

Deteksi Tepi Danau Pada Citra Satelit Menggunakan Metode Canny

Koko Pradityo¹, Budi Darma Setiawan², Randy Cahya Wihandika³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹kk.pradityo@gmail.com, ²s.budidarma@ub.ac.id, ³rendicahya@ub.ac.id

Abstrak

Danau adalah sebuah fitur darat yang berperan penting dalam kehidupan manusia. Perubahan pada danau dapat memengaruhi keadaan lingkungan sekitar serta kehidupan masyarakat yang berada disekitarnya. Salah satu cara untuk mengetahui perubahan kondisi danau dilakukan dengan melakukan deteksi tepi permukaan danau pada citra satelit untuk mengetahui perubahan luas danau tersebut. Penerapan algoritme yang tepat dan optimasi lebih lanjut dapat membantu analisis keadaan danau. Aplikasi menerapkan algoritme deteksi tepi Canny untuk melakukan deteksi tepi permukaan danau pada citra satelit. Algoritme segmentasi berdasarkan metode color thresholding diterapkan untuk melakukan optimasi pada performa deteksi tepi. Hasil pengujian menunjukkan bahwa deteksi tepi pada citra satelit dengan hanya menggunakan metode Canny menghasilkan error rate sebesar 57%, sementara proses segmentasi dengan color thresholding meningkatkan kinerja deteksi tepi sebesar 67%.

Kata kunci: danau, citra satelit, deteksi tepi, metode Canny, segmentasi, color thresholding

Abstract

Lake is an important land feature for human life. Changes in a lake's condition would affect the environment and the people living nearby. One of the method being used to detect changes in lake's condition is by using an edge detection of the lake based on satellite image for further analysis such as measuring the change in lake's total area. Appropriate implementation and optimization of such algorithm can lead to a better analysis of the lake's condition. In this research, the system implemented Canny Edge Detection algorithm to detect the edge of a lake on a satellite image. A segmentation algorithm based on color thresholding is used to improve the edge detection algorithm. The test result shows that Canny Edge Detection algorithm has 57% error detection rate, while segmentation process using color thresholding improves the detection performance by 67%.

Keywords: Canny edge detection, lake, satellite image, segmentation, color thresholding

1. PENDAHULUAN

Danau merupakan sebuah fitur daratan yang memiliki peran penting bagi kehidupan manusia dan ekosistem yang berada di sekitarnya. Perubahan terhadap kondisi danau seperti volume danau dapat berakibat negatif pada kehidupan manusia, seperti turunnya debit air irigasi serta suplai air bersih untuk warga. Untuk itu, pemantauan kondisi danau harus terus dilakukan.

Salah satu cara untuk memantau kondisi danau adalah dengan menggunakan pencitraan satelit. Dari citra satelit, pakar dapat melakukan berbagai hal, diantaranya deteksi tepi danau.

Dalam bidang pengolahan citra satelit, danau merupakan salah satu objek yang cukup

menarik untuk diteliti. Bentuk danau yang diskrit dan kontras antara permukaan danau dan permukaan tanah disekitarnya menjadikan danau sebagai objek deteksi tepi citra yang baik.

Pada penelitian ini, peneliti menggunakan algoritme deteksi tepi Canny untuk melakukan deteksi tepi danau pada citra satelit. Algoritme ini dipilih oleh peneliti karena telah terbukti memiliki kinerja paling baik diantara algoritme deteksi tepi lainnya (Nadernejad, 2008). Lebih lanjut, peneliti menggunakan sebuah algoritme segmentasi berdasarkan warna (*color thresholding*) untuk meningkatkan kinerja deteksi tepi danau.

Proses deteksi tepi permukaan danau memiliki beberapa potensi manfaat yang signifikan, seperti mengetahui keliling dan luas suatu danau. Informasi tersebut dapat digunakan

sebagai acuan untuk menentukan perubahan kondisi suatu danau dalam rentang waktu tertentu.

2. PENELITIAN TERDAHULU

Pada penelitian terdahulu yang membandingkan beberapa metode deteksi tepi, Sharifi (2002) mengungkapkan bahwa pada kondisi citra yang memiliki lebih banyak *noise*, algoritme deteksi tepi Canny memiliki kinerja yang lebih baik dibandingkan dengan metode lain seperti ISEF, Canny, Sobel, Kirsch, Marr-Hildreth, Sobel serta Laplacian.

Sementara Liu H (2004) dalam penelitiannya menggunakan algoritme Canny dan metode *thresholding* lokal adaptif untuk melakukan deteksi tepi pantai secara otomatis. Kesimpulan dari penelitian tersebut mengungkap bahwa salah satu faktor dalam menjaga akurasi deteksi tepian pantai adalah dengan cara menerapkan segmentasi.

Penelitian lain yang dilakukan oleh Kulkarni (2012) mendapatkan kesimpulan bahwa sebuah algoritme segmentasi multi-level terhadap ketiga komponen warna (merah, hijau, dan biru) memiliki kinerja yang baik ketika citra yang diolah memiliki perbedaan warna yang diskrit.

3. DATA EKSPERIMEN

Penelitian ini menggunakan data masukan berupa citra satelit LANDSAT 4 dan 5 yang didapatkan oleh peneliti dari situs USGS (<http://earthexplorer.usgs.gov>). Dataset citra yang diambil merupakan satelit LANDSAT 4-5 TM C1 Level-1 dengan format TIFF. Resolusi citra sebelum dilakukan *pre-processing* adalah 8031 x 6931 piksel dengan kedalaman warna 8-bit. Citra danau yang diambil merupakan citra danau yang berada pada wilayah NKRI.



Gambar 1. Contoh citra satelit yang diolah

4. ALGORITME DETEKSI TEPI CANNY

Detektor tepi Canny adalah sebuah operator deteksi tepi yang menggunakan algoritme banyak-tingkat untuk mendeteksi berbagai

macam tepian pada gambar. Langkah-langkah deteksi tepi Canny dapat dijelaskan sebagai berikut:

1. Penerapan filter Gaussian

Langkah ini dilakukan untuk menghilangkan *noise* yang terdapat pada citra. Filter Gaussian yang umum digunakan adalah persamaan (1).

$$H_{ij} = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(i-(k+1))^2 + (j-(k+1))^2}{2\sigma^2}\right); \quad (1)$$

dengan $1 \leq i, j \leq (2k + 1)$

Penjelasan variabel:

i, j = koordinat piksel yang diperiksa
 k = ukuran kernel gaussian
 σ = sigma (deviasi standar)

2. Menemukan intensitas gradien citra

Algoritme Canny menggunakan filter untuk mendeteksi tepian horizontal, vertikal, dan diagonal pada sebuah citra. Pada operator deteksi tepi, nilai kembalian merupakan derivatif pertama dalam bentuk arah horizontal (G_x) dan vertikal (G_y). Pada penelitian ini, digunakan operator Sobel untuk mendapatkan derivatif pertama. Tepi dan arah gradien dapat ditentukan dengan persamaan (2).

$$G = \sqrt{G_x^2 + G_y^2} \quad (2)$$

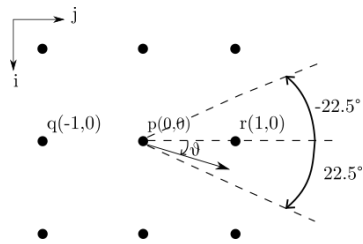
$$\Theta = \text{atan2}(G_y, G_x)$$

Penjelasan variabel:

G = Intensitas gradien piksel
 G_x = gradien komponen horizontal
 G_y = gradien komponen vertikal
 Θ = Nilai arah gradien

3. Penekanan *non-maximum*

Penekanan *non-maximum* digunakan untuk menipiskan garis tepian yang sudah didapatkan dari ketiga langkah sebelumnya. Penekanan ini dilakukan dengan mencari nilai gradien piksel tersebut lalu membandingkannya dengan piksel tetangganya yang termasuk dalam besaran sudut gradien piksel tersebut.



Gambar 2. Penekanan non-maksima

Dari Gambar 1, apabila piksel yang diperiksa adalah piksel p , dan nilai gradien yang didapatkan bernilai diantara $-22,5^\circ$ dan $22,5^\circ$, maka nilai dari nilai intensitas p akan dibandingkan dengan nilai intensitas dari piksel r yang terletak pada dalam jangkauan gradien piksel p . Pada Implementasinya, nilai gradien dibagi dalam empat bagian sebagai berikut:

- Horizontal, bila nilai gradien terletak diantara $22,5^\circ$ hingga $-22,5^\circ$ atau gradien terletak antara $157,5^\circ$ dan $-157,5^\circ$
- Vertikal, bila nilai gradien terletak diantara $-112,5^\circ$ hingga $-67,5^\circ$ atau gradien terletak antara $67,5^\circ$ dan $112,5^\circ$
- $+45$, bila nilai gradien terletak diantara $-67,5^\circ$ hingga $-22,5^\circ$ atau gradien terletak antara $112,5^\circ$ dan $157,5^\circ$
- -45 , bila nilai gradien terletak diantara $-157,5^\circ$ hingga $-112,5^\circ$ atau gradien terletak antara $22,5^\circ$ dan $67,5^\circ$

Untuk mengubah keluaran dari persamaan (2) kedalam satuan derajat, maka digunakan persamaan (3).

$$G = \text{atan}\left(\frac{G_y}{G_x}\right) * \frac{180}{\pi} \quad (3)$$

- Menerapkan *thresholding* ganda *Threshold* yang digunakan dalam algoritme ini digunakan untuk menentukan apakah sebuah tepi diklasifikasikan sebagai sebuah tepian lemah atau kuat. Apabila nilai gradien sebuah piksel lebih besar daripada *threshold* atas, maka piksel tersebut diklasifikasikan sebagai tepi kuat. Apabila nilai gradien piksel berada diantara nilai *threshold* atas dan

threshold bawah, maka piksel tersebut merupakan sebuah tepian lemah.

- Melacak tepian dengan *hysteresis*
Proses ini dilakukan untuk menjaga tepian agar kontinu. Tepian lemah yang terdeteksi dan bertetangga secara langsung dengan sebuah tepian kuat akan ditandai sebagai tepian kuat.

5. COLOR THRESHOLDING

Color Thresholding atau *thresholding* warna merupakan salah satu metode segmentasi citra. Pada umumnya segmentasi dilakukan dengan terlebih dahulu mengubah citra yang akan diolah kedalam bentuk *grayscale*, lalu menerapkan metode segmentasi yang diinginkan pada citra tersebut. Pada kasus deteksi tepi danau pada citra satelit, warna permukaan danau dengan daratan cenderung memiliki perbedaan yang diskrit, sehingga segmentasi dengan menggunakan metode *thresholding* warna sebelum melakukan deteksi tepi dapat membantu menghilangkan noise yang terdeteksi sebagai tepi.

Untuk melakukan *thresholding* warna pada citra, dapat digunakan persamaan (4).

$$g(x,y) = \begin{cases} f(x,y), & 0 \leq \text{red}(x,y) \leq tR, \\ 0, & \text{red}(x,y) > tR. \end{cases}$$

$$g(x,y) = \begin{cases} f(x,y), & 0 \leq \text{green}(x,y) \leq tG, \\ 0, & \text{green}(x,y) > tG. \end{cases} \quad (2.1)$$

$$g(x,y) = \begin{cases} f(x,y), & 0 \leq \text{blue}(x,y) \leq tB, \\ 0, & \text{blue}(x,y) > tB. \end{cases}$$

Penjelasan variabel:

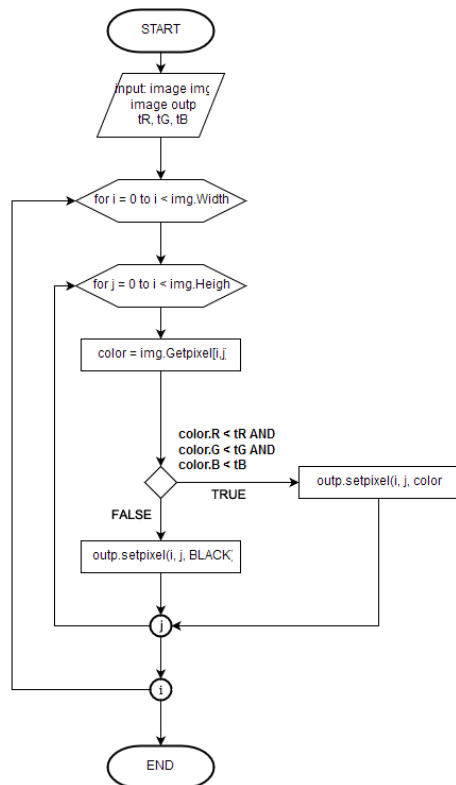
x, y = koordinat piksel citra yang diperiksa

tR = *threshold* komponen warna merah

tG = *threshold* komponen warna hijau

tB = *threshold* komponen warna biru

Color thresholding yang dilakukan pada aplikasi ini bertujuan untuk memisahkan wilayah danau dari permukaan tanah. Dengan segmentasi ini, wilayah permukaan tanah ditandai dengan warna hitam sementara wilayah danau tidak mengalami perubahan. Nilai *threshold* atau batasan untuk tiap komponen warna dimasukkan secara manual oleh pengguna dengan memperhatikan histogram warna yang ditampilkan pada aplikasi.

Gambar 3. Diagram alir *color thresholding*

6. GROUND TRUTH

Dalam konteks penginderaan jarak jauh (*remote sensing*) *Ground Truth* mengacu pada informasi yang didapatkan langsung dari lapangan. Secara spesifik, *ground truth* mengacu pada proses dimana sebuah piksel dari sebuah citra satelit dibandingkan dengan fitur nyata yang terdapat di lapangan untuk memverifikasi konten dari piksel tersebut.

Menurut Jianhong (2011), terdapat dua cara untuk menghasilkan *ground truth*. Cara pertama adalah dengan menggunakan instrumen dengan akurasi tinggi untuk mendapatkan pengukuran yang tepat. Cara kedua adalah dengan pembuatan data secara manual oleh pakar pada kasus dimana hasil penginderaan tidak dapat diukur oleh alat secara akurat. Pada penelitian ini, citra *ground truth* yang dibuat secara manual oleh peneliti digunakan sebagai acuan sebagai pembandingan yang digunakan untuk menguji akurasi citra keluaran dari aplikasi.

Untuk mengetahui nilai akurasi atau *error rate* dari citra keluaran, dilakukan pencocokan dengan metode *ground truth*. Citra keluaran dibandingkan sebuah citra *ground truth* (selanjutnya akan disebut sebagai citra *key*) dengan tepian permukaan danau telah ditandai secara manual oleh peneliti.

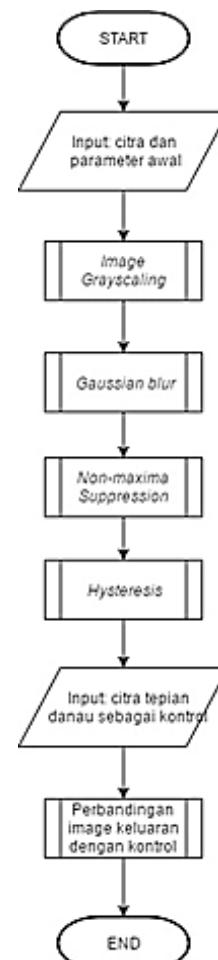
Proses pembuatan citra *key* (*ground truth*) melibatkan proses menandai tepian permukaan danau secara manual per piksel untuk setiap gambar yang akan dijadikan *key*, serta melakukan dilasi menggunakan aplikasi *imageJ* untuk tepian sebanyak 1 piksel untuk mengakomodasi translasi tepian pada citra keluaran aplikasi.

Gambar 4. Sampel citra *key*

7. IMPLENTASI SISTEM

Perancangan sistem didasarkan pada analisis kebutuhan dan data yang telah diperoleh. Fokus utama pada sistem adalah kemudahan *input* parameter serta kejelasan hasil *output* yang dilakukan sehingga dapat dianalisis oleh peneliti.

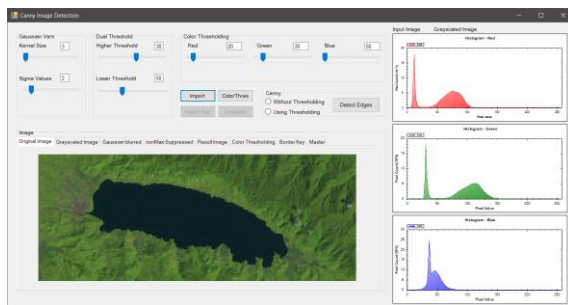
Alur kerja aplikasi dapat digambarkan dengan menggunakan diagram alir pada Gambar 5.



Gambar 5. Diagram alir aplikasi deteksi tepi

8. HASIL DAN ANALISIS

Dalam penelitian ini, penulis berhasil melakukan deteksi tepi danau pada citra satelit menggunakan metode Canny. Aplikasi yang dibuat mampu menerima masukan pengguna berupa citra awal, nilai parameter dan citra *key (ground truth)* serta menghasilkan keluaran berupa citra dengan tepian danau yang telah dideteksi tepinya serta jumlah *error rate* deteksi tepi untuk citra yang diolah.



Gambar 6. Tampilan antarmuka aplikasi

Dari aplikasi yang telah dibuat, dilakukan analisis meliputi pengujian terhadap perubahan parameter Gaussian dan *threshold* yang digunakan dan perbandingan antara hasil deteksi tepi dengan melibatkan proses segmentasi *thresholding* warna serta hasil deteksi tepi tanpa melakukan proses segmentasi.



Gambar 7. Sampel hasil keluaran aplikasi

8.1 Metode Pengujian

Pengujian algoritme deteksi tepi Canny dilakukan untuk mengetahui tingkat keberhasilan algoritme dalam menyelesaikan permasalahan yaitu mendeteksi tepi permukaan danau pada citra satelit masukan.

Untuk mengetahui nilai parameter Gaussian serta *threshold* yang optimal, dilakukan proses deteksi tepi dengan menggunakan parameter yang berbeda-beda. Aplikasi juga melakukan perbandingan antara citra keluaran dari proses yang hanya melibatkan deteksi tepi Canny dan keluaran citra hasil dari deteksi tepi yang telah melalui proses pre-processing dengan segmentasi menggunakan *thresholding* warna.

8.2 Pengujian parameter Gaussian dan *threshold*

Proses pengujian ini dilakukan untuk mengetahui nilai parameter Gaussian dan *threshold* yang optimal untuk setiap sampel citra yang dijadikan sebagai masukan. Nilai awal parameter ini ditetapkan secara manual, yaitu ukuran kernel Gaussian 3, nilai sigma 1, *threshold* bawah 30, dan *threshold* atas 50.

Pada pengujian ini, digunakan 10 citra sampel yang diambil dari citra satelit LANDSAT 4 dan 5, dan telah dilakukan pre-processing sebelumnya sehingga citra satelit memiliki format channel RGB. Hasil dari pengujian nilai parameter dapat dilihat pada Tabel 2.

Tabel 2. Pengujian parameter gaussian dan *threshold*

G	Parameter							
kS	dS	tA	tB	TD	ED	TM	RE	
A	3	1	50	30	2412	1704	708	70,65
A	5	5	35	20	1650	1023	627	62,00
B	3	1	50	30	911	467	444	51,26
B	3	3	35	15	1744	1044	700	59,86
C	3	1	50	30	7935	6993	942	88,13
C	5	2	36	12	5351	4461	890	83,37
D	3	1	50	30	1894	1364	530	72,02
D	5	3	26	10	2410	1848	562	76,68
E	3	1	50	30	3712	2908	804	78,34
E	5	4	50	30	1630	917	713	56,26
F	3	1	50	30	1069	347	722	32,46
F	5	4	38	30	818	192	626	23,47
G	3	1	50	30	16623	14163	2460	85,20
G	7	6	50	40	3893	1886	2007	48,45
H	3	1	50	30	2122	987	1135	46,51
H	5	5	45	30	1400	422	978	30,14
I	3	1	50	30	2034	800	1234	39,33
I	5	5	40	30	1706	523	1183	30,66
J	3	1	50	30	9912	8307	1605	83,81
J	5	5	55	30	2318	853	1465	36,80
Rata-rata:								57,70
Deviasi Standar:								21,19

Keterangan tabel:

- G = label citra
- kS = kernel size (ukuran kernel)
- dS = sigma (deviasi standar)
- tA = *threshold* atas
- tB = *threshold* bawah

TD = Tepian terdeteksi (piksel)
 ED = Error terdeteksi (piksel)
 TM = Tepian mutlak terdeteksi (piksel)
 RE = Rasio *error* (%)

Berdasarkan Tabel 2, nilai parameter Gaussian berpengaruh terhadap proses menghilangkan noise citra yang dapat disalahartikan oleh algoritme Canny sebagai tepian. Hal ini konsisten pada setiap kasus, dan terlihat paling jelas pada kasus citra G, dimana perubahan parameter Gaussian dengan cara menaikkan ukuran kernel serta memperbesar sigma secara signifikan menghilangkan noise pada citra yang diolah. Perubahan parameter Gaussian pada citra G menyebabkan penurunan nilai *error* sebesar 86%.

Kasus-kasus lain menunjukkan sebuah pola yang berulang, yaitu menaikkan parameter Gaussian berguna untuk menghilangkan noise pada permukaan tanah yang berpotensi dideteksi secara salah oleh algoritme Canny sebagai tepian. Nilai *threshold* bawah dapat diturunkan untuk meningkatkan toleransi batasan nilai piksel yang dianggap sebagai tepian.

Dari Tabel 2 dapat diketahui rasio *error* pada deteksi tepi menggunakan algoritme Canny berkisar antara 23,4% hingga 70%, dengan nilai rata-rata *error* sebesar 57,7% dan deviasi standar sebesar 21,2%. Hal ini dapat diartikan bahwa dari seluruh piksel yang dideteksi oleh algoritme sebagai tepian, jumlah noise yang ikut terhitung sebagai tepi lebih banyak daripada tepian yang sebenarnya. Percobaan untuk menurunkan jumlah noise dengan cara menaikkan parameter Gaussian berakibat turunnya jumlah tepian sebenarnya yang terdeteksi, dan menyebabkan tepian permukaan danau yang terdeteksi menjadi diskontinu.

8.3 Pengujian dengan melibatkan segmentasi

Pada pengujian ini, dilakukan perbandingan antara hasil citra keluaran tanpa melibatkan proses segmentasi dengan hasil citra keluaran yang telah melalui proses segmentasi sebelum dilakukan proses deteksi tepi. Tabel 3 Menunjukkan hasil pengujian dengan menggunakan segmentasi.

Berdasarkan Tabel 3, dapat dilihat bahwa proses segmentasi dengan menggunakan *thresholding* warna secara signifikan meningkatkan kinerja algoritme deteksi Canny. Lebih lanjut, Tabel 3 menunjukkan bahwa kinerja algoritme deteksi tepi lebih bergantung

pada parameter *threshold* warna yang digunakan. parameter Gaussian dan *threshold* Canny memiliki efek yang tidak signifikan terhadap kinerja deteksi tepi.

Tabel 3. Pengujian dengan melibatkan segmentasi

G	Parameter				Threshold			Error
	kS	dS	tA	tB	R	G	B	
A	5	5	35	20	-	-	-	1023
A	3	1	30	10	20	40	50	170
B	3	3	35	15	-	-	-	1044
B	5	1	50	30	45	90	92	771
C	5	2	36	12	-	-	-	4461
C	3	2	40	35	35	35	80	444
D	5	3	26	10	-	-	-	1848
D	3	2	50	30	35	60	90	227
E	5	4	50	30	-	-	-	917
E	3	2	30	10	40	35	77	103
F	5	4	38	30	-	-	-	192
F	3	5	35	20	30	50	80	94
G	7	6	50	40	-	-	-	1886
G	3	2	35	20	35	50	100	458
H	5	5	45	30	-	-	-	422
H	3	2	45	30	40	55	80	247
I	5	5	40	30	-	-	-	523
I	5	3	40	30	50	70	85	339
J	5	5	55	30	-	-	-	853
J	3	1	40	30	25	45	75	99

Proses segmentasi dapat memisahkan bagian permukaan danau dan secara langsung melakukan penekanan terhadap fitur permukaan tanah, sehingga *noise* yang terdapat pada permukaan tanah tidak akan diproses oleh algoritme deteksi tepi Canny. Proses segmentasi dengan *thresholding* warna menghasilkan tepian danau yang lebih kontinu.

Tabel 4. Penurunan *error* setelah proses segmentasi

Citra	Error (piksel)		Selisih	Penurunan <i>error</i> (%)
	tS	dS		
A	1023	170	853	83,38
B	1044	771	273	26,15
C	4461	444	4020	90,05
D	1828	227	1601	87,58
E	917	103	814	88,77
F	192	94	98	51,04
G	1886	458	1428	75,72
H	442	247	195	44,12

I	523	339	184	35,18
J	853	99	754	88,39
Rata-rata				67,04

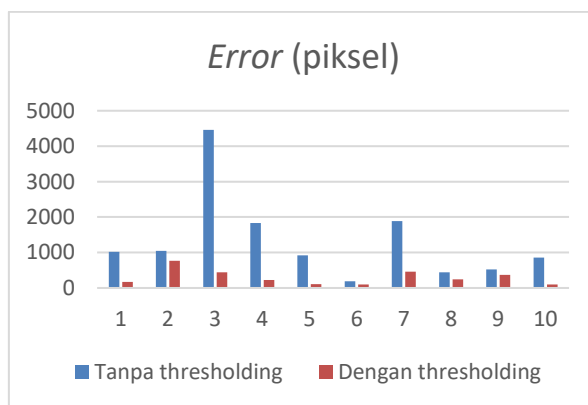
Keterangan label:

tS = deteksi tepi tanpa segmentasi

dS = deteksi tepi dengan segmentasi

Berdasarkan Tabel 4, dapat dilihat bahwa kinerja deteksi tepi meningkat antara 26% hingga 90% setelah diterapkan proses segmentasi. Rata-rata peningkatan kinerja adalah 67%.

Gambar 8 menunjukkan grafik perbandingan nilai error hasil deteksi tepi dengan proses segmentasi dan tanpa proses segmentasi.



Gambar 8. Perbandingan jumlah error

9. KESIMPULAN

Dari penelitian yang dilakukan, dapat ditarik kesimpulan bahwa algoritme Canny dapat digunakan untuk melakukan deteksi tepi permukaan danau pada citra satelit dengan catatan bahwa nilai parameter Gaussian dan *threshold* tidak mempunyai besaran tertentu yang dapat diaplikasikan pada setiap kasus secara umum. Meski demikian, besaran parameter Gaussian memiliki sebuah pola berulang yaitu bertambahnya ukuran *kernel* Gaussian berpengaruh dalam proses reduksi *noise* pada citra masukan yang akan diolah oleh algoritme Canny, sedangkan besaran parameter *threshold* berpengaruh dalam menentukan toleransi piksel tepian sehingga membantu membentuk tepian yang bersifat kontinu.

Penerapan algoritme Canny dalam proses deteksi tepi permukaan danau pada citra satelit memiliki nilai rata-rata kesalahan deteksi sebesar 57,7% dengan deviasi standar 21,2%. Lebih lanjut, penerapan segmentasi citra

menggunakan metode *color thresholding* meningkatkan kinerja deteksi tepi permukaan danau pada citra satelit dengan rata-rata peningkatan kinerja sebesar 67,04% dan nilai rata-rata kesalahan deteksi tepi sebesar 23,4%.

Penelitian lebih lanjut dapat dilakukan dalam hal otomatisasi penentuan nilai parameter Gaussian serta *threshold* atas dan bawah untuk algoritme Canny. Hal ini dapat dilakukan dengan memasukkan sampel citra melalui proses *Machine Learning* Selanjutnya, perlu dilakukan penelitian tentang manfaat segmentasi dalam deteksi tepi dengan menggunakan metode segmentasi selain metode *color thresholding*. Penelitian tentang segmentasi berbasis warna juga perlu dilakukan untuk penentuan nilai *threshold* warna dalam segmentasi. Hal ini dapat dilakukan dengan cara mencari nilai *threshold* pada histogram tiap channel warna menggunakan konsep *Local Maximum*.

10. DAFTAR PUSTAKA

- Ehsan Nadernejad, Sara Sharifzadeh & Hamid Hassanpour, 2008. *Edge Detection Techniques: Evaluations and Comparisons*. Applied Mathematical Sciences, Vol. 2. no. 31, pp 1507 – 1520
- Mohsen Sharifi, Mahmoud Fathy, Maryam Tayefeh Mahmoudi, 2002. *A Classified and Comparative Study of Edge Detection Algorithms*. *Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC '02)*
- H. Liu dan K. C. Jezek, 2004. *Automated extraction of coastline from satellite imagery by integrating Canny edge detection and locally adaptive thresholding methods*. *International Journal of Remote Sensing*. Vol. 25, no. 5, pp 937 – 958.
- Nilima Kulkarni, 2012. *Color Thresholding Method for Image Segmentation of Natural Images*. *International Journal of Image, Graphics and Signal Processing*, Vol 1, pp. 28 – 34
- Alifiya Jahagirdar, Dr. Manisha Patil, Dr. Vrushsen Pawar & Dr. Vilas Kharat, 2016. *Comparative Study of Satellite Image Edge Detection Techniques*. *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 4, Issue 5.
- Campbell, J. B. 2002. *Introduction to Remote Sensing*. New York London: The Guilford Press
- Azriel Rosenfeld, 1969. *Picture Processing by Computer*, New York: Academic Press.

- Umbaugh, Scott E. 2010. *Digital image processing and analysis : human and computer vision applications with CVPtools* (2nd ed.). Boca Raton, FL: CRC Press
- John Canny, 1986. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. PAMI-8, no. 6, pp 679 – 698.
- World Wide Web Consortium, 2000. *Techniques For Accessibility Evaluation And Repair Tools* diakses melalui <https://www.w3.org/TR/AERT#color-contrast> [diakses pada 19 November 2016]
- Underwater Research Group, 2011. *Ground Truth for Image Processing* diakses melalui <http://urrg.eng.usm.my/index.php?view=article&id=449:ground-truth-for-image-processing> [Diakses pada 21 April 2017]