

APLIKASI *DIGITAL MATTING* MENGGUNAKAN METODE *BAYESIAN APPROACH*

Rudy Adipranata¹⁾, Kartika Gunadi²⁾, Novita Halim

^{1,2)} Program Studi Teknik Informatika, Universitas Kristen Petra

Jl. Siwalankerto 121-131, Surabaya

e-mail : rudya@peter.petra.ac.id

Abstrak

Penggunaan aplikasi *image processing* semakin diminati oleh banyak orang untuk berbagai keperluan. Dari sebuah gambar yang sederhana dapat dibuat gambar yang menarik. Proses *matting* bertujuan untuk mengambil sebuah atau beberapa bagian dalam gambar yang akan digabungkan dengan gambar lainnya.

Dalam penelitian ini dikembangkan aplikasi untuk melakukan proses *matting*. Proses awal pada aplikasi adalah *color quantization* untuk membentuk kelompok-kelompok warna yang kemudian akan diproses dengan metode *Bayesian* dan akan menghasilkan *alpha matte* dari gambar. *Alpha matte* ini yang kemudian dipakai untuk menggabungkan gambar asal dengan gambar yang baru.

Dari hasil pengujian yang dilakukan, hasil gambar *composite* sangat dipengaruhi oleh *trimap* yang dibuat. Kecepatan proses bergantung pada banyaknya *unknown region* dan juga *neighborhood*. Untuk ketegasan gambar hasil dipengaruhi oleh *minimum variance*.

Kata Kunci : *Matting, Compositing, Bayesian*

1. PENDAHULUAN

Matting adalah sebuah metode untuk mengambil *foreground* (objek) dari sebuah gambar untuk kemudian digabungkan (*compositing*) dengan gambar yang lain. *Matting* ini merupakan salah satu teknologi yang awalnya dikembangkan untuk produksi film dan video. Sebelum berkembang gambar digital, proses *matting* ini dilakukan dengan menggunakan film. Karena kualitas gambar pada film semakin berkurang saat melakukan pengkopian gambar, maka dikembangkan *digital matting*. Beberapa contoh metode *digital matting* yang dikembangkan pada awalnya adalah *blue-screen matting* dan *rotoscoping*. Metode ini membutuhkan *blue screen* dengan *lighting* yang terkontrol dan orang yang berpengalaman dalam pengambilan gambar.

Hal-hal yang kurang dari metode tradisional *digital matting* adalah tidak dapat melakukan proses *matting* pada gambar natural. Selain itu juga sulit untuk mendapatkan detail dari *foreground* yang ingin diambil dengan baik, contohnya adalah bila *foreground* yang ingin diambil adalah helai rambut yang sangat tipis. Dengan perkembangan teknologi yang semakin maju, maka orang-orang mulai mengenalkan berbagai macam algoritma untuk mengatasi permasalahan-permasalahan tersebut. Beberapa contoh algoritma *matting* adalah *Knockout* (Berman, 2000), algoritma *matting* yang dikembangkan oleh Ruzon dan Tomazi (Ruzon, 2000), dan *Bayesian matting* yang dikembangkan oleh Chuang, Curless, Salesin, dan Szeliski (Chuang, 2001).

Algoritma *Knockout* dan algoritma Ruzon dan Tomazi didesain untuk menghasilkan garis pada transisi *foreground* dan *background*, sehingga kurang sesuai untuk transparansi dan siluet yang tipis. *Bayesian matting* dikembangkan untuk mengatasi masalah tersebut. Dengan *Bayesian matting*, *background* dari gambar yang ingin diambil *foreground*-nya tidak harus *background* dengan satu warna saja. Selain itu *Bayesian matting* juga mampu mengambil *foreground* dengan detail yang cukup baik. Pada *Bayesian matting* dihitung distribusi dari *foreground* dan *background* yang kemudian akan diselesaikan dengan menggunakan *Bayes's rule*.

2. TINJAUAN PUSTAKA

2.1 *Trimap*

Trimap adalah sebuah gambar *pre-segmented* yang terdiri dari tiga bagian yaitu *foreground*, *background*, dan *unknown*. *Foreground* adalah bagian yang ingin diambil dari gambar. *Background* adalah bagian yang akan dihilangkan dari gambar, sedangkan *unknown* adalah bagian dari gambar dimana pixel-pixel dalam bagian ini akan diproses untuk mendapatkan nilai *opacity*. Terdapat dua macam *trimap*, yaitu *stroke trimap* dan *filled trimap* (Digital Film, 2011).

2.1.1 *Stroke Trimap*

Stroke trimap terdiri dari goresan tipis yang menunjukkan *foreground* dan *background*. Pixel yang tidak ditandai akan dianggap sebagai *unknown*. Contoh *trimap* dapat dilihat pada Gambar 1 (Digital Film, 2011).



Gambar 1. Stroke Trimap

2.1.2 Filled Trimap

Filled trimap menunjukkan sebuah gambar yang sepenuhnya berisi ketiga bagian segmen yaitu *foreground*, *background*, dan *unknown*. *Filled trimap* jika diproses akan lebih cepat dan akurat bila dibandingkan dengan *stroke trimap*. Contoh *filled trimap* dapat dilihat pada Gambar 2 (Chuang, 2001).



Gambar 2. Filled Trimap

2.2 Gaussian Function

Pada dimensi satu, fungsi *gaussian* adalah fungsi probabilitas kepadatan dari distribusi normal. Fungsi *gaussian* juga sering disebut sebagai *frequency curve*. Fungsi *gaussian* pada dimensi satu dapat dilihat pada persamaan (1) (Wolfram, 2012).

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (1)$$

Fungsi *gaussian* pada dimensi dua adalah fungsi distribusi untuk variasi x dan y yang tidak berkorelasi yang mempunyai *bivariate normal distribution* dan *standard deviasi* yang sama $\sigma = \sigma_x = \sigma_y$. Fungsi gaussian untuk dimensi dua terdapat pada persamaan (2) (Wolfram, 2012):

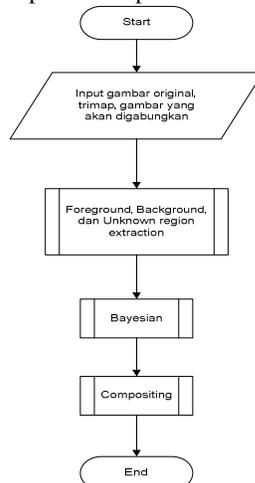
$$f(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{[(x-\mu_x)^2 + (y-\mu_y)^2]/(2\sigma^2)}{2\pi\sigma^2}} \quad (2)$$

2.3 Color Quantization

Banyak peralatan yang hanya dapat menampilkan warna dalam jumlah yang terbatas. Keterbatasan dari peralatan tersebut membuat sulit untuk menampilkan gambar natural yang memiliki rentang warna yang luas. Untuk itu maka warna-warna tersebut perlu dikuantisasi dengan sebuah *palette* dengan jumlah yang terbatas. Orchard dan Bouman mengajukan algoritma *color quantization* dengan menggunakan *color palette design* dan *binary tree palette design* untuk meminimalkan *total square error* (TSE) antara *pixel* yang asli dengan *pixel* hasil dari *palette* (Orchard, 1991).

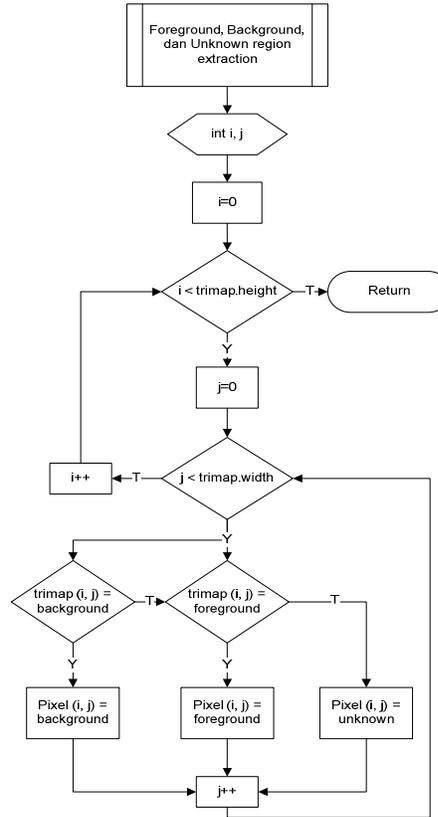
3. DESAIN APLIKASI

Desain aplikasi yang dikembangkan dapat dilihat pada Gambar 3.



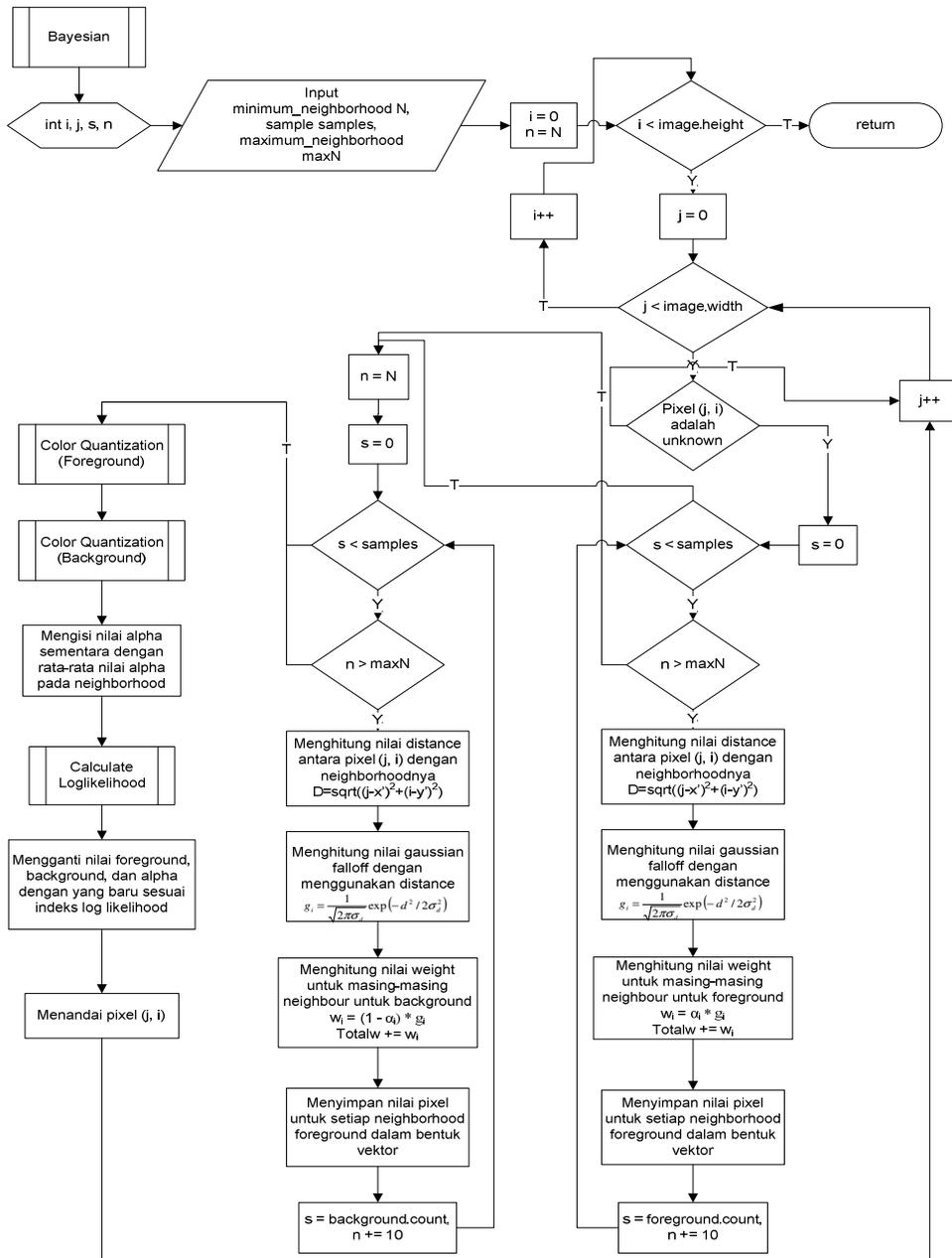
Gambar 3. Diagram Alir Desain Aplikasi

Proses *foreground*, *background* dan *unknown region extraction* digunakan untuk mengetahui sebuah piksel pada gambar original termasuk *foreground*, *background*, atau *unknown* menurut gambar *trimap*. Diagram alir proses ini dapat dilihat pada Gambar 4.



Gambar 4. Diagram Alir Proses *Foreground*, *Background*, dan *Unknown Region Extraction*

Bayesian adalah proses utama aplikasi untuk menentukan nilai *alpha matte* yang akan digunakan untuk menggabungkan *foreground* dengan gambar yang baru. Proses *Bayesian* hanya dilakukan pada bagian *unknown* saja. Untuk setiap piksel yang diproses akan ditentukan *neighborhood*-nya. Kemudian setiap *neighborhood* akan diberi *weight* masing-masing untuk *foreground* dan *background*. Setelah itu *foreground* dan *background* akan diproses dalam *Color Quantization* untuk menghasilkan kelompok-kelompok warna yang mempunyai *mean* dan *covariance*. Untuk setiap kelompok *foreground* dan *background* yang terbentuk akan dihitung nilai *foreground*, *background*, dan *alpha* sementara yang akan digunakan untuk menghitung nilai *log likelihood* pada proses *Calculate Loglikelihood*. Setelah didapatkan nilai *log likelihood* yang paling besar maka nilai *foreground*, *background*, dan *alpha* akan diganti dengan *foreground*, *background*, dan *alpha* yang baru. Proses ini berjalan terus sampai semua daerah *unknown* telah diproses. Diagram alir untuk proses *Bayesian* dapat dilihat pada Gambar 5.



Gambar 5. Diagram Alir Bayesian

4. HASIL DAN PEMBAHASAN

Dari hasil pengembangan aplikasi *digital matting* yang telah dilakukan, dilakukan pengujian yang meliputi:

- Pengujian terhadap parameter *maximum neighborhood*.
- Pengujian terhadap parameter *maximum iteration*.
- Pengujian terhadap parameter *minimum variance*.
- Pengujian dengan gambar *trimap* yang berbeda-beda.
- Pengujian terhadap ketebalan *unknown region trimap* yang berbeda-beda.

Maximum neighborhood (MaxN) adalah parameter yang digunakan untuk membatasi nilai *neighborhood*. Jika *minimum sample* yang diperoleh *neighborhood* baik untuk *foreground* maupun *background* tidak terpenuhi maka nilai *neighborhood* akan bertambah sampai batas *maximum neighborhood*. Hasil pengujian parameter *maximum neighborhood* untuk gambar pada Tabel 1 dapat dilihat pada Tabel 2.

Tabel 1. Gambar Original dan *Trimap* untuk Pengujian Parameter *Maximum Neighborhood*

Gambar Original	<i>Trimap</i>	Keterangan
		Ukuran : 527 x 413 <i>Unknown</i> Region : 28.936

Tabel 2. Hasil Pengujian Parameter *Maximum Neighborhood*

<i>MaxN</i>	<i>Alpha Matte</i>	<i>Composite</i>	Waktu
335			794.526 ms
775			1.892.089 ms
1105			1.925.193 ms

Dapat dilihat dari hasil pengujian pada Tabel 2 bahwa *maximum neighborhood* berpengaruh pada *alpha matte* saat daerah *unknown* pada *trimap* cukup besar. Selain itu, semakin besar *maximum neighborhood* membuat proses memakan waktu yang lebih lama.

Parameter *maximum iteration (MaxI)* berguna untuk membatasi iterasi yang dilakukan untuk proses perhitungan *likelihood* pada setiap pasang kelompok *foreground* dan *background* yang terbentuk. Hasil pengujian parameter *maximum iteration* dari gambar pada Tabel 3 dapat dilihat pada Tabel 4.

Tabel 3. Gambar Original dan *Trimap* untuk Pengujian Parameter *Maximum Iteration*

Gambar Original	<i>Trimap</i>	Keterangan
		Ukuran : 527 x 413 <i>Unknown</i> Region : 28.936

Tabel 4. Hasil Pengujian Parameter *Maximum Iteration*

<i>MaxI</i>	<i>Alpha Matte</i>	<i>Composite</i>	Waktu
110			1.836.819 ms

550			1.880.826 Ms
1100			1.838.254 ms

Parameter *minimum variance (MinVar)* adalah parameter untuk membatasi kelompok *foreground* dan *background* yang terbentuk. Bila nilai maksimal *eigenvalue* dari kelompok-kelompok yang terbentuk lebih kecil dari *minimum variance* maka iterasi untuk membentuk kelompok yang baru akan dihentikan. Pengujian parameter *minimum variance* dari gambar pada Tabel 5 dapat dilihat pada Tabel 6.

Tabel 5. Gambar Original dan *Trimap* untuk Pengujian Parameter *Minimum Variance*

Gambar Original	<i>Trimap</i>	Keterangan
		Ukuran : 270 x 377 <i>Unknown Region</i> : 20.295

Tabel 6. Hasil Pengujian Parameter *Minimum Variance*

<i>MinVar</i>	<i>Alpha Matte</i>	<i>Composite</i>	Waktu
0,005			1.859.485 ms
0,05			1.451.215 ms
0,5			1.263.406 ms

Dari hasil pengujian pada Tabel 6 semakin besar nilai *minimum variance* maka *alpha matte* yang terbentuk semakin tegas. Selain itu, semakin kecil nilai *minimum variance* maka waktu proses semakin lama. Hal ini disebabkan karena kelompok warna *foreground* dan *background* yang terbentuk dipengaruhi oleh *minimum variance*.

Trimap yang dibentuk cukup menentukan hasil proses dari gambar. Gambar *trimap* ini merupakan *input* dari user. Pengujian pada Tabel 7 menggunakan *trimap* yang berbeda untuk sebuah gambar yang sama. Masing-masing *trimap* digunakan untuk mengambil *foreground* yang berbeda.

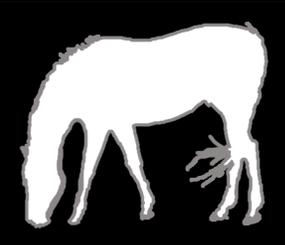
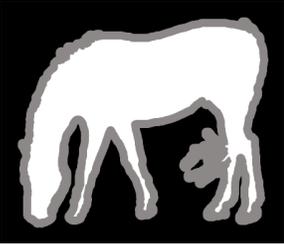
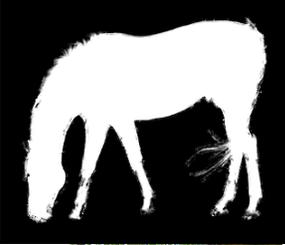
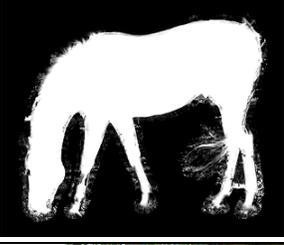
Tabel 7. Pengujian Menggunakan *Trimap* Berbeda Pada Gambar yang Sama

Gambar Original	 Ukuran : 854 x 633			
<i>Trimap</i>				
<i>Alpha Matte</i>				
<i>Composite</i>				
	Unknown Region : 31.845 Waktu : 2.288.269 ms		Unknown Region : 32.259 Waktu : 2.385.224 ms	

Pengujian untuk ketebalan *unknown region* pada *trimap* dapat dilihat pada Tabel 8. Ketegasan *alpha matte* dipengaruhi oleh ketebalan *unknown region* pada *trimap*. Dari Tabel 8 dapat diketahui bahwa semakin tebal *unknown region* maka semakin jelas *alpha matte* yang terbentuk.

Tabel 8. Pengujian Ketebalan *Unknown Region* pada *Trimap*

Gambar Original	 Ukuran : 460 x 389			
-----------------	--	--	--	--

Trimap		
Alpha Matte		
Composite		
	Unknown Region : 19.204 Waktu : 1.353.573 ms	Unknown Region : 39.479 Waktu : 2.684.683 ms

5. KESIMPULAN

Dari hasil pengujian dapat disimpulkan bahwa *trimap* sebagai *input* dari *user* mempengaruhi hasil *alpha matte* dan lama proses. Semakin presisi *trimap* yang dibuat maka *alpha matte* yang dihasilkan juga akan semakin jelas dan bila *unknown region* pada *trimap* semakin banyak jumlahnya maka proses akan semakin lama. Jumlah *maximum neighborhood* mempengaruhi waktu proses serta bentuk *alpha matte*. Semakin besar *maximum neighborhood* semakin lama waktu proses. Nilai *minimum variance* berpengaruh pada *alpha matte* yang terbentuk dimana semakin besar nilai *minimum variance* maka *alpha matte* yang terbentuk semakin tegas. Selain itu, semakin kecil nilai *minimum variance* maka waktu yang dibutuhkan semakin lama. Ketegasan *alpha matte* dipengaruhi oleh ketebalan *unknown region* pada *trimap* dimana semakin tebal *unknown region* maka semakin jelas *alpha matte* yang terbentuk.

DAFTAR PUSTAKA

- A. Berman, A. Dadourian, and P. Vlahos., 2000, *Method for removing from an image the background surrounding a selected object*, U.S. Patent 6,134,346.
- Digital Film Tools., 2011, *What is trimap?*, http://support.digitalfilmtools.com/support/index.php?_m=knowledgebase&_a=viewarticle&kbarticleid=78
- Chuang, Y., Curless, B., Salesin, D., and Szeliski, R., 2001, *A bayesian approach to digital matting*, Proceedings of IEEE CVPR, pp. 264-271
- M. A. Ruzon and C. Tomasi., 2000, *Alpha estimation in natural images*, In CVPR2000, pages 18–25, June 2000.
- Wolfram Mathworld, 2012, *Gaussian function*, <http://mathworld.wolfram.com/GaussianFunction.html>
- Orchard, M. T. dan Bouman, C. A., 1991, *Color quantization of images*, IEEE Transactions on Signal Processing, 39(12):2677– 2690.