

SEGMENTASI CITRA REMOTE SENSING LAUT DENGAN METODE CLUSTERING DBSCAN

I Made Suwija Putra

Magister Teknik Elektro, Program Pascasarjana, Universitas Udayana
Kampus Sudirman, Denpasar, Bali

*Email: putrasuwija@gmail.com

Abstrak

Kebutuhan terhadap citra jarak jauh di bidang kelautan ini sangat diperlukan terutama dalam bidang pemetaan wilayah zona kedalaman laut ataupun bidang keteknikan yang lain. Oleh karena itu dibutuhkan sistem yang dapat menentukan wilayah zona kedalaman laut melalui citra digital berupa citra penginderaan jauh (*remote sensing*) yang tentunya diharapkan lebih efisien dari cara konvensional yang sudah ada untuk bisa memetakan zona wilayah kelautan karena hanya membutuhkan sebuah citra input dengan objek laut dari penginderaan jarak jauh. Penelitian ini melakukan identifikasi zona kedalaman laut dari citra menggunakan metode DBSCAN (*Density-Based Spatial Clustering of Applications with Noise*). Metode DBSCAN diberikan proses pra pengolahan citra baik itu penyesuaian ukuran citra dan kuantisasi warna citra sebelum dilakukan proses DBSCAN agar proses metode DBSCAN bisa bekerja lebih efisien. Proses DBSCAN nantinya akan menghasilkan zona kedalaman laut yang sesuai dengan citra yang ingin diketahui tanpa harus menentukan jumlah *cluster* terlebih dahulu atau zona kedalaman laut yang ada pada citra yang diujikan. Tingkat keberhasilan metode DBSCAN yang diberikan proses pra pengolahan berupa kuantisasi warna ini berhasil memberikan informasi mengenai zona kedalaman laut sesuai dengan citra *ocean* yang ingin diketahui dengan rata-rata persentase keberhasilan sebesar 96,3 persen dari beberapa percobaan citra *input* yang telah dilakukan pada penelitian ini.

Kata Kunci : DBSCAN, laut, segmentasi, kuantisasi warna, *cluster*

1. PENDAHULUAN

Segmentasi citra adalah salah satu permasalahan dasar pada dunia komputer. Segmentasi citra (*image segmentation*) mempunyai arti membagi suatu citra menjadi wilayah-wilayah yang homogen berdasarkan kriteria keserupaan yang tertentu antara tingkat keabuan suatu *pixel* dengan tingkat keabuan *pixel-pixel* tetangganya, kemudian hasil dari proses segmentasi ini akan digunakan untuk proses tingkat tinggi lebih lanjut yang dapat dilakukan terhadap suatu citra, misalnya proses klasifikasi citra dan proses identifikasi objek [1]. Segmentasi merupakan teknik untuk membagi suatu citra menjadi beberapa daerah (*region*) di mana setiap daerah memiliki kemiripan atribut [2]. Metode segmentasi citra dapat diklasifikasikan ke dalam empat pendekatan (metode), salah satunya adalah segmentasi dengan teknik berbasis *region*. Pada segmentasi citra yang menggunakan teknik berbasis *region*, *pixel* yang berdekatan pada daerah yang sama memiliki fitur visual yang sama seperti level keabuan, nilai warna, atau teksturnya. Teknik berbasis *region* sangat tergantung pada pemilihan kriteria homogeneitas, daripada harus menentukan parameter homogeneitasnya, teknik *Seeded Region Growing* (SRG) diatur oleh sejumlah *seed*. Jika diketahui *seed*, SRG akan mencoba menemukan segmentasi citra yang akurat ke dalam daerah dengan properti dimana setiap komponen daerah yang saling berhubungan

bertemu dengan salah satu *seed*-nya [3]. Teknik ini yang sebagai dasar dari teknik segmentasi *region*.

Zona kedalaman laut merupakan wilayah-wilayah laut yang dibedakan berdasarkan tingkat kedalaman yang mewakilinya [4]. Ini sangat penting untuk diketahui sebagai informasi bahari dalam suatu negara terlebih lagi salah satu negara itu mempunyai luas perairan lebih banyak dari luas daratan. Informasi wilayah zona kedalaman laut tentunya sangat berguna sebagai potret dari keadaan geografis dari perairan suatu negara untuk yang nantinya bisa dipakai sebagai bahan penunjang untuk kebijakan dalam peningkatan pengembangan potensi sektor kelautan seperti perikanan, perhubungan laut dan pertambangan. Pada saat ini untuk menentukan wilayah zona kedalaman laut menggunakan metode pengukuran konvensional yaitu teknik bandul timah hitam dan *echo sounder*. Teknik ini bisa menghasilkan data kedalaman laut yang nantinya data tersebut membutuhkan proses selanjutnya yaitu pengumpulan dan pemetaan sesuai dengan ukuran-ukuran kedalamannya sehingga bisa dibuatkan zona kedalaman laut berdasarkan rentang ukuran kedalaman yang dipetakan sebelumnya. Kedua teknik tersebut membutuhkan proses yang panjang untuk sampai bisa menghasilkan sebuah pemetaan zona kedalaman laut dikarenakan perlunya banyak peralatan yang dibutuhkan dan banyak melakukan pengukuran untuk bisa memetakan seluruh wilayah laut untuk dapat menghasilkan informasi zona

kedalaman laut. Oleh karena itu dibuatlah sistem yang dapat menentukan wilayah zona kedalaman laut melalui citra digital berupa citra penginderaan jauh (*remote sensing*) yang diharapkan bisa lebih efisien karena hanya membutuhkan sebuah citra *input* dengan objek laut dari penginderaan jarak jauh selain itu juga bisa dijadikan alternatif informasi wilayah zona kedalaman laut selain informasi yang sudah ada sebelumnya.

Metode *DBSCAN* yang dikembangkan berdasarkan algoritma *density* (kepadatan) yaitu menumbuhkan area-area dengan kepadatan yang cukup tinggi ke dalam *cluster-cluster* dan menemukan *cluster-cluster* dalam bentuk yang sembarang dalam suatu *database spatial* yang memuat *noise*. Metode ini dipakai dikarenakan *DBSCAN* bisa mengatasi kelemahan dari metode segmentasi citra yang berbasis *region* yaitu *region growing*, sehingga nantinya diharapkan dapat menentukan zona kedalaman laut yang cepat dan tepat sesuai dengan citra laut yang diujikan. Metode ini bisa membedakan data yang termasuk *noise* dan yang termasuk dalam satu *cluster* yang sama. Metode ini menganggap *cluster* sebagai suatu area yang berisi objek yang padat atau sesak, yang dipisahkan oleh area yang memiliki kepadatan rendah (merepresentasikan *noise*). Semua objek yang tidak masuk ke dalam *cluster* manapun dianggap sebagai *noise*. *DBSCAN* bisa berkerja sesuai dengan nilai dari 2 parameter input awal yaitu *MinPts* adalah minimal banyak items dalam suatu *cluster* dan *Eps* adalah nilai untuk jarak antar item [5].

Metode *DBSCAN* nantinya akan diberikan proses kuantisasi warna. Proses kuantisasi warna ini sangat penting untuk menampilkan gambar dengan banyak warna pada perangkat yang hanya dapat menampilkan jumlah warna terbatas, biasanya karena keterbatasan memori, dan memungkinkan kompresi secara efisien untuk jenis citra yang akan diujikan yaitu citra *ocean* dari *database Ocean Data View*.

Penelitian ini untuk dapat menguji apakah metode *DBSCAN* cukup handal untuk proses penentuan zona kedalaman laut dan informasi yang diberikan dengan metode *DBSCAN* yang diberi dengan pra pengolahan berupa kuantisasi warna mendekati sempurna dari keadaan di dunia nyata dengan input parameter nilai *Eps* dan *MinPts* yang telah ditentukan yaitu masing-masing 100 dan 100.

2 DENSITY-BASED SPATIAL CLUSTERING OF APPLICATIONS WITH NOISE (DBSCAN)

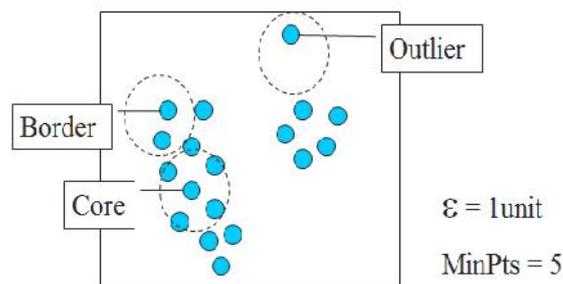
DBSCAN merupakan algoritma menumbuhkan area-area dengan kepadatan yang cukup tinggi ke dalam *cluster-cluster* and menemukan *cluster-cluster* dalam bentuk yang sembarang dalam suatu *database spatial* yang memuat *noise* [6]. *DBSCAN* mendefinisikan *cluster* sebagai himpunan maksimum dari titik-titik kepadatan yang terkoneksi (*density-*

connected). Semua objek yang tidak masuk ke dalam *cluster* manapun dianggap sebagai *noise*.

DBSCAN menentukan sendiri jumlah *cluster* yang akan dihasilkan sehingga kita tidak perlu lagi untuk menentukan jumlah *cluster* yang diinginkan [7], tetapi memerlukan 2 *input* lain, yaitu:

- MinPts*: minimal banyak items dalam suatu *cluster*
- Eps*: nilai untuk jarak antar-items yang menjadi dasar pembentukan *neighborhood* dari suatu titik item.

Neighborhood yang terletak di dalam radius (ϵ) disebut ϵ -*neighborhood* dari objek data. Jika ϵ -*neighborhood* dari suatu objek berisi paling sedikit suatu angka yang minimum, *MinPts* dari suatu objek, objek tersebut disebut *core object*. *Neighborhood* dari *border points* berisi jauh lebih sedikit *items* daripada *neighborhood* dari *core points*. Suatu *border point* bisa jadi termasuk ke dalam lebih dari 1 *core object*. Berikut ini gambar yang menunjukkan mana yang merupakan *border point* dan mana yang merupakan *core point* contoh dengan menggunakan *MinPts*=5 dan *Eps*=1.



Gambar 1. Core dan Border

Menurut definisi, ada 2 jenis titik (*points*) dalam suatu *cluster*: di dalam *cluster* (*core points*) dan di tepian *cluster* (*border points*) di mana *neighborhood* dari *border points* berisi jauh lebih sedikit *items* daripada *neighborhood* dari *core points*. Suatu *border point* bisa jadi termasuk ke dalam lebih dari 1 *cluster*.

DBSCAN menelusuri *cluster-cluster* dengan memeriksa ϵ -*neighborhood* (*Eps-neighborhood*) dari tiap-tiap point dalam *database*. Jika ϵ -*neighborhood* dari point *p* mengandung lebih dari *MinPts*, *cluster* baru dengan *p* sebagai *core object* diciptakan.

Kemudian *DBSCAN* secara iteratif mengumpulkan secara langsung objek-objek *density-reachable* dari *core object* tersebut, dimana mungkin melibatkan penggabungan dari beberapa *cluster-cluster* yang *density-reachable*.

a. Directly density-reachable

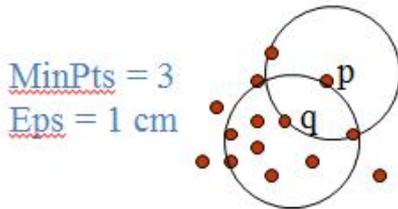
Sebuah titik *item* dikatakan *directly density-reachable* dari titik lainnya jika jarak di antara mereka tidak lebih dari nilai *Eps*. *Directly density-reachable* = titik *q* dikatakan *directly density-*

reachable dari titik p jika titik q adalah $\in Neps(p)$ dan p adalah *core point*.

$$NEps(p): \{q \text{ belongs to } D \mid dist(p,q) \leq Eps \quad (1)$$

Jarak dari titik ke titik lainnya tidak lebih dari nilai *Eps*.

$$Core \ object \ p: |Neps(p)| \geq MinPts \quad \dots\dots\dots (2)$$

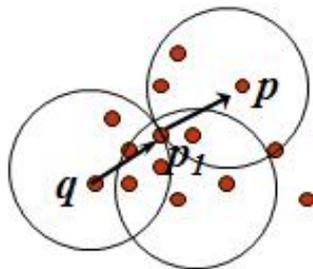


Gambar 2. Directly density-reachable

b. Density-reachable

Sebuah titik *item* dikatakan *density-reachable* dari titik *item* yang lain jika ada suatu rantai yang menghubungkan keduanya yang berisi hanya titik-titik yang *directly density-reachable* dari titik-titik sebelumnya.

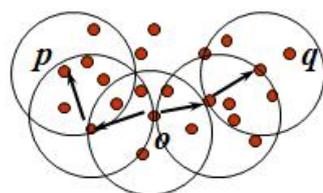
Suatu objek p adalah *density reachable* dari objek q dengan respek ke \in dan *MinPts* dalam suatu set objek D jika terdapat suatu rantai objek p_1, p_2, \dots, p_n , dimana $p_1 = q$ dan $p_n = p$, di mana p_{i+1} *density reachable* secara langsung dari p_i dengan respek ke \in dan *MinPts*, untuk $1 \leq i \leq n$, p_i anggota D.



Gambar 3. Density-reachable

c. Density-connected

Sebuah obyek p adalah *density-connected* terhadap obyek q dengan memperhatikan *Eps* dan *MinPts* dalam set obyek D, jika ada sebuah obyek o elemen D sehingga p dan q keduanya *density-reachable* dari o dengan memperhatikan *Eps* dan *MinPts*.



Gambar 4. Density-connected

2.1 Algoritma DBSCAN

Kunci dari algoritma *DBSCAN* adalah bahwa untuk setiap titik dari sebuah *cluster*, *neighborhood* dari radius yang diberikan harus mengandung setidaknya jumlah minimum poin, yaitu, kepadatan *neighborhood* harus melebihi beberapa *threshold* ditetapkan[8]. Algoritma ini membutuhkan dua parameter masukan:

- a. *Eps*, radius yang menentukan batas daerah *neighborhood* dari titik (*Eps-neighborhood*);
- b. *MinPts*, jumlah minimum poin yang harus ada di *Eps-neighborhood*.

Adapun urutan algoritma dari *DBSCAN* secara umum memiliki 6 langkah yaitu [9]:

1. Pilih point p awal secara acak;
2. Ambil semua *point* yang *density reachable* terhadap titik p;
3. Jika p adalah *core point* maka *cluster* terbentuk;
4. Jika p adalah *border point*, tidak ada yang merupakan hubungan *density-reachable* dari p dan *DBSCAN* akan mengunjungi *point* selanjutnya dari *database*;
5. Lanjutkan proses sampai semua *point* telah diproses. Hasil yang didapatkan tidak tergantung dari urutan dari proses *point* yang diambil.

2.2 Proses Kerja DBSCAN

DBSCAN mencari *region* tersegmentasi pada citra berwarna dengan mengecek jangkauan spasial (*SpatialEps*) dan jangkauan warna (*ColorEps*) dari semua *pixel* pada citra. Jika terdapat sejumlah minimum *pixel* (*MinPts*) yang termasuk dalam jangkauan *pixel* tersebut, baik secara spasial maupun warna, maka sebuah *cluster* baru terbentuk dengan *pixel* tersebut sebagai intinya (*core pixel*) [10]. Kemudian *DBSCAN* secara iteratif akan mengumpulkan *pixel* yang terkoneksi secara spasial dan memiliki kesamaan warna dengan *pixel* inti. Proses berhenti ketika tidak ada lagi *pixel* yang dapat ditambahkan ke dalam *cluster*.

Metode segmentasi citra berwarna dengan *DBSCAN* tidak akan membentuk *region* tersegmentasi dari *pixel* yang memiliki kesamaan warna namun secara signifikan terpisah dalam spasial. Metode *DBSCAN* dirancang di tiga dimensi warna yaitu di masing-masing RGB.

Berdasarkan definisi dan spesifikasi di atas, dapat dilakukan proses segmentasi dengan algoritma sebagai berikut.

- a. Cari *pixel* yang tidak berlabel dalam citra dan membuat *pixel* inti saat ini dan wilayah inti saat ini. *order* adalah dari sudut kiri atas ke bagian bawah sudut citra.
- b. Jika p *pixel* inti ditemukan, sebuah *cluster* baru dibuat. Kemudian, secara iterasi mengumpulkan *pixel* yang tidak berlabel yang merupakan *density-connected* dengan p, dan berikan label *pixel* ini dengan label *cluster* yang sama.

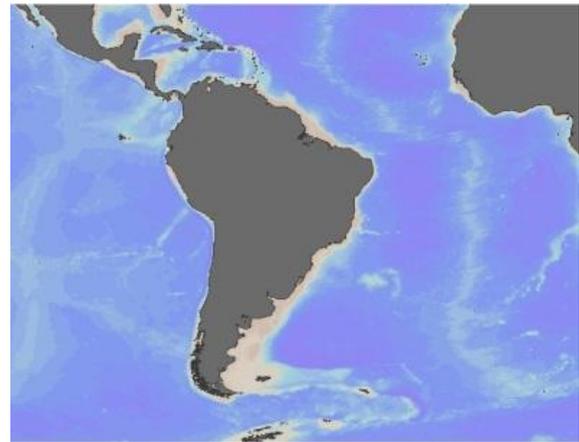
- c. Jika masih ada yang ada *core pixel* dalam citra, kembali ke langkah kedua.
- d. Untuk *pixel* yang tidak termasuk dalam *cluster*, menggabungkan mereka dengan *cluster* yang berbatasan dengan mereka dan memiliki kesamaan nilai warna rata-rata tertinggi dengan mereka.
- e. Beri label setiap *cluster* yang ditemukan dalam citra sebagai *region* segmentasi. Setelah proses segmentasi di atas, *pixel* diletakkan ke dalam *cluster* yang berbeda dan bentuk yang berbeda dari daerah yang tersegmentasi dari suatu citra.
- f. Setiap *cluster* yang terbentuk akan dilakukan pencocokan dengan *cluster* acuan kedalaman laut yang telah ditetapkan di awal.

Penentuan parameter *SpatialEps* dan *ColorEps* akan ditentukan untuk citra.

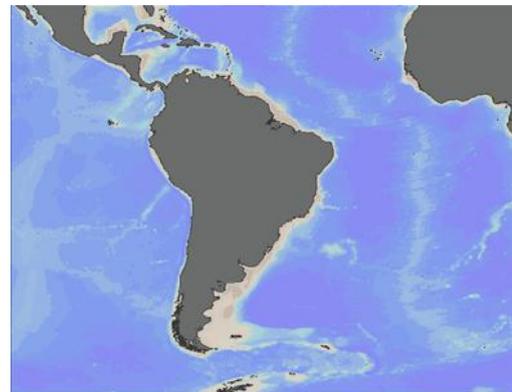
3 PROSES KUANTISASI WARNA

Metode *DBSCAN* menggunakan perbedaan nilai RGB (*Red, Green, Blue*) untuk memisahkan bagian-bagian citra yang akan diuji. Langkah proses awal untuk segmentasi *clustering DBSCAN* adalah ketika citra uji diujikan ke sistem maka akan dilakukan pra pengolahan citra awal yakni mengubah citra menjadi ukuran (*resize*) menjadi citra dengan lebar (*width*) 400 *pixel* dan tinggi (*height*) yang menyesuaikan dengan resolusi citra sebelumnya. Hal ini bertujuan untuk menyeragamkan ukuran citra input [11]. Citra uji diperoleh dari memproses data uji yang berasal dari *ocean database* dari aplikasi *Ocean Data View* yang merupakan aplikasi *Open Source* [12]. Data uji yang dipergunakan adalah sejumlah data dalam bentuk *image* atau citra yang diperoleh dengan cara *men-capture* citra yang terdapat pada *software Ocean Data View* yang merupakan perangkat lunak yang dibuat oleh R. Schlitzer berfungsi untuk menampilkan hasil eksplorasi dari *oceanografi* dan tampilan geo-referensi [13], yang difokuskan kepada objek laut. Data uji didapatkan dengan cara proses *cropping* yang kemudian disimpan dalam bentuk *file* gambar sebagai nantinya sebagai *input* untuk sistem.

Citra uji yang telah siap dalam bentuk *file* gambar ini kemudian dibuka oleh sistem guna untuk diproses pada tahap selanjutnya. dari proses utama dari sistem yaitu proses *DBSCAN*. Masing-masing metode ini menghasilkan catatan waktu untuk menghasilkan *output* berupa informasi zona kedalaman laut yang berbeda.



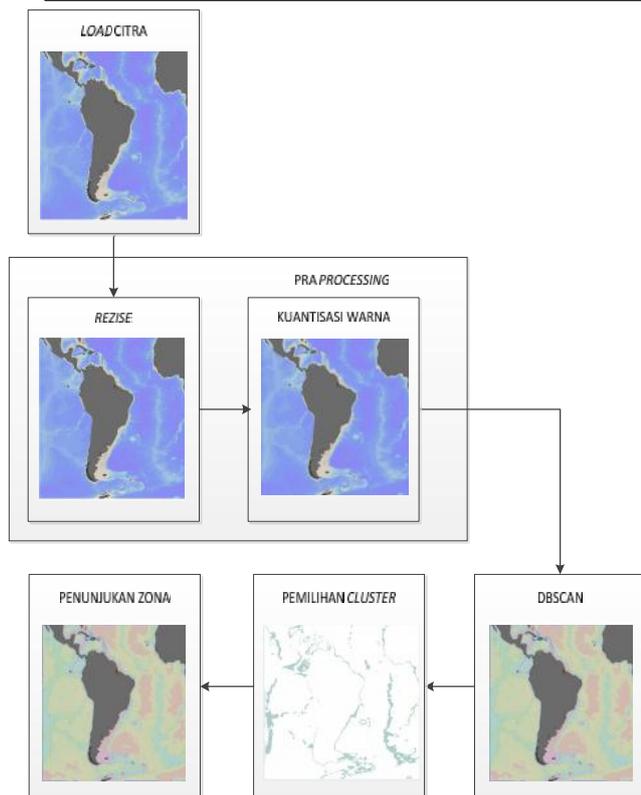
(a)



(b)

Gambar 5. Contoh citra sebelum(a) dan sesudah(b) kuantisasi

Tahapan selanjutnya adalah melakukan normalisasi warna atau dinamakan *Color Quantization* berguna untuk menampilkan gambar dengan banyak warna pada perangkat yang hanya dapat menampilkan jumlah warna terbatas, biasanya karena keterbatasan memori, dan memungkinkan kompresi secara efisien untuk jenis gambar tertentu. yang dimana setiap citra yang dipilih harus melakukan *Color Quantization* sebelum lanjut pada proses tahapan utama yaitu segmentasi *clustering DBSCAN* yang kemudian akan bisa didapatkan *output* berupa informasi zona kedalaman laut yang terdapat pada citra yang diujikan. Langkah selengkapnya dapat digambarkan dengan *DBSCAN* pada Gambar 6 di bawah ini.



Gambar. 6. Bagan proses segmentasi clustering *DBSCAN* dengan kuantisasi warna

Pemberian label *cluster* dalam bentuk warna dilakukan untuk menampilkan *cluster* yang ingin dilihat dan disajikan dengan warna yang berbeda sesuai dengan tingkat zona kedalaman laut. Pada pemberian label *cluster* sangat bergantung dari tahapan segmentasi pada *DBSCAN* karena setelah proses segmentasi dan *clustering* dilakukan *DBSCAN* maka label bisa diberikan disetiap *region* segmentasi yang terbentuk. *Cluster* zona kedalaman laut yang ditentukan saat ini adalah berjumlah 9 *cluster* zona kedalaman laut berdasarkan warna kedalaman laut yang diwakilkan.

4 HASI DAN PEMBAHASAN

Penelitian menggunakan 10 citra uji dimana masing-masing citra memiliki karakteristik objek citra *ocean* yang berbeda-beda artinya setiap citra terdapat *noise* yang dalam hal ini merupakan sebuah pulau atau daratan pada citra *ocean*.

Citra uji diberikan perlakuan yang sama dalam hal prapengolahan citra sesuai dengan tahapan pada Gambar 6, hasil akhir citra uji yang didapatkan nantinya akan berupa segmentasi *clustering* masing-masing zona kedalaman laut yang ada pada citra uji tersebut. Pada saat proses akhir selesai waktu yang dibutuhkan untuk menghasilkan output citra akan dicatat guna mengetahui dan nantinya akan dibandingkan citra mana yang menghasilkan waktu

yang lebih cepat dan hasil nilai akurasi yang lebih baik.

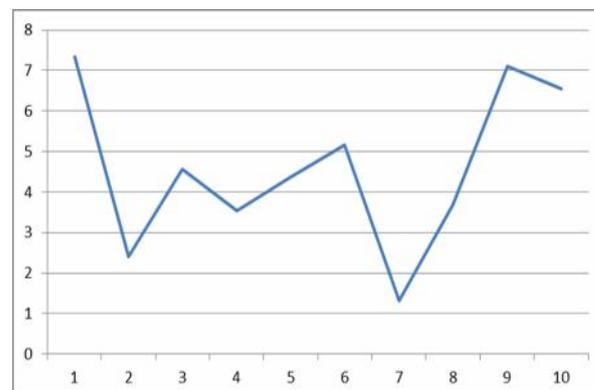
4.1 Pegujian untuk Mendapatkan Kecepatan Waktu Proses Segmentasi *DBSCAN*

Pengujian dilakukan dengan mencatat nilai waktu yang keseluruhan yang dibutuhkan oleh setiap citra uji dalam proses mendapatkan segmentasi *clustering* menggunakan proses *DBSCAN* sesuai dengan Gambar 6. Waktu yang didapatkan terlihat pada aplikasi sistem langsung.

Adapun untuk percobaan ini nilai awal parameter *input* untuk metode *DBSCAN* yang dipakai adalah nilai *Eps*=100 dan nilai *MinPts*=100.

Pengujian dilakukan pada *laptop* atau *notebook* *Toshiba* L635 dengan *processor* *Intel(R) Core(TM) i7 CPU M620 @2.67GHz*, *RAM* 8GB *DDR3*, *Hard Disk* 500 GB, *VGA* *ATI RADEON 512 MB*.

Hasil perbandingan waktu 10 citra uji dapat dilihat pada gambar 7 berikut. Pada gambar 7 menunjukkan bahwa untuk menghasilkan informasi segmentasi zona kedalaman laut membutuhkan waktu yang berbeda-beda. Perbedaan ini ditentukan dari banyaknya variasi objek yang terdapat dalam citra uji. Semakin banyak variasi objek, maka semakin banyak waktu yang dibutuhkan untuk melakukan keseluruhan segmentasi citra uji begitu juga dengan sebaliknya semakin sedikit variasi objek yang diberikan maka semakin cepat proses segmentasi selesai dilakukan. Ini dikarenakan variasi objek yang berbeda akan memberikan proses pada *DBSCAN* lebih banyak karena untuk mengetahui apakah salah satu objek itu *noise* atau *cluster*, *DBSCAN* akan mengecek masing-masing objek yang kemudian dicocokkan dengan masing-masing karakter objek [12]. Jika banyak kemiripan objek dengan *database* zona kedalaman laut maka akan dibentuk satu *cluster*, namun jika tidak terdapat kemiripan karakter dari *database* zona kedalaman laut maka disebut dengan *noise*.



Gambar 7. Grafik kecepatan waktu yang dibutuhkan oleh segmentasi masing-masing citra uji dengan metode *DBSCAN*

4.2 Pegujian untuk Mendapatkan Akurasi Proses Segementasi DBSCAN

Pengujian dilakukan dengan menggunakan metode observatif atau pengamatan dengan menggunakan data dari *database Ocean Data View* sebagai acuan untuk mengetahui apakah hasil *cluster* segmentasi yang terbentuk sudah sesuai dengan zona kedalaman laut yang terbentuk terhadap warna yang diwakilkan untuk dilihat kecocokannya dengan *database Ocean Data View*. Hubungan kecocokan tersebut dapat dilihat pada Tabel 1 di bawah ini.

Tabel 1: Relasi zona kedalaman laut antara *database Ocean Data View* dengan warna cluster segmentasi zona kedalaman laut yang terbentuk oleh metode DBSCAN

Zona Kedalaman Laut	Database Ocean Data View	Warna Clustering
Zona Literal atau pesisir		
Zona Neritik		
Zona Epipelagik		
Zona Mesopelagik		
Zona Batiopelagik		
Zona Abisalpelagik		
Zona Hadalpelagik		
Zona Oseanik		
Zona Oseanik Sangat Dalam		

Akurasi yang dimaksud disini meliputi ketepatan hasil *cluster* segmentasi yang terbentuk oleh masing-masing kuantisasi warna yang mengoptimasi metode DBSCAN yang disesuaikan dengan acuan *database Ocean Data View* sehingga *cluster* yang terbentuk sudah bisa mewakili zona kedalaman laut yang sesuai dan indikator akurasi yang lain adalah kemampuan dari pra pengolahan untuk mengoptimasi DBSCAN agar bisa membedakan *noise* dari hasil segmentasi yang terbentuk. *Noise* yang dimaksudkan adalah objek lain yang bukan merupakan objek laut dalam citra yang diujikan seperti contohnya adalah pulau atau daratan.

Sepuluh citra uji juga akan dinilai keberhasilan mendapatkan hasil berupa zona kedalaman laut yang sesuai dengan yang telah ditentukan pada Gambar 7. Keberhasilan ini dilihat dari masih adanya atau tidak *noise* yang mengikuti tiap-tiap *cluster*. Hasil penelitian ini yang menggunakan citra uji dari *Ocean DataView* dengan menggunakan metode DBSCAN yang diberikan proses pra pengolahan berupa kuantisasi warna dapat dilihat pada Tabel 2 berikut. Dalam tabel 2 berisikan hasil dari pengujian akurasi segmentasi clustering DBSCAN dengan menggunakan optimasi kuantisasi warna. Masing-masing proses menggunakan 10 citra uji yang sama dengan memiliki karakteristik objek di setiap citra yang berbeda-beda.

Penilaian disini menggunakan sistem penilaian dengan besaran persen antara 0% sebagai nilai terendah dan 100% sebagai nilai tertinggi untuk masing-masing indikator.

Table 2: Presentase Keberhasilan 10 Citra Uji

Citra Uji	Persentase Keberhasilan Segmentasi (%)
Citra 1	99.20
Citra 2	81.25
Citra 3	98.58
Citra 4	98.26
Citra 5	98.72
Citra 6	98.87
Citra 7	90.16
Citra 8	99.34
Citra 9	99.46
Citra 10	99.38
Rata-rata	96.32

Proses keberhasilan akurasi ini menerapkan proses operasi pencocokan nilai *pixel* citra sebelum

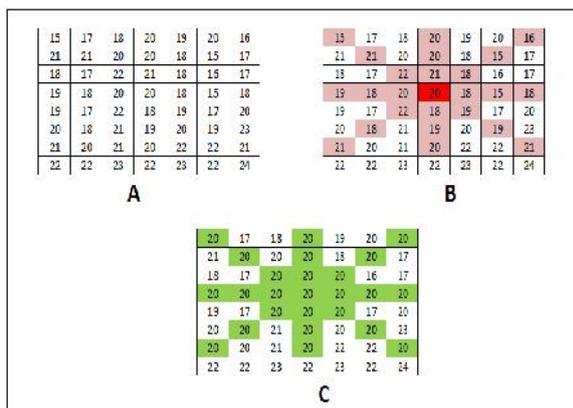
dan sesudah citra input diproses atau dengan kata lain membandingkan nilai pixel yang memiliki koordinat sama sebelum maupun sesudah segmentasi DBSCAN, dimana citra hasil segmentasi akan dilakukan operasi pengurangan terhadap citra asli. Jika hasil pengurangan nilai pixel itu adalah sama dengan 0 atau dengan kata lain tidak ada perubahan nilai pixel baik itu sebelum ataupun sesudah segmentasi maka cluster tersebut merupakan pixel yang tidak bisa tersegmentasi atau noise kecuali pixel itu merupakan core point, sedangkan jika hasilnya ada beberapa selisih maka merupakan pixel yang berhasil tersegmentasi yang cocok dengan zona kedalaman laut pada citra ocean sehingga pixel dalam cluster tersebut direpresentasikan dengan rentang warna yang sesuai dengan warna kedalaman citra tersebut yang telah di-setting pada sistem.

Hal ini dilakukan untuk setiap cluster lainnya, hingga didapatkan pixel mana yang tidak menghasilkan nilai 0, maka pixel tersebut tidak direpresentasikan putih. Pixel yang muncul ini merupakan noise yang salah atau pixel yang tidak mendapatkan cluster.

Pixel tersebut kemudian akan dihitung berapa jumlah pixel yang masih ada, guna untuk mendapatkan presentase keberhasilan akurasi. Rumus presentase akurasi tersebut adalah

$$Keberhasilan = \left(\frac{\text{Jumlah pixel yang tersegmentasi}}{\text{Jumlah pixel keseluruhan}} \right) \times 100\% \dots\dots (3)$$

Persamaan (3) di atas untuk variable pixel yang tersegmentasi tersebut didapatkan secara otomatis oleh sistem yang telah bisa menentukan segmentasi mana yang pixel mana bisa tersegmentasi, berikut ini adalah simulasi dari cara sistem menentukan pixel mana yang tersegmentasi terlihat pada Gambar 8 di bawah ini.



Gambar 8. Contoh simulasi cara kerja persentase keberhasilan segmentasi dengan metode DBSCAN

Gambar 8 A, B dan C adalah salah satu contoh yang bisa mengilustrasikan cara sistem untuk mendapatkan nilai akurasi yang direpresentasikan dengan kotak-kotak dengan berisi nilai setiap pixel, sebagai contoh mengitung nilai akurasi dimulai dengan menentukan point awal secara acak dalam hal ini dipakai nilai pixel di tengah-tengah citra sebagai

titik point atau pixel dengan highlight merah (nilai = 20), kemudian daerah mencari keanggotaannya hanya dilakukan secara horizontal dan vertical sehingga terlihat membentuk sebuah satu daerah dengan nilai pixel masing-masing terlihat pada Gambar 8.A

Gambar 8 menggunakan 2 input parameter utama sebagai langkah awal untuk menentukan titik point awal bekerja yaitu Eps = 3 dan MinPts = 9, Eps= 3, artinya dalam jarak nilai selisih pixel sebanyak 3 pixel untuk bisa dibentuk satu kesatuan yang bernama cluster harus memenuhi syarat yaitu adanya keanggotaan pixel yang density reachable dengan pixel tetangganya dengan bias mengikat keanggotaan sebanyak minimal 9 (MinPts=9) keanggotaan pixel untuk bisa membentuk 1 cluster.

Gambar 8 A sesuai dengan algoritma dari DBSCAN memilih titik point awal secara acak, dalam hal contoh ini adalah titik pixel di tengah-tengah citra yaitu pixel yang memiliki nilai 20. Dari titik pusat ini kemudian akan dicari pixel yang memiliki yang masuk kriteria dengan jarak 3 pixel dan memiliki density reachable ke semua arah yang berdekatan dengan pixel titik pusat. Hasilnya terdapat pada Gambar 8 B, dimana dimisalkan didapatkan titik yang telah memenuhi syarat yang diberi tanda warna hijau seperti pada Gambar 8 B, hasil tersebut sudah bisa dijadikan 1 cluster dikarenakan sudah memenuhi syarat minimal keanggotaan yaitu MinPts = 9, sehingga pixel tersebut yang telah menjadi satu cluster dirubah menjadi nilai yang sama yaitu nilai yang telah ditentukan untuk mewakili cluster, dalam contoh ini digunakan nilai=20, seperti yang terlihat pada Gambar 8 C yang selanjutnya itu bernama border point.

Core point telah mendapatkan border point yang sesuai inilah yang merupakan 1 cluster yang selanjutnya pixel yang lain di luar border point yang bukan merupakan keanggotaan cluster merupakan noise dari citra atau dengan kata lain merupakan pixel yang tidak bisa tersegmentasi.

Sehingga dengan penjabaran tersebut maka untuk mendapatkan nilai akurasi dari sebuah citra ocean yang tersegmentasi dengan metode DBSCAN adalah menggunakan Persamaan (1).

Implementasi Persamaan (1) untuk citra input yang telah tersegmentasi oleh metode DBSCAN yang dioptimasi oleh kuantisasi warna. Dari contoh simulasi pada Gambar 8 dapat dijabarkan menjadi

$$Keberhasilan = \left(\frac{\text{Jumlah pixel yang tersegmentasi}}{\text{Jumlah pixel keseluruhan}} \right) \times 100 \%$$

$$Keberhasilan = \left(\frac{25}{56} \right) \times 100 \%$$

$$Keberhasilan = 44,64 \%$$

Dari hasil perhitungan dengan Persamaan (3) diatas didapatkan hasil pesentase keberhasilan 44,64% hasil ini tidak mencerminkan tingkat akurasi sistem keseluruhan. Hasil keberhasilan ini sangat

berpengaruh terhadap nilai yang diberikan pada 2 input utama metode *DBSCAN* yaitu *Eps* dan *MinPts*, sehingga semakin tepat nilai 2 parameter ini diberikan semakin besar nilai akurasi yang didapatkan.

Contoh cara kerja dengan proses tersebut yang sama dilakukan terhadap semua citra uji yang lainnya dengan menggunakan cara yang sama kemudian hasil rata keberhasilan masing-masing citra uji dihitung nilai rata-rata keberhasilan keseluruhan metode *DBSCAN* dengan diberikan pra pengolahan kuantisasi warna dengan menggunakan rumus rata-rata pada *Persamaan* (4), seperti di bawah ini

$$a = \frac{\sum_{i=1}^n x_i}{n} \dots\dots\dots (4)$$

$$= 96.32\%$$

Keberhasilan segmentasi *clustering* zona *clustering* sesuai dengan data pada Tabel 2 dengan perhitungan pada *Persamaan* (4) di atas memberikan gambaran bahwa metode *DBSCAN* yang diberikan pra pengolahan kuantisasi warna memiliki tingkat keberhasilan yang besar karena dari 10 citra uji didapatkan rata-rata keseluruhan keberhasilan metode *DBSCAN* adalah sebesar 96.32% yang artinya hampir mendekati sempurna dari nilai teratas 100% dalam kemampuan untuk menentukan zona kedalaman laut dari citra *Ocean Data View*.

Tingkat keberhasilan ini menggambarkan segmentasi *DBSCAN* yang dioptimasi dengan kuantisasi warna memberikan segmentasi yang lebih jelas dan berhasilnya metode untuk menghasilkan segmentasi *cluster* yang sesuai dengan zona kedalaman laut yang diwakilkan sesuai dengan warna-warna segmentasi pada Tabel 1.

5. KESIMPULAN

Metode segmentasi *clustering DBSCAN* yang diberikan pra pengolahan kuantisasi warna memiliki kemampuan rata-rata yang cukup baik dengandi buktikan hasil penelitian ini memperoleh rata-rata presentase 96.32% untuk segmentasi dengan karakteristik dari citra *ocean* sehingga bisa menghasilkan informasi berupa zona kedalaman laut dari sebuah citra *remote sensing ocean*. Dengan sifatnya yang berupa metode *clustering* yang dimana bisa langsung menentukan jumlah *cluster* yang akan didapatkan sehingga sangat membantu kinerja yaitu kecepatan dalam mendapatkan informasi yang diinginkan.

6. DAFTAR PUSTAKA

- [1] Rinaldi Munir, " *Pengolahan Citra Digital Dengan Pendekatan Algoritmik*". Penerbit Andi Offset: Yogyakarta 2004.
- [2] Darma Putra, " *Pengolahan Citra Digital*", Penerbit Andi Offset: Yogyakarta, 2009.

- [3] Xu L, Jackowski M., Goshtasby A, " *Segmentation of Skin Cancer Images*", USA, 1999.
- [4] J.W. Hedgpeth, " *Teratise of Marine ecology and Paleocology*". New York; Waverly press, 1957
- [5] Dasu T, Johnson T, " *Exploratory Data Mining and Data Cleaning*" Wiley Pub. Inc., Indianapolis: Indiana, 2003.
- [6] Han, Jiawei & Kamber, Micheline, " *Data Mining – Concepts and Techniques*", Simon Fraser University, USA: Morgan Kaufmann, 2001.
- [7] Ye Qixiang, Gao Wen, Zeng Wei, " *Color Image Segmentation Using Density-Based Clustering*", IEEE-ICASSP, 2003, 345-348.
- [8] Sander J, Ester M, Kriegel H P, Xu X, " *Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and its Applications*, in: *Data Mining and Knowledge Discovery*", an Int. Journal, Kluwer Academic Publishers, Vol. 2, No. 2, 1998, pp. 169-194. Haixun Wang, Carlo.
- [9] Ester M, Kriegel H.-P, Sander J. and Xu X., " *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise*". Proc. 2nd Int. Conf. On Knowledge Discovery and Data Mining. Portland, OR, 1996, 226-231.
- [10] Celebi Emre, Aslandogan Alp, Bergstresser Paul, " *Mining Biomedical Images with Density-based Clustering*", Dept. of Computer Science & Engineering, University of Texas at Arlington, 2003.
- [11] Gonzalez Rafael, Wintz Paul, " *Digital Image Processing*". Addison-Wesley Publishing Company, Inc., 1999.
- [12] *Ocean Data View Home*, [Http://odv.awi.de/](http://odv.awi.de/) diakses tanggal 1 mei 2012
- [13] Moreira Adriano, Santos Maribel Y and Carneiro Sofia, " *Density-based clustering algorithms – DBSCAN and SNN*", University of Minho – Portugal, 2005.