

Pembuatan Aplikasi Objek Removal dengan Menggunakan *Exemplar-Based Inpainting*

Filbert Sugianto Manunggal¹, Liliana², Kartika Gunadi³

Program Studi Teknik Informatika, Fakultas Teknologi Industri, Universitas Kristen Petra
Siwalankerto 121-131, Surabaya, Indonesia

Email: s.manunggal@gmail.com¹, lilian@petra.ac.id², kgunadi@petra.ac.id³

ABSTRAK

Teknologi saat ini telah berkembang dengan pesat sehingga memudahkan pengambilan gambar. Namun seringkali terdapat objek yang tidak diinginkan pada gambar yang diambil. Masalah dapat timbul apabila objek tersebut dihilangkan dari gambar, karena akan menghasilkan ruang kosong pada gambar tersebut.

Untuk mengatasi masalah tersebut, pengisian ruang kosong (*target region*) pada gambar dengan menggunakan metode *novel based exemplar* sebagai metode untuk pengisian gambar. Keunggulan dari metode ini adalah penggunaan urutan pengisian gambar yang dipengaruhi oleh nilai *isophote* dan jumlah *source region*.

Hasil pengujian menunjukkan bahwa ukuran atau bentuk penyeleksian objek, gradasi, pembiasan warna sangat berpengaruh terhadap hasil *inpainting*. Prioritas sangat mempengaruhi pengambilan pada *source region* yang dicari. Gradien yang jelas tanpa dipengaruhi oleh pembiasan, gradasi warna dan blur akan membuat hasil *inpainting* yang alami.

Kata Kunci

Cross Isophote, Deteksi garis, *Inpainting*, *Novel based*, *Exemplar based*.

ABSTRACT

Technological development is so rapid, followed by the ease of taking pictures. The problem is the void caused by undesired object being erased.

To overcome the problem, filling the target void region within the image by using the novel based exemplar method as a method to fill the void. The main advantage of using this method is the usage of order filling that is dependent on the value of the isophote and the number of source region.

The result shows that the size or the shape of the object selection, gradation, diffusion of color, and blur significantly affects the result of the inpainting. Priority plays a significant role in picking the color in the source region. Clear gradient not affected by light diffusion, color gradation, or blur will make a "natural" inpainting result.

Keywords:

Cross Isophote, Edge Detection, *Inpainting*, *Novel based*, *Exemplar based*.

1. LATAR BELAKANG

Dewasa ini penyimpanan dalam bentuk digital semakin banyak digunakan karena lebih mudah diakses, dialihkan, dimanipulasi, dan lain-lain. Penggunaan kertas yang terlalu berlebihan dapat merusak atmosfer yang ada di bumi, disebabkan karena kertas merupakan hasil dari kayu, bambu, tumbuhan lain yang mendingung serat. Penghematan kertas pada penyimpanan digital didapatkan dengan dapat memperbaiki penyimpanan digital tersebut sebelum penyimpanan tersebut dicetak.

Penyimpanan digital yang dikenal banyak orang adalah penyimpanan gambar. Penyimpanan gambar dibutuhkan karena banyak media komunikasi *mobile* yang sudah dilengkapi kamera digital yang dapat membantu pengambilan gambar digital setiap saat. Selain digunakan pada kamera media komunikasi *mobile*, gambar digital juga banyak digunakan pada media-media elektronik lainnya seperti laptop, komputer, tablet, dan lain-lain.

Berkembangnya penyimpanan gambar didukung dengan aplikasi-aplikasi yang membuat penyimpanan itu sangat berguna. Aplikasi-aplikasi tersebut membantu dalam mengubah ukuran gambar, memanipulasi efek *brightness* dan *contrast* pada gambar, mengubah warna menjadi *black and white*, dan masih banyak lagi. Pengolahan gambar sangat membantu dalam mendapatkan gambar yang diinginkan.

Pengambilan gambar dengan menggunakan kamera digital semakin banyak digunakan dengan adanya kamera pada *mobile device* dan alat elektronik lainnya. Banyak gambar yang didapatkan dimanipulasi baik dalam segi pencahayaan maupun segi objek gambar sedemikian rupa hingga mendapat hasil gambar yang diinginkan. Gambar yang ingin dimanipulasi dalam segi objek gambar adalah gambar yang mempunyai objek yang tidak diinginkan atau dihapus. Namun manipulasi gambar ini dapat mengisi informasi-informasi bagian objek pada gambar yang dihapus sehingga tidak nampak dihapus.

2. TEORI

Inpainting merupakan suatu pengolahan gambar untuk mengatasi suatu objek yang tidak diinginkan.. Algoritma digital *inpainting* merupakan teknik dasar yang digunakan oleh restorator profesional di bidang seni lukis. Algoritma *inpainting* merupakan algoritma yang cukup konvensional. Terdapat banyak algoritma yang telah dikembangkan untuk menyempurnakan *inpainting*, diantaranya adalah *exemplar-based inpainting*, *the fast marching method*, *linear structure* dan masih banyak lagi yang lainnya.

Metode-metode tersebut melakukan pencarian dari daerah sumber untuk menentukan daerah yang paling mirip dengan daerah yang akan diisi. Pencarian pada suatu daerah dibutuhkan informasi-informasi pixel daerah sekitar. Pencarian tersebut dibantu dengan *sliding window* yang mempunyai ukuran yang variatif, dimulai dengan *sliding window* yang mempunyai ukuran 3x3, 5x5, 7x7 dan 9x9. Semakin banyaknya informasi daerah sekitar yang merupakan dari bagian daerah sumber mempunyai pengaruh yang baik terhadap pencarian.

Algoritma yang menggunakan order filing sebagai proses pengisian adalah *exemplar-based inpainting*. Informasi-informasi daerah sekitar sangat mempengaruhi pengisian, dengan kata lain bahwa prioritas pengisian bagian yang mempunyai pengaruh yang paling besar adalah pencarian sumber dengan keakuratan yang terbaik. Persamaan yang digunakan dalam proses pencarian dapat dilihat sebagai berikut:

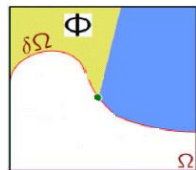
$$P(x,y) = C(x,y) * D(x,y) \quad (1)$$

x dan y pada persamaan pencarian prioritas merupakan koordinat sebuah titik yang akan dilakukan pengisian pada koordinat tersebut. Tentu semua itu dibutuhkan bantuan dari daerah sumber dan *sliding window*. *Sliding window* yang digunakan dalam *exemplar-based inpainting* adalah 9x9. [1],[2],[6]

Fungsi C pada persamaan pencarian prioritas merupakan intensitas daerah sumber pada window 9x9 dibandingkan target region. Sedangkan Fungsi dari D adalah persamaan untuk mencari nilai informasi-informasi yang terdapat pada suatu koordinat tersebut. Informasi-informasi dapat dinyatakan dengan sebuah nilai yang mempunyai perbandingan lurus dengan pencarian edge detection. Pengaruh daerah yang mempunyai hubungan dengan yang lain mempunyai prioritas untuk terlebih dahulu diisi. Namun faktor intensitas sumber dan informasi-informasi mempunyai perbandingan yang seimbang.[1]

$$\phi = I - \Omega \quad (2)$$

Ω merupakan symbol daerah yang akan dicari atau daerah yang telah dihapus atau daerah yang telah diseleksi, I adalah gambar keseluruhan. Sedangkan ϕ adalah daerah sumber dimana informasi-informasi yang hilang akan didapatkan dari daerah sumber.



Gambar 1 Daerah Sumber dan Daerah Pencarian

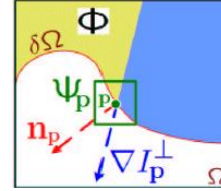
Pada Gambar 1, terdapat variabel yang belum didefinisikan. Variabel $\delta\Omega$ adalah variabel kontur dimana garis merah sebagai penanda dari kontur atau perbatasan antara daerah sumber dengan daerah yang dicari atau dihapus.

Setelah kontur, daerah yang dihapus, dan daerah yang tidak dihapus telah diketahui. Perhitungan prioritas pada setiap titik yang terdapat pada kontur akan dilakukan seperti pada persamaan berikut:

$$P(p) = C(p) * D(p) \quad (3)$$

p merupakan sebuah titik yang terdapat pada kontur. Perhitungan ini membutuhkan window 9x9 sebagai bantuan untuk mendapatkan informasi sekitar. Persamaan diatas dapat dijabarkan lagi sebagai berikut:

$$D(p) = \frac{\nabla U p^\perp \cdot n_p}{\alpha}$$



Gambar 2 Inisialisasi Variabel

Fungsi D adalah fungsi untuk mengukur nilai isophote dari kontur ke daerah yang dihapus. Perhitungan isophote dilakukan setiap kali iterasi. Setiap melakukan Update maka perhitungan isophote perlu dihitung kembali dikarenakan terjadi perubahan informasi-informasi pada *sliding window*, dimana kontur sekitar mengalami pengaruh isophote dan fungsi C juga. [6],[7],[10] Fungsi D juga dipengaruhi N_p , sebagai normal point yang mempunyai arah ke daerah yang telah dihapus. Sedangkan pada fungsi C melihat pada intensitas sumber. Semakin banyak nilai daerah sumber maka nilai C semakin besar pada titik tersebut. Nilai sumber yang dilihat berdasarkan pada *sliding window* tiap titik yang melintasi kontur.[1],[2],[8]

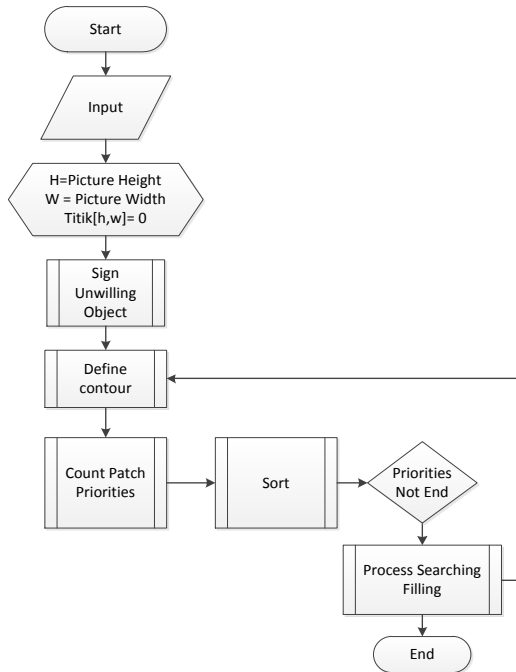
Kontur mempunyai beberapa fungsi yang akan digunakan dalam perhitungan prioritas, pencarian sumber yang akan diputari oleh *sliding window*. Proses pencarian sumber menggunakan perhitungan MSE (Mean Square Error) seperti pada persamaan berikut:

$$\sum \frac{(Wp - Wq)^2}{n} \quad (4)$$

Wp merupakan *sliding window* dengan ukuran 9x9 yang berada pada koordinat kontur. Dimana Wp adalah bagian yang membutuhkan informasi. Sedangkan Wq adalah bagian yang mempunyai kemiripan yang hampir sama dengan Wp . Setelah pencarian telah dilakukan ada beberapa hal yang dilakukan seperti pada Gambar 3.[9]

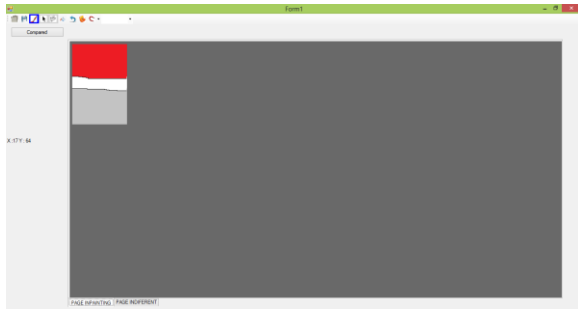
3. DESAIN SISTEM

Pembuatan aplikasi ini mempunyai beberapa proses yang terlibat didalamnya. Diantaranya adalah seleksi objek yang tidak diinginkan, pembuatan kontur, perhitungan nilai prioritas dan pencarian pada daerah sumber sebagai patch paling yang mirip dengan patch yang sedang dicari.

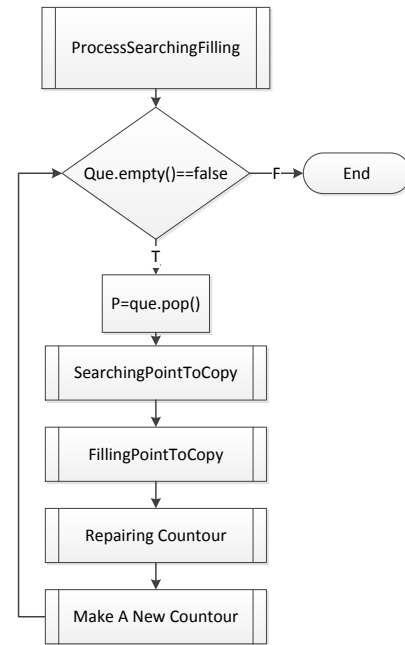


Gambar 3 Diagram Alir Sistem Secara Garis Besar

Proses pertama yang akan dilakukan adalah seleksi objek yang tidak diinginkan. Seleksi pada objek yang tidak diinginkan adalah menghapus objek yang tidak diinginkan proses penyeleksian objek diikuti dengan pembuatan kontur. Namun proses pembuatan kontur selalu dilakukan secara kontinu ketika target region berubah menjadi source region. Proses tersebut dapat dilihat pada Gambar 2.



Gambar 4 Pembuatan Seleksi Objek dan Kontur



Gambar 5 Diagram Alir Proses Pencarian Sumber

Pada Gambar 5 menunjukkan bahwa selama Target region masih ada maka kontur pun akan terus menerus dibuat secara berkelanjutan. Sesuai dengan arti kontur adalah perbatasan anantara daerah sumber dengan daerah yang dihapus. Untuk mengetahui daerah sumber didapatkan suatu persamaan sebagai berikut:

1. Identifikasi kontur.
2. Hitung Prioritas
3. Temukan Prioritas yang paling tinggi
4. Cari daerah sumber yang paling mirip
5. Update Cp dan Dp
6. Perbaharui kontur
7. Ulangi step 1-6 sampai kontur tidak ditemukan








Tabel 1 Pengisian dari daerah sumber ke daerah yang dicari
Sumber : [1]

4. PENGUJIAN

Percobaan pada *exemplar-based inpainting* dilihat berdasarkan 4 hal, antara lain:

1. Pengujian terhadap pemberian *masking* dengan ukuran atau bentuk yang berbeda pada gambar yang sama.
2. Pengujian terhadap pemberian *masking* pada gambar sampai menuju ke tepi gambar.
3. Pengujian terhadap pemberian *masking* yang *source region* di sekitarnya mempunyai pengaruh gradasi warna, ketajaman gambar yang kurang kuat atau *blur*, pembiasan warna.
4. Pengujian terhadap pemberian *masking* yang *source region* di sekitarnya tidak mempunyai pengaruh gradasi warna, ketajaman gambar yang kurang kuat atau *blur*, pembiasan warna.














4.1. Pengujian terhadap Pemberian Masking dengan Ukuran atau Bentuk yang Berbeda Pada Gambar yang Sama

Sumber Gambar				
No	Masking	Detail		Result
a		Masking	10.59%	
		Error	1733	
		Time	11:37	
b		Masking	12.53%	
		Error	2069	
		Time	12:53	
c		Masking	13.36%	
		Error	2519	
		Time	13:44	

Tabel 2 Pengujian dengan perbedaan masking

Bentuk dan ukuran *masking* mempengaruhi hasil dari *inpainting*. Perpotongan edge menyebabkan pengaruh tersebut terjadi.








4.2. Pengujian terhadap Pemberian Masking Menuju Pada Tepi Gambar

Sumber Gambar				
No	Masking	Detail		Result
a		Masking	16.37%	
		Error	0	
		Time	0:10	
b		Masking	9.87%	
		Error	0	
		Time	0:05	
c		Masking	13.58%	
		Error	0	
		Time	0:08	
d		Masking	17.87%	
		Error	1621	
		Time	0:23	
e		Masking	13.59%	
		Error	0	
		Time	0:07	
f		Masking	56.7%	
		Error	0	
		Time	00:30	

Tabel 3 Pengujian Masking Menuju Tepi

Pemberian *masking* sampai menuju tepi pada Tabel 3 menunjukkan bahwa semua kontur mempunyai nilai prioritas yang sama. Sehingga semua kontur mempunyai hak yang sama dalam pengisian setiap iterasi.






4.3. Pengujian Terhadap Masking yang Mempunyai Pengaruh



Sumber Gambar				
No	Masking	Detail		Result
a		Masking	28.81%	
		Error	544.32	
		Time	09:38	
b		Masking	10.13%	
		Error	249.6	
		Time	01:24	
c		Masking	3.38%	
		Error	17.09	
		Time	01:24	

Tabel 4 Pengujian Gambar yang mempunyai pengaruh

Nilai error pada pengujian Tabel 4 menunjukan bahwa nilai error bergantung pada segmen masking yang diinputkan. Semakin banyak segmen masking pada gambar maka pengaruh untuk menghasilkan nilai error semakin semakin besar semakin banyak. Segmen yang mempunyai pengaruh cahaya mempunyai pengaruh yang sangat besar seperti pada Gambar a.

4.4. Pengujian Terhadap Pemberian Masking Yang Source Region yang Tidak Mempunyai Pengaruh







Sumber Gambar				
No	Masking	Detail		Result
a		Masking	11.77%	
		Error	1593	
		Time	10:20	
b		Masking	14.53%	
		Error	707.55	
		Time	15:32	

c		Masking	12.38%	
		Error	763.14	
		Time	12:29	

Tabel 5 Pengujian Gambar yang Tidak Mempunyai Pengaruh

Nilai error pada Tabel 5 tidak berbanding lurus dengan besar *masking*. Segmen *masking* mempengaruhi dalam mendapatkan nilai error. Segmen pada Gambar b merupakan satu segmen masking yaitu terdapat pada rumput. Sehingga mendapatkan nilai terkecil pada besar masking 14.53% dengan perbandingan besar *masking* dengan *error*.

4.5. Pengujian Terhadap Ukuran Masking

Sumber Gambar			
Gambar Masking			
Big Of Picture		300x199	
Big Of Masking		7.78%	
No	Detail		Result
a	Error	113.44	
	Time	02:16	
	Big Of Patch	9x9	
b	Error	113.44	
	Time	02:09	
	Big Of Patch	7x7	
c	Error	154.68	
	Time	02:09	
	Big Of Patch	5x5	
d	Error	165.83	
	Time	00:04	
	Big Of Patch	3x3	

Tabel 4 Pengujian Ukuran Masking

Ukuran masking mempunyai pengaruh pada nilai kesamaan yang terdekat. Semakin dekat titik kesamaan dengan titik yang dicari akan mempengaruhi waktu pencarian.

5. KESIMPULAN

Berdasarkan sistem yang telah dikembangkan dan hasil pengujian yang telah dilakukan, dapat disimpulkan beberapa hal sebagai berikut:

- Efek blur, sinar, gradasi dan pembiasan cahaya sangat mempengaruhi prioritas. Bagian-bagian efek pencahayaan selalu mendapatkan prioritas yang tinggi. Sehingga bagian yang seharusnya mempunyai prioritas yang lebih tinggi akan kemungkinan menghilang.
- Nilai *error* dipengaruhi oleh besar *masking*, efek blur, sinar, gradasi dan pembiasan warna.
- Semakin kecil masking maka membuat nilai *error* yang dihasilkan lebih kecil. Namun nilai *error* juga dipengaruhi oleh pencarian pada gambar.
- Nilai *error* juga dipengaruhi oleh pemberian *masking*. Pemberian *masking* yang mempunyai lebih dari satu segmen mempunyai *error* lebih banyak, dikarenakan pencarian pada banyak segmen sangat dipengaruhi oleh prioritas, efek warna gambar dan kualitas gambar.
- Pencarian pada gambar dipengaruhi oleh kualitas resolusi pada gambar. Semakin besar resolusi gambar maka pencarian gambar semakin jelas, karena semakin besar resolusi gambar maka jumlah pixel semakin banyak yang membuat pixel lebih detail daripada jumlah pixel yang lebih kecil.
- Ukuran semakin kecil membuat pencarian semakin detail karena pencarian pada *patch* lebih jelas.
- Waktu proses *inpainting* dipengaruhi oleh besar *masking*, besar gambar, ukuran *patch* dan pencarian *patch* yang terdekat. Semakin dekat pencarian yang ditemukan semakin cepat waktu proses yang dikerjakan.

- Gambar dengan kualitas yang lebih besar mempunyai nilai *error* yang lebih kecil, nilai *error* dibandingkan dengan besar gambar. Namun pemberian masking pada segmen sangat mempengaruhi pada hasil nilai *error*. Pemberian *masking* pada objek yang sama dengan ukuran atau bentuk yang berbeda mempengaruhi hasil *inpainting*
- Bentuk *masking* yang memiliki rongga pada bagian tengah sangat mempengaruhi hasil dikarenakan informasi-informasi bagian tengah belum terlalu lengkap.

6. DAFTAR PUSTAKA

- [1] A. Efros and W.T. Freeman. (2001) Image quilting for texture synthesis and transfer. In *Proc. ACM Conf. Comp. Graphics (SIGGRAPH)*, pp. 341–346, Eugene Fiume.
- [2] G.V.R. Sagar, S.Kashif hussain. An Image Inpainting Technique Based on 8-Neighborhood Fast Sweeping Method. Dept. of Electronics & Communication Engineering, faculty of engineering, G.Pulla Reddy Engineering College, Andhra Pradesh, India
- [3] G.V.R. Sagar, S.Kashif hussain. An Image Inpainting Technique Based on 8-Neighborhood Fast Sweeping Method. Dept. of Electronics & Communication Engineering, faculty of engineering, G.Pulla Reddy Engineering College, Andhra Pradesh, India
- [4] Wu, Jiyong., Ruan, Qiuqi. (2006). *Object Removal By Cross Isophotes Exemplar-based Inpainting*.
- [5] Jacobs, David. (2005). *Image Gradients*.
- [6] Aujol, Jean-Fran_cois. (2010). *Exemplar-based inpainting from a variational point of view*.
- [7] Wu, Jiyong., Ruan, Qiuqi. (2006). A Novel Exemplar-Based Image Completion Model.
- [8] Wu, Jiyong., Ruan, Qiuqi., AN, Gaoyun. (2010). Exemplar-Based Image Completion Model Employing PDE Corrections
- [9] Edwards, Stephen A. (2012). *Fundamentals of Computer Systems Bresenham's Line Algorithm in Hardware*.