

IMPLEMENTASI KOMPRESI CITRA DIGITAL MENGGUNAKAN KUANTISASI DAN K-MEANS CLUSTERING PADA FITUR BLUCARE DI APLIKASI BLUCAMPUS UNIVERSITAS BUDI LUHUR BERBASIS ANDROID

Kaishananda Dwi Takanwiedi¹⁾, Painem²⁾

Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur

Jl. Raya Ciledug, Petukangan Utara, Kebayoran Lama, Jakarta Selatan 12260

E-mail : ktakanwiedi@gmail.com¹⁾, painem@budiluhur.ac.id²⁾

Abstrak

Pada era serba digital sekarang ini pertukaran informasi menjadi hal yang sangat penting dan menjadi kebutuhan umum masyarakat. Salah satu contoh informasi yang dipertukarkan dapat berupa citra digital (gambar). Pertukaran informasi dapat dilakukan melalui perangkat yang terkoneksi dengan internet yang melibatkan pihak ketiga sebagai perantara (server). Citra digital memiliki berbagai macam ukuran. Untuk pertukaran informasi yang berupa citra digital yang memiliki ukuran yang besar, dibutuhkan ruang penyimpanan besar untuk menyimpan citra digital dan membutuhkan waktu pengiriman yang lebih lama ke server. Untuk mengatasi masalah ini diperlukan suatu metode yang dapat mengatasi masalah ukuran citra digital yang besar. Salah satu solusi untuk mengatasi masalah tersebut adalah dengan melakukan proses pemampatan (kompresi) pada citra digital. Kompresi adalah proses pemampatan atau pengurangan suatu ukuran data dari ukuran yang besar ke ukuran yang lebih kecil. Untuk melakukan kompresi dibutuhkan algoritma kompresi. Algoritma kompresi tersebut diharapkan memiliki memiliki rasio kompresi yang tinggi dan juga kecepatan di dalam proses pemampatannya. Di dalam proses pertukaran informasi, sebelum citra digital tersebut dikirimkan ke server, akan dilakukan proses pemampatan (kompresi) terhadap citra digital, sehingga citra digital tersebut memiliki ukuran yang lebih kecil. Citra digital yang sudah dikompres kemudian dikirim ke sever. Dari server, citra digital tersebut akan dikirim ke penerima yang dituju. Algoritma kompresi dapat membuat proses pemampatan citra berjalan dengan cepat dan memiliki rasio kompresi yang tinggi. Di dalam penelitian ini penulis menggabungkan dua algoritma kompresi untuk mengkompresi citra digital, yaitu algoritma Kuantisasi dan algoritma K-means clustering yang diimplementasikan pada fitur BLuCare di aplikasi BluCampus Universitas Budi Luhur berbasis Android. Kedua algoritma tersebut akan digunakan secara bergantian dimana citra digital akan kompres dengan algoritma Kuantisasi. Selanjutnya citra digital hasil kuantisasi akan di kompres dengan algoritma K-means clustering. Algoritma tersebut memiliki rata-rata waktu untuk pemampatan (kompresi) citra sebesar 43,732 detik, dan memiliki rata-rata rasio pemampatan (kompresi) citra sebesar 95,65%.

Kata kunci: Kompresi, Citra Digital, Kuantisasi, K-Means, Android.

1. PENDAHULUAN

Pengiriman informasi tidak hanya terbatas pada teks saja, namun dapat berupa audio dan visual. Citra digital merupakan salah satu informasi yang berperan penting dalam penyajian data visual. Perkembangan teknologi terutama *smartphone* Android juga mempengaruhi citra digital yang dihasilkan, semakin bagus kamera *smartphone* maka citra digital yang dihasilkan semakin tinggi resolusi kualitas dan ukuran citra digital, sehingga membutuhkan memori penyimpanan yang besar dan waktu pengiriman yang lebih lama. Solusi untuk menyelesaikan masalah tersebut dengan melakukan pemampatan (kompresi).

Untuk melakukan kompresi terhadap citra digital, perlu menggunakan algoritma kompresi yang memiliki rasio kompresi yang tinggi dan kecepatan dalam proses kompresi. Oleh karena itu dibuat fitur BLuCare pada aplikasi BluCampus Universitas Budi Luhur yang di dalamnya terdapat modul untuk melakukan proses kompresi terhadap citra digital. Di dalam fitur BLuCare digunakan algoritma kompresi Kuantisasi dan K-Means *Clustering*.

Dari uraian di atas apabila permasalahan tersebut dapat terselesaikan, maka tujuan dari penelitian ini yaitu mengimplementasikan algoritma Kuantisasi dan K-Means *Clustering* untuk melakukan kompresi terhadap citra digital dengan rasio kompresi yang tinggi dan kecepatan dalam proses kompresi dapat tercapai. Jadi, proses kinerja aplikasi *BluCampus* terutama dalam fitur *BLuCare* dapat bekerja secara optimal karena ukuran foto yang digunakan menjadi lebih kecil dibanding dengan ukuran aslinya. Sehingga memudahkan dalam proses penyimpanan dan pengiriman data ke *server*.

Dalam penelitian ini metode penelitian yang digunakan meliputi studi literatur, analisis data, perancangan sistem dan pengujian sistem.

2. METODE PENELITIAN

2.1. Metodologi Penelitian

Dalam penelitian ini, beberapa metode digunakan untuk memperoleh informasi yang diperlukan dan menyelesaikan masalah yang

ditemui. Adapun metode – metode ini sebagai berikut :

a. Studi literatur

Metode ini digunakan untuk memperoleh pembelajaran data atau informasi dengan cara mengumpulkan berbagai refensi baik itu dalam bentuk makalah, jurnal, serta referensi lainnya untuk mendapatkan informasi yang dibutuhkan.

b. Analisis Data

Metode ini digunakan untuk menganalisis Algoritma kompresi yang digunakan yaitu algoritma Kuantisasi dan K-Means *Clustering*, serta teknik-teknik yang digunakan.

c. Perancangan Sistem

Metode ini digunakan untuk merancang sistem aplikasi untuk mengimplementasikan metode algoritma Kuantisasi dan K-Means *Clustering* dengan menggunakan bahasa pemrograman JAVA berbasis Android.

d. Pengujian Sistem

Metode ini dilakukan dengan menguji dan mengecek jalannya program.

Ukuran citra digital atau dalam hal ini foto, yang sangat besar, menjadi masalah dalam berbagai macam proses pengiriman atau pertukaran informasi berupa foto. Hal itu karena dibutuhkan waktu yang cukup lama untuk mengirim foto tersebut. Permasalahan tersebut juga terjadi di dalam fitur BluCare aplikasi BluCampus yang terdapat modul untuk mengirim laporan beserta foto tentang laporan tersebut. Jika ukuran foto yang digunakan besar, maka akan memperlambat proses laporan tersebut.

Dari permasalahan yang ada maka diperlukan proses kompresi foto di dalam fitur BluCare aplikasi BluCampus, yang mampu mengkompresi ukuran foto yang digunakan dengan rasio kompresi yang tinggi dan waktu kompresi yang cepat. Sehingga, masalah ukuran foto yang besar dapat diatasi.

Aplikasi kompresi foto ini menggunakan algoritma Kuantisasi dan K-Means *Clustering*. Algoritma Kuantisasi dan K-Means *Clustering* termasuk ke dalam *lossy compression*. Dengan menggunakan kedua algoritma tersebut, dapat mengurangi ukuran foto dari ukuran semula. Sehingga dapat melancarkan atau mempercepat proses pembuatan laporan di dalam fitur BluCare aplikasi BluCampus.

Proses pemampatan bertujuan untuk mengurangi duplikasi data sehingga memori yang digunakan menjadi lebih sedikit. *Lossless Compression* dan *Lossy Compression* merupakan dua penggolongan teknik dalam kompresi [1]. Beberapa faktor menjadikan kompresi terhadap citra menjadi tepat agar antara *pixel* dengan *pixel* tetangganya tidak terdapat korelasi (*Redudancy*). Untuk menghitung tingkat keberhasilan dalam proses pemampatan (kompresi) dapat dituliskan sebagai berikut :

$$\text{Compression ratio} = \left(1 - \frac{\text{compressed size}}{\text{uncompressed size}} \right) \times 100\%$$

Dalam matematika proses pemetaan satu set besar nilai input ke (dihitung) set yang lebih kecil disebut kuantisasi. Contoh khas proses kuantisasi adalah pembulatan dan bisa juga pemotongan, menjadikan kuantisasi sebagai inti dari algoritma kompresi *lossy*. Kuantisasi merupakan inheren non-linear dan proses ireversibel (karena nilai output yang sama dibagi oleh beberapa nilai masukan, tidak memungkinkan pada umumnya untuk memulihkan nilai masukan yang tepat ketika diberikan hanya nilai output) karena pemetaannya yang *many-to-few* (banyak-ke-beberapa).

Kuantisasi citra merupakan sebuah teknik pengelompokan nilai tingkat keabuan citra ke dalam beberapa level. Semakin kecil skala yang digunakan maka akan semakin kecil pula resolusi citra yang bisa menyebabkan citra terlihat buram, patah-patah atau tidak jelas. Warna dari sebuah citra diperoleh dari besar kecilnya intensitas cahaya yang ditangkap oleh sensor. Sampai saat ini belum ada satu sensor pun yang mampu menangkap seluruh gradasi warna yang terdapat dari alam.

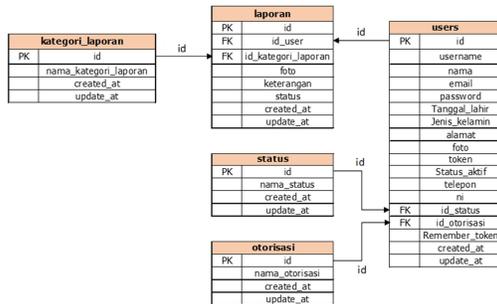
K-Means didasarkan pada penentuan jumlah awal kelompok dengan mengidentifikasi nilai *centroid* awalnya, karena itulah k-means merupakan algoritma dengan pembagian partisi [2]. Penggunaan proses secara berulang-ulang untuk mendapatkan basis data *cluster* membutuhkan jumlah *cluster* awal yang diinginkan sebagai masukan dan menghasilkan titik *centroid* akhir sebagai output. K-Means memilih pola k sebagai titik awal *centroid* secara acak atau random. Jumlah iterasi untuk mencapai *cluster centroid* akan dipengaruhi oleh calon *cluster centroid* awal secara random. Sehingga didapat cara dalam pengembangan algoritma dengan menentukan *centroid cluster* yang dilihat dari kepadatan data awal yang tinggi agar mendapatkan kinerja yang lebih tinggi. Titik - titik *centroid* akan dihasilkan yang kemudian dijadikan tujuan dari algoritma K-Means. Setelah iterasi terhenti, semua obyek yang berada di dalam suatu dataset menjadi anggota dari suatu *cluster* atau dataset tersebut. Untuk menentukan nilai *cluster* dibutuhkan proses pencarian terhadap seluruh obyek untuk menemukan *cluster* dengan jarak terdekat ke obyek yang kemudian seluruh item terdekat akan dikelompokkan ke *cluster* terdekat berdasarkan jaraknya, itulah inti Algoritma K-Means. Rumus Euclidean Distance digunakan untuk menghitung jaraknya berdasarkan titik pusat awal. Data-data yang memiliki jarak pendek atau dekat terhadap *centroid* atau titik pusat akan membuat sebuah *cluster*, begitu terus menerus hingga tidak ada lagi perubahan yang terjadi.

(1)

Dengan menggabungkan algoritma kuantisasi dan k-means clustering diharapkan aplikasi dapat bekerja optimal sesuai fungsinya karena ukuran citra akan menjadi lebih kecil dari ukuran aslinya dengan rasio kompresi yang tinggi dan kecepatan dalam proses kompresi.

2.2. LRS (Logical Record Structure)

Berikut ini adalah representasi dari tabel-tabel yang digunakan di dalam database pada fitur BluCare aplikasi BluCampus ini :



Gambar 1. LRS Aplikasi

2.3. Spesifikasi Basis Data

Rincian dari tabel-tabel yang digunakan akan dijelaskan pada tabel berikut ini :

Tabel 1. Tabel Kategori Laporan

Field	Type	Length (byte)	Keterangan
id	Int	10	Id kategori
nama_kategori_laporan	Varchar	100	Nama kategori
created_at	Date Time	8 byte	Tanggal dibuat
update_at	Date Time	8 byte	Tanggal diubah

Tabel 2. Tabel Laporan

Field	Type	Length (byte)	Keterangan
id	Int	10	Id laporan
id_user	Int	10	Id user
id_kategori_laporan	Int	10	Id kategori
foto	Varchar	191	Url untuk melihat foto
keterangan	text	200	Keterangan laporan
status	Int	4	Status laporan
created_at	Date Time	8 byte	Tanggal dibuat
update_at	Date Time	8 byte	Tanggal diubah

Tabel 3. Tabel Otorisasi

Field	Type	Length (byte)	Keterangan
id	Int	10	Id otorisasi
nama_otorisasi	Varchar	100	Nama otorisasi
created_at	Date Time	8 byte	Tanggal dibuat
update_at	Date Time	8 byte	Tanggal diubah

Tabel 4. Tabel Status

Field	Type	Length (byte)	Keterangan
id	Int	10	Id status
nama_status	Varchar	100	Nama status
created_at	Date Time	8 byte	Tanggal dibuat
update_at	Date Time	8 byte	Tanggal diubah

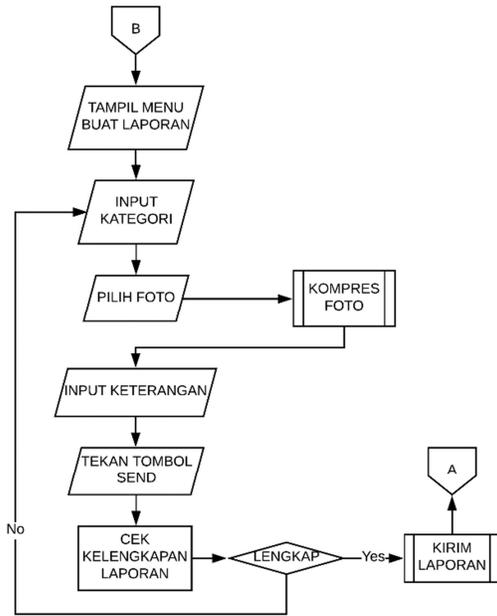
Tabel 5. Tabel Users

Field	Type	Length (byte)	Keterangan
id	Int	10	Id user
username	Varchar	35	Username dari user
nama	Varchar	100	Nama user
email	Varchar	100	Email user
password	Varchar	100	Password user
Tanggal_lahir	Date Time	8 byte	Tanggal lahir user
Jenis_kelamin	Varchar	1	Kelamin user
alamat	Varchar	100	Alamat user
foto	Varchar	191	Url untuk melihat foto
token	Varchar	191	Token akun dari user
Status_aktif	int	4	Status akun user
telepon	Varchar	15	Nomor telepon user
ni	Varchar	14	Nomor induk user
id_status	int	10	Id untuk status user
id_otorisasi	int	10	Id untuk otorisasi user
Remember_token	Varchar	100	Token pengingat akun user
created_at	Date Time	8 byte	Tanggal dibuat

update_at	Date Time	8 byte	Tanggal diubah
-----------	-----------	--------	----------------

2.4. Flowchart dan algoritma Program

Berikut ini merupakan rancangan *flowchart* dan algoritma pemrograman pada form Buat Laporan serta *flowchart* dan algoritma untuk proses kompresi foto menggunakan Kuantisasi dan K-Means *Clustering*, berikut *flowchart* dan algoritmanya :



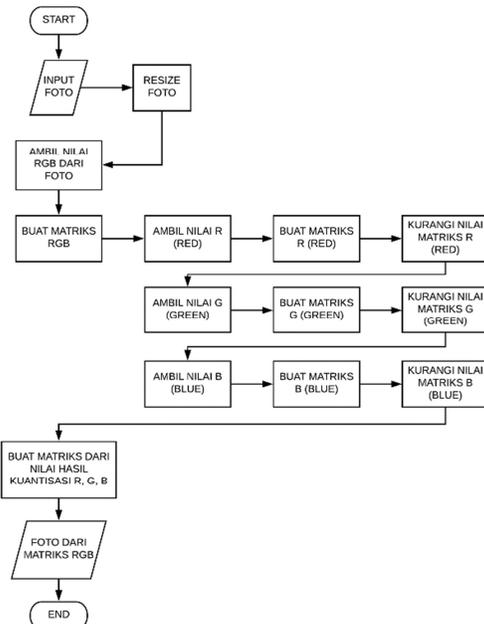
Gambar 2. Flowchart Form Buat Laporan

Flowchart pada gambar 1 menjelaskan alur pembuatan laporan dimana *user* mengisi semua *form* yang tersedia sesuai kejadian yang terjadi, kemudian menyertakan foto kejadian masalah jika memungkinkan. Lalu mengirim laporan tersebut. Proses kompresi foto akan berlangsung secara otomatis setelah *user* memilih foto yang akan digunakan untuk laporan. Berikut ini adalah algoritma dari *flowchart* di atas :

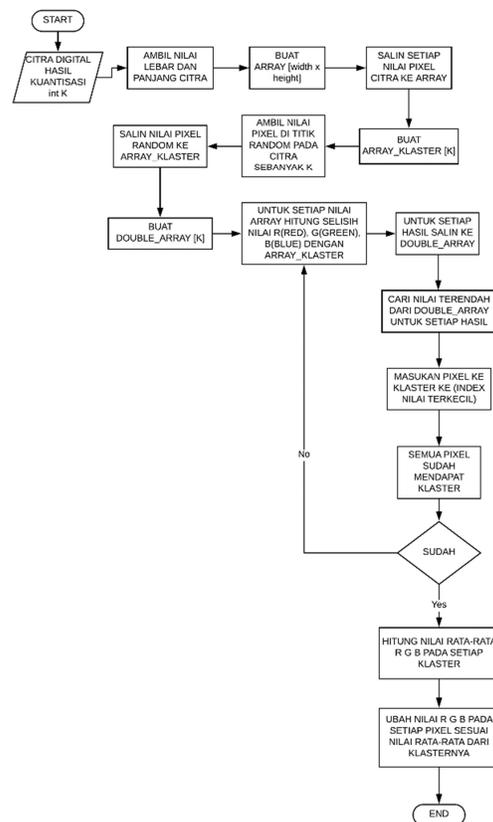
- | | |
|-----|------------------------------|
| 1. | Start |
| 2. | Tampil Menu Buat Laporan |
| 3. | Input Kategori |
| 4. | Pilih foto |
| 5. | Kompres Foto |
| 6. | Input Keterangan |
| 7. | Tekan tombol SEND |
| 8. | If data laporan lengkap then |
| 9. | Kirim laporan |
| 10. | Else |
| 11. | Lengkapi data laporan |
| 12. | End If |

Proses kompresi menggunakan dua algoritma secara bergantian, dimana pertama foto akan dikompresi dengan algoritma Kuantisasi, kemudian

hasilnya akan dikompresi dengan algoritma K-Means *Clustering*. *Flowchart* algoritma Kuantisasi akan ditunjukkan oleh gambar 2 dan *flowchart* K-Means *Clustering* akan ditunjukkan oleh gambar 3 :



Gambar 3. Flowchart Kuantisasi



Gambar 4. Flowchart K-Means Clustering

Algoritma di bawah ini akan menjelaskan proses Kuantisasi :

1. Start
2. *Input* foto
3. *Resize* foto
4. Simpan nilai RGB
5. Salin nilai RGB ke matriks RGB
6. Simpan nilai R (red) dari masing-masing nilai matriks RGB ke matriks R
7. Simpan nilai G (green) dari masing-masing nilai matriks RGB ke matriks G
8. Simpan nilai B (blue) dari masing-masing nilai matriks RGB ke matriks B
9. Kurangi masing-masing nilai matriks R ke nilai yang ditentukan
10. Kurangi masing-masing nilai matriks G ke nilai yang ditentukan
11. Kurangi masing-masing nilai matriks B ke nilai yang ditentukan
12. Gabungkan matriks R, matriks G, matriks B menjadi matriks RGBbaru
13. Ubah nilai RGB pada foto sesuai nilai matriks RGBbaru
14. End

Berikut ini adalah algoritma yang menjelaskan proses K-Means *Clustering*.

1. Start
2. Input foto hasil kuantisasi dan int k
3. W=lebar foto
4. H=panjang foto
5. Salin setiap nilai pixel pada foto ke ArrayPixel[W*H]
6. Ambil nilai random pada ArrayPixel sebanyak k buah
7. Salin semua nilai random ke ArrayK[k]
8. Setiap nilai pada ArrayK menjadi satu kluster
9. Untuk masing-masing nilai ArrayPixel :
10. Bandingkan dengan semua nilai ArrayK
11. Hitung selisih nilai R(red), G(green), B(blue)
12. Salin hasil ke DoubleArray[k]
13. Cari nilai terendah DoubleArray
14. Masukkan nilai ArrayPixel ke dalam kluster
15. If semua pixel sudah mendapat kluster then
16. Continue
17. Else
18. Kembali ke langkah 9
19. End if
20. Hitung nilai rata-rata R(red), G(green), B(blue) pada setiap kluster
21. Ubah nilai R(red), G(green), B(blue) pada masing-masing nilai ArrayPixel sesuai nilai rata-rata klusternya
22. Ubah nilai pixel foto sesuai nilai

ArrayPixel23.
23. End

3. HASIL DAN PEMBAHASAN

3.1. Tampilan Layar



Gambar 5. Tampilan Layar Menu Buat Laporan

Gambar di atas merupakan tampilan layar yang menggambarkan menu Buat Laporan. Di dalam menu ini terdapat *form* yang harus *user* isi untuk melengkapi laporan, diantaranya *form* kategori dan *form* keterangan. Kemudian ada *form* foto yang dapat *user* isi jika memungkinkan dengan *klik* tombol AMBIL FOTO. Setelah foto dipilih oleh *user*, proses kompresi akan berlangsung secara otomatis. Tombol SEND berfungsi untuk mengirim laporan yang sudah dibuat. Tombol SEND hanya akan berfungsi jika *form* kategori dan *form* keterangan sudah diisi. Setelah laporan berhasil terkirim, semua data yang sudah diisi dihalaman ini akan terhapus secara otomatis, sehingga *user* dapat langsung menggunakan untuk membuat laporan kembali.

3.2. Uji Coba Program

Berikut ini adalah hasil dari pengujian terhadap proses kompresi foto, pengujian dilakukan dengan menggunakan 5 foto. Dari hasil pengujian ini dapat dilihat hasil dari rata-rata rasio kompresi dan kecepatan waktu proses kompresi. Berikut tabel hasilnya :

Tabel 6. Hasil Percobaan Kompresi

NO	FOTO ASLI	FOTO HASIL	WAKTU	RASIO
1			1 menit 1,686 detik	Ukuran asli : 1974 KB Ukuran hasil kompresi : 88 KB

				Rasio : 95%
2			55,910 detik	Ukuran asli : 1529 KB Ukuran hasil kompresi : 47 KB Rasio : 97%
3			33,062 detik	Ukuran asli : 1890 KB Ukuran hasil kompresi : 79 KB Rasio : 96%
4			41,446 detik	Ukuran asli : 2085 KB Ukuran hasil kompresi : 101 KB Rasio : 95%
5			24,566 detik	Ukuran asli : 1532 KB Ukuran hasil kompresi : 61 KB Rasio : 96%
RATA - RATA			43,732 detik	95.65%

Dari hasil pengujian, proses kompresi dengan menggabungkan algoritma Kuantisasi dan algoritma K-Means Clustering menghasilkan rata-rata waktu kompresi sebesar 43,732 detik dan rata-rata rasio kompresinya sebesar 95,65%.

3.3. Tanggapan Pengguna

Dari 4 kategori aspek penilaian yaitu functionality, reliability, usability, dan efficiency skor dari 22 responden dapat dilihat pada Tabel 2.

Tabel 7. Tabel Skor Responden

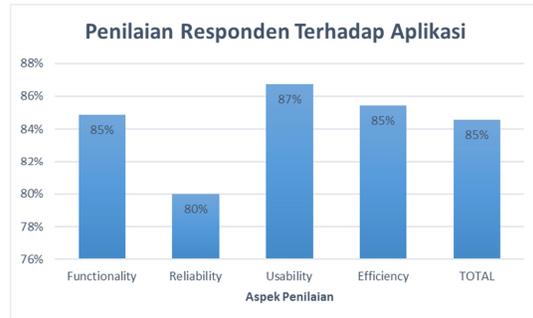
Aspek Penilaian	Skor Responden				
	5	4	3	2	1
	SS	S	R	TS	STS
Functionality	37	69	9	-	-
Reliability	17	35	17	-	-
Usability	35	53	4	-	-
Efficiency	34	49	9	-	-
Jumlah	123	206	39	-	-

Pada Tabel 3 hasil penilaian responden terhadap aplikasi dari 4 kategori aspek penilaian yaitu *functionality*, *reliability*, *usability*, dan *efficiency*. Total skor aktual untuk aspek

functionality sebesar 83% ,aspek *reliability* 81%, aspek *usability* 86%,dan aspek *efficiency* 88%, dan total skor aktual keseluruhan sebesar 84%. Gambar 5 menunjukkan diagram penilaian responden aplikasi dari 4 aspek penilaian dan total keseluruhan.

Tabel 8. Tabel Penilaian Responden

Aspek Penilaian	Skor Aktual					Total Skor Aktual	Skor Ideal	%
	5	4	3	2	1			
	SS	S	R	TS	STS			
Functionality	185	276	27	-	-	488	575	85%
Reliability	85	140	51	-	-	276	345	80%
Usability	175	212	12	-	-	399	460	87%
Efficiency	170	196	27	-	-	393	460	85%
TOTAL	615	824	117	-	-	1.556	1.840	85%



Gambar 6. Diagram Penilaian Responden

3.4. Kelebihan Program

- Aplikasi sesuai kebutuhan dan harapan pengguna dalam tujuan dari fitur *BluCare* itu sendiri, yaitu memudahkan dalam membuat laporan.
- Laporan yang dikirim dari menu Buat Laporan dengan laporan yang dilihat pada menu Lihat Laporan masih akurat atau sama.
- Foto hasil kompresi jika dibandingkan dengan foto asli tidak terlalu signifikan perubahannya.
- Aplikasi sangat mudah dimengerti oleh pengguna.
- Aplikasi memiliki tampilan yang mudah dipahami.
- Aplikasi Sangat mudah dioperasikan.

3.5. Kekurangan Program

- Ketika pengujian aplikasi, di beberapa *smarthphone* mengalami *crash* atau hang. Hal tersebut kemungkinan terjadi karena perbedaan spesifikasi dimana spesifikasi *smartphone* yang digunakan lebih rendah.
- Proses kompresi foto berjalan cukup lama. Hal ini mungkin terjadi karena foto yang digunakan memiliki ukuran yang sangat besar, sehingga proses kompresi berjalan cukup lama.
- Di dalam laporan yang dibuat oleh responden, banyak yang tidak mencantumkan lokasi tempat terjadinya masalah. Hal tersebut karena responden

tidak menemukan tempat untuk menginput lokasi masalah.

4. KESIMPULAN

- a) Dengan menggunakan algoritma Kuantisasi dan K-means *clustering* untuk mengkompresi foto, ukuran dari foto yang digunakan memiliki ukuran yang lebih kecil dibandingkan dengan ukuran aslinya.
- b) Dengan menggunakan algoritma Kuantisasi dan K-means *clustering* untuk mengkompresi foto, didapatkan hasil rata-rata rasio kompresi sebesar 95,65%.
- c) Dengan menggunakan algoritma Kuantisasi dan K-means *clustering* untuk mengkompresi foto, didapatkan hasil rata-rata waktu untuk proses kompresi sebesar 43,732 detik.

5. DAFTAR PUSTAKA

- [1] Sutoyo, T, dkk. (2009) 'Teori Pengolahan Citra Digital', Penerbit Andi, Yogyakarta hal 9 - 27.
- [2] Madhulatha, T., (2012) 'An Overview On Clustering Methods'.
- [3] Safaat, Nazrudin. (2011) 'Android Pemrograman Aplikasi Mobile Smartphone Dan Tablet PC Berbasis Android'. Informatika, Bandung.