

IMPLEMENTASI *BOUNDARY BASED SEGMENTATION* UNTUK MENGEKSTRASI KONTUR SAPI MADURA

Rosida Vivin Nahari¹, Achmad Jauhari², Riza Alfita³

^{1,2}Teknik Informatika, Fakultas Teknik, Universitas Trunojoyo

³Teknik Elektro, Fakultas Teknik, Universitas Trunojoyo

E-mail: ¹rosida_vn@yahoo.com, ²jauhariaja@gmail.com, ³yogya_001@yahoo.co.id

Abstrak – Pada tahap *preprocessing* penelitian yang bertujuan untuk menganalisa citra digital sapi Madura, sangat bergantung pada keberhasilan dalam mengekstrasi kontur dari objek citra sapi Madura itu sendiri. *Boundary based segmentation* menjadi bagian penting dalam mengekstrasi kontur karena proses ini merupakan langkah awal pemisahan *foreground* objek sapi dengan *background*, sehingga selanjutnya dapat dilakukan analisis dan penentuan berat badan sapi. Pada studi ini dilakukan pemrosesan *Boundary based Segmentation* dilakukan dengan memanfaatkan algoritma Canny untuk mempermudah tahap ekstraksi kontur yang menggunakan *active contour model*. Studi kasus yang digunakan adalah pada citra sapi Madura. Algoritma Canny merupakan salah satu algoritma deteksi tepi modern yang memiliki kelebihan mendeteksi dan melokalisasi dengan baik, serta respon yang jelas. Pemilihan parameter deteksi tepi Canny sangat mempengaruhi hasil dari tepian yang dihasilkan. Parameter-parameter tersebut diantaranya nilai standar deviasi gaussian dan nilai ambang. Algoritma Canny berjalan dengan 5 proses, yaitu, *smoothing*, *finding gradient*, *non-maksimum-suppression*, *double thresholding*, dan *Edge Tracking by hysteresis*. Hasil Algoritma ini menunjukkan tepi yang terdeteksi gambar yang disoroti di mana ada perubahan signifikan pada nilai intensitasnya. Hanya ada satu respon untuk tiap tepi, sehingga mudah dideteksi dan tidak menimbulkan kerancuan pada tahap *active contour model* selanjutnya, sehingga dihasilkan satu perbedaan *boundary* yang signifikan

Kata Kunci — Deteksi Canny, Kontur, *Boundary Segmentation*.

1. PENDAHULUAN

Tahap awal dalam proses segmentasi objek sapi adalah ekstraksi kontur. Ketidaksinambungan intensitas citra sebagai indikator batasan-batasan objek [1], Hal ini dapat saja terjadi disebabkan oleh faktor perubahan refleksi permukaan citra. Hal ini mempengaruhi hasil interpretasi citra yang dilakukan. Melalui pengamatan tersebut, pada penelitian ini *boundary based segmentation* diterapkan bersamadengan *active contour models* sehingga diharapkan mampu meningkatkan efektivitas proses perkiraan awal kontur citra sapi dan pemastian kesinambungan titik-titik kontur dengan baik. Sehingga hasil segmentasi citra sapi yang diharapkan menjadi lebih baik dengan tingkat ketepatan bentuk kontur yang lebih akurat. Sehingga pemanfaatan perangkat citra untuk mendukung penelitian kami dalam proses analisis berat badan sapi, dapat dilakukan lebih maksimal.

Active contour models dikenal dengan istilah “snake” adalah suatu kurva dinamik yang digunakan untuk merunut batas-batas (*boundary*) objek pada citra, merupakan model pendeteksian kontur yang mampu membentuk kontur objek secara akurat. Dengan menerapkan fungsi *energy-minimizing*, diawali dengan penempatan *deformable curve* pada objek, *snake energy* mencari titik pada citra yang memiliki energi paling minimum, hal itu akan membuat kurva tertarik ke titik tersebut. Pada akhir proses, *deformable curve* akan menempati *boundary* dari objek yang diinginkan di dalam citra, sehingga informasi mengenai titik-titik *deformable curve* dapat diketahui.

Snake algorithm merupakan algoritma dasar *active contour models* [3]. Algoritma ini membangkitkan kurva dengan melakukan

perpindahan gerak seperti gerakan ular untuk menemukan *boundary* objek di dalam citra. Dimulai dari titik awal, ular mengubah bentuk (dirinya) sendiri ke dalam penyesuaian dengan bentuk kontur terdekat yang paling mencolok.

2. METODE PENELITIAN

2.1. Ekstrasi Kontur

Rangkaian pixel-pixel tepi yang membentuk batas daerah (region boundary) disebut kontur (contour) [JAI95]. Kontur dapat terbuka atau tertutup. Kontur tertutup berkoresponden dengan batas yang mengelilingi suatu daerah lihat Gambar 9.1(a). Pixel-pixel di dalam daerah dapat ditemukan dengan filling algorithm. Area Boundary berguna untuk mendeskripsikan bentuk objek dalam tahap analisis citra. Sedangkan kontur terbuka dapat berupa fragmen garis atau bagian dari batas daerah yang tidak membentuk sirkuit.

Kontur dapat berupa senarai tepi (edge list) atau berupa kurva. Senarai tepi merupakan himpunan terurut pixel-pixel tepi. Contoh kontur adalah rangkaian pixel tepi yang membentuk garis yang dapat direpresentasikan hanya dengan sebuah persamaan garis lurus. Representasi semacam inimenyederhanakan perhitungan selanjutnya seperti arah dan panjang garis.

Algoritma ekstrasi kontur terdapat dua tahapan utama. Tahap pertama adalah preprocessing untuk mengekstrasi satu set poin berorientasi dari input gambar. Tahap kedua menentukan kontur diantara titik yang berorientasi menggunakan geometri. Tahap kedua tersebut terdiri dari beberapa langkah yaitu titik disaring dengan teknik pengelompokan. Selanjutnya titik dihubungkan berdasarkan kedekatan dan orientasi ke jalur yang mewakili kontur,. Terakhir, jalur yang disederhanakan dengan mengurangi jumlah titik yang mereka miliki [6].

Output : Serangkaian titik boundary dari objek sapi

Input : citra biner yang berisi objek yang sudah tersegmentasi dan direpresentasikan sebagai titik hitam



Gambar 1. Inputan Citra Sapi yang sudah dalam bentuk Binary Image

5 tahap proses ekstrasi kontur citra sapi

1. Preprocessing : menghilangkan derau (*noise*)
2. *Boundary based Segmentation*
3. Deteksi kontur
4. Menghapus titik poin yang redundan
5. Menyederhanakan bentuk kontur

3. HASIL DAN PEMBAHASAN

Kehadiran derau dalam objek penelitian dalam hal ini citra digital tidak bisa dipungkiri. Kehadirannya bisa menyebabkan kesalahan dalam mengestimasi informasi yang terkandung dalam citra tersebut. Derau sebenarnya memiliki variasi yang beragam, namun beberapa jenis derau telah dideskripsikan secara matematis untuk mengetahui “tingkah laku” nya dan bagaimana cara mengatasinya. Tentu bila secara matematis telah dirumuskan, maka kita bisa mengembangkan “anti derau” secara matematis pula. Sebuah citra yang telah tercampur derau akan memiliki persamaan umum, yakni:

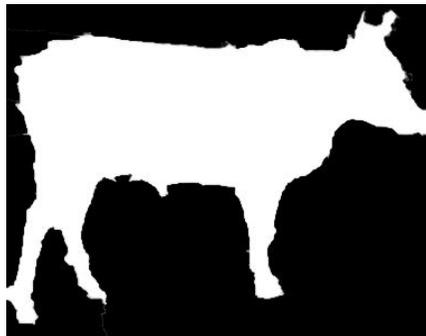
$$g(x,y) = f(x,y) + n(x,y)$$

derau bisa tampak pada kawasan ruang-waktu atau juga dalam kawasan frekuensi. Bila memang sudah tampak dalam kawasan ruang-waktu, maka bisa langsung digunakan tapis untuk menghilangkan derau tersebut. Namun, derau bisa saja hanya tampak pada kawasan frekuensi. Maka perlu diberlakukan alihragam untuk mendeteksi derau tersebut untuk mengetahui jenisnya. Lalu diberlakukanlah tapis yang sesuai. Dalam

kawasan frekuensi citra memiliki bentuk persamaan:

$$G(u,v) = F(u,v) + N(u,v)$$

Derau yang bersifat kontinu dapat dilihat dari sifat periodisitasnya melalui kawasan frekuensi. Tentu saja kita tau dibutuhkan alihragam fourier untuk mengubah kawasan ke frekuensi. Bila derau terbukti periodis, maka yang dilakukan adalah mengurangi derau dari citra untuk mendapatkan estimasi citra asli atau original. Dengan menggunakan spatial filtering, derau bisa di kurangi hanya derau tersebut bersifat additive atau tambahan. Dengan arti lain, bahwa kita memberikan derau tambahan pada citra original lalu dilakukan penapisan dengan menggunakan filter-filter tertentu.



Gambar 2. Citra yang sudah dihilangkan noisanya

Penelitian ini menggunakan *arithmetic mean filter* untuk menghilangkan *noise* pada citra sapi. Filter ini akan menghitung rerata dari citra berderau $g(x,y)$ dalam area yang dikenai oleh jendela. Daerah yang dikenai jendela ini disebut juga sebagai subimage yang dilambangkan dengan S_{xy} . Sehingga diharapkan tidak akan menimbulkan kebingungan ketika simbol ini digunakan dalam menyebutkan subimage dalam artikel ini. Bila dimisalkan bahwa $f(x,y)$ merupakan nilai intensitas setelah di tapis, maka rumus *arithmetic mean filter* adalah:

$$f(x,y) = \frac{1}{mn} \sum_{(s,t) \in S_{xy}} g(s,t)$$

m dan n merupakan baris dan kolom dari sebuah matriks filter. Biasanya matriks

ini berukuran ganjil x ganjil agar ada poros tengah karena hasil perhitungan rerata akan diberikan pada titik tengah dari matriks

Pada contoh ini matriks penapis yang akan digunakan berukuran 3×3 yang semuanya bernilai 1. Matriks ini gunanya untuk mencuplik matriks S_{xy} dari matriks original yang kemudian dioperasikan dengan rumusan *arithmetic mean filter* :

3	6	7
1	2	9
5	8	4

$$\begin{aligned} \text{Maka hasil filternya} &= \frac{1}{(2)(2)} (3 + 6 + 7 + 9 + \\ &\quad 4 + 8 + 5 + 1) \\ &= \frac{1}{4} (43) \\ &= 10.75 \\ &\approx 11 \end{aligned}$$

Filter jenis ini, bisa digunakan untuk menapis citra yang mengandung derau acak seperti *uniform* atau *Gaussian*

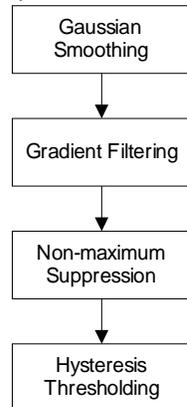
Maka matriks S_{xy} hasil filter adalah :

3	6	7
1	11	9
5	8	4

Pencarian *Boundary* objek sapi pada citra dapat dilakukan dengan cara memanfaatkan proses deteksi tepi (*edge detection*). Algoritma deteksi tepi yang digunakan pada penelitian ini adalah Algoritma deteksi tepi Canny.

Operator Canny, yang dikemukakan oleh John Canny pada tahun 1986, terkenal sebagai operator deteksi tepi yang optimal. Algoritma ini memberikan tingkat kesalahan yang rendah, melokalisasi titik-titik tepi (jarak piksel-piksel tepi yang ditemukan deteksi dan tepi yang sesungguhnya sangat pendek), dan hanya memberikan satu tanggapan untuk satu tepi. Menurut Green (2012), terdapat enam langkah yang

dilakukan untuk mengimplementasikan deteksi tepi Canny.



Gambar 3. Diagram Deteksi Tepi Canny

Algoritma Canny dijabarkan sebagai berikut ini :

Langkah 1

Pertama, dilakukan penipisan terhadap citra dengan tujuan untuk menghilangkan derau. Hal ini dapat dilakukan dengan menggunakan filter Gaussian dengan cadar sederhana. Cadar yang digunakan berukuran jauh lebih kecil daripada ukuran citra.

1 / 115

2	4	5	4	2
4	9	12	9	4
5	12	15	12	5
4	9	12	9	4
2	4	5	4	2

Gambar 4. Cadar untuk Filter Gaussian

Langkah 2

Setelah penghalusan gambar terhadap derau dilakukan, dilakukan proses untuk mendapatkan kekuatan tepi (edge strength). Hal ini dilakukan dengan menggunakan operator Gaussian. Selanjutnya, gradien citra dapat dihitung melalui rumus :

$$|G| = |G_x| + |G_y|$$

Langkah 3

Langkah ketiga berupa penghitungan arah tepi. Rumus yang digunakan untuk keperluan ini :

$$theta = \tan^{-1}(G_y, G_x)$$

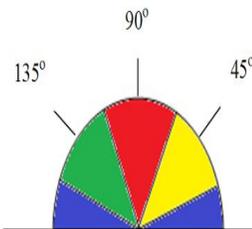
Langkah 4

Setelah arah tepi diperoleh, perlu menghubungkan antara arah tepi dengan sebuah arah yang dapat dilacak dari citra. Sebagai contoh, terdapat susunan piksel berukuran 5 x 5 seperti terlihat pada Gambar 2. Dengan melihat piksel “a” tampak bahwa a hanya memiliki 4 arah berupa 0 derajat, 45 derajat, 90 derajat, dan 135 derajat.

X	X	X	X	X
X	X	X	X	X
X	X	a	X	X
X	X	X	X	X
X	X	X	X	X

Gambar 5 . piksel berukuran 5x5

Selanjutnya, arah tepi yang diperoleh akan dimasukkan ke dalam salah satu kategori dari keempat arah tadi berdasarkan area yang tertera pada Gambar . Berikut adalah aturan konversi yang berlaku



Gambar 6. Area mengkonversi arah tepi ke dalam kategori salah satu arah dari 00, 45, 90, 135 derajat

- Semua arah tepi yang berkisar antara 0 dan 22,5 serta 157,5 dan 180 derajat (warna biru) diubah menjadi 0 derajat.
- Semua arah tepi yang berkisar antara 22,5 dan 67,5 derajat (warna kuning) diubah menjadi 45 derajat.
- Semua arah tepi yang berkisar antara 67,5 dan 112,5 derajat (warna merah) diubah menjadi 90 derajat.

- Semua arah tepi yang berkisar antara 112,5 dan 157,5 derajat (warna hijau) diubah menjadi 135 derajat.

Langkah 5

Setelah arah tepi diperoleh, penghilangan non-maksimum dilaksanakan. Penghilangan non-maksimum dilakukan di sepanjang tepi pada arah tepi dan menghilangkan piksel-piksel (piksel diatur menjadi 0) yang tidak dianggap sebagai tepi. Dengan cara seperti itu, diperoleh tepi yang tipis.

Langkah 6

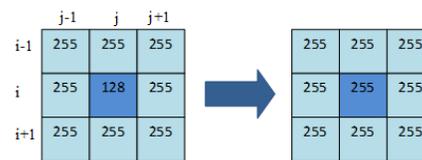
Langkah keenam berupa proses yang disebut *hysteresis*. Proses ini menghilangkan garis-garis yang seperti terputus-putus pada tepi objek. Caranya adalah dengan menggunakan dua ambang T1 dan T2. Lalu, semua piksel citra yang bernilai lebih besar daripada T1 dianggap sebagai piksel tepi. Selanjutnya, semua piksel yang terhubung dengan piksel tersebut dan memiliki nilai lebih besar dari T2 juga dianggap sebagai piksel tepi.

Bagian penting yang perlu dijelaskan adalah penghilangan non-maksimum dan peng-ambangan histeresis. Penghilangan non-maksimum dilakukan dengan mula-mula menyalin isi larik Grad (yang berisi besaran gradien) ke Non_max. Selanjutnya, penghilangan non-maksimum dilaksanakan dengan memperhatikan dua titik tetangga yang terletak pada arah tepi (yang tersimpan dalam Theta). Misalnya, arah tepi adalah 0. Apabila titik yang menjadi perhatian mempunyai koordinat (r, c), dua titik tetangga berupa (r, c-1) dan (r, c+1). Apabila gradien titik perhatian lebih besar daripada gradien kedua tetangga, nilainya akan dipertahankan. Sebaliknya, jika nilai titik perhatian lebih kecil daripada nilai salah satu atau kedua gradien tetangga, nilainya akan diabaikan (diubah menjadi nol).

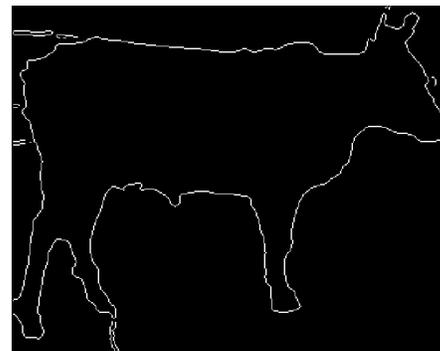
Pengambangan histeresis dilakukan dengan melibatkan dua ambang T1 (ambang bawah) dan ambang T2 (ambang atas). Nilai yang kurang dari T1 akan diubah menjadi hitam (nilai 0) dan nilai yang lebih dari T2 diubah menjadi putih (nilai 255). Lalu, bagaimana nilai yang lebih dari atau sama dengan T1 tetapi kurang dari T2? Oleh karena itu, untuk sementara nilai pada posisi

seperti itu diberi nilai 128, yang menyatakan nilai abu-abu atau belum jelas, akan dijadikan 0 atau 255.

Selanjutnya, dilakukan pengujian untuk mendapatkan kondisi seperti tercantum pada Gambar 5. Apabila kondisi seperti itu terpenuhi, angka 128 diubah menjadi 255. Proses pengujian seperti itu dilakukan sampai tidak ada lagi perubahan dari nilai 128 menjadi 255. Tahap selanjutnya, semua piksel yang bernilai 128 yang tersisa diubah menjadi nol.



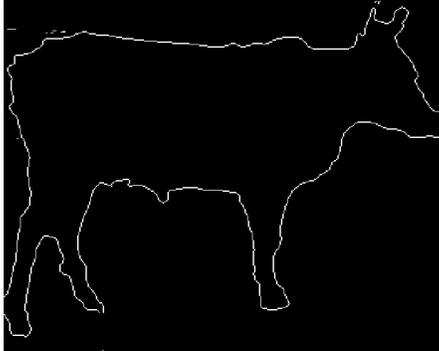
Gambar . Pengujian untuk mengubah nilai 128 menjadi 255



Gambar 7. Citra Hasil deteksi tepi dengan Canny

Setelah proses deteksi tepi, maka untuk mendapatkan letak koordinat dari piksel yang dilabeli dengan graylevel "1" dengan cara mengikuti arah kontur dari piksel yang bergray level "1". Langkah awal yang dilakukan adalah menentukan *starting point*. Kemudian menentukan arah pergerakan dengan syarat harus bisa menghindari alur yang melingkar. Cara untuk menghindari alur yang melingkar adalah dengan memanfaatkan bentuk matrik dalam mendeteksi searah jam. Serta memanfaatkan teknik pemvektoran. Hasil dari teknik ini masih mengandung beberapa titik yang redundan. Hal ini dapat

dihilangkan dengan tetap mengikuti alur sesuai start dan end poin.



Gambar 8. Hasil Ekstrasi Kontur Sapi Madura

4. SIMPULAN

Berdasarkan penelitian yang sudah dilakukan menghasilkan kesimpulan, yaitu:

1. .Pemilihan parameter nilai standar deviasi gaussian dan nilai ambang deteksi tepi Canny sangat mempengaruhi hasil dari tepian yang dihasilkan.
2. Hasil Algoritma ini menunjukkan tepi yang terdeteksi gambar yang disoroti terdapat perubahan signifikan pada nilai intensitasnya. Hanya ada satu respon untuk tiap tepi, sehingga mudah dideteksi dan tidak menimbulkan kerancuan pada tahap *active contour model* selanjutnya, sehingga dihasilkan satu perbedaan boundary yang signifikan

5. SARAN

Saran-saran untuk untuk penelitian lebih lanjut adalah dapat dilakukan tahap optimasi dalam proses penjelajahan *boundary segmentation* sehingga mempercepat proses komputasi.

DAFTAR PUSTAKA

- [1] S. Vijayarani, M. Vinupriya. 2013. Performance Analysis of Canny and Sobel

Edge Detection Algorithm In Image Mining, IJIRCCCE

- [2] J. A. M. Saif, A. A. M. Alkhubati, dkk, 2013. Image Segmentation Using Edge Detection and Thresholding, 13th IACIT
- [3] M Kass, et all, . 1988. Snakes: Active Contour Models (Int J Comput Vis), h 25.
- [4] Zhuowei, Hu.,(2006), Advances In The Application Of Mathematical Morphology In Spatial Data Processing and Analysis, ISPRS Commission VII Mid-term Symposium "Remote Sensing: From Pixels to Processes", Enschede, the Netherlands..
- [5] Khrisna, Dika Adi, Achmad Hidayatno, and R. Rizal Isnanto. Identifikasi Objek Berdasarkan Bentuk Dan Ukuran. Diss. Jurusan Teknik Elektro Fakultas Teknik, 2011.
- [6] Leroy, Bertrand, Isabelle L. Herlin, and Laurent D. Cohen. 1996."Multi-resolution algorithms for active contour models." ICAOS'96. Springer Berlin Heidelberg, 58-65.