

Aplikasi Edit Foto Background Dengan Menggunakan Metode K-Means Clustering

Ihsan Lubis¹, Herlina Andriani Simamora², Sayuti Rahman³, Rosyidah Siregar⁴, Husni Lubis⁵

^{1,5} Program Studi Sistem Informasi, ^{2,3,4} Program Studi Teknik Informatika
Universitas Harapan Medan
Jl. HM. Jhoni No. 70 Medan, Indonesia

¹ Ihsan.lubis@gmail.com, ² herlina.drialey@gmail.com,
³ masay.ram@gmail.com, ⁴ rosyidah_siregar.unhar@harapan.ac.id,
⁵ husni.lubis82@gmail.com

Abstract

Background photos become one of the most important things in every human activity. Where each background color contains its own meaning, depending on when the conditions where we use the background color. The background color is indispensable in the interests of humans in all matters relating to administration, for example, in the scope of work, school, and others. The background color change can be done by clustering the image. The initial stage done in changing the background color is to cluster the image first so that the background object can be found. In this study, the authors apply the k-means method to divide the objects into several parts. In this application before changing the background color we have to determine the number of clusters first, then the background color is changed according to the desired cluster. This application is designed using Matlab version 2010a. Test results ten images can be concluded that the success of background color changes can be done with a background that has one color while the background that has different colors can not change.

Keywords: Image, Background, K-means, Clustering, Processing Image, Matlab 2010a

1. PENDAHULUAN

Kebutuhan akan warna *background* foto yang berbeda-beda sering dirasakan seperti saat melamar pekerjaan, mencetak sertifikat dengan melampirkan foto, Membedakan warna *background* berdasarkan jenis kelamin dan sebagainya. Hal ini merupakan hal yang sangat mudah bagi mereka yang mampu menggunakan aplikasi yang banyak tersedia. Namun bagi kalangan awam hal ini sangat sulit dan memakan banyak waktu. Untuk itu perlu adanya sebuah aplikasi yang mampu mengubah *background* dengan cepat dan mudah. Dengan memanfaatkan clustering untuk menemukan wilayah *background* dan mengganti warna sesuai dengan keinginan pengguna.

Clustering merupakan pembagian suatu citra menjadi wilayah-wilayah yang berdasarkan pada kriteria kesamaan antara tingkat keabuan suatu pixel dengan tingkat keabuan pixel-pixel yang lainnya, pengelompokan clustering berdasarkan kedekatan dari suatu karakteristik sample yang ada. contohnya memisahkan suatu citra yaitu objek dan *background*. Sebuah *background* adalah suatu gambar yang di dalamnya terdapat objek-objek yang tidak bergerak (*obyek static*), perubahan-perubahan yang mungkin terjadi pada *background* adalah perubahan intensitas cahaya dari siang hari ke sore hari, perubahan bayangan benda yang terdapat pada *background* yang diakibatkan oleh perubahan posisi matahari, perubahan posisi benda pada *background*, penambahan benda dalam *background*, dan sebagainya.

Metode clustering yang digunakan pada penelitian ini menggunakan metode K-means. K-means merupakan salah satu metode data *clustering* non hirarki yang mempartisi data ke dalam cluster atau kelompok sehingga data yang mempunyai karakteristik yang sama akan dikelompokkan ke dalam satu *cluster* yang sama dan data yang mempunyai karakteristik yang berbeda akan dikelompokkan ke dalam kelompok yang lain.

Dalam penelitian ini peneliti merancang aplikasi untuk menganalisis tingkat keberhasilan dalam mengedit suatu *background* pada citra menggunakan metode k-means, dan mengubah warna

background tersebut. Pada penelitian ini menggunakan pemrograman matlab, dan foto dengan *background* yang berbeda-beda.

1.1 Clustering

Clustering adalah penyatuan sekelompok data dan mempunyai korelasi karakteristik sejenis atau sekelompok data yang memiliki kesamaan. *Clustering* juga merupakan suatu cara menganalisa data dengan cara mengelompokkan objek kedalam kelompok-kelompok berdasarkan suatu kesamaan tertentu. Proses *group* data berbeda dengan *Clustering*. Dalam *group*, pengelompokan dalam data harus sama, jadi sudah dipastikan yang termasuk dalam *group* itu dan memiliki karakter yang sama[1].

1.2 Citra

Citra adalah suatu representasi (gambaran), kemiripan, atau imitasi dari suatu objek sebagai salah satu komponen multimedia yang memegang peranan yang sangat penting sebagai bentuk informasi visual. Citra terbagi 2 yaitu citra yang bersifat analog dan citra yang bersifat digital. Citra analog adalah citra yang bersifat kontinu seperti gambar pada monitor televisi, foto sinar X, foto yang tercetak di kertas foto, lukisan, pemandangan alam, hasil CT-Scan dan lain sebagainya. Sedangkan pada citra digital merupakan citra yang dapat diolah oleh komputer. Citra digital dapat ditulis dalam bentuk matriks sebagai berikut[2].

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \vdots & \vdots & \ddots & \vdots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix}$$

Gambar 1. Nilai matriks pada citra digital

Secara matematis citra digital dapat dituliskan sebagai fungsi intensitas $f(x,y)$, dimana nilai x (baris) dan y (kolom) merupakan koordinat posisi dan $f(x,y)$ adalah nilai fungsi pada setiap titik (x,y) yang menyatakan besar intensitas citra atau tingkat keabuan atau warna dari pixel di titik tersebut.

1.3 Pengolahan Citra

Pengolahan citra merupakan suatu metode yang digunakan untuk memanipulasi atau memproses gambar dalam bentuk 2 dimensi. Pengolahan citra juga disebut suatu bentuk pengolahan atau pemrosesan sinyal dengan input berupa gambar (*image*) dan ditransformasikan menjadi gambar yang lain sebagai keluarannya dengan teknik tertentu. Pengolahan citra dilakukan untuk memperbaiki kesalahan data sinyal gambar yang terjadi akibat transmisi dan selama akuisisi sinyal, serta meningkatkan kualitas penampakan gambar agar mudah diinterpretasi oleh sistem penglihatan manusia baik dengan melakukan manipulasi dan juga penganalisan terhadap gambar[3].

2 METODE PENELITIAN

Metodologi penelitian yang dilakukan dalam penyelesaian penelitian ini meliputi:

1. Studi Literatur
Pada tahap ini dilakukan dengan cara mencari informasi serta menggali pengetahuan yang berhubungan dengan penelitian ini, baik melalui buku, jurnal, internet, dan juga dari sumber lainnya yang mendukung perancangan sistem berdasarkan landasan teori yang ada.
2. Pengumpulan Data
Pada tahap ini dilakukan dengan cara mengumpulkan semua data, yaitu berupa foto-foto yang akan dicluster dan dirubah warna *background*-nya.
3. Desain atau Pemodelan Sistem

Pada tahap ini akan dibuat perancangan aplikasi untuk mengedit sebuah *background* pada foto, dan merubah warna *background*. Warna *background* yang dirubah adalah merah, biru, dan hitam putih dengan memanfaatkan metode K-Mean clustering untuk mengisolasi bagian *background* yang terdeteksi.

4. Pengujian

Pada tahap ini dilakukan pengujian terhadap sistem yang telah dibangun untuk melihat keluaran dari setiap tahapan yang dilakukan pada sistem.

K-Means merupakan salah satu metode data *clustering* non hirarki yang berusaha mempartisi data ke dalam *cluster* atau kelompok sehingga data yang memiliki karakteristik yang sama dikelompokkan dalam satu *cluster* yang sama dan data yang mempunyai karakteristik berbeda dikelompokkan ke dalam kelompok yang lain.

Tujuan dari *clustering* adalah untuk meminimalisasi *objective function* yang diset dalam proses *clustering*, yang berusaha meminimalisasi variasi di dalam suatu *cluster* dan memaksimalkan variasi antar *cluster*.

Langkah-langkah Algoritma *K-Means* adalah sebagai berikut :

- 1) Pilih jumlah *cluster* k .
- 2) Inisialisasi k pusat *cluster* secara random
- 3) Alokasikan semua data / objek ke *cluster* terdekat. Kedekatan dua objek ditentukan berdasarkan jarak kedua objek tersebut. Demikian juga kedekatan suatu data ke *cluster* tertentu ditentukan jarak antara data dengan pusat *cluster*, sehingga diperlukan perhitungan jarak antara data dengan pusat *cluster*. Jarak tersebut digunakan untuk menentukan suatu data tersebut masuk dalam *cluster* yang mana. Untuk menghitung jarak semua data ke setiap titik pusat *cluster* dapat menggunakan rumus sebagai berikut.

$$D(i, j) = \sqrt{(x_{li} - x_{lj})^2 + (x_{ki} - x_{kj})^2} \quad (1)$$

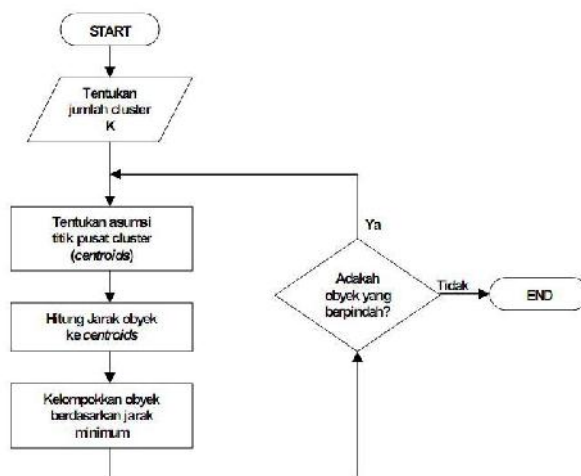
Dimana:

$D(i, j)$ = Jarak data ke i ke pusat *cluster* j

X_{ki} = Data ke i pada atribut data ke k (koordinat objek)

X_{kj} = Titik pusat ke j pada atribut ke k (koordinat centroid)

1. Hitung kembali pusat *cluster* dengan keanggotaan *cluster* yang sekarang. Pusat *cluster* adalah rata-rata dari semua data / objek dalam *cluster* tertentu. Ulangi lagi setiap objek memakai pusat *cluster* yang baru. Jika pusat *cluster* tidak berubah lagi maka proses *clustering* selesai. Atau, kembali ke langkah 3 sampai pusat *cluster* tidak berubah lagi.



Gambar 2. Algoritma k-means clustering

Berikut ini adalah langkah-langkah contoh dari perhitungan metode k-means

1. Masukkan nilai-nilai RGB dari *pixel-pixel* yang ada pada gambar. Misalkan citra berukuran 4×3 .

Tabel 1. Nilai Pixel Warna

Pixel	R	G	B
1	44	39	43
2	135	124	128
3	183	174	177
4	188	182	184
5	191	189	190
6	195	193	196
7	196	194	197
8	196	194	197
9	194	193	198
10	191	192	196
11	160	156	170
12	160	155	162

2. Kelompokkan nilai RGB menggunakan Algoritma *K-means*. Pada langkah ini dilakukan hal-hal berikut:

- a. Inisialisasi jumlah *cluster* yang ingin dibentuk sebanyak 3, yang merepresentasikan citra. Bayangan dari citra dan latar belakang (*background*). Tujuan dari penentuan jumlah *cluster* adalah untuk menentukan jumlah inisial pusat *cluster* yang akan dibangkitkan.
- b. Random nilai RGB dari 3 *centroid* (titik pusat *cluster*) awal berdasarkan jangkauan (*range*) nilai minimum dan maksimum RGB dari *pixel-pixel* yang ada pada gambar.

R = 44 s/d 196

G = 39 s/d 194

B = 43 s/d 198

Misalkan diperoleh nilai *centroid* / pusat *cluster* dari masing-masing *cluster* seperti pada tabel 2.

Tabel 2. *Centroid* / Pusat *Cluster*

cluster	R	G	B
1	44	39	43
2	195	193	196
3	160	155	162

- c. Hitung jarak nilai RGB setiap pixel terhadap masing-masing *centroid* / pusat *cluster* tersebut dengan menggunakan rumus 1 yaitu Rumus *Euclidean distance*. Berikut ini hasil dari perhitungan jarak nilai RGB pada pixel.

Jarak RGB pixel ke-1 dengan pusat *cluster* adalah:

$$C(1,1) = \sqrt{(44-44)^2 + (39-39)^2 + (43-43)^2} = 0 \quad C(1,2) = \sqrt{(44-195)^2 + (39-193)^2 + (43-196)^2} = 264.4352$$

$$C(1,3) = \sqrt{(44-160)^2 + (39-155)^2 + (43-162)^2} = 202.6647$$

Jarak RGB pixel ke-6 dengan pusat *cluster* adalah:

$$C(6,1) = \sqrt{(195-44)^2 + (193-39)^2 + (196-43)^2} = 264.4352 \quad C(6,2) = \sqrt{(195-195)^2 + (193-193)^2 + (196-196)^2} = 0$$

$$C(6,3) = \sqrt{(195-160)^2 + (193-155)^2 + (196-162)^2} = 61.84658$$

Jarak RGB pixel ke-12 dengan pusat *cluster* adalah:

$$C(12,1) = \sqrt{(160-44)^2 + (155-39)^2 + (162-43)^2} = 202.6647 \quad C(12,2) = \sqrt{(160-195)^2 + (155-193)^2 + (162-196)^2} = 61.84658$$

$$C(12,3) = \sqrt{(160-160)^2 + (155-155)^2 + (162-162)^2} = 0$$

Tabel 3. Hasil Nilai Jarak Pada Pixel

Data	C1	C2	C3
1	0	264.4352	202.6647
2	150.768	113.9517	52.36411
3	235.5886	29.42788	33.39162
4	247.1154	17.72005	44.68781
5	256.3552	8.246211	53.86093
6	264.4352	0	61.84658
7	266.1672	1.732051	63.57673
8	266.1672	1.732051	63.57673
9	265.0302	2.236068	62.41795
10	261.5856	4.123106	59.04236
11	208.024	57.18391	8.062258
12	202.6647	61.84658	0

- d. *Object clustering* dilakukan berdasarkan hasil perhitungan di atas, dimana setiap *pixel* menjadi anggota suatu *cluster* yang *Euclidean distance* dari nilai RGB *pixel* tersebut terhadap centroid-nya minimum.

Tabel 4. RGB Centroid Cluster

Pixel	R	G	B	Cluster
1	44	39	43	I
2	135	124	128	III
3	183	174	177	II
4	188	182	184	II
5	191	189	190	II
6	195	193	196	II
7	196	194	197	II
8	196	194	197	II
9	194	193	198	II
10	191	192	196	II
11	160	156	170	III
12	160	155	162	III

- e. Hitung posisi pusat *cluster* baru berdasarkan rata-rata anggota *cluster*. Untuk *cluster* pertama, ada 1 data yaitu data pertama, sehingga :

$$R = 44, G = 39, B = 43$$

Untuk *cluster* kedua, ada 8 data yaitu data ke 3, 4, 5, 6, 7, 8, 9, dan data ke-10, sehingga:

$$R = \frac{183 + 188 + 191 + 195 + 196 + 196 + 194 + 191}{8} = 192 \quad G = \frac{174 + 182 + 189 + 193 + 194 + 194 + 193 + 192}{8} = 189$$

$$B = \frac{177 + 184 + 190 + 196 + 197 + 197 + 198 + 196}{8} = 192$$

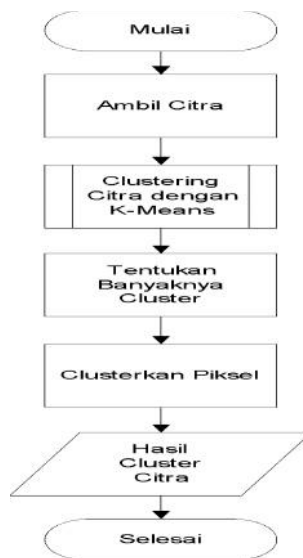
Untuk *cluster* ketiga, ada 3 data yaitu data ke 2, 11 dan data ke 12, sehingga :

$$R = \frac{135 + 160 + 160}{3} = 152 \quad G = \frac{124 + 156 + 155}{3} = 145 \quad B = \frac{128 + 170 + 162}{3} = 154$$

Tabel 5. Centroid / Pusat Cluster

Cluster	R	G	B
1	44	39	43
2	192	189	192
3	152	145	154

- f. Periksa apakah pixel-pixel mengalami perpindahan *cluster*. Jika masih ada, maka lanjutkan ke langkah c. Jika tidak, maka berarti komputasi telah mencapai stabilitas.



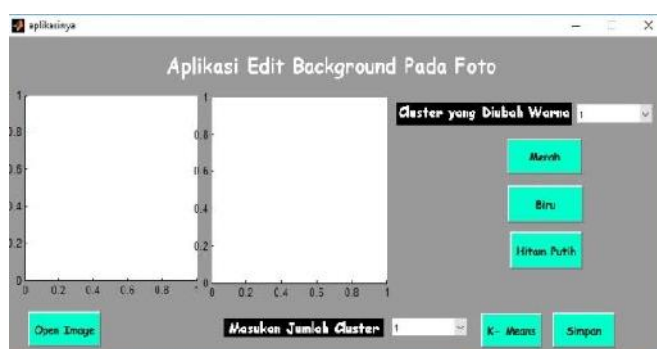
Gambar 3. Flowchart sistem

Pada Gambar 3, di atas merupakan dari flowchart sistem yang berjalan, yang langkah pertamanya adalah mengambil suatu citra dari file, langkah kedua mengclusterkan citra (membagi citra) dengan metode k-means, langkah ketiga menentukan banyaknya jumlah cluster yang diinginkan, selanjutnya citra akan di cluster dan langkah terakhir adalah tampilan citra yang sudah di cluster akan terlihat diakses pada matlab.

3. HASIL DAN PEMBAHASAN

Pada bagian ini, diuraikan hasil perancangan aplikasi dan pembahasan dari metode yang telah diuraikan di atas. Berikut ini hasil dari perancangan aplikasi edit foto pada *background* menggunakan metode k-means adalah sebagai berikut :

1) Tampilan Awal Aplikasi



Gambar 4. Tampilan awal aplikasi

Seperti pada Gambar 4, terdapat 2 popup menu yaitu memilih banyaknya jumlah cluster dan pengklasteran citra yang diubah. Terdapat 6 pushbutton yaitu button open image digunakan untuk mencari file citra yang ingin diclusterkan, button k-means digunakan untuk pengklasteran citra, button merah digunakan untuk mengubah warna *background* menjadi warna merah, button biru digunakan

untuk mengubah warna *background* menjadi warna biru, button hitam putih digunakan untuk merubah foto menjadi hitam putih, button simpan digunakan untuk menyimpan hasil dari cluster.

2) Tampilan Pengambilan Citra



Gambar 5. Tampilan pengambilan citra

Pada Gambar 5, sebelum menjalankan aplikasi terlebih dahulu dipilih citra yang diinginkan untuk diclusterkan dan merubah warna *background*.

3) Tampilan Pengklasteran Citra K-Means



Gambar 6. Hasil cluster citra untuk background warna merah



Gambar 7. Hasil cluster citra dengan background berbeda warna

Pada gambar 6, 7 merupakan tampilan dari pengklasteran citra, yang dimana fungsinya untuk membagi citra menjadi beberapa bagian, dari hasil cluster tersebut dapat diketahui dibagian mana atau dijumlah cluster mana yang *background* dapat dirubah.

4) Perubahan Warna *Background*



Gambar 8. Perubahan warna merah *background*

Pada Gambar 8, adalah hasil dari perubahan warna merah pada *background* tersebut yang pengclusteran citranya terdapat pada Gambar 6.



Gambar 9. Perubahan warna biru *background*

Pada Gambar 9, adalah hasil dari perubahan warna biru pada *background* tersebut.



Gambar 10. Perubahan warna *background* dengan *background* berbeda warna

Pada Gambar 10 adalah tampilan hasil dari perubahan warna pada *background* yang dimana foto tersebut memiliki *background* yang berbeda warna.

5) Tampilan Perubahan Foto Hitam Putih



Gambar 11. Perubahan foto hitam putih

Gambar 11, adalah tampilan dari perubahan foto menjadi hitam putih.

Pengujian dilakukan untuk mengetahui mampu atau tidak aplikasi ini dalam mengcluster citra, merubah warna *background*, merubah warna pada foto, dan berhasilkah aplikasi edit foto pada *background* dengan menggunakan metode k-means saat dijalankan. Untuk menguji sistem diperlukan foto dengan *background* satu warna dan foto dengan *background* berbeda warna. Berikut pada Tabel 6, merupakan hasil pengujian dari pengklasteran citra dan perubahan warna *background*.

Tabel 6. Hasil Perubahan Warna *Background*

NO	FOTO	JUMLAH CLUSTER	B. MERAH	B. BIRU	HITAM PUTIH
1		3	Berhasil	Berhasil	Berhasil
		4	Berhasil	Berhasil	
		5	Berhasil	Berhasil	
		6	Berhasil	Berhasil	
2		3	Tidak Berhasil	Tidak Berhasil	Berhasil
		4	Tidak Berhasil	Tidak Berhasil	
		5	Tidak Berhasil	Tidak Berhasil	
		6	Tidak Berhasil	Tidak Berhasil	
3		3	Tidak Berhasil	Tidak Berhasil	Berhasil
		4	Tidak Berhasil	Tidak Berhasil	
		5	Tidak Berhasil	Tidak Berhasil	
		6	Tidak Berhasil	Tidak Berhasil	
4		3	Tidak Berhasil	Tidak Berhasil	Berhasil
		4	Tidak Berhasil	Tidak Berhasil	
		5	Tidak Berhasil	Tidak Berhasil	
		6	Tidak Berhasil	Tidak Berhasil	
5		3	Berhasil	Berhasil	Berhasil
		4	Berhasil	Berhasil	
		5	Berhasil	Berhasil	
		6	Berhasil	Berhasil	
6		3	Tidak Berhasil	Tidak Berhasil	Berhasil
		4	Tidak Berhasil	Tidak Berhasil	
		5	Tidak Berhasil	Tidak Berhasil	
		6	Tidak Berhasil	Tidak Berhasil	
7		3	Tidak Berhasil	Tidak Berhasil	Berhasil
		4	Tidak Berhasil	Tidak Berhasil	
		5	Tidak Berhasil	Tidak Berhasil	
		6	Tidak Berhasil	Tidak Berhasil	
8		3	Berhasil	Berhasil	Berhasil
		4	Berhasil	Berhasil	
		5	Berhasil	Berhasil	
		6	Berhasil	Berhasil	
9		3	Berhasil	Berhasil	Berhasil
		4	Tidak Berhasil	Tidak Berhasil	
		5	Tidak Berhasil	Tidak Berhasil	
		6	Tidak Berhasil	Tidak Berhasil	

Dari pelaksanaan pengujian sistem, didapatkan hasil sebagai berikut :

- 1) Perubahan warna pada *background* dapat berjalan apabila citra yang memiliki *background* yang sama.
- 2) Pengujian sistem dilakukan dengan menggunakan 9 citra.
- 3) Semakin banyak jumlah cluster yang digunakan, maka semakin bagus hasil pengklasteran yang didapat. Semakin sedikit jumlah *cluster* yang digunakan, maka hasil pengklasteran akan semakin sulit didapat.
- 4) Tingkat keberhasilan dalam merubah warna *background* yaitu sebesar 100% jika citra tersebut memiliki warna *background* yang sama.

4. PENUTUP

Pada bagian ini diuraikan kesimpulan dan saran dari aplikasi edit foto pada *background* dengan menggunakan metode k-means.

4.1 Kesimpulan

Setelah melakukan studi literatur, analisis perancangan, dan pengujian terhadap aplikasi aplikasi edit foto pada *background* dengan menggunakan metode k-means, maka didapat kesimpulan antara lain sebagai berikut:

1. Dari hasil implementasi sistem bahwa metode *k-means* mampu mengklaster citra dan memisahkan objek dan *background* dengan baik.
2. Perubahan warna *background* dikatakan berhasil apabila *background* yang mempunyai satu warna, dan sebaliknya apabila *background* yang mempunyai berbeda warna dikatakan tidak dapat berubah.

3. Keberhasilan perubahan warna pada *background* dengan citra yang memiliki *background* yang sama dengan menggunakan metode k-means yaitu 100%.
4. Aplikasi ini dapat digunakan dengan baik dalam mengganti *background* dan pengguna lebih mudah mengganti *background* foto.

4.2 Saran

Adapun saran yang ingin disampaikan peneliti adalah sebagai berikut:

1. Untuk penelitian selanjutnya, Pengklasteran citra dapat dilakukan tanpa menentukan jumlah *cluster* terlebih dahulu.
2. Untuk penelitian selanjutnya, perubahan warna *background* dapat dilakukan dengan menggunakan *background* yang mempunyai berbeda warna.

BAHAN REFERENSI

- [1] N. A. Fitriani, *MATH unesa*, vol. 2, no. 6, pp. 2–6, 2017.
- [2] S. Rahman and M. ulfayani T, *Perancangan Aplikasi Identifikasi Biometrika Telapak Tangan Menggunakan Metode Freeman Chain Code*, *CESS (Journal Comput. Eng. Syst. Sci.*, vol. Vol. 2, no. No.2, pp. 64–73, 2017.
- [3] T. T. Pamungkas, R. R. Isnanto, and A. A. Zahra, *Pengenalan Plat Nomor Kendaraan Menggunakan Metode Template Matching Dan Jarak Canberra*, *Transient*, vol. 3, no. 2, pp. 166–173, 2014.
- [4] A. P. A.M and Murinto, *Segmentasi Citra Batik Berdasarkan Fitur Tekstur Menggunakan Metode Metode Filter Gabor dan K-Means Clustering*, *J. Inform.*, vol. 10, no. 1, pp. 1173–1179, 2016.
- [5] Atina, *Segmentasi Citra Paru Menggunakan Metode k-Means Clustering*, 2017.
- [6] A. Solichin and A. Harjoko, *Metode Background Subtraction untuk Deteksi Obyek Pejalan Kaki pada Lingkungan Statis*, *Jur. Ilmu Komput. dan Elektron. Fak. MIPA, Univ. Gajah Mada, Yogyakarta*, pp. 1–6, 2013.
- [7] D. Apriliyani and Murinto, *Analisis Perbandingan Teknik Segmentasi Citra Digital Menggunakan Metode Level Set*, *J. Inform.*, vol. 7, no. 2, pp. 802–810, 2013.
- [8] D. Putra, *Pengolahan Citra*. 2010.
- [9] Suma'inna and D. Alam, *Kompresi Citra Berwarna Menggunakan*, *J. Mat. Integr.*, vol. 10, no. 1, pp. 55–62, 2014.
- [10] Andri, Paulus, N. P. Wong, and T. Gunawan, *SEGMENTASI BUAH MENGGUNAKAN METODE K-MEANS*, vol. 15, no. 2, pp. 91–100, 2014.
- [11] S. Q. Nugroho and R. A. Pramunendar, *Pengelompokan Kayu Kelapa Menggunakan Algoritma K-Means Berdasarkan Tekstur Citra Kayu Kelapa Dua Dimensi (2D)*, pp. 1–5, 2015.
- [12] N. R. Apriyanti, R. A. Nugroho, and O. Soesanto, *ALGORITMA K-MEANS CLUSTERING DALAM*, vol. 2, no. 2, pp. 1–13, 2015.
- [13] Nasution, M. I. P., Andriana, S. D., Syafitri, P. D., Rahayu, E., & Lubis, M. R. (2016). Mobile device interfaces illiterate. In *Proceedings of the 2015 International Conference on Technology, Informatics, Management, Engineering and Environment, TIME-E 2015*. <http://doi.org/10.1109/TIME-E.2015.7389758>