

BAB 6

KESIMPULAN DAN SARAN

Pada bagian ini, diberi kesimpulan yang dapat ditarik dari penelitian ini. Selain itu juga ada saran untuk mengembangkan aplikasi pengenalan warna ini selanjutnya.

6.1 Kesimpulan

Kesimpulan dari penelitian ini dapat ditarik dari hasil pengujian terhadap perangkat lunak. Pengujian yang telah dilakukan terhadap perangkat lunak ada dua, yaitu pengujian fungsional yang dapat dilihat pada bagian 5.2.1 dan pengujian eksperimental yang dapat dilihat pada bagian 5.2.2. Dari kedua pengujian tersebut beberapa kesimpulan yang dapat diambil adalah:

1. Dari empat jenis *format file* citra, yang dapat diproses dengan baik hanya tiga yaitu JPG/JPEG, PNG, dan BMP. *Format file* GIF dapat dibaca oleh perangkat lunak akan tetapi warnanya akan selalu dinyatakan hitam. Selain itu pada *format file* PNG yang memiliki transparansi, sebelum diproses transparansi tersebut akan dianggap memiliki nilai RGB 0.
2. Citra yang memiliki ukuran *width* atau *height* lebih dari sama dengan 3000 piksel memiliki kemungkinan untuk membuat perangkat lunak *not responding* akibat terlalu lama untuk menampilkan hasil ke *user interface* atau bahkan menyebabkan *error* berupa *Java Out of Memory*.
3. Semakin besar ukuran citra, akan semakin lama prosesnya terutama yang menggunakan fungsi pengelompokan dengan Algoritma K-Means. Oleh karena itu untuk pelatihan disarankan menggunakan citra dengan ukuran kurang dari 1000 piksel.
4. Fitur mengelompokkan citra berdasarkan warna adalah cara memisahkan sekelompok warna. Fitur ini dapat diterapkan menggunakan Algoritma K-Means dimana banyak pengelompokan tergantung pada nilai *k*.
5. Banyak pengelompokan atau nilai *k* yang ideal berada diantara 2-20. Hal ini dikarenakan agar tidak membuat proses terlalu lama dan performa perangkat lunak terlalu berat.
6. Hasil pengelompokan citra berdasarkan warna bergantung pada jumlah *k* yang diberikan. Semakin tinggi nilai *k* diberikan, citra yang dikelompokkan akan memberi hasil semakin serupa dengan aslinya.
7. Fitur menentukan nama warna dapat diterapkan menggunakan Standar Deviasi untuk menghasilkan *range* batas bawah nilai warna sampai batas atas nilai warna. Untuk menghasilkan *range* nilai warna tergantung pada banyak data yang diproses.
8. Fitur mengenali nama warna tergantung pada nilai warna tersebut berada pada suatu *range* nilai-nilai warna. Jika tidak ada dalam *range* warna apapun maka akan diidentifikasi sebagai *unknown*.

9. Penentuan nama warna yang benar sangat tergantung pada manusia untuk mengajari perangkat lunak. Oleh sebab itu, perbedaan pendapat antara sesama manusia dapat mempengaruhi tingkat akurasi dari proses penentuan nama warna.
10. Fitur memeriksa keberadaan warna pada citra tidak menggunakan cara mengelompokkan citra berdasarkan warna, akan tetapi melakukan pemeriksaan per piksel. Hal ini dapat menyebabkan warna yang terdapat dalam satu piksel dan tidak kelihatan pada mata manusia, dinyatakan ada pada citra tersebut.
11. Fitur mencari citra-citra dari kumpulan citra dengan warna tertentu dinilai secara *precision* dan *recall*. Dimana *precision* menentukan tingkat akurasi dari perangkat lunak dalam mengembalikan jawaban dan *recall* menentukan ketepatan perangkat lunak dalam mengembalikan jawaban. Kedua nilai ini dihitung dalam persen.
12. Keakuratan dan ketepatan dari fitur mencari citra-citra dari kumpulan citra dengan warna tertentu juga tergantung pada banyak jenis data warna yang ada dan banyak data yang diproses. Semakin banyak jenis data warna yang ada, akurasi dalam mengembalikan hasil yang di dalamnya terdapat warna yang relevan akan semakin tinggi. Akan tetapi ketepatannya dalam mengembalikan hasil yang di dalamnya adalah semua yang relevan menurut mata manusia akan lebih rendah. Sebaliknya, semakin rendah banyak jenis warna nilai akurasi akan menurun akan tetapi nilai ketepatannya akan meningkat.

6.2 Saran

Berikut adalah beberapa saran untuk dapat mengembangkan skripsi ini lebih lanjut:

1. Pada skripsi ini resolusi citra dibatasi 2700 x 2700 karena keterbatasan *memory* oleh Java. Untuk penelitian lebih lanjut mungkin dapat dilakukan dengan resolusi lebih tinggi oleh karena citra yang ditangkap oleh teknologi kamera baik pada *smartphone* atau kamera-kamera lain pada saat ini rata-rata sudah memiliki resolusi minimal 4000 piksel.
2. Dalam menggunakan Algoritma K-Means pada penelitian berikutnya, disarankan untuk mencari cara agar penentuan *centroid* tidak terlalu acak.
3. Memperbanyak data untuk perangkat lunak pelajari dan proses.
4. Pada skripsi ini proses yang dilakukan oleh perangkat lunak menggunakan *pixel-based measurement*. Hal ini mengakibatkan jika terdapat satu piksel yang berbeda warna di antara piksel-piksel di sekitarnya, perangkat lunak dapat mengidentifikasi citra tersebut memiliki warna yang sebenarnya tidak ada. Oleh karena itu untuk penelitian lebih lanjut proses pengenalan warna pada citra dapat menggunakan *window-based measurement*. *Window-based measurement* digunakan untuk memeriksa area pada citra dibandingkan hanya memeriksa masing-masing piksel saja. Proses ini dilakukan untuk menghilangkan piksel yang berbeda dalam satu area sehingga perangkat lunak tidak mendeteksi warna tersebut menjadi bagian dari citra.

DAFTAR REFERENSI

- [1] Hidayatullah, P. (2017) *Pengolahan Citra Digital : Teori dan Aplikasi Nyata*. Informatika Bandung, Bandung.
- [2] Madenda, S. (2015) *Pengolahan Citra dan Video Digital: Teori, Aplikasi, dan Pemrograman Menggunakan MATLAB*. Erlangga, Jakarta.
- [3] Zaitoun, N. M. dan Aqel, M. J. (2015) Survey on image segmentation techniques. *Procedia Computer Science*, **65**, 797–806.
- [4] Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L., dan Zurada, J. (2017) *Artificial Intelligence and Soft Computing: 16th International Conference, ICAISC 2017, Zakopane, Poland, June 11-15, 2017, Proceedings*, Lecture Notes in Computer Science, . 1. Springer International Publishing.
- [5] IDRISSE, N. dan MARBOHA, A. (2013) Review of color image segmentation. *Third international symposium on Automatic Amazigh processing* , **?**, 15–20.
- [6] S, S. dan L, B. V. (2016) Color image segmentation using quad tree based image decomposition techniques. *International Journal of Advanced Research Trends in Engineering and Technology*, **3**, 196–201.
- [7] Dubey, S. R., Dixit, P., Singh, N., dan Gupta, J. P. (2013) Infected fruit part detection using k-means clustering segmentation technique. *Ijimai*, **2**, 65–72.
- [8] Jones, M. T. (2008) *Artificial intelligence: a systems approach*. Laxmi Publications, Ltd.
- [9] Wu, X. dan Kumar, V. (2009) *The Top Ten Algorithms in Data Mining* Chapman & Hall/CRC Data Mining and Knowledge Discovery Series. Taylor & Francis.
- [10] Witten, I. H., Moffat, A., dan Bell, T. C. (1999) *Managing gigabytes: compressing and indexing documents and images*. Morgan Kaufmann.
- [11] Burney, S. A. dan Tariq, H. (2014) K-means cluster analysis for image segmentation. *International Journal of Computer Applications*, **96**.
- [12] Shmmala, F. A. dan Ashour, W. (2013) Color based image segmentation using different versions of k-means in two spaces. *Global Advanced Research Journal of Engineering, Technology and Innovation (GARJETI)*, **1**, 030–041.