**ESTIMASI BERAT BADAN SAPI OTOMATIS MENGGUNAKAN TEKNIK PENGOLAHAN CITRA**

****

**TUGAS AKHIR**

*Disusun dalam rangka memenuhi salah satu persyaratan*

*Untuk menyelesaikan program Strata-1 Departemen Teknik Informatika*

*Fakultas Teknik Universitas Hasanuddin*

*Makassar*

**Disusun Oleh:**

**MUH. SURYA ALIF UTAMA**

**D421 15 001**

**DEPARTEMEN TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS HASANUDDIN**

**MAKASSAR**

**2019**

# LEMBAR PENGESAHAN

**“ESTIMASI BERAT BADAN SAPI OTOMATIS MENGGUNAKAN TEKNIK PENGOLAHAN CITRA”**

Disusun Oleh:

|  |  |
| --- | --- |
| **MUH. SURYA ALIF UTAMA** | **D421 15 001** |

Skripsi ini telah dipertahankan pada Ujian Akhir Sarjana tanggal 6 Agustus 2019.

Diterima dan disahkan sebagai salah satu syarat memperoleh gelar Sarjana Teknik (S.T) pada Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

Makassar, 6 Agustus 2018

Disetujui Oleh:

|  |  |
| --- | --- |
| Pembimbing I,  Dr. Indrabayu, S.T., M.T., M.Bus.Sys NIP. 19750716 200212 1 004 | Pembimbing II,  Dr.Eng. Intan Sari Areni, S.T., M.T.  NIP. 19750203 200012 2 002 |
| Diterima dan disahkan oleh:  Ketua Departemen S1 Teknik Informatika  Dr. Amil Ahmad Ilham, S.T., M.IT  NIP. 19731010 199802 1 001 | |

# KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa karena berkat rahmat dan karunia-Nya sehingga tugas akhir yang berjudul “ESTIMASI BERAT BADAN SAPI OTOMATIS MENGGUNAKAN TEKNIK PENGOLAHAN CITRA” ini dapat diselesaikan sebagai salah satu syarat dalam menyelesaikan jenjang Strata-1 pada Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

Penulis menyadari bahwa dalam penyusunan dan penulisan laporan tugas akhir ini tidak lepas dari bantuan, bimbingan serta dukungan dari berbagai pihak, dari masa perkuliahan sampai dengan masa penyusunan tugas akhir. Oleh karena itu, penulis dengan senang hati menyampaikan terima kasih kepada:

1. Kedua Orang tua penulis, Bapak H. Taufan Kadir, S.H. dan Ibu Hj. Arny, S.H. yang selalu memberikan dukungan, doa, dan semangat serta selalu sabar dalam mendidik penulis sejak kecil;
2. Bapak Dr. Indrabayu S.T., M.T., M.Bus.Sys., selaku pembimbing I dan Ibu Dr.Eng. Intan Sari Areni, S.T., M.T., selaku pembimbing II yang selalu menyediakan waktu, tenaga, pikiran dan perhatian yang luar biasa untuk mengarahkan penulis dalam penyusunan tugas akhir;
3. Bapak Dr. Amil Ahmad Ilham, ST., M.IT., selaku Ketua Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin atas bimbingannya selama masa perkuliahan penulis;
4. Bapak Ir. Christoforus Yohannes dan Ibu Dr. Ir. Ingrid Nurtanio, M.T. selaku dosen Departemen Teknik Elektro dan Informatika Universitas Hasanuddin atas bimbingan, nasehat dan wejangan terkait perkuliahan dan kehidupan;
5. Para sahabat, teman-teman dan kakak-kakak AIMP *Research Group* FT UH yang telah memberikan begitu banyak bantuan selama penelitian, pengambilan data dan diskusi *progress* penyusunan tugas akhir;
6. Teman-teman Hypervisor FT UH atas dukungan dan semangat yang diberikan selama ini;
7. Para sahabat *awardee* Charoen Pokphand Best Student Appreciation Batch III yang telah senantiasa meluangkan waktu untuk berdiskusi dan memberikan masukan selama penelitian;
8. Segenap Staf Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin yang telah membantu penulis.
9. Orang-orang berpengaruh lainnya yang tanpa sadar telah menjadi inspirasi penulis.

Akhir kata, penulis berharap semoga Allah SWT. berkenan membalas segala kebaikan dari semua pihak yang telah banyak membantu. Semoga Tugas Akhir ini dapat memberikan manfaat bagi pengembangan ilmu. Aamiin.

Wassalam

Makassar, Agustus 2019

# ABSTRAK

Sapi merupakan salah satu hewan dengan nilai jual tinggi yang banyak diternakkan di Indonesia. Berat badan sapi sangat penting diperhatikan karena berat badan merupakan indikator penting dalam menentukan harga jual. Terdapat beberapa cara konvensional yang biasa digunakan untuk mengukur berat badan sapi, yaitu dengan menggunakan timbangan sapi digital yang harganya mahal, pita ukur yang membutuhkan kontak langsung dengan sapi, dan menaksir berat badan sapi yang hanya didasarkan pada intuisi tanpa adanya validasi. Untuk itu dibuat sistem yang dapat mengestimasi berat badan sapi menggunakan teknik pengolahan citra. Terdapat tiga skenario pada penelitian ini yang didasarkan pada jarak pengambilan data, yaitu 1.5 meter, 2 meter, dan 2.5 meter. Rata-rata *Mean Absolute Error* yang dihasilkan pada masing-masing skenario adalah 9.59 kg, 10.90 kg, dan 11.81 kg berturut-turut. Nilai parameter yang mempengaruhi segmentasi adalah rasio Rasio *low threshold* pada *canny edge detection* (RC), Rasio *threshold* pada *distance transform* (RDT), Jumlah iterasi pada operasi morfologi dengan *structuring element* berbentuk *disc* dengan ukuran 5x5 (IC). MAE terkecil pada skenario 1 adalah 6.61 kg menggunakan regresi linear dan 6.1 menggunakan *Support Vector Regression* kernel linear dengan RDT = 0.6, RDT = 0.15, dan IC = 4. Fitur terbaik yang digunakan dalam mengestimasi berat badan sapi adalah panjang badan sapi dalam *pixel* dengan nilai koefisien korelasi Pearson dan kombinasi panjang badan dan luas penampang samping sapi dengan nilai .

**Kata kunci:** berat badan sapi, pengolahan citra, *canny edge detection, distance transform, connected component labeling,* regresi linear, *support vector regression.*

# DAFTAR ISI

[LEMBAR PENGESAHAN ii](#_Toc16201644)

[KATA PENGANTAR iii](#_Toc16201645)

[ABSTRAK v](#_Toc16201646)

[DAFTAR ISI vi](#_Toc16201647)

[DAFTAR GAMBAR x](#_Toc16201648)

[DAFTAR TABEL xiv](#_Toc16201649)

[BAB I PENDAHULUAN 1](#_Toc16201650)

[1.1. Latar Belakang 1](#_Toc16201651)

[1.2. Rumusan Masalah 3](#_Toc16201652)

[1.3. Tujuan Penelitian 3](#_Toc16201653)

[1.4. Manfaat Penelitian 4](#_Toc16201654)

[1.5. Batasan Masalah 4](#_Toc16201655)

[1.6. Metode Penulisan 5](#_Toc16201656)

[1.7. Sistematika Penulisan 5](#_Toc16201657)

[BAB II TINJAUAN PUSTAKA 7](#_Toc16201658)

[2.1. Landasan Teori 7](#_Toc16201659)

[2.1.1. Hewan Ternak Sapi 7](#_Toc16201660)

[2.1.2. Hubungan antara Dimensi Ukuran Tubuh dengan Berat Badan Sapi 8](#_Toc16201661)

[2.1.3. Pengukuran Berat Badan Sapi 8](#_Toc16201662)

[2.1.4. Visi Komputer 13](#_Toc16201663)

[2.1.5. Pengolahan Citra 14](#_Toc16201664)

[2.1.6. Regresi Linear 26](#_Toc16201665)

[2.1.7. *Support Vector Regression* 28](#_Toc16201666)

[2.2. Penelitian Terkait 31](#_Toc16201667)

[2.2.1. *Beef Cattle Weight Determine by Using Digital Image Processing* (Pradana dkk, 2016) 31](#_Toc16201668)

[2.2.2. *Estimation of bodyweight from Body Measurements and Determination of Body Measurements on Limousin Cattle Using Digital Image Analysis* (Ozkaya dkk, 2016) 31](#_Toc16201669)

[2.2.3. Estimasi Berat Karkas Sapi Berdasarkan Segmentasi *K-Means Clustering* dengan Menggunakan Klasifikasi *Multiclass SVM* (Alkautsar dkk, 2016) 32](#_Toc16201670)

[2.2.4. Estimasi Berat Karkas Sapi Berdasarkan Segmentasi *Mean Shift* dengan Klasifikasi *Support Vector Machine Linear* (Henarta dkk, 2016) 32](#_Toc16201671)

[2.2.5. Estimasi Berat Karkas dan Klasifikasi Daging Sapi Berdasarkan Deteksi Tepi *Canny, Discrete Wavelet Transform,* dan *Support Vector Machine* (Renasari dkk, 2017) 33](#_Toc16201672)

[BAB III METODOLOGI PENELITIAN 34](#_Toc16201673)

[3.1. Tahapan Penelitian 34](#_Toc16201674)

[3.2. Waktu dan Lokasi Penelitian 35](#_Toc16201675)

[3.3. Instrumen Penelitian 36](#_Toc16201676)

[3.4. Teknik Pengambilan Data 36](#_Toc16201677)

[3.4.1. Data Citra Sapi 37](#_Toc16201678)

[3.4.2. Data Dimensi Tubuh Sapi 37](#_Toc16201679)

[3.5. Perancangan Sistem 38](#_Toc16201680)

[3.5.1. *Input* *Image* 40](#_Toc16201681)

[3.5.2. *Resize* Citra 41](#_Toc16201682)

[3.5.3. *Grayscaling* 42](#_Toc16201683)

[3.5.4. *Otsu’s Thresholding* 43](#_Toc16201684)

[3.5.5. *Canny Edge Detection* 47](#_Toc16201685)

[3.5.6. *Laplacian of Gaussian* 51](#_Toc16201686)

[3.5.7. *Distance Transform* 52](#_Toc16201687)

[3.5.8. *Connected Component Labeling* 56](#_Toc16201688)

[3.5.9. *Closing Operation* 58](#_Toc16201689)

[3.5.10. *Bounding Box* Berdasarkan *Euclidean Distance* 59](#_Toc16201690)

[3.5.11. *Bounding Box* Berdasarkan Rentang Centroid 62](#_Toc16201691)

[3.5.12. Prediksi Berat Badan Sapi Menggunakan Regresi Linear 63](#_Toc16201692)

[3.5.13. Prediksi Berat Badan Sapi Menggunakan SVR 66](#_Toc16201693)

[3.6. Analisis Kerja Sistem 67](#_Toc16201694)

[BAB IV HASIL DAN PEMBAHASAN 69](#_Toc16201695)

[4.1. Hasil Penelitian 69](#_Toc16201696)

[4.2. Pembahasan 75](#_Toc16201697)

[4.2.1. Hubungan RC dengan MAE 76](#_Toc16201698)

[4.2.2. Hubungan RDT dengan MAE 77](#_Toc16201699)

[4.2.3. Hubungan IC dengan MAE 79](#_Toc16201700)

[4.2.4. Perbandingan Hasil Segmentasi Menggunakan *Canny* dan *Laplacian of Gaussian* 81](#_Toc16201701)

[4.2.5. Perbandingan Hasil *Bounding Box* Menggunakan *Euclidean Distance* dan Rentang *Centroid* 82](#_Toc16201702)

[4.2.6. Perbandingan *Mean Absolute Error* Menggunakan Fitur Panjang, Keliling Penampangdan Luas Badan Samping 83](#_Toc16201703)

[4.2.7. Perbandingan *Mean Absolute Error* Menggunakan *Support Vector Regression* dan Regresi Linear 85](#_Toc16201704)

[BAB V PENUTUP 87](#_Toc16201705)

[DAFTAR PUSTAKA 89](#_Toc16201706)

[LAMPIRAN 92](#_Toc16201707)

# DAFTAR GAMBAR

[**Gambar 2.1.** Timbangan Sapi Manual (Stajnko dkk, 2019) 10](#_Toc16201708)

[**Gambar 2.2.** Timbangan Sapi Digital (Stajnko dkk, 2019) 10](#_Toc16201709)

[**Gambar 2.3.** Rondo Tape (www.hoechstmass.com) 11](#_Toc16201710)

[**Gambar 2.4.** Weigh Tape (bainbridgevet.com.au) 11](#_Toc16201711)

[**Gambar 2.5.** Representasi Citra dalam Matriks 15](#_Toc16201712)

[**Gambar 2.6.** Contoh Matriks Citra RGB (Courtney, 2001) 17](#_Toc16201713)

[**Gambar 2.7.**  Hasil Konversi dengan Lightness Method (Cook, 2009) 18](#_Toc16201714)

[**Gambar 2.8.** Hasil Konversi dengan Average Method (Cook, 2009) 19](#_Toc16201715)

[**Gambar 2.9.** Hasil Konversi dengan Luminosity Method (Cook, 2009) 20](#_Toc16201716)

[**Gambar 2.10.** Jenis-jenis Tepi: (a) Tepi Curam; (b) Tepi Landai; (c) Tepi Curam dengan Derau 22](#_Toc16201717)

[**Gambar 2.11.** Perubahan Citra Sebelum dan Sesudah Penerapan Distance Transform (Fisher dkk, 2003) 23](#_Toc16201718)

[**Gambar 2.12.** Perubahan Matriks Sebelum dan Sesudah Penerapan Distance Transform (Fisher dkk, 2003) 23](#_Toc16201719)

[**Gambar 2.13.** Contoh Operasi Hit pada Dilasi 24](#_Toc16201720)

[**Gambar 2.14.** Contoh Operasi Fit pada Erosi 25](#_Toc16201721)

[**Gambar 2.15.** Perbandingan Hasil Ekstraksi Blob dengan 4-connectivity dan 8-connectivity. 26](#_Toc16201722)

[**Gambar 2.16.** Hasil Persamaan Regresi dari Masing-masing Model (www.scikit-learn.org) 29](#_Toc16201723)

[**Gambar 2.17**. Garis Merah: Boundary Lines; Garis Biru: Hyper Plane; Titik Hijau: Poin Data (www.medium.com) 30](#_Toc16201724)

[**Gambar 2.18.** Ilustrasi Persamaan Hyper Plane, Boundary Line, dan Deviasi (www.medium.com) 31](#_Toc16201725)

[**Gambar 3.1.** Diagram Tahapan Penelitian 34](#_Toc16201755)

[**Gambar 3.2.** (a) Posisi Kamera terhadap Objek Penelitian (b) Salah Satu Citra Hasil Pengambilan Data dari Samping 37](#_Toc16201756)

[**Gambar 3.3.** Batas Pengukuran Panjang Badan Sapi 38](#_Toc16201757)

[**Gambar 3.4.** Gambaran Umum Sistem 39](#_Toc16201758)

[**Gambar 3.5.** Flowchart Perancangan Sistem Secara Keseluruhan 40](#_Toc16201759)

[**Gambar 3.6.** Contoh Citra Masukan 41](#_Toc16201760)

[**Gambar 3.7.** Citra Hasil Resize 42](#_Toc16201761)

[**Gambar 3.8.** Citra Hasil Grayscaling 43](#_Toc16201762)

[**Gambar 3.9.** a) Histogram Citra; b) Citra dengan 6 Derajat Keabuan (Greensted, 2010) 44](#_Toc16201763)

[**Gambar 3.10.** Potongan Histogram dengan Derajat Keabuan pada Sumbu X dan Frekuensi pada Sumbu Y: a) Background; b) Foreground (Greensted, 2010) 45](#_Toc16201764)

[**Gambar 3.11.** Hasil Perhitungan Within Class Variance pada semua nilai (Greensted, 2010) 46](#_Toc16201765)

[**Gambar 3.12.** Pembagian Sudut Gradien Penentu Arah 48](#_Toc16201766)

[**Gambar 3.13.** Non-maximum Suppression 49](#_Toc16201767)

[**Gambar 3.14.** Ilustrasi double thresholding dengan calon tepi pada sumbu X dan nilai threshold pada sumbu Y 50](#_Toc16201768)

[**Gambar 3.15.** Citra Hasil Canny Edge Detection 51](#_Toc16201769)

[**Gambar 3.16.** Citra Hasil Laplacian of Gaussian 52](#_Toc16201770)

[**Gambar 3.17.** Citra Hasil Distance Transform. 54](#_Toc16201771)

[**Gambar 3.18.** Citra Hasil Global Thersholding 55](#_Toc16201772)

[**Gambar 3.19.** Citra Keluaran Connected Component Labeling. 57](#_Toc16201773)

[**Gambar 3.20.** Hasil Pengurangan Derau 58](#_Toc16201774)

[**Gambar 3.21.** Sebelum dan Sesudah Proses Konvolusi pada Operasi Closing 59](#_Toc16201775)

[**Gambar 3.22.** Hasil Operasi Closing Menutup Lubang yang Ada 59](#_Toc16201776)

[**Gambar 3.23.** Bounding Box Tanpa Derau 60](#_Toc16201777)

[**Gambar 3.24.** Bounding Box dengan Min Max Blob 61](#_Toc16201778)

[**Gambar 3.25.** Bounding Box dengan Euclidean Distance 62](#_Toc16201779)

[**Gambar 3.26.** Citra Hasil Bounding Box dengan Rentang Centroid 63](#_Toc16201780)

[**Gambar 3.27.** Representasi Data yang Dapat dan Tidak Dapat Dipisah Secara Linear 66](#_Toc16201781)

[**Gambar 4.1.** Perbandingan Citra Keluaran dengan RC yang Berbeda: a) RC = 0.1; b) RC = 0.3; c) RC = 0.5; d) RC = 0.7 76](#_Toc16201782)

[**Gambar 4.2.** Grafik Hubungan RC dengan MAE 77](#_Toc16201783)

[**Gambar 4.3.** Perbandingan Citra Keluaran dengan RDT yang Berbeda: a) RC = 0.11; b) RC = 0.15; c) RC = 0.17; d) RC = 0.19 78](#_Toc16201784)

[**Gambar 4.4.** Grafik Hubungan RDT dengan MAE 79](#_Toc16201785)

[**Gambar 4.5.** Perbandingan Hasil Bounding Box: a) RDT 0.15; b) RDT 0.17 79](#_Toc16201786)

[**Gambar 4.6.** Grafik Hubungan IC dengan MAE 80](#_Toc16201787)

[**Gambar 4.7.** Citra keluaran dengan IC 5 80](#_Toc16201788)

[**Gambar 4.8.** Perbandingan Hasil Segmentasi dan MAE Canny dan LoG 82](#_Toc16201789)

[**Gambar 4.9.** Hasil Pemberian Rentang yang Teralu Besar 83](#_Toc16201790)

[**Gambar 4.10.** Grafik Perbandingan MAE Terhadap Empat Fitur 84](#_Toc16201791)

[**Gambar 4.11.** Perbandingan Hasil Ekstraksi Keempat Fitur 84](#_Toc16201792)

[**Gambar 4.12.** Grafik Perbandingan Metode Estimasi Berat Badan Sapi 85](#_Toc16201793)

# DAFTAR TABEL

[**Tabel 4.1.** Hasil Perbandingan Rata-rata MAE pada Tiga Skenario 70](#_Toc16201794)

[**Tabel 4.2.** Perhitungan Persamaan Regresi pada Data dengan Jarak 1.5 Meter (RC = 0.6, RDT = 0.15, IC = 4) 70](#_Toc16201795)

[**Tabel 4.3.** Hasil Estimasi Berat Badan Sapi Menggunakan Data Uji 72](#_Toc16201796)

[**Tabel 4.4.** Hasil MAE Data Uji 73](#_Toc16201797)

[**Tabel 4. 5.** Perbandingan Prediksi Regresi Linear dan SVR 73](#_Toc16201798)

# BAB I PENDAHULUAN

# Latar Belakang

Sapi merupakan salah satu hewan dengan nilai jual tinggi yang banyak diternakkan di Indonesia. Bobot badan sapi sangat penting diperhatikan karena pada dasarnya bobot badan merupakan indikator penting dalam memperbesar harga jual sehingga memperbesar pendapatan (Rahmah dkk, 2016). Terdapat beberapa cara konvensional yang biasa digunakan untuk mengukur berat badan sapi. Mulai dari menimbang sapi secara langsung di atas timbangan, menggunakan pita ukur, hingga menaksir berat badan sapi. Menimbang sapi secara langsung tidaklah efisien baik dari segi mobilitas, aksesibilitas, dan afordabilitas. Peternak sapi yang tidak memiliki timbangan hanya melakukan taksiran untuk menentukan berat badan sapi. Apabila peternak belum terbiasa dalam menaksir berat badan sapi, maka bisa saja terjadi kesalahan penaksiran dan menyebabkan kerugian.

Timbangan yang digunakan untuk mengukur berat badan sapi memiliki ukuran yang sangat besar sehingga membutuhkan tenaga ekstra untuk dipindah-pindahkan. Peternak juga harus mengarahkan sapi ke atas timbangan sehingga memiliki kemungkinan terjadinya kecelakaan pada peternak. Selain itu, jika sapi dipaksa untuk naik ke atas timbangan maka dapat membuat sapi stres (Lafesto dan Letik 2017). Harga yang sangat mahal membuat tidak semua peternak dapat membeli timbangan tersebut. Hal ini yang menyebabkan para peternak terutama yang berada di pedesaan tidak menggunakan timbangan untuk mengukur berat badan sapi.

Untuk itulah, maka dalam penelitian ini dibuat sebuah sistem untuk memprediksi berat badan sapi menggunakan teknik pengolahan citra yang otomatis, akurat dan efisien sehingga tidak merugikan konsumen dan peternak.

Beberapa penelitian yang telah dilakukan sebelumnya, mengambil dua citra masukan untuk memprediksi berat badan sapi yaitu citra depan untuk mengambil fitur lingkar dada dan citra samping untuk mengambil fitur panjang badan. Adapun pada penelitian ini, fitur yang digunakan merupakan panjang badan sapi yang diambil dari bagian samping. Fitur ini dipilih karena citra bagian samping sapi dapat lebih mudah diambil oleh pengguna dibandingkan citra bagian depan untuk diambil fitur lingkar dadanya. Untuk mengambil citra bagian depan sapi, maka pengguna sebisa mungkin harus membuat kepala sapi tunduk ke bawah, sehingga bagian lingkar dadanya terlihat. Hal ini dianggap akan menyulitkan pengguna.

Berdasarkan Lafesto dan Letik pada tahun 2017, McNitt pada tahun 1983 telah menemukan adanya hubungan yang linear di antara panjang badan, lingkar dada dan berat badan sapi. Hasil pengukuran fisik sapi dapat menentukan berat badannya sehingga dapat meningkatkan kinerja sistem dalam hal penentuan berat.

Terdapat beberapa penelitian yang telah menerapkan teknik pengolahan citra untuk mengukur berat badan ternak. Zein Hanni Pradana pada tahun 2016 mengambil fitur dari citra depan dan samping yang kemudian akan dimasukkan ke dalam metode regresi linear untuk mengestimasi berat badan sapi. Penelitian ini menghasilkan akurasi sebesar 73,2%. Muhammad Taufiq Alkautsar dkk pada tahun 2016 menggunakan kombinasi *K-Means Clustering* dan *Active Contour Model* untuk ekstraksi fitur lebar dada dan panjang badan. Algoritma *Multiclass* *Support Vector Machine* (SVM) digunakan untuk memprediksi berat badan sapi yang menghasilkan akurasi sebesar 87,53%. Mutia Henarta dkk pada tahun 2016 mengekstrak fitur dari citra samping dan depan menggunakan segmentasi Mean Shift. Penelitian ini menghasilkan akurasi sebesar 89% dengan metode SVM *Linear* untuk klasifikasi. Mona Renasari dkk pada tahun 2017 menggunakan teknik segmentasi yang berbeda yaitu menggunakan *Discrete Wavelet Transform* dan menghasilkan akurasi sebesar 86,1%. Listianto Raharjo dkk pada tahun 2019 menggunakan metode *Deformable Template* pada citra sapi yang menghasilkan akurasi sebesar 76,1905%.

# Rumusan Masalah

Rumusan masalah yang akan diuraikan dalam penelitian ini antara lain:

1. Bagaimana pengaruh nilai *threshold* terhadap hasil segmentasi citra sapi pada lingkungan peternakan?
2. Bagaimana unjuk kerja sistem estimasi berat badan sapi dengan menggunakan teknik pengolahan citra?

# Tujuan Penelitian

Tujuan dari penelitian ini adalah :

1. Untuk menganalisis pengaruh nilai *threshold* terhadap hasil segmentasi citra sapi pada lingkungan peternakan.
2. Untuk mengetahui unjuk kerja sistem estimasi berat badan sapi dengan menggunakan teknik pengolahan citra.

# Manfaat Penelitian

Dengan dilakukannya penelitian ini, diharapkan manfaat yang didapatkan antara lain:

1. Bagi masyarakat baik peternak dan konsumen, penelitian ini dapat digunakan sebagai alat bantu untuk memperkirakan berat badan sapi dengan efisien.
2. Bagi peneliti, penelitian ini dapat digunakan untuk menambah pengetahuan dan kemampuan di bidang visi komputer khususnya dalam memperkirakan berat badan sapi menggunakan teknik pengolahan citra.
3. Bagi institusi pendidikan, penelitian ini dapat digunakan sebagai referensi ilmiah untuk penelitian-penelitian selanjutnya.

# Batasan Masalah

Adapun batasan masalah dalam penelitian ini adalah:

1. Objek penelitian hanya berupa sapi Bali.
2. Fitur citra yang digunakan merupakan fitur yang diekstrak dari citra bagian samping.
3. Citra sapi diambil dari tiga jarak yang berbeda yaitu 1,5 meter, 2 meter, dan 2,5 meter.
4. Pengambilan data dilakukan pada siang hari dengan kondisi cahaya yang cukup.
5. Kamera yang digunakan untuk pengambilan data memiliki resolusi 13 *megapixel*.

# Metode Penulisan

Terdapat beberapa metode yang digunakan pada penelitian ini yaitu:

1. Metode Pengambilan Data

Metode pengambilan data dilakukan dengan pengambilan data objek sapi secara langsung dengan mendatangi Rumah Pemotongan Hewan di Antang.

1. Studi Literatur

Studi literatur dilakukan dengan mengumpulkan informasi yang terkait dengan penelitian ini dari berbagai sumber seperti buku, internet, dan sumber lainnya.

1. Diskusi dan Konsultasi

Diskusi dan konsultasi dilakukan dengan mengadakan tanya jawab secara langsung kepada dosen pembimbing dan pihak-pihak profesional lainnya yang berhubungan dengan penelitian ini.

# Sistematika Penulisan

Laporan penelitian ini dibagi menjadi lima bab yang tersusun secara sistematis sebagai berikut:

1. **PENDAHULUAN**

Bab ini menjelaskan tentang latar belakang diangkatnya judul penelitian estimasi berat badan sapi otomatis menggunakan teknik pengolahan citra, disertai dengan rumusan masalah, tujuan dan manfaat penelitian, batasan masalah, metode penulisan, dan sistematika penulisan.

1. **TINJAUAN PUSTAKA**

Bab ini membahas tentang teori-teori umum yang berkaitan dengan konsep dasar ekstraksi fitur citra dan estimasi berat badan dengan menggunakan teknik pengolahan citra.

1. **METODOLOGI PENELITIAN**

Bab ini memberikan gambaran mengenai perancangan sistem estimasi berat badan sapi dengan teknik pengolahan citra beserta konsep perancangannya.

1. **HASIL DAN PEMBAHASAN**

Bab ini berisi tentang hasil pengolahan data serta pembahasan yang disertai tabel hasil penelitian.

1. **PENUTUP**

Bab ini berisi tentang kesimpulan yang didapatkan berdasarkan hasil penelitian yang telah dilakukan serta saran-saran untuk pengembangan lebih lanjut.

# BAB II TINJAUAN PUSTAKA

# Landasan Teori

## Hewan Ternak Sapi

Sapi merupakan hewan ungu lata besar yang paling sering diternakkan yang termasuk ke dalam anggota suku Bovinae dan anak suku Bovinae. Di Indonesia, sapi merupakan komoditi utama bahan pangan. Adapun pemeliharaan sapi diperuntukkan untuk dimanfaatkan susu dan dagingnya. Hasil sampingan lainnya yang dapat dimanfaatkan adalah kulit, tanduk, jeroan, serta kotorannya. Ternak sapi menghasilkan 50% kebutuhan daging di dunia, 95% kebutuhan susu, dan kulitnya menghasilkan sekitar 85% (Murtidjo 1990).

Di beberapa tempat, sapi juga digunakan sebagai penggerak alat transportasi, pengolah lahan tanaman seperti membajak sawah, dan alat industri lainnya seperti peremas tebu. Dengan banyaknya manfaat dari ternak sapi, sapi telah menjadi bagian dari berbagai kebudayaan sejak lama.

Kebanyakan sapi ternak merupakan keturunan dari jenis liar yang dikenal sebagai Auerochse atau Urochse, yang sudah punah di Eropa sejak 1627. Namun, terdapat beberapa spesies sapi liar lain yang keturunannya didomestikasi, termasuk sapi bali yang juga diternakkan di Indonesia.

## Hubungan antara Dimensi Ukuran Tubuh dengan Berat Badan Sapi

Penelitian mengenai hubungan antara parameter pengukuran tubuh dengan berat badan sapi bukanlah hal yang baru dalam dunia peternakan. Hal ini dikarenakan pentingnya mencari solusi alternatif dalam mengestimasi berat badan sapi tanpa menggunakan timbangan yang kurang ekonomis dan kurang praktis terutama di pedesaan. Padahal berat badan sapi merupakan parameter utama dalam penentuan harga sapi.

Berdasarkan Zurahmah dan The (2011), beberapa penelitian telah melaporkan adanya hubungan antara dimensi ukuran tubuh pada sapi dengan bobot badannya, sehingga menghasilkan suatu formula untuk mengestimasi bobot badan pada umur dan jenis kelamin tertentu (Sumaedi dkk., 2001; Maskyadji, 1997; Clufran, 1976; Saleh, 1982). Francis dkk pada tahun 2002 juga telah membuktikan adanya korelasi yang kuat dan positif antara bobot badan dengan panjang badan (*r* = 0,90) maupun dengan lingkar dada (*r* = 0,96) pada sapi yang diteliti. Nani Zurahmah dan Enos The (2011) melakukan penelitian terhadap sapi Bali yang membuktikan bahwa ukuran lingkar dada dan panjang badan merupakan penduga bobot badan terbaik ( = 76.8%). McNitt (1974) juga menyimpulkan bahwa panjang badan, lingkar dada, dan berat badan dari ternak sapi memiliki hubungan yang linear.

## Pengukuran Berat Badan Sapi

Dalam menentukan berat badan sapi, terdapat berbagai metode yang dapat dilakukan, mulai dari menggunakan timbangan yang menghasilkan akurasi yang sangat tinggi hingga melakukan penaksiran dengan menggunakan pendekatan tertentu.

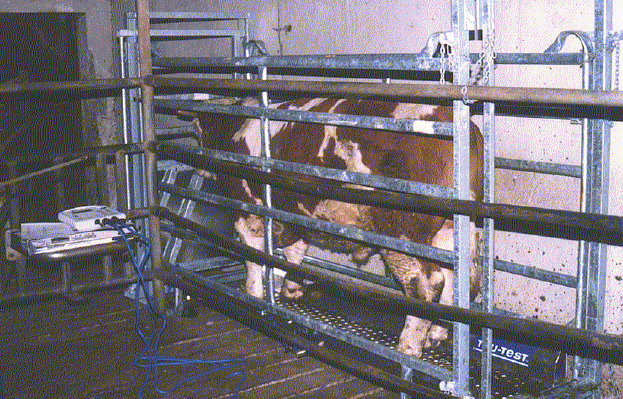
1. Timbangan

Mengukur berat badan sapi menggunakan timbangan merupakan metode yang paling lama digunakan dalam dunia peternakan. Metode ini menghasilkan tingkat akurasi yang sangat tinggi untuk keperluan komersial. Terdapat dua jenis timbangan sapi yang dapat digunakan, timbangan manual seperti pada **gambar 2.1**. maupun timbangan digital seperti pada **gambar 2.2**.

Seperti namanya, timbangan manual memerlukan kalibrasi secara manual ketika sapi berdiri di atasnya dengan tenang kemudian peternak baru dapat membaca berat badan yang dikeluarkan. Sedangkan timbangan digital dapat melakukan kalibrasi dan pembacaan berat badan secara otomatis. Baik timbangan sapi manual dan digital memerlukan peternak untuk mengarahkan sapi satu per satu ke atas timbangan. Hal ini dapat memakan waktu yang banyak dan membuat sapi stres. Terlebih lagi harga timbangan sapi yang relatif mahal sehingga tidak dapat dijangkau oleh seluruh peternak.



**Gambar 2.1.** Timbangan Sapi Manual (Stajnko dkk, 2019)



**Gambar 2.2.** Timbangan Sapi Digital (Stajnko dkk, 2019)

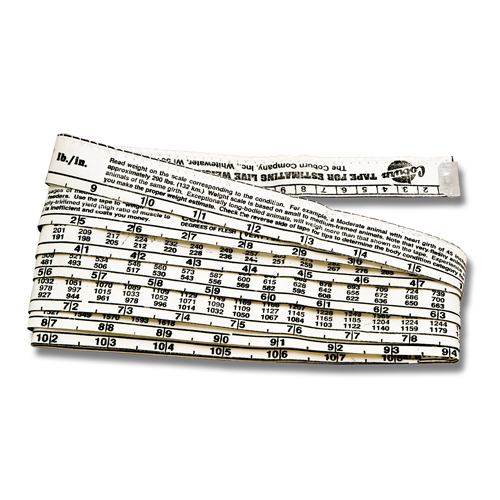
1. Pita

Penggunaan pita (*tape*) khusus merupakan alternatif untuk mengestimasi berat badan sapi. Terdapat dua macam pita yang sering digunakan dalam mengestimasi berat badan sapi, yaitu *rondo tape* seperti yang terlihat pada **gambar 2.3**. dan *weigh tape* seperti yang terlihat pada **gambar 2.4**.

Untuk menggunakan pita-pita ini, maka sapi pertama-tama harus dibuat berdiri dengan posisi normal dengan keempat kaki diatur membentuk persegi di bawah badan. Pita kemudian dilingkarkan dengan kencang tepat di bagian belakang dari sepasang kaki depan sapi pada lingkar dada terkecil. Kemudian, pita tersebut akan menunjukkan estimasi berat badan sapi berdasarkan ukuran lingkar dada sapi yang diukur.

**

**Gambar 2.3.** *Rondo Tape* (www.hoechstmass.com)

**

**Gambar 2.4.** *Weigh Tape* (bainbridgevet.com.au)

1. Formula

Beberapa penelitian telah membenarkan bahwa terdapat hubungan yang linear antara dimensi tubuh sapi dengan berat badannya. Hal ini mendorong penelitian lanjutan lainnya untuk mencari korelasi eksak di antara parameter-parameter ini. Dari penelitian-penelitian ini dihasilkan beberapa formula yang hingga kini masih digunakan dalam mengestimasi berat badan sapi.

Adapun formula-formula tersebut adalah:

1. Schoorl Denmark:

|  |  |
| --- | --- |
|  | (2.1) |

1. Schoorl Indonesia:

|  |  |
| --- | --- |
|  | (2.2) |

1. Winter Eropa:

|  |  |
| --- | --- |
|  | (2.3) |

1. Winter Indonesia:

|  |  |
| --- | --- |
|  | (2.4) |

1. Lambourne:

|  |  |
| --- | --- |
|  | (2.5) |

Keterangan:

BB = Berat Badan

PB = Panjang Badan

LD = Lingkar Dada

Jumlah formula yang dapat digunakan dalam mengestimasi berat badan sapi yang cukup beragam disebabkan karena penelitian-penelitian tersebut menggunakan sapi dengan jenis yang berbeda-beda. Sehingga standar ciri tubuh yang dihasilkan juga berbeda. Beberapa formula bisa memiliki deviasi yang sangat besar karena jenis sapi yang digunakan tidak sama.

## Visi Komputer

Visi komputer (*computer vision*) merupakan ilmu yang mempelajari bagaimana komputer dibuat sedemikian rupa sehingga dapat mengenali objek yang diamati. Visi komputer terdiri dari metode-metode untuk memperoleh, mengolah, dan menganalisis informasi dari satu citra atau kumpulan citra. Objek yang dimaksudkan tidak terbatas pada objek asli, tetapi bisa berupa citra dan video. Pengenalan dilakukan melalui sistem sensor, yang mana kamera merupakan sistem sensor yang paling umum digunakan dalam pengaplikasian visi komputer. Seperti halnya dengan konsep kecerdasan buatan yang meniru konsep kecerdasan manusia, konsep visi komputer juga meniru bagaimana manusia melihat sehingga komputer dapat mengambil informasi yang terdapat dalam citra tersebut.

Visi komputer terdiri dari dua bagian, yaitu:

1. Pengolahan citra (*image processing*): merupakan bagian di mana kualitas citra ditingkatkan.
2. Pengenalan pola (*pattern recognition*): merupakan proses untuk mengelompokkan data masukan baik berupa data numerik maupun citra secara otomatis berdasarkan fitur yang dimiliki.

Aplikasi visi komputer sangat fleksibel sehingga dapat diterapkan pada berbagai bidang, seperti:

1. *Augmented reality*
2. Pengenalan tulisan
3. Pengenalan dan pendeteksian wajah
4. Inspeksi kualitas pada industri
5. Sistem transportasi cerdas
6. Analisis citra medis
7. Keamanan
8. Dan lain sebagainya.

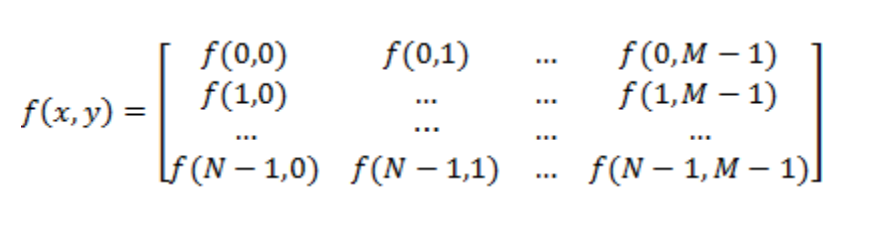
## Pengolahan Citra

Pengolahan citra adalah metode untuk melakukan beberapa operasi terhadap citra. Tujuan dilakukannya pengolahan citra adalah untuk meningkatkan kualitas citra maupun untuk mengekstrak informasi yang penting dari citra tersebut. Di dalam mengolah sebuah citra, terdapat berbagai algoritma yang dapat diterapkan untuk menghasilkan keluaran yang lebih baik. Keluaran yang baik akan mempengaruhi hasil dari proses yang akan dilakukan selanjutnya.

1. **Citra Digital**

Citra merupakan kombinasi antara titik, garis, bidang, dan warna yang mewakili suatu objek atau benda. Sedangkan citra digital merupakan keluaran yang dihasilkan melalui perangkat pencitraan digital seperti kamera dan dapat disimpan oleh komputer digital.

Untuk menerjemahkan citra menjadi angka-angka yang dapat dipahami oleh komputer, maka citra dibagi menjadi bagian-bagian kecil yang disebut sebagai *pixel* (*picture elements*). Dalam setiap *pixel*, perangkat pencitraan merekam sebuah angka ataupun sekumpulan angka yang merepresentasikan beberapa properti pada *pixel* yang dituju, seperti intensitas cahayanya maupun warnanya. Angka-angka ini diatur dalam sebuah baris-baris dan kolom-kolom *array* yang merujuk pada posisi vertikal dan horizontal dari *pixel*-*pixel* yang ada dalam citra.



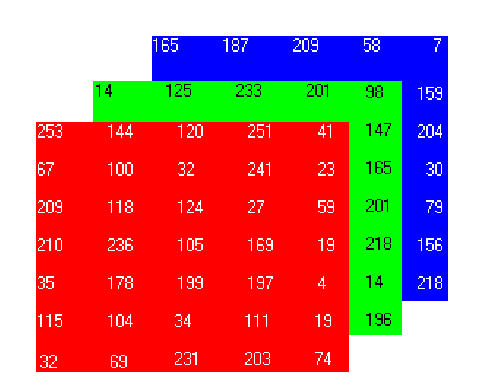
**Gambar 2.5.** Representasi Citra dalam Matriks

Menurut Darmawan dan Saptiani (2010), citra digital mengandung elemen-elemen dasar. Elemen-elemen dasar yang paling penting diuraikan sebagai berikut:

1. Kecerahan (*Brightness*) merupakan intensitas cahaya rata-rata dari suatu area yang melingkupinya.
2. Kontras (*Contrast*) merupakan sebaran terang (*lightness*) dan gelap (*darkness*) di dalam sebuah citra. Citra dengan kontras rendah komposisi citranya sebagian besar terang atau sebagian besar gelap. Citra dengan kontras yang baik, komposisi gelap dan terangnya, tersebar merata.
3. Kontur (*Contour*) merupakan keadaan yang ditimbulkan oleh perubahan intensitas pada *pixel*-*pixel* tetangga, sehingga dapat dideteksi tepi objek di dalam citra.
4. Warna (*Color*) merupakan persepsi yang dirasakan oleh sistem visual manusia terhadap panjang gelombang cahaya (λ) yang dipantulkan oleh objek. Warna-warna yang dapat ditangkap oleh mata manusia merupakan kombinasi cahaya dengan panjang berbeda. Kombinasi yang memberikan rentang warna paling lebar adalah *red* (R), *green* (G) dan *blue* (B).
5. Bentuk (*Shape*) merupakan properti intrinsik dari objek tiga dimensi, dengan pengertian bahwa bentuk merupakan properti intrinsik utama untuk visual manusia. Umumnya citra yang dibentuk oleh manusia merupakan 2D, sedangkan objek yang dilihat adalah 3D.
6. Tekstur (*Texture*) merupakan distribusi spasial dari derajat keabuan di dalam sekumpulan *pixel*-*pixel* yang bertetangga.
7. **Citra Warna (*True Color*)**

Pada citra warna, setiap *pixel* yang terdapat di dalamnya mewakili warna yang merupakan kombinasi dari tiga warna dasar yaitu merah (*red*), hijau (*green*), dan biru (*blue*)*.* Biasanya warna-warna dasar ini disebut sebagai *channel*. Setiap warna dasar menggunakan penyimpanan 8 bit = 1 *byte*, yang berarti setiap warna mempunyai gradasi 255 warna. Berarti setiap *pixel* mempunyai kombinasi warna sebanyak 28. 28. 28 = 224 = 16 juta warna lebih. Itu sebabnya format ini dinamakan *true color* karena mempunyai jumlah warna yang cukup besar sehingga bisa dikatakan hampir mencakup semua warna di alam.

Penyimpanan *true color* di dalam memori berbeda dengan citra *grayscale*. Setiap *pixel* dari citra *grayscale* yang terdiri atas 256 gradasi warna hanya diwakili oleh 1 *byte*. Sedangkan 1 *pixel* citra *true color* diwakili oleh 3 *byte*, di mana masing-masing *byte* merepresentasikan tiap R, G, dan B.

**

**Gambar 2.6.** Contoh Matriks Citra RGB (Courtney, 2001)

1. **Citra *Grayscale***

Sesuai dengan namanya, jenis citra ini memiliki gradasi warna hitam dan putih, yang menghasilkan efek warna abu-abu. Intensitas berkisar antara 0 sampai dengan 255. Nilai 0 menyatakan hitam dan 255 menyatakan putih. Citra RGB dengan matriks penyusun citra yang sebelumnya 3 matriks akan berubah menjadi 1 matriks saja.

Citra *grayscale* merupakan citra yang nilai intensitas *pixel*nya didasarkan pada derajat keabuan. Pada citra *grayscale* 8-bit, derajat warna hitam sampai dengan putih dibagi ke dalam 256 derajat keabuan di mana warna hitam sempurna direpresentasikan dengan nilai 0 dan putih sempurna dengan nilai 255. Citra RGB dapat dikonversi menjadi citra *grayscale* sehingga dihasilkan hanya satu kanal warna. Terdapat tiga metode yang digunakan untuk mengonversi citra RGB ke *grayscale*:

1. *Lightness Method*

Metode ini mengambil rerata pada intensitas warna yang paling mencolok dan paling tidak mencolok dengan persamaan sebagai berikut:

|  |  |
| --- | --- |
|  | (2.6) |



**Gambar 2.7.**  Hasil Konversi dengan *Lightness Method* (Cook, 2009)

1. *Average Method*

Metode rerata (*average method*) merupakan metode yang paling sederhana. Nilai *pixel* dari tiap-tiap *channel* diambil dan dirata-ratakan. Sehingga persamaan dari metode ini yaitu:

|  |  |
| --- | --- |
|  | (2.7) |

**

**Gambar 2.8.** Hasil Konversi dengan *Average Method* (Cook, 2009)

1. *Luminosity Method*

Metode ini memberikan nilai beban-beban tertentu pada *channel-channel* tertentu sesuai dengan kebutuhan pengguna. Manusia lebih sensitif terhadap warna hijau dibandingkan warna-warna lainnya, jadi warna biru biasanya memiliki bobot yang paling besar. Adapun salah satu contoh persamaan dari metode ini adalah:

|  |  |
| --- | --- |
|  | (2.8) |



**Gambar 2.9.** Hasil Konversi dengan *Luminosity Method* (Cook, 2009)

1. ***Thresholding***

*Thresholding* merupakan salah satu teknik dasar dalam melakukan segmentasi citra. *Thresholding* digunakan untuk mengonversi citra *grayscale* menjadi citra biner. Citra biner hanya terdiri dari dua intensitas warna yaitu hitam yang memiliki representasi nilai 0 dan putih yang memiliki representasi nilai 1. Sehingga jenis citra in hanya membutuhkan 1 bit memori untuk menyimpan kedua warna ini.

*Thresholding* secara umum digunakan untuk memisahkan beberapa objek target dari latar belakangnya. Objek target biasanya direpresentasikan dengan warna putih. Adapun algoritma yang digunakan pada *global thresholding* ditunjukkan pada persamaan (2.9).

|  |  |
| --- | --- |
| Untuk semua *pixel* : | (2.9) |

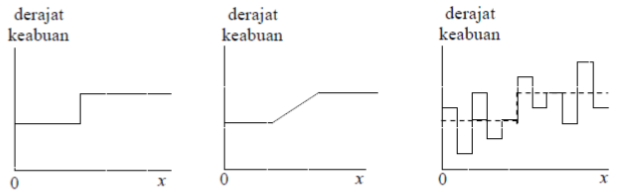
Berdasarkan persamaan (2.9), jika nilai *pixel* pada koordinat lebih besar atau sama dengan nilai ambang, *T,* yang telah ditentukan sebelumnya, maka nilai tersebut akan diubah menjadi 1 dengan warna putih. Sedangkan jika nilai *pixel* pada koordinat lebih kecil dibandingkan dengan nilai ambang, *T,* maka nilai tersebut akan diubah menjadi 0 dengan warna hitam.

1. **Deteksi Tepi**

Faktor kunci dalam mengekstraksi ciri adalah kemampuan mendeteksi keberadaan tepi (*edge*) dari objek di dalam citra. Setelah tepi objek diketahui, langkah selanjutnya dalam analisis citra adalah segmentasi, yaitu mereduksi citra menjadi objek atau region, misalnya memisahkan objek-objek yang berbeda dengan mengekstraksi batas-batas objek (*boundary*). Langkah terakhir dari analisis citra adalah klasifikasi, yaitu memetakan segmen-segmen yang berbeda ke dalam kelas objek yang berbeda pula.

Ada tiga macam tepi yang terdapat di dalam citra digital. Ketiganya adalah:

1. Tepi curam memiliki perubahan intensitas yang tajam. Arah tepi berkisar 90°.
2. Tepi landai atau disebut juga tepi lebar, yaitu tepi dengan sudut arah yang kecil. Tepi landai dapat dianggap terdiri dari sejumlah tepi-tepi lokal yang lokasinya berdekatan.
3. Tepi yang mengandung derau (*noise*) umumnya memiliki tepi yang tidak konsisten.



**Gambar 2.10.** Jenis-jenis Tepi: (a) Tepi Curam; (b) Tepi Landai; (c) Tepi Curam dengan Derau

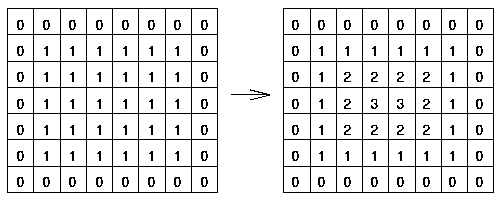
1. ***Distance Transform***

*Distance transform* (transformasi jarak) merupakan operator yang normalnya hanya diaplikasikan terhadap citra biner.Algoritma *distance transform* merupakan algoritma yang bagus untuk berbagai aplikasi seperti pengolahan citra, visi komputer, pengenalan pola, analisis bentuk, dan geometri komputasi (Arcelli dkk, 2011).

Citra yang dihasilkan oleh transformasi ini adalah citra grayscale yang terlihat mirip dengan citra masukannya, kecuali intensitas keabuan dari titik-titik di dalam daerah objek (*foreground*)diubah sedemikian rupa untuk menunjukkan jarak ke batas terdekat dari setiap titik seperti pada gambar 2.11. Jika dilihat dalam bentuk matriks, maka penerapan *distance transform* terhadap citra biner, dapat dilihat pada gambar 2.12.

**Gambar 2.11.** Perubahan Citra Sebelum dan Sesudah Penerapan Distance Transform (Fisher dkk, 2003)



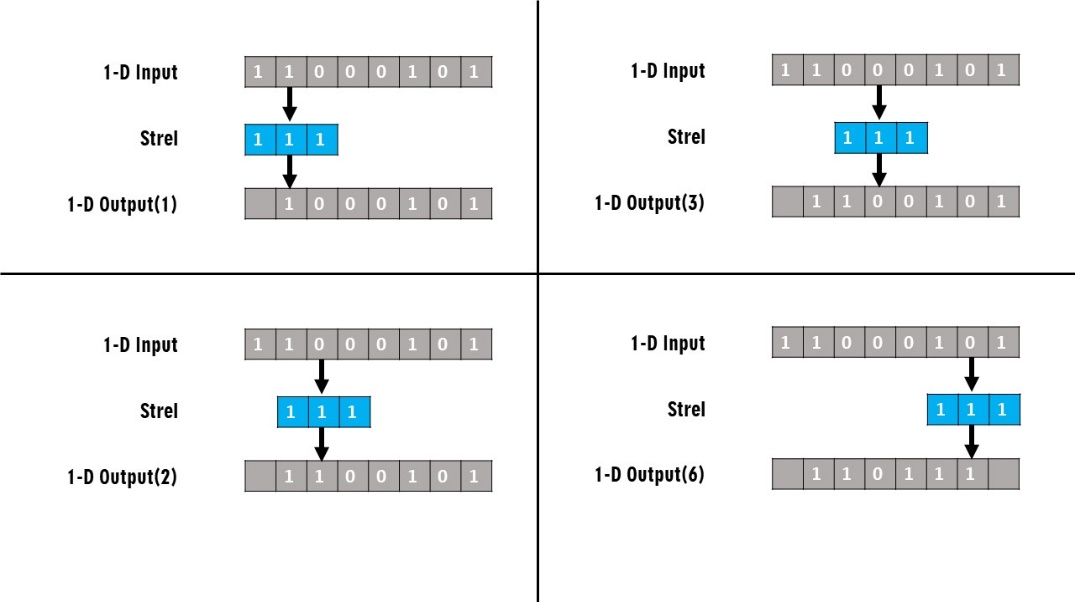
**Gambar 2.12.** Perubahan Matriks Sebelum dan Sesudah Penerapan Distance Transform (Fisher dkk, 2003)

1. **Operasi Morfologi Citra**

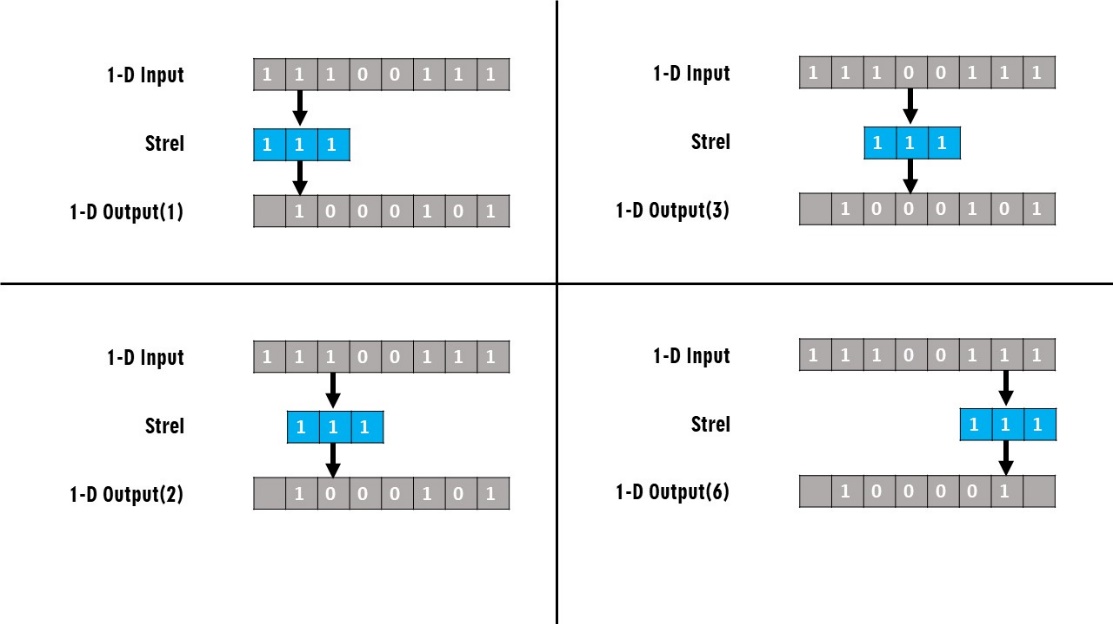
Operasi morfologi citra adalah teknik pengolahan citra yang didasarkan pada bentuk segmen atau region dalam citra. Operasi morfologi citra dilakukan dengan mengkonvolusi *structuring element* (strel) pada citra biner. Ukuran dan bentuk strel dapat ditentukan sesuai dengan citra keluaran yang diinginkan.

Terdapat beberapa jenis dari operasi morfologi citra seperti dilasi (*dilation*), erosi (*erosion*), pembukaan (*opening*), dan penutupan (*closing*). Dilasi bertujuan untuk memperbesar objek. Erosi bertujuan untuk mengecilkan objek. *Opening* merupakan operasi erosi yang diikuti oleh dilasi yang bertujuan untuk menghilangkan noise yang ada di luar objek. *Closing* merupakan operasi dilasi yang diikuti oleh erosi yang bertujuan untuk menghilangkan derau yang ada di dalam objek dengan menutup lubang yang ada di dalam objek.

Dalam menerapkan morfologi pada citra, terdapat dua operasi dasar yang digunakan, yaitu operasi *hit* dan operasi *fit.* Jika terdapat minimal satu dari nilai strel 1 yang cocok dengan masukan yang juga memiliki nilai *pixel* 1, maka akan diberikan keluaran 1. Selain itu, akan diberikan keluaran 0. Operasi ini merupakan operasi *hit* yang diilustrasikan pada **gambar** **2.13**. Jika semua nilai strel 1 cocok dengan masukan yang juga memiliki nilai *pixel* 1, maka akan diberi keluaran 1. Selain itu, akan diberikan keluaran 0. Operasi ini merupakan operasi *fit* yang diilustrasikan pada **gambar 2.14.**



**Gambar 2.13.** Contoh Operasi *Hit* pada Dilasi

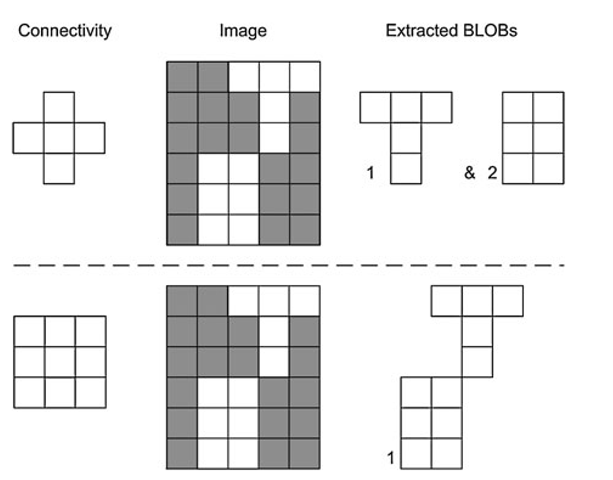


**Gambar 2.14.** Contoh Operasi *Fit* pada Erosi

1. ***Binary Large Object* (Blob)**

Secara umum, blob merupakan kumpulan data biner yang saling terhubung yang dikumpulkan dalam satu entitas. Blob dapat berupa gambar, audio, maupun objek multimedia lainnya. Dalam pengolahan citra digital, sebuah blob merujuk ke bagian dari citra yang memiliki properti yang konstan atau mendekati konstan sehingga bisa dianggap serupa satu sama lain. Blob pada citra terdiri atas kumpulan *pixel* yang saling terhubung. Penentuan kedua *pixel* terhubung atau tidak tergantung pada *connectivity* yaitu penentuan *pixel*-*pixel* mana yang bertetangga dan mana yang bukan.

Terdapat dua jenis *connectivity* yang paling sering digunakan yaitu *8-connectivity* dan *4-connectivity*. *8-connectivity* lebih akurat dibandingkan dengan *4-connectivity*. Tetapi, *4-connectivity* sering kali digunakan karena membutuhkan komputasi yang lebih sedikit dan memproses citra lebih cepat. Efek yang dihasilkan dari dua jenis *connectivity* ini diilustrasikan pada **gambar 2.15**.



**Gambar 2.15.** Perbandingan Hasil Ekstraksi Blob dengan 4-*connectivity* dan 8-*connectivity*.

## Regresi Linear

Regresi linear merupakan teknik analitik prediktif dasar yang menggunakan data historis untuk memprediksi variabel keluaran. Model regresi linear sendiri memiliki banyak aplikasi yang bisa diterapkan di dunia nyata seperti pada bidang ekonomi untuk memprediksi pertumbuhan ekonomi, pada bisnis untuk memprediksi penjualan produk dan performa pegawai, pada ilmu sosial untuk memprediksi kecenderungan politik dari jenis kelamin atau ras tertentu, pada bidang kesehatan untuk memprediksi tekanan darah dari berat badan, dan masih banyak lagi.

Regresi linear digunakan untuk menggambarkan data dan menjelaskan hubungan antara variabel-variabel yang ada. Terdapat dua jenis variabel yang ada pada model regresi linear:

1. Variabel masukan atau prediktor merupakan variabel yang membantu memprediksi nilai dari variabel keluaran. Variabel ini biasa diberi simbol .
2. Variabel keluaran merupakan variabel yang ingin diprediksi. Variabel ini biasa diberi simbol .

Inti dari analisis regresi adalah untuk mencari dan mencocokkan sebuah garis ke dalam plot pencar (*scatter plot*) yang memetakan variabel-variabel ini. Untuk mengestimasi menggunakan regresi linear, maka digunakan persamaan yang terdapat pada persamaan (2.10).

|  |  |
| --- | --- |
|  | (2.10) |

Di mana merupakan nilai yang diprediksi berdasarkan pada persamaan linear sebelumnya. Tujuan dari regresi linear adalah untuk mencari nilai yang signifikan secara statistik dari parameter dan yang dapat mengurangi deviasi di antara dan .

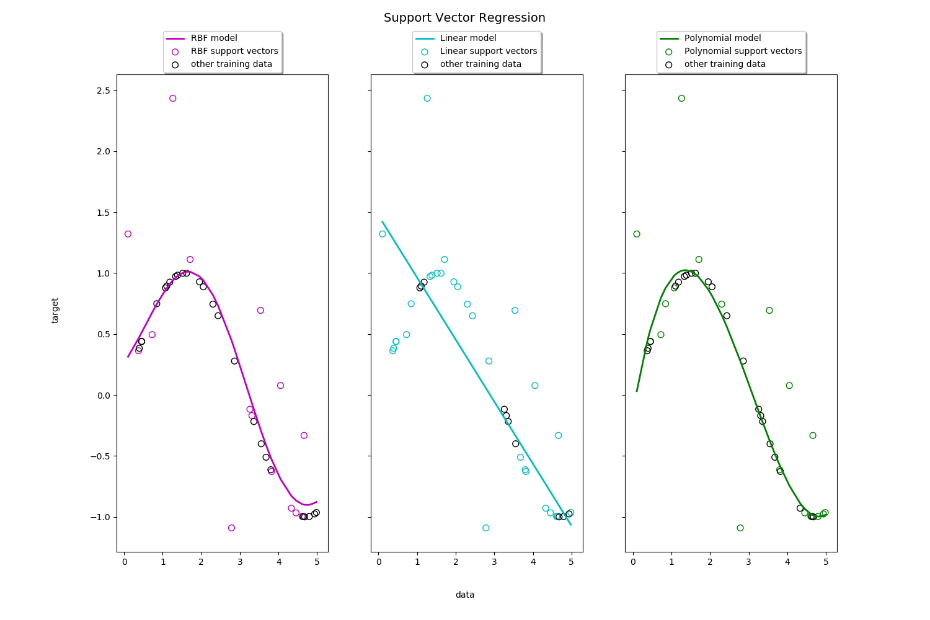
Untuk menentukan nilai dari parameter dan , maka digunakan metode yang disebut sebagai *ordinary least squares.* Tujuan dari metode ini adalah untuk mencari nilai dari parameter dan yang meminimalkan jumlah perbedaan kuadrat antara dan . Adapun persamaan yang digunakan untuk menentukan parameter-parameter ini dapat dilihat pada persamaan (2.11).

|  |  |
| --- | --- |
|  | (2.11) |

Dimana merupakan rata-rata dari nilai dan merupakan rata-rata dari nilai .

## *Support Vector Regression*

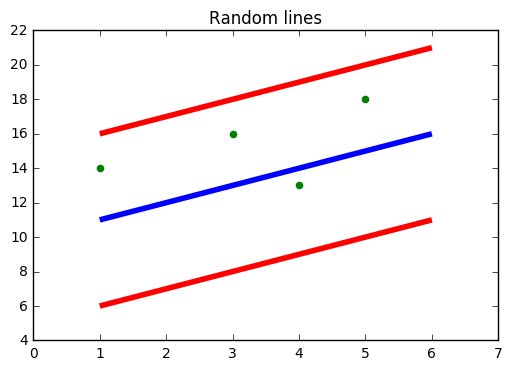
*Support Vector Regression* (SVR) merupakan metode regresi yang memiliki prinsip kerja seperti *Support Vector Machine* (SVM). Algoritma ini merupakan pengembangan dari algoritma regresi linear yang hanya dapat menghasilkan persamaan linear saja. Pada SVR terdapat beberapa model persamaan yang dapat dihasilkan dari proses pelatihan. Model tersebut adalah model linear untuk data yang memiliki hubungan yang linear serta model *Radial Basis Function* (RBF) dan model polinomial untuk data yang memiliki hubungan yang non-linear. Adapun contoh plot persamaan yang didapatkan pada masing-masing model dapat dilihat pada **gambar 2.16.**



**Gambar 2.16.** Hasil Persamaan Regresi dari Masing-masing Model (www.scikit-learn.org)

Adapun beberapa istilah yang perlu dipahami dalam menggunakan algoritma SVR ini dapat diilustrasikan pada **gambar 2.17.** dan penjelasan sebagai berikut.

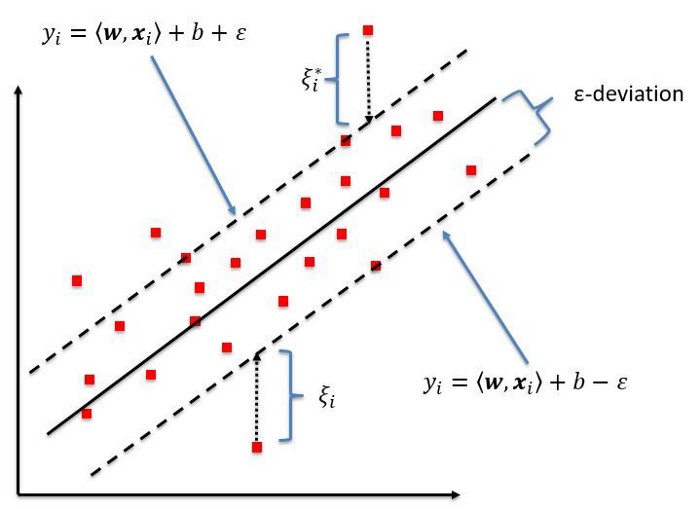
* + - 1. *Kernel*, merupakan fungsi yang digunakan untuk memetakan data dimensi rendah menjadi data dimensi tinggi.
      2. *Hyper Plane,* merupakan garis yang akan menentukan nilai target atau nilai prediksi.
      3. *Boundary Line,* merupakan dua garis selain *hyper plane* yang membuat margin. *Support vector* bisa saja berada tepat di *boundary line* atau di luar *boundary line.*
      4. *Support Vectors,* merupakan poin data yang paling dekat dengan *boundary line*.



**Gambar 2.17**. Garis Merah: *Boundary Lines;* Garis Biru: *Hyper Plane;* Titik Hijau: Poin Data (www.medium.com)

Jika *hyper plane* yang digunakan memiliki model linear, maka fungsi yang dihasilkan dapat dilihat pada persamaan (2.10). Sedangkan persamaan (2.12) menunjukkan persamaan yang membentuk dua *boundary line* yang masing-masing mengapit *hyper plane.* Adapun ilustrasi dari persamaan-persamaan ini dapat dilihat pada **gambar 2.18.** *Boundary line* merupakan batas toleransi *error* yang akan membatasi data poin mana saja yang memiliki nilai eror yang kecil yang akan diambil untuk dimasukkan ke dalam proses pelatihan. Sehingga menghasilkan model yang lebih baik.

|  |  |
| --- | --- |
|  | (2.12) |



**Gambar 2.18.** Ilustrasi Persamaan *Hyper Plane, Boundary Line,* dan Deviasi (www.medium.com)

# Penelitian Terkait

## *Beef Cattle Weight Determine by Using Digital Image Processing* (Pradana dkk, 2016)

Pradana, dkk pada tahun 2016 mengambil fitur lingkar dada dan panjang sapi melalui citra yang didapatkan dari sisi depan dan samping sapi. Peneliti menggunakan metode regresi linear untuk memperkirakan berat badan sapi yang menghasilkan akurasi sebesar 73,2%.

## *Estimation of bodyweight from Body Measurements and Determination of Body Measurements on Limousin Cattle Using Digital Image Analysis* (Ozkaya dkk, 2016)

Ozkaya, dkk pada tahun 2016 melakukan pengukuran pada panjang badan, tinggi badan, lebar dada, dan tinggi kemudi baik secara manual maupun menggunakan *Digital Image Analysis* (DIA). Akurasi perbandingan keduanya menghasilkan 98% untuk tinggi badan, 97% untuk tinggi kemudi, 94% untuk lebar dada dan 90,6% untuk panjang badan. *Regression Analysis* yang digunakan untuk mengetahui hubungan antara luas badan dan berat badan menghasilkan sebesar 61.5%, sedangkan dengan menggunakan semua fitur yang ada menghasilkan sebesar 88.7%.

## Estimasi Berat Karkas Sapi Berdasarkan Segmentasi *K-Means Clustering* dengan Menggunakan Klasifikasi *Multiclass SVM* (Alkautsar dkk, 2016)

Alkautsar dkk. pada tahun 2016 menggunakan metode segmentasi citra dengan kombinasi *K-Means Clustering* dan *Active Contour Model* untuk mendapatkan fitur lingkar dada dan panjang badan sapi. Untuk memprediksi berat badan sapi peneliti menggunakan *Multiclass* SVM yang menghasilkan akurasi sebesar 87,53% pada skala citra 0,5.

## Estimasi Berat Karkas Sapi Berdasarkan Segmentasi *Mean Shift* dengan Klasifikasi *Support Vector Machine Linear* (Henarta dkk, 2016)

Penelitian ini menggunakan *mean shift* untuk melakukan segementasi pada citra bagian depan dan bagian samping sapi. Fitur seperti panjang badan dan lingkar dada yang telah diekstrak akan dikonversi ke satuan centimeter yang kemudian akan dilakukan estimasi bobot dengan rumus Schoorl Indonesia, Schoorl Denmark dan Winter. Akurasi tertinggi yang didapatkan pada penelitian ini adalah 89%.

## Estimasi Berat Karkas dan Klasifikasi Daging Sapi Berdasarkan Deteksi Tepi *Canny, Discrete Wavelet Transform,* dan *Support Vector Machine* (Renasari dkk, 2017)

Renasari dkk pada tahun 2017 menggunakan metode segmentasi citra dengan *Canny Edge Detection* dan *Discrete Wavelet Transform* untuk mendapatkan fitur lebar dada dan panjang badan sapi yang didapatkan dari citra bagian depan dan samping. Untuk memprediksi berat badan sapi peneliti menggunakan SVM yang menghasilkan akurasi sebesar 86.1% untuk estimasi bobot dan 85% untuk klasifikasi.

# BAB III METODOLOGI PENELITIAN

# Tahapan Penelitian

Sistem yang diusulkan pada penelitian ini terdiri dari segmentasi citra untuk mengekstrak fitur dan algoritma regresi linear untuk mengestimasi berat badan sapi. Adapun tahapan-tahapan yang akan dilakukan pada penelitian ini dapat dilihat pada **gambar 3.1**.

**Gambar 3.1.** Diagram Tahapan Penelitian

Berdasarkan diagram penelitian pada **gambar 3.1** dapat dijelaskan sebagai berikut:

1. Studi literatur merupakan tahap awal dalam penelitian ini. Tahap ini dilakukan untuk mengumpulkan penelitian terkait ekstraksi fitur pada citra ternak dan metode untuk mengestimasi berat badannya.
2. Pengambilan data dibagi menjadi dua bagian, pengambilan data citra dan data hasil pengukuran ciri tubuh sapi. Citra sapi diambil menggunakan kamera *smartphone* dan ciri tubuh sapi diambil menggunakan meteran dan timbangan.
3. Analisis data bertujuan untuk mengetahui citra yang memiliki kualitas yang baik untuk diproses selanjutnya. Citra-citra tersebut kemudian diurutkan berdasarkan urutan pengambilan data pengukuran ciri tubuh sapi sehingga tidak ada data yang tertukar.
4. Penentuan metode ekstraksi fitur disesuaikan berdasarkan kondisi citra yang telah diambil sebelumnya. Fitur yang digunakan dalam penelitian ini adalah panjang badan yang didapatkan dari citra sapi bagian samping. Proses segmentasi dilakukan dengan menggunakan gabungan antara *canny edge detection*, *distance transform,* dan *connected component labeling*. Fitur panjang, keliling badan, dan luas badan samping dalam satuan *pixel* akan dimasukkan ke dalam algoritma regresi linear dan SVR untuk proses pelatihan dan pengujian.
5. Langkah awal yang dilakukan dalam mendesain sistem adalah membuat *flowchart* sistem penelitian secara umum. Perancangan sistem dilakukan menggunakan bahasa pemrograman Python.
6. Uji coba sistem dilakukan untuk mengetahui seberapa akurat sistem yang telah dibuat. *Mean Absolute Error* (MAE) digunakan untuk memvalidasi sistem.
7. Tahap akhir dalam penelitian ini adalah melakukan penulisan laporan dalam bentuk skripsi sebagai bahan publikasi.

# Waktu dan Lokasi Penelitian

Penelitian ini dilakukan selama 10 bulan yang dimulai sejak disetujuinya proposal penelitian ini pada akhir bulan September 2018 hingga proses pelaporan hasil penelitian ini pada bulan Juli 2019. Penelitian ini dilakukan di Laboratorium Kecerdasan Buatan Departemen Teknik Informatika, Fakultas Teknik, Universitas Hasanuddin.

Pengambilan data dilakukan di Rumah Pemotongan Hewan, kecamatan Manggala, kota Makassar. Lokasi ini dipilih karena ketersediaan objek penelitian dalam hal ini sapi.

# Instrumen Penelitian

Instrumen penelitian yang digunakan pada penelitian ini meliputi :

1. *Software*
2. Windows 10 x64
3. Framework Jupyter Notebook
4. *Hardware*
5. Laptop Acer, RAM 8 GB, *Processor* Intel(R) Core(TM) i5-4210U CPU @ 1,70 GHz 2,40 GHz
6. *Smartphone* Xiaomi Redmi Note 2 beresolusi 13 MP
7. Timbangan sapi digital.
8. Pita ukur

# Teknik Pengambilan Data

Terdapat dua macam data yang diambil pada penelitian ini, data citra sapi dan data dimensi tubuh sapi.

## Data Citra Sapi

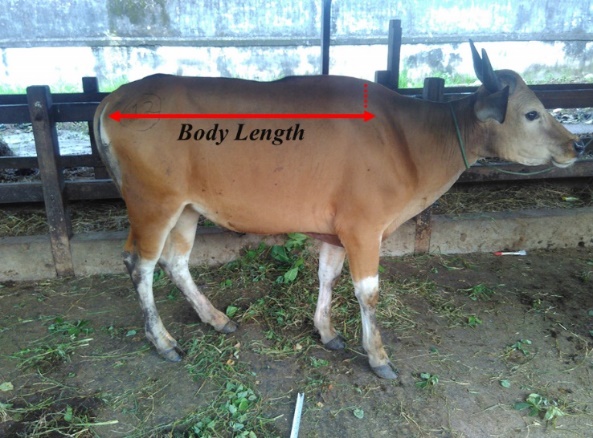
Pengukuran jarak antara sapi dan kamera *smartphone* dilakukan terlebih dahulu sehingga memenuhi kriteria jarak tertentu yang telah ditentukan sebelumnya, yaitu 1,5 meter, 2 meter, dan 2,5 meter. Setelah sesuai, maka dilakukan pengambilan gambar dengan sudut 90° antara kamera dan permukaan tanah. Tinggi kamera disesuaikan sedemikian rupa sehingga sapi berada di tengah *frame.* Ilustrasi pengambilan data dapat dilihat pada **gambar 3.2**.



**Gambar 3.2.** (a) Posisi Kamera terhadap Objek Penelitian (b) Salah Satu Citra Hasil Pengambilan Data dari Samping

## Data Dimensi Tubuh Sapi

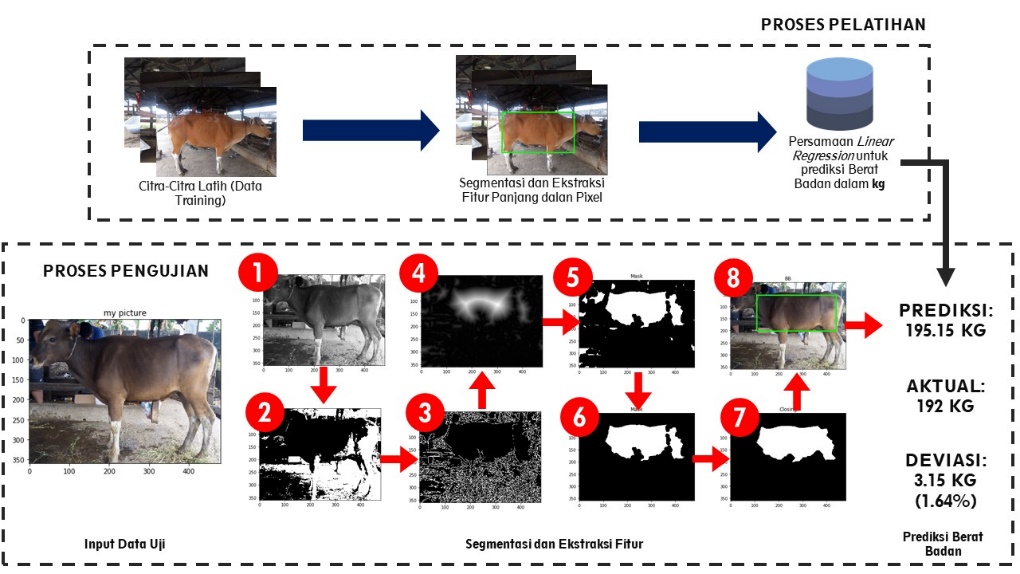
Pengukuran ciri tubuh sapi berupa berat badan dan panjang badan masing-masing dilakukan menggunakan timbangan digital dan pita ukur. Sapi yang ingin ditimbang diarahkan ke atas timbangan dan berat badannya dicatat. Panjang badan sapi diukur menggunakan pita ukur dari pundak hingga tulang ekor sapi seperti pada **gambar 3.3**.



**Gambar 3.3.** Batas Pengukuran Panjang Badan Sapi

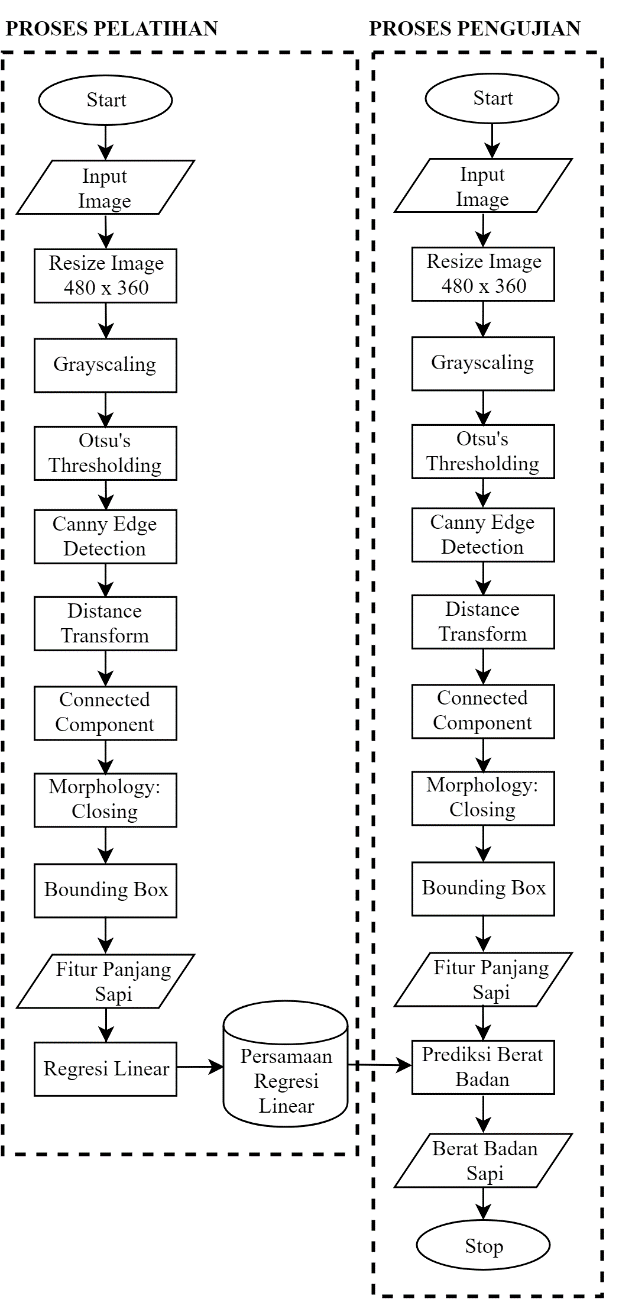
# Perancangan Sistem

Sistem ini secara umum dibagi menjadi dua bagian, pelatihan dan pengujian yang dapat dilihat pada **gambar 3.4**. Baik pelatihan dan pengujian akan melewati proses *pre-processing* dan segmentasi untuk selanjutnya akan diekstrak fiturnya berupa panjang badan sapi dalam *pixel*. Dalam proses pelatihan, fitur yang telah diekstrak dari setiap citra akan dimasukkan ke dalam algoritma regresi linear yang akan menghasilkan persamaan regresi. Persamaan regresi inilah yang digunakan untuk memprediksi berat badan sapi pada proses pengujian.



**Gambar 3.4.** *Gambaran Umum Sistem*

Adapun *flowchart* sistem secara keseluruhan dapat dilihat pada **gambar 3.5**.



**Gambar 3.5.** *Flowchart* Perancangan Sistem Secara Keseluruhan

Berdasarkan **gambar** **3.5**, perancangan sistem estimasi berat badan sapi dapat diuraikan sebagai berikut.

## *Input* *Image*

Langkah pertama adalah mempersiapkan data citra badan sapi bagian samping yang akan menjadi masukan. Dalam penelitian ini, data masukan diambil menggunakan kamera *smartphone* dan menghasilkan spesifikasi sebagai berikut:

1. Resolusi : 4160 x 3120 *pixel*
2. Ekstensi : jpg
3. Rasio : 4:3

Data masukan merupakan hasil pengambilan yang dilakukan menggunakan kamera belakang Xiaomi Redmi Note 2 dengan resolusi 13 *megapixel*. Pengambilan data dilakukan pada saat kondisi cahaya yang cukup dan dilakukan pada siang hari. Adapun contoh data yang diambil sebagaimana ditunjukkan pada **gambar 3.6**.



**Gambar 3.6.** Contoh Citra Masukan

## *Resize* Citra

Tahap selanjutnya adalah memastikan semua citra masukan memiliki dimensi yang sama. Persamaan dimensi citra dilakukan untuk menyamakan jumlah *pixel* yang ada di setiap citra. Kepadatan *pixel* yang berbeda-beda akan menyebabkan bias terhadap hasil prediksi. Ukuran citra yang ditetapkan pada penelitian ini adalah 480 x 360. *Resize* citra juga dilakukan untuk mempercepat waktu proses. Hasil citra yang telah melewati proses *resize* dapat dilihat pada **gambar 3.7**.



**Gambar 3.7.** Citra Hasil *Resize*

## *Grayscaling*

Setelah semua data memiliki ukuran yang sama, maka langkah selanjutnya adalah mengubah citra yang memiliki *colorspace* RGB menjadi *colorspace grayscale*. Proses ini dilakukan dengan menggunakan metode *luminosity*. Adapun contoh perhitungan nilai *grayscale* setiap *pixel* menggunakan persamaan (3.1).

|  |  |
| --- | --- |
|  | (3.1) |

Keterangan:

1. *Gray* = nilai derajat keabuan
2. R = nilai pada *channel Red*
3. G = nilai pada *channel Green*
4. B = nilai pada *channel Blue*

Misalkan didapatkan matriks citra sebagai berikut di mana *pixel* merupakan sebuah vektor tiga dimensi yang mewakili setiap channel yang ada dan memiliki nilai [55, 57, 72]. Maka, nilai derajat keabuan dari *pixel* adalah:

Adapun hasil konversi citra RGB ke citra *grayscale* dapat dilihat pada **gambar 3.8**.



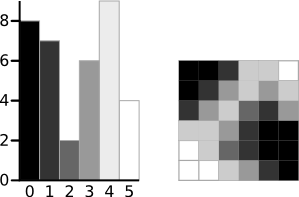
**Gambar 3.8.** Citra Hasil *Grayscaling*

## *Otsu’s Thresholding*

Citra yang telah dikonversi menjadi citra *grayscale*, hanya berisikan nilai 0-255 tiap *pixel*nya. Nilai *threshold* merupakan nilai yang menentukan nilai-nilai berapa saja yang akan diubah menjadi 0 untuk warna hitam jika nilainya berada di bawah nilai *threshold* dan 1 untuk warna putih jika nilainya berada di atas nilai *threshold*.

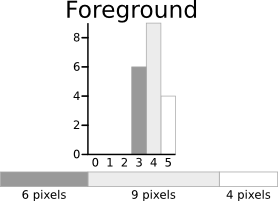
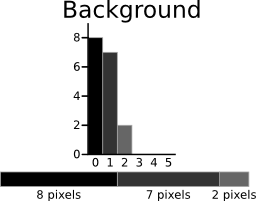
Pada penelitian ini, digunakan metode *otsu’s thresholding* sehingga pembagian histogram citra *grayscale* ke dalam dua daerah yang berbeda dilakukan secara otomatis tanpa membutuhkan penetapan nilai ambang. Pencarian nilai ambang optimal dari gambar masukan dapat dilakukan dengan menelusuri seluruh nilai ambang yang mungkin (dari 0 hingga 255). Adapun pencarian nilai ambang optimal ini dilakukan sebagai berikut:

1. Misalkan terdapat citra dengan potongan matriks 6x6 seperti pada **gambar 3.9**. Dari nilai intensitas 0-5, akan dicari nilai *within class variance* yang paling kecil dengan mencoba satu per satu nilai T-nya secara berurutan.



**Gambar 3.9.** a) Histogram Citra; b) Citra dengan 6 Derajat Keabuan (Greensted, 2010)

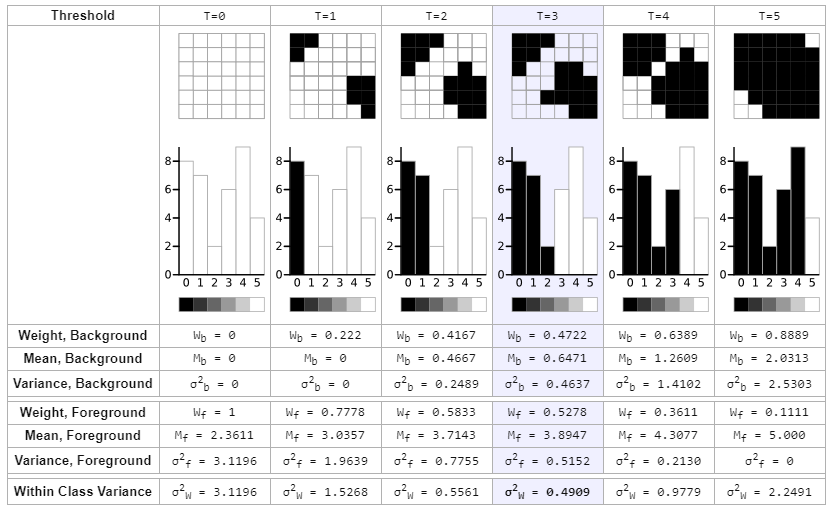
1. Jika pada perhitungan ini nilai , maka semua *pixel* dengan nilai akan dianggap sebagai *background* dan semua *pixel* dengan nilai akan dianggap sebagai *foreground.* Sehingga, histogram *background* dan *foreground* akan terlihat seperti pada **gambar 3.10**.



**Gambar 3.10.** Potongan Histogram dengan Derajat Keabuan pada Sumbu X dan Frekuensi pada Sumbu Y: a) *Background;* b) *Foreground* (Greensted, 2010)

1. Setelah *background* dan *foreground* dibagi berdasarkan nilai yang sedang diuji, maka nilai *Weight, Mean,* dan *Variance* akan dihitung pada masing-masing *background* dan *foreground.*

1. Langkah selanjutnya adalah menghitung *Within-Class Variance* dengan menjumlahkan *background variance* dan *foreground variance* yang dikalikan dengan *weight* masing-masing.
2. Mencari nilai *Within Class Variance* pada yang lain dengan mengikuti langkah 1-4. Adapun hasil perhitungan terhadap nilai T lainnya dapat dilihat pada **gambar 3.11**.



**Gambar 3.11.** Hasil Perhitungan *Within Class Variance* pada semua nilai (Greensted, 2010)

1. Langkah terakhir adalah mencari nilai terkecil. Pada contoh di atas, nilai yang terkecil dihasilkan oleh nilai. Sehingga nilai ambang yang digunakan pada contoh matriks di atas adalah.

## *Canny Edge Detection*

Setelah citra biner dihasilkan, maka akan dilakukan *canny edge detection* untuk menentukan tepi-tepi dari objek yang ada pada citra. Perubahan nilai intensitas keabuan yang besar dalam jarak yang singkat akan membentuk tepi. Adapun tahapan dalam melakukan *canny edge detection* adalah:

1. Menghilangkan derau yang ada pada citra *grayscale* dengan mengimplementasikan filter Gaussian. Proses ini menghasilkan citra yang tampak sedikit buram sehingga garis-garis halus akan terfilter dan menyisakan garis-garis tegas yang merupakan tepi objek yang sebenarnya. Adapun salah satu contoh filter Gaussian 5×5 dengan nilai σ = 1.4 dapat dilihat pada persamaan (3.2) di mana *asterisk* (\*)menandakan operasi konvolusi.

|  |  |
| --- | --- |
|  | (3.2) |

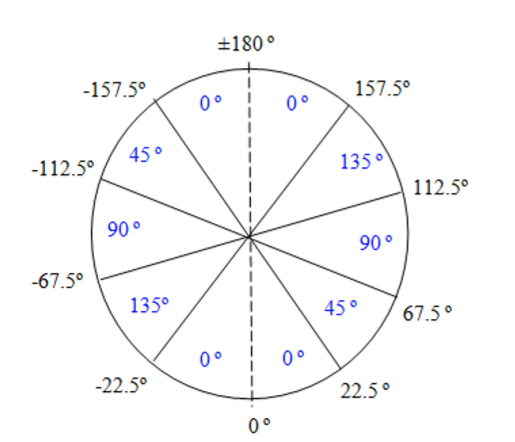
1. Mencari gradien intensitas dari citra dengan menggunakan operator Sobel. Proses ini menghasilkan turunan pertama dari arah horizontal () dan vertikal () dengan menggunakan persamaan (3.3).

|  |
| --- |
|  |
|  |
|  | (3.3) |

1. Menentukan arah tepi yang ditemukan dengan menggunakan persamaan (3.4).

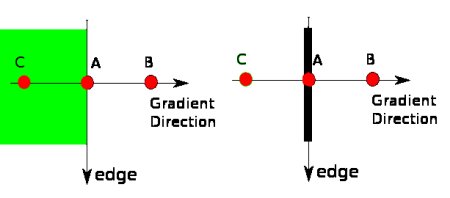
|  |  |
| --- | --- |
|  | (3.4) |

Sudut arah tepi dibulatkan ke salah satu dari keempat sudut yang mewakili vertikal, horizontal, dan dua diagonal (0°, 45°, 90°, 135°) seperti pada **gambar 3.12**. Sebagai contoh, nilai pada [0°, 22.5°] akan dipetakan ke 0°.



**Gambar 3.12.** Pembagian Sudut Gradien Penentu Arah

1. Memperkecil garis tepi yang dihasilkan pada proses sebelumnya dengan menerapkan *non-maximum suppression.* Berdasarkan **gambar 3.13**, *pixel* A berada di tepi pada arah vertikal. *Pixel* B dan C berada pada arah gradien. Kemudian *pixel* A akan dibandingkan dengan *pixel* B dan C untuk melihat apakah terbentuk *local maximum*. Jika terdapat *local maximum*, maka nilai *pixel* akan disimpan dan melangkah ke tahap selanjutnya, jika tidak, maka titik A akan dikonversi menjadi 0 (bukan tepi).



**Gambar 3.13.** *Non-maximum Suppression*

1. Menerapkan *double thresholding* untuk menghilangkan derau dan mempertegas yang mana yang termasuk tepi dan yang mana yang bukan termasuk tepi. Terdapat dua jenis nilai *threshold* yang digunakan, *high threshold* dan *low threshold*. Berdasarkan **gambar 3.14**, semua *pixel* yang bernilai lebih dari nilai *high threshold,* maka akan dianggap sebagai tepi yang kuat seperti pada *pixel* A. Semua *pixel* yang bernilai kurang dari nilai *low threshold*, maka akan dianggap sebagai bukan tepi dan akan diatur menjadi nilai 0. Semua *pixel* yang berada di antara nilai *low threshold* dan *high threshold*, akan dianggap sebagai tepi jika terhubung dengan tepi yang kuat seperti pada *pixel* C, jika tidak, maka akan dianggap sebagai bukan tepi seperti pada *pixel* B.



**Gambar 3.14.** Ilustrasi *double thresholding* dengan calon tepi pada sumbu X dan nilai *threshold* pada sumbu Y

Pada penelitian ini, kedua nilai *threshold* yang digunakan berbeda dari satu citra ke citra lainnya. Nilai dari *high threshold* yang digunakan pada tahap ini adalah nilai yang didapatkan pada tahap *otsu’s thresholding*  sebelumnya. Sedangkan nilai dari *low threshold* berasal dari nilai *high threshold* yang dikalikan dengan rasio tertentu. Rasio inilah yang akan ditentukan untuk mendapatkan hasil yang terbaik melalui proses *trial & error*. Definisi dari hasil terbaik pada tahap ini adalah objek sapi dapat terlihat dengan jelas dengan sedikit atau tanpa tepi pada badan sapi. Adapun pada penelitian ini, nilai rasio yang digunakan adalah 0.3 dengan perbandingan hasil citra yang dapat dilihat pada **gambar 3.15**.



**Gambar 3.15.** Citra Hasil *Canny Edge Detection*

## *Laplacian of Gaussian*

Tahapan ini merupakan tahapan deteksi tepi untuk membandingkan metode deteksi mana yang menghasilkan hasil segmentasi yang lebih baik. *Laplacian of Gaussian* (LoG) merupakan filter turunan yang digunakan untuk menemukan perubahan nilai intensitas yang signifikan pad citra. Karena filter turunan selalu sensitif terhadap derau, maka citra dihaluskan terlebih dahulu menggunakan filter Gaussian sebelum menerapkan Laplacian. Kedua proses inilah yang disebut sebagai *Laplacian of Gaussian*. Adapun persamaan yang digunakan pada operator Laplacian dapat dilihat pada persamaan (3.5).

|  |  |
| --- | --- |
|  | (3.5) |

Berbeda dengan *canny,* LoG tidak membutuhkan dua *threshold* untuk menentukan *weak edge, strong edge,* dan *non-edge*. Hal ini dikarenakan LoG menerapkan filter turunan, di mana nilainya diambil dari nilai turunan intensitas tepi berdasarkan operator Sobel. Dengan tidak adanya penentuan nilai *threshold,* maka proses *otsu’s thresholding* tidak perlu dilakukan jika menggunakan LoG. Adapun hasil citra yang dihasilkan dapat dilihat pada **gambar 3.16.**



**Gambar 3.16.** Citra Hasil *Laplacian of Gaussian*

## *Distance Transform*

*Distance transform* digunakan untuk menentukan *peak* (puncak) dan *valley* (lembah) dari masing-masing bagian yang tidak memiliki tepi yang ada dalam citra biner masukan. Bagian yang tidak memiliki tepi pada bagian sebelumnya diasumsikan sebagai sebuah objek. Semakin besar bagian yang tidak memiliki tepi, maka semakin tinggi *peak* yang dihasilkan. Sebaliknya, jika bagian yang tidak memiliki tepi berukuran kecil, maka *peak* yang dihasilkan juga pendek. Dari pusat *peak* ke seluruh tepi objek akan terbentuk *valley* yang memiliki intensitas *pixel* yang berbeda-beda. Semakin jauh dari *peak* maka semakin sedikit nilai intensitas *pixel* yang dihasilkan.

Pada penelitian ini, diterapkan *euclidean distance* untuk menentukan jarak *pixel* yang bernilai 0 dengan tepi terdekat yang bernilai 1. Adapun proses pada tahap ini adalah.

* + - 1. Misalkan terdapat matriks 4x4 yang merupakan potongan matriks dari citra hasil *canny* pada **gambar 3.15.**

Untuk membentuk *peak* maka semua *pixel* dengan nilai intensitas 0 akan dicari *euclidean distance*-nya terhadap *pixel* dengan nilai intensitas 1 yang diasumsikan sebagai sebuah tepi menggunakan persamaan (3.6).

|  |  |
| --- | --- |
|  | (3.6) |

* + - 1. Adapun koordinat dari matriks di atas adalah:

Berdasarkan persamaan (3.6), maka jarak *pixel* (1,1) ke *pixel* tepi terdekat (3,1) adalah:

Sedangkan jarak *pixel* (2,3) ke *pixel* tepi terdekat (3,2) adalah:

* + - 1. Mengubah nilai intensitas 0 pada *pixel* dengan koordinat (1,1) dengan nilai intensitas 2. Begitu pula dengan nilai intensitas 0 pada *pixel* dengan koordinat (2,3) akan diganti dengan nilai intensitas 1.414.
      2. Mencari nilai *euclidean distance* terhadap semua *pixel* dengan intensitas 0.

Sehingga citra keluaran yang dihasilkan pada tahap ini akan terlihat seperti pada **gambar 3.17**.



**Gambar 3.17.** Citra Hasil *Distance Transform.*

Tahap ini dikombinasikan dengan *global thresholding*, di mana citra hasil *distance transform* yang merupakan citra *grayscale*, akan diubah kembali menjadi citra biner. Untuk itu digunakan nilai *threshold*. Adapun tahapannya adalah.

Intensitas tertinggi (*peak*) akan diambil dengan memanggil fungsi *.max()*  pada variabel *bwdist* yang menyimpan seluruh nilai intensitas *pixel* pada citra*.*

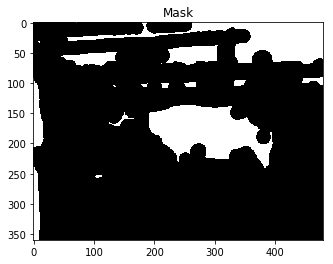
Nilai *threshold* akan ditetapkan berdasarkan persamaan (3.7).

|  |  |
| --- | --- |
|  | (3.7) |

Setelah nilai *threshold* didapatkan, maka semua *pixel* yang memiliki nilai di bawah *threshold* akan diubah nilainya menjadi 0. Semua *pixel* yang memiliki nilai di atas atau sama dengan *threshold* akan diubah nilainya menjadi 1 seperti persamaan (3.8).

|  |  |
| --- | --- |
|  | (3.8) |

Sehingga citra yang dihasilkan adalah citra dengan dua derajat keabuan 0 dan 1 seperti pada **gambar 3.18.**



**Gambar 3.18.** Citra Hasil *Global Thersholding*

## *Connected Component Labeling*

Pada tahap ini, dilakukan analisis struktural dari tiap-tiap bagian yang saling terhubung. Tahap ini juga disebut sebagai *labeling* karena tiap-tiap bagian yang terhubung akan diberi label sehingga terpisah satu sama lain.

*Connected component labeling* memindai citra dan bagian-bagian *pixel*nya berdasarkan *pixel connectivity*. Terdapat dua jenis *connectivity,* 4-*connectivity* dan 8-*connectivity*. Kedua jenis *connectivity* ini, tidak terlihat perbedaannya secara kasat mata karena ukuran citra yang memiliki banyak *pixel*. Sehingga pada penelitian ini digunakan jenis *connectivity default*, yaitu 8-connectivity.

Semua *pixel* pada *connected component* memiliki intensitas *pixel* yang sama dan saling terhubung satu sama lain. Operator *connected component labeling* memindai citra dengan bergerak di sepanjang baris hingga mencapai titik *p* (di mana *p* merupakan *pixel* yang akan diberikan label) dengan *V* = {1}. Kemudian, akan dilakukan pengujian terhadap keempat tetangga dari *p* yang telah dilakukan pemindaian sebelumnya. Keempat tetangga dari *p*, adalah *pixel* tetangga di sebelah kiri, di atas, dan kedua diagonal atas. Berdasarkan informasi ini, maka proses labeling dari *pixel* *p* akan terjadi sebagai berikut:

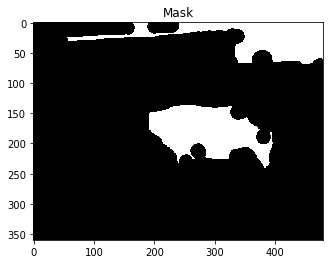
1. Jika keempat tetangga *pixel*nya adalah 0, tetapkan label baru untuk p.
2. Jika hanya satu *pixel* tetangga yang memiliki *V* = {1}, tetapkan label yang sama untuk p.
3. Jika lebih dari satu *pixel* tetangga memiliki *V* = {1}, tetapkan salah satu label untuk p dan catat kesetaraannya.

Setelah pemindaian selesai, pasangan label yang ekuivalen diurutkan ke dalam masing-masing kelas. Sebagai langkah terakhir, pemindaian kedua dilakukan, di mana setiap label diganti dengan label yang ditetapkan untuk masing-masing kelas. Untuk tampilan, label dapat ditampilkan menggunakan tingkat keabuan atau warna yang berbeda. Hasil dari proses ini dapat dilihat pada **gambar 3.19**. Derajat keabuan yang berbeda menunjukkan perbedaan pada tiap-tiap bagian yang dianggap saling terhubung.



**Gambar 3.19.** Citra Keluaran *Connected Component Labeling*.

Setelah semua bagian telah diberi label, maka proses pengurangan derau dapat dilakukan. Bagian-bagian kecil yang dianggap buka bagian dari badan sapi akan dihilangkan dengan cara mengatur batas luasan label. Pada penelitian ini semua label yang memiliki luasan kurang dari 2000 akan dihilangkan sehingga akan terlihat pada **gambar 3.20**.

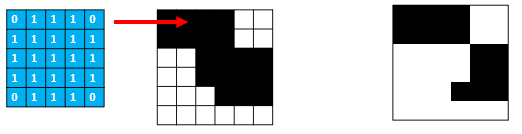


**Gambar 3.20.** Hasil Pengurangan Derau

## *Closing Operation*

Citra yang dihasilkan pada tahap sebelumnya terkadang masih memiliki beberapa kekurangan, misalnya terdapat lubang atau bahkan terdapat bagian sapi yang saling terpisahkan. Untuk mengatasi hal ini, maka dilakukan operasi morfologi dengan menggunakan *closing*. *Closing* dipilih karena operator ini dapat mengisi lubang lebih baik daripada dilasi dan dapat mempertahankan bentuk aslinya. Adapun *structuring element* (strel) yang digunakan pada penelitian ini berbentuk *disc* dengan ukuran 5 x 5 serta iterasi sebanyak 4 kali. Hal ini dipilih berdasarkan *trial & error*.

**Gambar 3.21** menunjukkan proses konvolusi strel terhadap citra masukan yang menghasilkan tertutupnya lubang yang memisahkan kedua blob. Adapun hasil keluaran citra yang telah melalui proses morfologi citra dapat dilihat pada **gambar 3.22**.



**Gambar 3.21.** Sebelum dan Sesudah Proses Konvolusi pada Operasi *Closing*

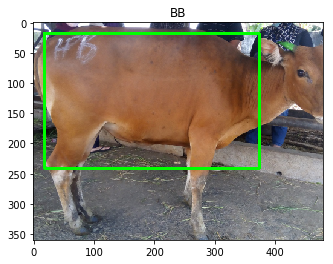
**

**Gambar 3.22.** Hasil Operasi *Closing* Menutup Lubang yang Ada

## *Bounding Box* Berdasarkan *Euclidean Distance*

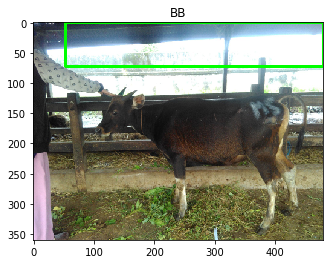
Pada tahap ini, blob yang telah terbentuk dari hasil tahap sebelumnya, akan diberikan *bounding box*. Adapun pada penelitian ini, telah dicoba dua teknik pemberian *bounding box*, yaitu dengan menentukan minimum dan maksimum luasan blob dan *euclidean distance* dengan mencari blob terdekat dengan pusat citra. Penentuan minimum dan maksimum luasan blob dilakukan pada tahap *labeling* seperti yang dijelaskan sebelumnya.

Jika hanya terdapat satu blob, maka pemberian *bounding box* dapat langsung dilakukan, dengan hasil seperti pada **gambar 3.23**. Tetapi, terdapat situasi di mana objek sapi memiliki ukuran yang kecil dan terdapat derau berupa *open space* dengan intensitas warna yang sama sehingga akan dianggap sebagai satu blob yang sama, seperti gambar 3.20 di atas.



**Gambar 3.23.** *Bounding Box* Tanpa Derau

Jika hanya mengandalkan batas minimum dan maksimum blob pada tahap sebelumnya, maka ada kemungkinan blob yang terfilter adalah objek yang bukan sapi yang melewati batas min dan max seperti pada **gambar 3.24**. Sehingga diperlukan pendekatan lain untuk mengatasi kasus ini.

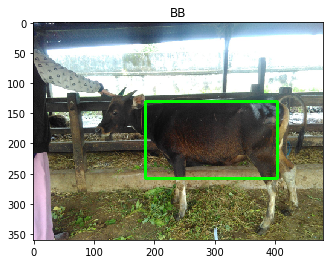


**Gambar 3.24.** *Bounding Box* dengan Min Max Blob

Karena teknik pengambilan data yang mengharuskan sapi berada di tengah *frame*, maka dilakukan pemilihan blob berdasarkan kedekatannya dengan pusat frame menggunakan *euclidean distance*. Tiap blob akan dicari titik pusatnya. Tiap titik pusat ini akan dicari *euclidean distance*-nya terhadap titik pusat citra menggunakan persamaan (3.6).

Misalnya pada **gambar 3.22,** koordinat titik pusat blob pertama adalah sedangkan koordinat titik pusat blob kedua adalah . Adapun koordinat titik pusat citra adalah . Sehingga nilai *euclidean distance* dari blob 1 ke titik pusat dan dari blob 2 ke titik pusat adalah:

Titik pusat blob yang menghasilkan *euclidean distance* paling kecil akan dianggap paling dengan titik pusat citra dan akan diberi *bounding box* seperti pada **gambar 3.25**.



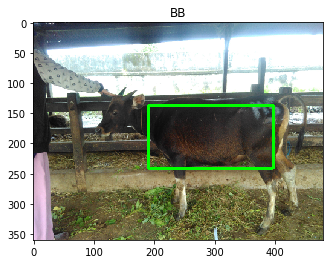
**Gambar 3.25.** *Bounding Box* dengan *Euclidean Distance*

Hasil dari *bounding box* ini kemudian akan diambil panjang *box* nya yang merepresentasikan panjang badan sapi dalam *pixel*. Fitur ini yang kemudian akan dimasukkan ke dalam algoritma untuk mengestimasi berat badan sapi.

## *Bounding Box* Berdasarkan Rentang Centroid

Tahapan ini digunakan untuk membandingkan hasil pemberian *bounding box* yang didapatkan pada tahap sebelumnya. Dengan mengandalkan teknik pengambilan data yang mengharuskan sapi berada di tengah *frame*, maka dilakukan pemilihan blob dapat dilakukan dengan mengatur rentang pada sumbu x dan sumbu y. Penetapan rentang pada sumbu x dan sumbu y didasarkan pada titik pusat citra yang dikurangi dengan margin tertentu. Adapun penentuan margin dilakukan berdasarkan *trial and error* untuk mendapatkan hasil yang baik. Jika titik pusat citra adalah dan margin yang ditetapkan adalah 100, maka *bounding box* akan diberikan jika titik pusat blob memenuhi persamaan (3.9). Contoh hasil pemberian *bounding box* dengan metode ini dapat dilihat pada **gambar 3.26.**

|  |  |
| --- | --- |
|  | (3.9) |



**Gambar 3.26.** Citra Hasil *Bounding Box* dengan Rentang Centroid

## Prediksi Berat Badan Sapi Menggunakan Regresi Linear

Regresi linear digunakan untuk memprediksi berat badan sapi dalam satuan kg dengan menggunakan satu variabel independen. Dalam penelitian ini, variabel independen yang digunakan adalah panjang sapi dalam *pixel*. Tahap ini terbagi menjadi dua bagian, yaitu proses pelatihan dan proses pengujian. Proses pelatihan menggunakan data panjang badan *pixel* dengan label yang berupa berat badan aktual. Pada penelitian ini, terdapat 33 data panjang badan dalam *pixel* untuk data latih dan 10 data panjang badan dalam *pixel* untuk data uji. Semua data panjang badan ini didapatkan dari hasil ekstraksi fitur pada tahap sebelumnya.

* + - 1. Proses Pelatihan

Proses pelatihan regresi linear bertujuan untuk menghasilkan parameter dan yang optimal berdasarkan data latih yang dimasukkan. Contohnya jika terdapat data sebagai berikut:

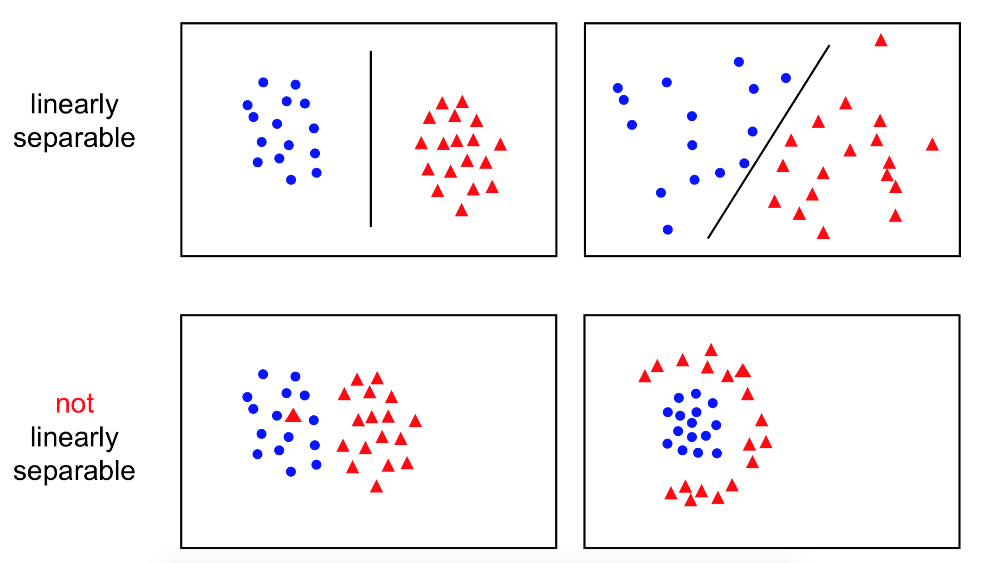
|  |  |
| --- | --- |
| **Panjang Pixel (X)** | **Berat Badan Aktual (Y)** |
| 276 | 195 |
| 442 | 255 |
| 464 | 256 |
| 371 | 237 |

* + - 1. Proses Pengujian

Proses pengujian regresi linear bertujuan untuk menghasilkan prediksi berdasarkan persamaan yang telah didapatkan pada proses pelatihan sebelumnya. Dengan mengambil nilai dan , yang dimasukkan ke dalam persamaan 2.10, maka berat badan sapi dapat diestimasi. Misalnya terdapat panjang badan dalam *pixel* dengan nilai 277. Maka estimasi nilai berat badan sapi adalah:

## Prediksi Berat Badan Sapi Menggunakan SVR

Metode ini diterapkan untuk membandigkan unjuk kerja sistem dengan metode regresi linear sebelumnya. Terdapat beberapa kernel yang dapat digunakan, yaitu linear, RBF, dan sigmoid. Kernel linear digunakan ketika data yang dimasukkan dapat dipisahkan secara linear, sedangkan kernel RBF dan sigmoid digunakan ketika data yang dimasukkan tidak dapat dipisahkan secara linear. Adapun representasi data yang dapat dipisah secara linear dan yang tidak dapat dipisah secara linear dapat dilihat pada **gambar 3.27.**



**Gambar 3.27.** Representasi Data yang Dapat dan Tidak Dapat Dipisah Secara Linear

Adapun pada kernel linear, persamaan yang digunakan untuk membuat *hyper plane* sama dengan fungi linear pada umumnya seperti pada persamaan (3.10).

|  |  |
| --- | --- |
|  | (3.10) |

Dimana adalah vektor bobot dan merupakan bias. Tujuan dari kernel ini adalah untuk menemukan fungsi yang memberikan nilai *error* yang paling minimum. Hasil dari sangat bergantung pada nilai parameter dan sehingga kernel ini akan mencari nilai-nilai optimal dai keuda parameter ini. Untuk kasus di mana data yang dimasukkan merupakan data non-linear, maka data masukan terlebih dahulu dipetakan oleh fungsi transformasi . Perubahan ini memerlukan kernel sehingga data dapat terpisahkan oleh *hyper plane*. Sehingga model linear baru yang digunakan sebagai fungsi regresi dapat dilihat pada persamaan (3.11).

|  |  |
| --- | --- |
|  | (3.11) |

# Analisis Kerja Sistem

Analisis kerja sistem estimasi berat badan sapi dilakukan dengan memanfaatkan *Mean Absolute Error* (MAE) atau *Mean Absolute Deviation* (MAD). MAE mengevaluasi metode estimasi yang digunakan dengan menjumlahkan deviasi absolut dari angka-angka prediksi dan aktual, kemudian dicari rata-ratanya. Persamaan MAE dapat dilihat pada persamaan (3.12).

|  |  |
| --- | --- |
|  | (3.12) |

MAE merupakan metrik yang memiliki rentang dari 0 hingga ∞. MAE dikelompokkan menjadi metrik dengan orientasi negatif, artinya nilai yang lebih rendah merupakan nilai yang lebih baik.

Sedangkan untuk melihat seberapa kuat korelasi antara fitur-fitur yang digunakan dengan berat badan yang dihasilkan, maka digunakan koefisien korelasi Pearson (). Metode ini menghitung korelasi dua variabel X dan Y, sehingga menghasilkan nilai di antara -1 dan +1. Di mana nilai 1 menunjukkan hubungan linear yang positif, 0 menunjukkan hubungan yang tidak linear, dan -1 menunjukkan hubungan linear yang negatif. Adapun persamaan yang digunakan untuk mencari nilai dapat dilihat pada persamaan (3.13), di mana merupakan fitur dan merupakan hasil estimasi.

|  |  |
| --- | --- |
|  | (3.13) |

# BAB IV HASIL DAN PEMBAHASAN

# Hasil Penelitian

Bab ini menyajikan hasil unjuk kerja sistem estimasi berat badan sapi menggunakan pengolahan citra. Penelitian ini menggunakan *Mean Absolute Error* (MAE)untuk mengetahui deviasi absolut rata-rata dari berat badan prediksi dan aktual. Adapun jumlah data yang digunakan adalah sebanyak 33 data latih dan 10 data uji. Semua data ini diambil berdasarkan tiga skenario yang didasarkan pada jarak yang berbeda yaitu skenario 1 pada jarak pengambilan gambar 1,5 meter, skenario 2 pada jarak pengambilan gambar 2 meter, dan skenario 3 pada jarak pengambilan gambar 2,5 meter. Terdapat tiga variabel penting yang mempengaruhi hasil segmentasi citra yang kemudian akan dilakukan proses *parameter* *tuning* dengan masing-masing sembilan nilai.

1. Rasio *low threshold* pada *canny edge detection* (RC) dengan nilai 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, dan 0.9.
2. Rasio *threshold* pada *distance transform* (RDT)dengan nilai 0.11, 0.12, 0.13, 0.14, 0.15, 0.16, 0.17, 0.18, dan 0.19.
3. Jumlah iterasi pada operasi morfologi dengan *structuring element* berbentuk *disc* dengan ukuran 5x5 (IC) dengan nilai 1, 2, 3, 4, 5, 6, 7, 8, dan 9.

Dengan masing-masing sembilan nilai, maka ketiga kombinasi parameter ini akan membutuhkan 729 kali iterasi pencarian MAE untuk satu skenario. Tujuan dilakukannya iterasi ini adalah untuk membandingkan hasil rata-rata MAE yang dihasilkan tiap kombinasi parameter pada tiap skenario dan menentukan jarak pengambilan gambar terbaik pada penelitian ini. Berdasarkan 729 iterasi, maka rata-rata MAE yang dihasilkan minimum pada 1,5 meter. Sehingga perhitungan pada bagian selanjutnya akan difokuskan pada skenario 1. Hasil perbandingan tiap skenario dapat dilihat pada **tabel 4.1**.

**Tabel 4.1.** Hasil Perbandingan Rata-rata MAE pada Tiga Skenario

|  |  |  |
| --- | --- | --- |
| **No.** | **Skenario** | **Rata-rata MAE (Kg)** |
| 1 | 1,5 Meter | 9,59 |
| 2 | 2 Meter | 10,90 |
| 3 | 2,5 Meter | 11,81 |

Adapun estimasi berat badan sapi diprediksi dengan algoritma regresi linear menggunakan 33 data panjang dalam *pixel* hasil segmentasi dan ekstraksi fitur. Adapun persamaan regresi dengan MAE yang paling minimum dapat dilihat pada **tabel 4.2**.

**Tabel 4.2.** Perhitungan Persamaan Regresi pada Data dengan Jarak 1.5 Meter (RC = 0.6, RDT = 0.15, IC = 4)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **X (Panjang Badan dalam Pixel)** | **Y (Berat Badan Aktual dalam Kg)** |  |  | **)** |  |
| 1 | 341 | 195 | -30.273 | -1.364 | 41.281 | 916.438 |
| 2 | 426 | 255 | 54.727 | 58.636 | 3209.008 | 2995.074 |
| 3 | 284 | 256 | -87.273 | 59.636 | -5204.628 | 7616.529 |
| 4 | 383 | 237 | 11.727 | 40.636 | 476.554 | 137.529 |
| 5 | 437 | 185 | 65.727 | -11.364 | -746.901 | 4320.074 |
| 6 | 359 | 234 | -12.273 | 37.636 | -461.901 | 150.620 |
| 7 | 376 | 237 | 4.727 | 40.636 | 192.099 | 22.347 |
| 8 | 330 | 238 | -41.273 | 41.636 | -1718.446 | 1703.438 |
| 9 | 404 | 237 | 32.727 | 40.636 | 1329.917 | 1071.074 |
| 10 | 362 | 101 | -9.273 | -95.364 | 884.281 | 85.983 |
| 11 | 442 | 282 | 70.727 | 85.636 | 6056.826 | 5002.347 |
| 12 | 324 | 196 | -47.273 | -0.364 | 17.190 | 2234.711 |
| 13 | 206 | 146 | -165.273 | -50.364 | 8323.736 | 27315.074 |
| 14 | 277 | 144 | -94.273 | -52.364 | 4936.463 | 8887.347 |
| 15 | 443 | 204 | 71.727 | 7.636 | 547.736 | 5144.802 |
| 16 | 312 | 175 | -59.273 | -21.364 | 1266.281 | 3513.256 |
| 17 | 363 | 90 | -8.273 | -106.364 | 879.917 | 68.438 |
| 18 | 413 | 178 | 41.727 | -18.364 | -766.264 | 1741.165 |
| 19 | 457 | 176 | 85.727 | -20.364 | -1745.719 | 7349.165 |
| 20 | 308 | 132 | -63.273 | -64.364 | 4072.463 | 4003.438 |
| 21 | 379 | 248 | 7.727 | 51.636 | 399.008 | 59.711 |
| 22 | 322 | 231 | -49.273 | 34.636 | -1706.628 | 2427.802 |
| 23 | 342 | 233 | -29.273 | 36.636 | -1072.446 | 856.893 |
| 24 | 418 | 226 | 46.727 | 29.636 | 1384.826 | 2183.438 |
| 25 | 309 | 145 | -62.273 | -51.364 | 3198.554 | 3877.893 |
| 26 | 462 | 229 | 90.727 | 32.636 | 2961.008 | 8231.438 |
| 27 | 432 | 159 | 60.727 | -37.364 | -2268.992 | 3687.802 |
| 28 | 403 | 224 | 31.727 | 27.636 | 876.826 | 1006.620 |
| 29 | 385 | 230 | 13.727 | 33.636 | 461.736 | 188.438 |
| 30 | 373 | 90 | 1.727 | -106.364 | -183.719 | 2.983 |
| 31 | 374 | 148 | 2.727 | -48.364 | -131.901 | 7.438 |
| 32 | 388 | 236 | 16.727 | 39.636 | 663.008 | 279.802 |
| 33 | 418 | 183 | 46.727 | -13.364 | -624.446 | 2183.438 |
| **Jumlah** | 12252 | 6480 | 6.82121E-13 | -3.41061E-13 | 25546.72727 | 109272.5455 |
| **Rata-rata** | 371.273 | 196.364 |

Berdasarkan data latih pada **tabel 4.2**, maka nilai dan bisa dihitung untuk kemudian digunakan sebagai persamaan regresi. Adapun masing-masing nilai dan yang akan digunakan adalah.

Sehingga persamaan regresi linear yang dihasilkan pada penelitian ini dapat dilihat pada persamaan (4.1).

|  |  |
| --- | --- |
|  | (4.1) |

Di mana merupakan estimasi berat badan sapi dalam satuan kilogram dan merupakan panjang badan sapi dalam satuan *pixel*. Dengan menggunakan persamaan (4.1), maka estimasi berat badan sapi menggunakan data uji menunjukkan hasil seperti pada **tabel 4.3.**

**Tabel 4.3.** Hasil Estimasi Berat Badan Sapi Menggunakan Data Uji

|  |  |  |
| --- | --- | --- |
| **No.** | **(Panjang Badan dalam Pixel)** | **(Berat Badan Prediksi dalam Kg)** |
| 1 | 369 | 195.83 |
| 2 | 335 | 187.88 |
| 3 | 380 | 198.40 |
| 4 | 331 | 186.95 |
| 5 | 352 | 191.86 |
| 6 | 346 | 190.46 |
| 7 | 427 | 209.39 |
| 8 | 332 | 187.18 |
| 9 | 324 | 185.31 |
| 10 | 353 | 192.09 |

Untuk menganalisis seberapa baik model yang telah dihasilkan, maka berat badan prediksi akan dicari deviasi absolutnya pada tiap poin data. Dengan menggunakan persamaan (3.8), maka didapatkan MAE sebesar 6.61 kg yang disajikan pada **tabel 4.4.**

**Tabel 4.4.** Hasil MAE Data Uji

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **(Berat Badan Aktual dalam Kg)** | **(Berat Badan Prediksi dalam Kg)** | **(Deviasi Absolut dalam Kg)** |
| 1 | 196 | 195.83 | 0.17 |
| 2 | 192 | 187.88 | 4.12 |
| 3 | 188 | 198.40 | 10.40 |
| 4 | 187 | 186.95 | 0.05 |
| 5 | 202 | 191.86 | 10.14 |
| 6 | 195 | 190.46 | 4.54 |
| 7 | 217 | 209.39 | 7.61 |
| 8 | 189 | 187.18 | 1.82 |
| 9 | 174 | 185.31 | 11.31 |
| 10 | 208 | 192.09 | 15.91 |
| ***Mean Absolute Error*** | | | 6.61 |

Jika hasil prediksi yang didapatkan menggunakan regresi linear dibandingkan dengan algoritma SVR kernel linear, maka dapat dilihat pada **tabel 4.5.** bahwa terdapat sedikit penurunan MAE jika menggunakan SVR.

**Tabel 4. 5.** Perbandingan Prediksi Regresi Linear dan SVR

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **(Berat Badan Aktual dalam Kg)** | **(Berat Badan Prediksi dalam Kg)** | **(Berat Badan Prediksi dalam Kg)** | **(Deviasi Absolut dalam Kg)** | **(Deviasi Absolut dalam Kg)** |
| 1 | 196 | 195.83 | 201.59 | 0.17 | 5.59 |
| 2 | 192 | 187.88 | 192.29 | 4.12 | 0.29 |
| 3 | 188 | 198.40 | 204.60 | 10.40 | 16.60 |
| 4 | 187 | 186.95 | 191.20 | 0.05 | 4.20 |
| 5 | 202 | 191.86 | 196.94 | 10.14 | 5.06 |
| 6 | 195 | 190.46 | 195.30 | 4.54 | 0.30 |
| 7 | 217 | 209.39 | 217.46 | 7.61 | 0.46 |
| 8 | 189 | 187.18 | 191.47 | 1.82 | 2.47 |
| 9 | 174 | 185.31 | 189.28 | 11.31 | 15.28 |
| 10 | 208 | 192.09 | 197.22 | 15.91 | 10.78 |
| ***Mean Absolute Error* Regresi Linear** | | | | 6.61 |  |
| ***Mean Absolute Error* SVR Kernel Linear** | | | |  | 6.10 |

Jika poin data ke 3 dan 10 yang memiliki jumlah deviasi paling tinggi pada **tabel 4.5.** dihilangkan dari data uji, maka nilai MAE yang dihasilkan dapat menurun menjadi 4.97 pada regresi linear dan 4.21 pada SVR. Hasil perbandingan prediksi setelah dua data poin dihilangkan, dapat dilihat pada **tabel 4.6.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **(Berat Badan Aktual dalam Kg)** | **(Berat Badan Prediksi dalam Kg)** | **(Berat Badan Prediksi dalam Kg)** | **(Deviasi Absolut dalam Kg)** | **(Deviasi Absolut dalam Kg)** |
| 1 | 196 | 195.83 | 201.59 | 0.17 | 5.59 |
| 2 | 192 | 187.88 | 192.29 | 4.12 | 0.29 |
| 3 | 187 | 186.95 | 191.20 | 0.05 | 4.20 |
| 4 | 202 | 191.86 | 196.94 | 10.14 | 5.06 |
| 5 | 195 | 190.46 | 195.30 | 4.54 | 0.30 |
| 6 | 217 | 209.39 | 217.46 | 7.61 | 0.46 |
| 7 | 189 | 187.18 | 191.47 | 1.82 | 2.47 |
| 8 | 174 | 185.31 | 189.28 | 11.31 | 15.28 |
| ***Mean Absolute Error* Regresi Linear** | | | | 4.97 |  |
| ***Mean Absolute Error* SVR Kernel Linear** | | | |  | 4.21 |

Untuk melihat fitur mana yang berkorelasi secara linear dengan berat badan sapi, maka digunakan koefisien korelasi Pearson (). Adapun hasil koefisien korelasi Pearson () untuk satu variabel, dua variabel, dan tiga variabel dapat dilihat pada **tabel 4.7.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Fitur (*pixel*)** | **Metode** | **MAE (Kg)** | **Koefisien Korelasi (** |
| 1 | PB | Regresi Linear | 6.61 | 0.72 |
| 2 | TB | Regresi Linear | 10.02 | 0.06 |
| 3 | KB | Regresi Linear | 9.19 | 0.60 |
| 4 | LB | Regresi Linear | 7.61 | 0.56 |
| 5 | PB, TB | Regresi Linear Multivariat | 7.41 | 0.49 |
| 6 | PB, KB | Regresi Linear Multivariat | 9.07 | 0.62 |
| 7 | PB, LB | Regresi Linear Multivariat | 6.77 | 0.73 |
| 8 | TB, KB | Regresi Linear Multivariat | 9.19 | 0.60 |
| 9 | TB, LB | Regresi Linear Multivariat | 10.87 | -0.16 |
| 10 | KB, LB | Regresi Linear Multivariat | 10.18 | 0.56 |
| 11 | PB, TB, KB | Regresi Linear Multivariat | 9.09 | 0.62 |
| 12 | PB, TB, LB | Regresi Linear Multivariat | 10.07 | 0.09 |
| 13 | TB, KB, LB | Regresi Linear Multivariat | 10.72 | 0.45 |
| 14 | PB, TB, KB, LB | Regresi Linear Multivariat | 11.13 | 0.44 |

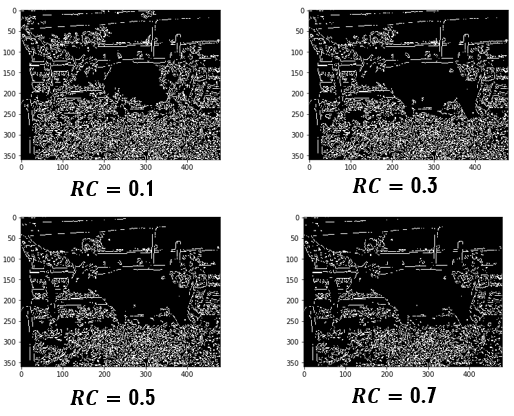
Dengan PB merupakan panjang badan, TB merupakan tinggi badan, KB merupakan keliling penampang samping, dan LB merupakan luas area badan dari samping.

# Pembahasan

Proses segmentasi citra merupakan proses yang sangat krusial pada penelitian ini. Alasannya adalah fitur panjang badan, keliling penampang samping, dan luas area samping sapi dalam satuan *pixel* merupakan fitur yang diekstrak sehingga perubahan yang kecil pada jumlah *pixel* yang diekstrak, bisa mempengaruhi hasil MAE. Sehingga pada penelitian ini disajikan hubungan antara parameter-parameter yang berpengaruh terhadap hasil segmentasi citra dengan MAE yang dihasilkan. Selain itu, disajikan juga perbandingan antara penggunaan fitur panjang badan, keliling penampang samping, dan luas area samping sapi

## Hubungan RC dengan MAE

Berdasarkan percobaan yang dilakukan, semakin kecil nilai RC maka semakin banyak tepi yang dihasilkan. Hal ini disebabkan karena nilai rasio yang kecil menyebabkan nilai *low threshold canny* yang dihasilkan menjadi kecil sehingga kemungkinan terdeteksinya *weak edge* dan *strong edge* lebih besar. **Gambar 4.1.** menyajikan perbedaan citra keluaran yang dihasilkan oleh nilai RC yang berbeda-beda. Tampak bahwa nilai RC yang kecil menyebabkan banyak tepi yang terdeteksi pada bagian badan sapi. Hal ini akan membuat hasil ekstraksi fitur panjang sapi akan semakin kecil dari yang seharusnya.



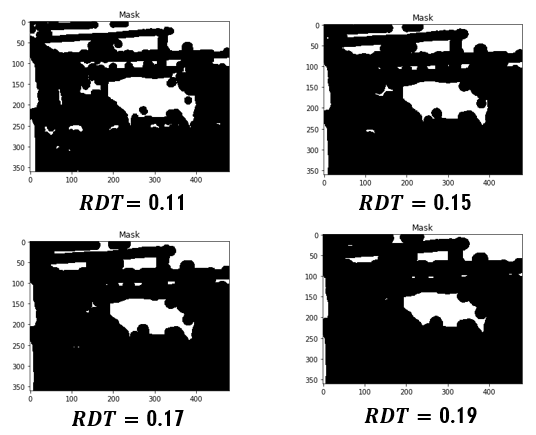
**Gambar 4.1.** Perbandingan Citra Keluaran dengan RC yang Berbeda: a) RC = 0.1; b) RC = 0.3; c) RC = 0.5; d) RC = 0.7

**Gambar 4.2.** menunjukkan bahwa MAE minimum yang dihasilkan berada pada nilai RC 0.6. Nilai di atas 0.6 akan memiliki lebih sedikit tepi dan sudah tidak memberikan pengaruh yang signifikan terhadap MAE. Hal ini dikarenakan hasil segmentasinya tidak memiliki perbedaan yang signifikan dan hasil panjang badan yang sama pada kebanyakan citra latih.

**Gambar 4.2.** Grafik Hubungan RC dengan MAE

## Hubungan RDT dengan MAE

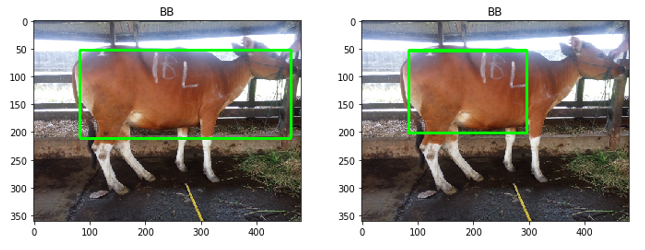
Berdasarkan percobaan yang dilakukan pada nilai RDT, didapatkan bahwa semakin besar nilai RDT maka semakin kecil blob yang dihasilkan. Hal ini disebabkan karena rasio yang besar akan menyebabkan nilai *threshold* yang dihasilkan juga besar. Sehingga akan semakin banyak *pixel* dengan intensitas di bawah nilai *threshold* yang akan dianggap sebagai *background* dan diberi nilai intensitas 0. **Gambar 4.3.** menyajikan perbedaan citra keluaran yang dihasilkan oleh nilai RDT yang berbeda-beda.



**Gambar 4.3.** Perbandingan Citra Keluaran dengan RDT yang Berbeda: a) RC = 0.11; b) RC = 0.15; c) RC = 0.17; d) RC = 0.19

**Gambar 4.4.** menunjukkan bahwa MAE minimum yang dihasilkan berada pada nilai RDT 0.15. Nilai di atas 0.16 cenderung tidak stabil dikarenakan terdapat perubahan yang signifikan pada hasil segmentasi citra yang menyebabkan berubahnya panjang badan dalam *pixel* secara drastis. **Gambar 4.5.** merupakan perbandingan hasil *bounding box* pada RDT 0.15 yang menghasilkan panjang 379 *pixel* dengan RDT 0.17 yang menghasilkan panjang 213 *pixel*.

**Gambar 4.4.** Grafik Hubungan RDT dengan MAE



**Gambar 4.5.** Perbandingan Hasil *Bounding Box*: a) RDT 0.15; b) RDT 0.17

## Hubungan IC dengan MAE

Berdasarkan percobaan yang telah dilakukan terhadap jumlah iterasi *closing*, didapatkan bahwa hanya terdapat beberapa citra yang terpengaruh pada proses *closing* ini. Sehingga grafik yang ada pada **gambar 4.6.** menunjukkan perubahan yang tidak terlalu signifikan antar beberapa nilai. Nilai IC yang terlalu sedikit tidak akan memberikan efek yang diinginkan, yaitu menyatukan dua bagian badan sapi yang terpisah oleh derau. Sedangkan nilai IC yang terlalu tinggi juga akan menyebabkan terjadinya penggabungan antara objek badan sapi dan objek non-sapi yang ada seperti pada **gambar 4.7.** Hal ini akan membuat panjang sapi menjadi lebih banyak dari yang seharusnya dan memberikan pengaruh terhadap MAE yang dihasilkan.

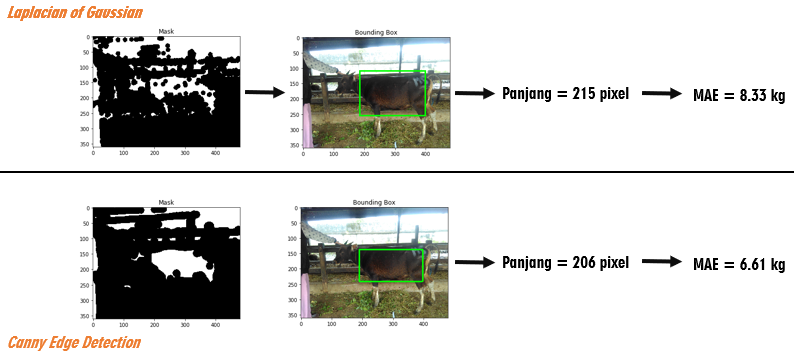
**Gambar 4.6.** Grafik Hubungan IC dengan MAE



**Gambar 4.7.** Citra keluaran dengan IC 5

## Perbandingan Hasil Segmentasi Menggunakan *Canny* dan *Laplacian of Gaussian*

Dibandingkan dengan LoG, *Canny* menghasilkan tepi citra yang lebih tegas. Citra yang dihasilkan menggunakan LoG menghasilkan banyak tepi yang kurang tegas sehingga memunculkan banyak tepi di area badan sapi. Hal ini akan menghasilkan keluaran yang destruktif dibandingkan dengan *canny*. Pada **gambar 4.8** setelah fitur panjang badan sapi diekstrak, maka terdapat perbedaan, pada LoG menghasilkan panjang 215 *pixel* dan *Canny* menghasilkan panjang 206 *pixel.* Sapi ini tergolong sapi yang kecil dengan berat badan 144 kg. Semakin panjang *pixel* nya maka semakin banyak angka berat badan yang dihasilkan. Sehingga *Canny* masih lebih baik dibandingkan dengan LoG. Terbukti pada hasil prediksi berat badan sapi ini pada proses pelatihan menggunakan LoG menghasilkan prediksi 190 kg, sedangkan *Canny* menghasilkan prediksi 157 kg. Selain itu, hasil MAE pada proses pengujian menghasilkan 8.33 kg untuk LoG dan 6.61 kg untuk *Canny.*

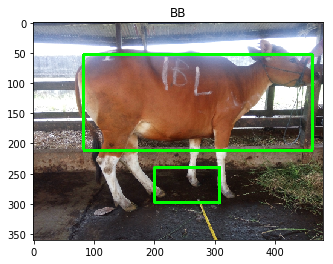


**Gambar 4.8.** Perbandingan Hasil Segmentasi dan MAE Canny dan LoG

## Perbandingan Hasil *Bounding Box* Menggunakan *Euclidean Distance* dan Rentang *Centroid*

Berdasarkan percobaan yang telah dilakukan terhadap kedua metode dalam menentukan *bounding box,* maka didapatkan bahwa pemberian *bounding box* menggunakan *euclidean distance* memerlukan 21,15 detik dan rentang *centroid* memerlukan 17,63 detik untuk memproses 33 data. Walaupun menggunakan rentang *centroid* memerlukan waktu komputasi yang lebih sedikit, tetapi terdapat beberapa hal yang perlu dipertimbangkan.

Dengan menggunakan rentang *centroid*, maka penetapan rentang yang tepat harus dilakukan secara manual menggunakan *trial and error*. Rentang yang terlalu besar, akan menyebabkan terdapat beberapa *bounding box* yang terdeteksi seperti pada **gambar 4.9.** Sedangkan, jika terlalu kecil, maka akan menyebabkan tidak terdeteksinya blob yang diinginkan sehingga menghasilkan nilai 0 pada fitur panjang badan sapi.



**Gambar 4.9.** Hasil Pemberian Rentang yang Teralu Besar

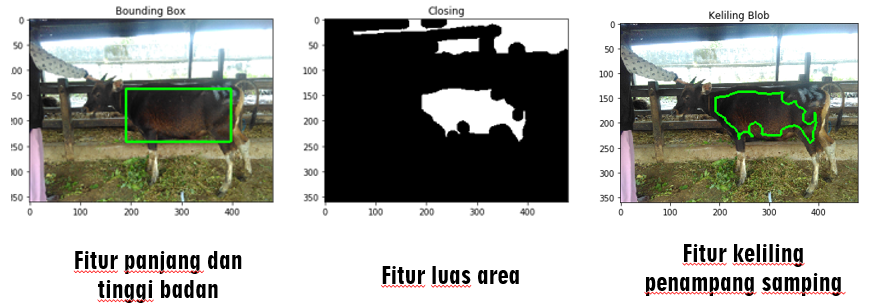
Sedangkan jika menggunakan *euclidean distance*,maka semua komputasi dan perhitungan parameter akan dilakukan oleh sistem. Sehingga pengguna tidak lagi harus menentukan rentang *centroid* secara manual.

## Perbandingan *Mean Absolute Error* Menggunakan Fitur Panjang, Keliling Penampangdan Luas Badan Samping

Tiga fitur diekstrak dalam satuan *pixel*, yaitu panjang badan, tinggi badan, keliling penampang, dan luas badan yang masing-masing didapatkan dari citra samping sapi. Setelah diekstrak, fitur-fitur ini masing-masing akan melewati proses pelatihan dan pengujian menggunakan regresi linear. Adapun hasil yang diperoleh dapat dilihat pada **gambar 4.10.**

**Gambar 4.10.** Grafik Perbandingan MAE Terhadap Empat Fitur

Berdasarkan percobaan yang telah dilakukan, didapatkan bahwa fitur panjang badan sapi merupakan fitur yang menghasilkan MAE yang paling minimum. Hal ini dikarenakan panjang badan dari samping merupakan fitur yang paling dapat diekstrak dengan baik dibandingkan kedua fitur lainnya. Berdasarkan **gambar 4.11**, didapatkan bahwa terdapat informasi yang hilang ketika mengekstrak luas area blob dan keliling penampang badan akibat proses segmentasi yang destruktif. Sedangkan fitur panjang badan, masih dapat diekstrak dengan baik karena mengambil panjang maksimal dari blob yang dideteksi.



**Gambar 4.11.** Perbandingan Hasil Ekstraksi Keempat Fitur

## Perbandingan *Mean Absolute Error* Menggunakan *Support Vector Regression* dan Regresi Linear

Perbandingan MAE dilakukan terhadap regresi linear, SVR kernel linear, SVR kernel RBF, dan SVR kernel *sigmoid*. Adapun nilai parameter yang digunakan pada SVR sama pada setiap kernel yaitu C=1.0 dan epsilon=11. Nilai yang digunakan merupakan nilai terbaik hasil *trial and error* yang menghasilkan MAE yang kecil. Adapun hasil yang didapatkan dari keempat percobaan tersebut pada citra uji dapat dilihat pada **gambar 4.12.**

**Gambar 4.12.** Grafik Perbandingan Metode Estimasi Berat Badan Sapi

Berdasarkan percobaan yang dilakukan terhadap beberapa metode regresi, didapatkan bahwa SVR dengan kernel linear menghasilkan MAE yang paling kecil. Walaupun hasilnya tidak terlalu jauh dibandingkan dengan hasil regresi linear. Sedangkan kedua kernel RBF dan *sigmoid* menghasilkan MAE yang cukup tinggi. Hal ini dikarenakan data yang dimasukkan merupakan data yang dapat dipisahkan secara linear. Sehingga baik regresi linear dan SVR kernel linear dapat memberikan hasil yang lebih baik dibandingkan SVR kernel RBF dan *sigmoid* yang dikhususkan untuk data masukan yang tidak dapat dipisahkan secara linear. Adapun perbedaan MAE antara regresi linear dan SVR kernel linear dikarenakan SVR mengaplikasikan *boundary line* sehingga memiliki parameter untuk memberikan nilai batas toleransi terhadap data mana saja yang akan digunakan dalam proses pengujian.

# BAB V PENUTUP

1. **Kesimpulan**

Dari hasil analisis yang telah dilakukan dalam pengujian sistem estimasi berat badan sapi otomatis menggunakan teknik pengolahan citra, dapat disimpulkan bahwa:

1. Segmentasi merupakan proses yang krusial dalam menghasilkan sistem dengan MAE yang minimum. Adapun metode segmentasi yang digunakan adalah gabungan antara *canny edge detection, distance transform,* dan *connected component labeling.* Hasil segmentasi terbaik didapatkan dengan menggunakan parameter RC = 0.6, RDT = 0.15, dan IC = 4.
2. Berdasarkan tiga skenario pengambilan data yang didasarkan pada jarak yang berbeda, maka skenario 1 dengan jarak pengambilan data 1.5 meter merupakan skenario terbaik dengan MAE terendah 6.61 kg menggunakan metode regresi linear dan 6.1 kg menggunakan SVR kernel linear. Di mana jarak ini akan mengurangi derau yang ada di sekeliling sapi. Adapun fitur terbaik yang digunakan dalam mengestimasi berat badan sapi menggunakan pengolahan citra adalah fitur panjang sapi dengan nilai . Karena jika dibandingkan dengan luas area badan dan keliling penampang samping sapi, fitur panjang ini yang paling tahan akan destruktifitas pada proses segmentasi. Sedangkan dengan menggunakan gabungan fitur panjang dan luas badan sapi menghasilkan nilai .
3. **Saran**

Sehubungan dengan selesainya proses pembuatan skripsi ini, penulis bermaksud menyampaikan beberapa saran kepada para pembaca yakni:

* + - 1. Proses pengambilan data sebisa mungkin dilakukan pada posisi sapi yang sama untuk semua poin data. Posisi yang ideal adalah kepala sapi terangkat ke depan secara horizontal sehingga hasil ekstraksi fitur panjang dalam *pixel* akan lebih mendekati dengan hasil pengukuran dimensi tubuh.
      2. Untuk mendapatkan hasil fitur yang akurat, maka sebaiknya terdapat tanda seperti garis pada badan sapi yang menjadi acuan dalam menentukan rasio konversi *pixel* ke cm.
      3. Perlu dilakukan percobaan terhadap proses segmentasi yang lain sehingga lebih adaptif terhadap lingkungan yang ekstrim di mana terdapat banyak objek non-sapi yang berada di sekitar sapi.
      4. Sistem yang telah dibuat oleh peneliti dapat dikembangkan dengan pemberian jarak relatif terhadap sapi sehingga kedepannya semua proses akan dapat berjalan secara otomatis pada jarak yang berbeda-beda dengan MAE yang kecil.

# DAFTAR PUSTAKA

Alkautsar, Muhammad Taufiq, Bambang Hidayat, dan Sjafril Darana. 2016. “Estimasi Berat Karkas Sapi Berdasarkan Segmentasi K-Means Clustering dengan Menggunakan Metode Klasifikasi Support Vector Machine Multiclass.” *e-Proceeding of Engineering* 1927-1936.

Arcelli, C., G. S. di Baja, dan L. Serino. 2011. “Distance-driven Skeletonization in Voxel Images.” *IEEE Transactions on Pattern Analysis and Machine Intelligence* 709-720.

Bhattacharyya, Indresh. 2018. *Medium.* 29 Juni. Diakses Agustus 1, 2019. https://medium.com.

Cook, John D. 2009. *John D. Cook Consulting.* 24 August. Diakses Juli 1, 2019. https://www.johndcook.com.

Courtney, Jane. 2001. “Application of Digital Image Processing to Marker-free Analysis of Human Gait.”

Darmawan, Aan, dan Angki Dwi Saptiani. 2010. “Perbandingan Metode Pergeseran Rata-Rata, Pergeseran Logaritma, Dan Alpha Blending Dalam Proses Metamorfosis Dari Dua Gambar Dijital.” *Maranatha Electrical Engineering Journal* 81-95.

Fisher, Robert, Simon Perkins, Ashley Walker, dan Erik Wolfart. 2003. *Hypermedia Image Processing Reference.* Diakses July 3, 2019. https://homepages.inf.ed.ac.uk/.

Francis, J., S. Sibanda, dan T. Kristensen. 2002. “Estimating Body Weight of Cattle Using Linear Body Measurements.” *Zimbabwe Veterinary Journal* (Zimbabwe Veterinary Journal ) 15-21.

Greensted, Andrew. 2010. *The Lab Book Pages.* 17 June. Diakses Juli 5, 2019. http://www.labbookpages.co.uk.

Henarta, Mutia, Bambang Hidayat, dan Sjafril Darana. 2016. “Estimasi Berat Karkas Sapi Berdasarkan Segmentasi Mean Shift dengan Klasifikasi Support Vector Machine Linear.” *e-Proceeding of Engineering* 1755-1762.

Kamal, Mazid, dan Ruri Suko Basuki. 2007. “Segmentasi Citra Daun Tembakau Berbasis Deteksi Tepi Menggunakan Algoritma Canny.” Semarang: Universitas Dian Nuswantoro.

Lafesto, Deddy B., dan Markus Daud Letik. 2017. “A Measuring Weight Model of Timor's Beef Cattle Based on Image.” *International Journal of Engineering and Technology* 677-688.

McNitt, J.I. 1974. *Livestock Husbandry Techniques.* Suffolk: The Chaucer Press.

Murtidjo, B.A. 1990. *Beternak Sapi Potong.* Kanisius: Yogyakarta.

Mustafid, Ahmad, dan Shofwatul Uyun. 2018. “Sistem Pengolahan Citra Digital untuk Menentukan Bobot Sapi Menggunakan Metode Titik Berat.” *Jurnal Teknologi Informasi dan Ilmu Komputer* 677-686.

Ozkaya, Serkan, Wojciech Neja, Sylwia Krezel-Czopek, dan Adam Oler. 2016. “Estimation of Bodyweight from Body Measurements and Determination of Body Measurements on Limousin Cattle Using Digital Image Analysis.” *Animal Production Science* 2060-2063.

Pradana, Zein Hanni, Bambang Hidayat, dan Sjafril Darana. 2016. “Beef Cattle Weight Determine by Using Digital Image Processing.” *International Conference on Control, Electronics, Renewable Energy and Communications* 179-184.

Raharjo, Listianto, Bambang Hidayat, dan Muhammad Fatah W. 2019. “Estimasi Bobot Ternak Sapi dengan Metode Deformable Template dan Klasifikasi Support Vector Machine Multiclass.” *e-Proceeding of Engineering* 712-719.

Rahmah, Ambar Aulia Fauziah, Rochadi Tawaf, dan Muhamad Fatah Wiyatna. 2016. “Hubungan Antara Kondisi Tubuh dan Bobot Badan dengan Harga Jual Sapi Pasundan.” (Universitas Padjajaran).

Renasari, Mona, Bambang Hidayat, dan Sjafril Darana. 2017. “Estimasi Berat Karkas dan Klasifikasi Daging Sapi Berdasarkan Deteksi Tepi Canny, Discrete Wavelet Transform, dan Support Vector Machine.” *e-Proceeding of Engineering* 1694-1701.

Scikit-learn Developers. t.thn. *Scikit-learn.* Diakses Agustus 3, 2019. https://scikit-learn.org.

Zurahmah, Nani, dan Enos The. 2011. “Pendugaan Bobot Badan Calon Pejantan Sapi Bali Menggunakan Dimensi Ukuran Tubuh.” *Buletin Peternakan* 160-164.

# LAMPIRAN