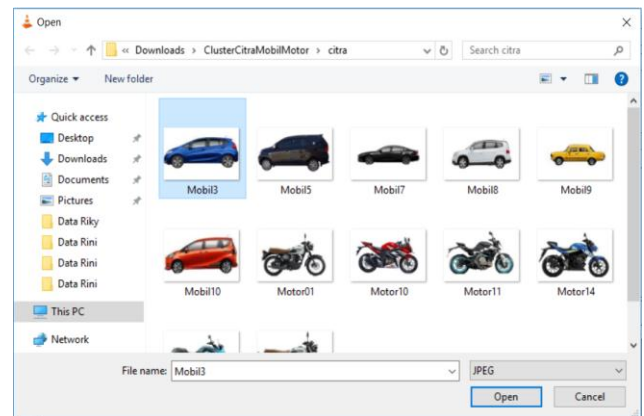
Pada tahap ini dilakukan ujicoba untuk memastikan bahwa aplikasi yang sudah dikembangkan dapat digunakan dengan baik. Antarmuka awal aplikasi dapat dilihat pada gambar 7. Pengguna dapat menggunakan aplikasi ini untuk mengelompokkan citra kendaraan motor dan mobil menggunakan algoritma K-Means. Untuk memulai prosesnya pengguna dapat menekan tombol “Masukan” untuk proses pemilihan citra dari *file*

yang terdapat dalam komputer. *File* citra yang telah dipilih ini kemudian akan diubah ke dalam *greyscale* lalu dicari batas tepinya menggunakan operator Prewitt kemudian di *crop* dan *zoom*. Setiap proses tersebut akan ditampilkan pada aplikasi ini. Setelah proses pengolahan citra selesai, data ini kemudian dimasukkan ke dalam basis data dengan cara menekan tombol “Simpan Data”.



Gambar 7. Desain Antarmuka

Pada gambar 8, diperlihatkan contoh pemilihan *file* citra untuk diproses pada aplikasi dan disimpan ke dalam basis data. Jendela pemilihan citra akan muncul ketika pengguna menekan tombol “Masukan”.



Gambar 8. Pemilihan File Citra













Pada gambar 9 adalah tampilan hasil dari pemilihan data citra dengan keterangan proses *input* citra sampai dengan teknik *cropping* dan *zoom*.



Gambar 9. Menampilkan Masukan Citra

merepresentasikan kelompok motor. Data uji coba sudah terlebih dahulu dimasukan dan melalui proses pengolahan citra *greyscale*, *crop* dan Prewitt. Hasil uji coba ditampilkan pada tabel 1.

Tabel 1. Hasil Pengelompokkan Citra

No	Citra	Nama Citra	Jenis Kendaraan	Hasil Kelompok
1		Mobil3	Mobil	C1
2		Mobil5	Mobil	C2
3		Mobil7	Mobil	C1
4		Mobil8	Mobil	C1
5		Mobil9	Mobil	C1
6		Mobil10	Mobil	C1
7		Motor1	Motor	C2
8		Motor10	Motor	C2
9		Motor11	Motor	C2
10		Motor14	Motor	C2
11		Motor19	Motor	C2
12		Motor21	Motor	C2

Pada proses pengelompokkan, sistem otomatis mengelompokkan data citra menjadi 2 kelompok, karena penelitian ini hanya mengelompokkan citra mobil dan citra motor. C1 merepresentasikan kelompok mobil dan C2 merepresentasikan kelompok motor. K-Means melakukan 4 iterasi dari hasil uji coba 12 citra kendaraan mobil dan motor, dengan pilihan centroid 1 adalah data nomor 3 dan centroid 2 adalah data nomor 8 yang dipilih secara acak oleh sistem, seperti terlihat pada gambar 14.

Pada tabel 1, data nomor 2 dengan nama citra Mobil5, terlihat data tersebut masuk ke dalam kelompok C2 (Motor), yang seharusnya data tersebut masuk ke dalam kelompok C1(Mobil). Citra tersebut mendapatkan hasil pengelompokkan yang tidak sesuai karena warna hitam yang terlalu dominan pada citra data nomor 2 ini. Dominannya warna hitam pada citra ini mengakibatkan hasil dari nilai operator Prewitt pada citra ini lebih cenderung mirip ke kelompok C2. Untuk 11 data citra lainnya mendapatkan hasil pengelompokkan yang sesuai.

6. KESIMPULAN

Setelah dilakukan pengujian pada aplikasi pengelompokkan citra kendaraan mobil dan motor menggunakan algoritma K-Means dan operator Prewitt dengan data yang diuji-cobakan berjumlah 12 data citra kendaraan, yang terdiri dari enam citra mobil dan enam citra motor berukuran 150 x 100 piksel. Aplikasi berjalan baik dengan menampilkan hasil sesuai perancangan. Selain itu aplikasi juga dapat menampilkan tahapan pengolahan citra mulai dari citra di masukan sampai menjadi citra Prewitt serta mendapatkan nilai piksel citra untuk disimpan di dalam basis data. Pada proses pengelompokkan aplikasi juga menampilkan beberapa tahapan proses yang digunakan hingga mendapatkan hasil akhir. *Centroid* akhir pada tahapan pengelompokkan juga ditampilkan di aplikasi yang telah dibangun.

Dari hasil uji coba 12 citra kendaraan mobil dan motor terdapat satu citra yang mendapatkan hasil tidak sesuai dari pengelompokkan. Kesalahan pengelompokkan pada citra ini lebih disebabkan oleh pra-proses citra yang tidak sempurna. Perlu ditambahkan pra proses citra seperti normalisasi citra agar tidak ada warna yang terlalu dominan pada setiap citra sebelum citra dilakukan proses pengelompokkan.

7. SARAN

Penelitian ini dapat dikembangkan lebih lanjut dengan menentukan nilai latar belakang citra asli tanpa harus diganti dengan latar belakang putih. Menyesuaikan posisi kendaraan di citra dalam segala arah dan posisi sehingga dapat mengurangi proses pengaturan citra yang dimasukan ke dalam basis data. Untuk Algoritma pengelompokkan dengan menggunakan algoritma pengelompokkan yang lainnya selain K-Means, agar waktu komputasi pengelompokkan dapat dibandingkan.

8. DAFTAR PUSTAKA

- Artfacts. 2013. The Evolution of Color Pigment Printing. <http://ww25.artfacts.org/artinfo/articals/evercolor.html>. (diakses tanggal 13 Nopember 2018).
- Bairam, U. and Green, P.J. 2018. Melon Crack Identification and Classification Using K-Means Clustering for Quality Inspection. *Colour and Visual Computing Symposium (CVCS)* (pp. 1-5). IEEE.
- Basuki, A. 2005. *Metode Numerik dan Algoritma Komputasi*. Yogyakarta : Andi.
- Chakravorty, P. 2018. *What Is a Signal? [Lecture Notes]*. IEEE Signal Processing Magazine, vol. 35, no. 5, pp. 175-177.
- Dhanachandra, N., Mangle, K., Chanu, Y.J., 2015. Image segmentation using K-Means clustering algorithm and subtractive clustering algorithm. *Procedia Computer Science*, 54, pp.764-771.
- Dim, Jules R., Tamio, T. 2013. *Alternative Approach for Satellite Cloud Classification: Edge Gradient Application*. *Advances in Meteorology*. : 1-8. doi:10.1155/2013/584816. ISSN 1687-9309.
- Febrianti, F., Hafiyusholeh, M. and Asyhar, A.H. 2016. Perbandingan Pengklusteran Data Iris Menggunakan Metode K-Means dan Fuzzy C-Means. *Jurnal Matematika "MANTIK"*, 2(1), pp.7-13.
- Gentle dan James, E. 2007. *Matrix transformations and factorizations*. Springer New York.
- Giyanto dan Heribertus. 2008. *Penerapan algoritma Clustering K-Means, K-Medoid, Gath Geva*. Yogyakarta: Universitas Gajah Mada.
- Han, J. 2012. *Data Mining: Concepts and Techniques (The Morgan Kaufmann Series in Data Management Systems) 3rd Edition*. Canada : Morgan Kaufmann.
- Irwansyah, E. 2015. *Advance Clustering: Teori dan Aplikasi*. Jakarta : DeePublish.
- Jang, J.S.R., Sun, C.T., Mizutani E. 2004. *Nuero-Fuzzy and Soft Computing*. Singapore : Pearson Education.
- Kaur, R., Singh, M., Singh, B. 2016. Comparative Analysis of Color Edge Detection Techniques Based on Fuzzy Logic. *International Journal of Engineering Sciences*.
- Kusrini dan Emha, L.T. 2009. *Algoritma Data Mining*. Yogyakarta : Andi
- McLachlan, G.J dan Peel, D. 2000. *Finite Mixture Models*. New York : John Wiley and Sons.
- Na, S., Xumin, L., Gong, Y. 2010. Research on k-means Clustering Algorithm: An Improved k-means Clustering Algorithm. *Third International Symposium on Intelligent Information Technology and Security Informatics*, Jingtangshan, 2010, pp. 63-67.
- Santosa, B. 2007. *Data Mining (Teori dan Aplikasi)*, Graha Ilmu : Yogyakarta.
- Shapiro, L.G dan George, S.C. 2002. *Computer Vision*. Prentice Hall.
- Swedia, E.R dan Cahyanti, M. 2015. *Algoritma Transformasi Ruang Warna*. Depok: Indie Publishing.
- Swedia, E.R., Septian, M.R.D., Cahyanti, M. 2017. Aplikasi Pendeteksi Rambu Lalu-Lintas Menggunakan Operator Sobel dan Metode Hamming. *Seminar Nasional Aplikasi Teknologi Informasi (SNATI)*.