

**PERANCANGAN SISTEM PENGOLAHAN CITRA DIGITAL UNTUK
MENENTUKAN BOBOT SAPI MENGGUNAKAN METODE *CANNY EDGE*
*DETECTION***

Skripsi
untuk memenuhi sebagian persyaratan
mencapai derajat Sarjana S-1
Program Studi Teknik Informatika



disusun oleh:

Ahmad Mustafid

12650021

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SUNAN KALIJAGA
YOGYAKARTA
2016**



Universitas Islam Negeri Sunan Kalijaga

FM-UINSK-BM-05-07/R0

PENGESAHAN SKRIPSI/TUGAS AKHIR

Nomor : UIN.02/D.ST/PP.01.1/1665/2016

Skripsi/Tugas Akhir dengan judul : Perancangan Sistem Pengolahan Citra Digital Untuk Menentukan Bobot Sapi Menggunakan Metode *Canny Edge Detection*

Yang dipersiapkan dan disusun oleh :
Nama : Ahmad Mustafid
NIM : 12650021
Telah dimunaqasyahkan pada : Selasa, 3 Mei 2016
Nilai Munaqasyah : A
Dan dinyatakan telah diterima oleh Fakultas Sains dan Teknologi UIN Sunan Kalijaga

TIM MUNAQASYAH :

Ketua Sidang

Dr. Shofwatul Uyun, M.Kom
NIP. 19820511 200604 2 002

Penguji I

M. Mustakim, M.T
NIP.19790331 200501 1 004

Penguji II

M. Didik R. Wahyudi, M.T
NIP. 19760812 200901 1 015

Yogyakarta, 11 Mei 2016
UIN Sunan Kalijaga
Fakultas Sains dan Teknologi
Dekan



Dr. Marziq Sa'id Nahdi, M.Si
NIP. 19550427 198403 2 001



SURAT PERSETUJUAN SKRIPSI/TUGAS AKHIR

Hal : Persetujuan Skripsi
Lamp : 1 Bendel Laporan Skripsi

Kepada
Yth. Dekan Fakultas Sains dan Teknologi
UIN Sunan Kalijaga Yogyakarta
di Yogyakarta

Assalamu'alaikum wr. wb.

Setelah membaca, meneliti, memberikan petunjuk dan mengoreksi serta mengadakan perbaikan seperlunya, maka kami selaku pembimbing berpendapat bahwa skripsi Saudara:

Nama : Ahmad Mustafid
NIM : 12650021
Judul Skripsi : Perancangan Sistem Pengolahan Citra Digital untuk Menentukan Bobot Sapi Menggunakan Metode *Canny Edge Detection*

sudah dapat diajukan kembali kepada Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Sunan Kalijaga Yogyakarta sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata Satu dalam Prodi Teknik Informatika

Dengan ini kami berharap agar skripsi/tugas akhir Saudara tersebut di atas dapat segera dimunaqsyahkan. Atas perhatiannya kami ucapkan terima kasih.

Wassalamu'alaikum wr.

wb.

Yogyakarta, 25 April 2016

Pembimbing

Dr. Shofwatul Uyan, S.T., M.Kom

NIP. 19820511 200604 2 002

PERNYATAAN KEASLIAN SKRIPSI

Yang bertanda tangan di bawah ini:

Nama : Ahmad Mustafid
NIM : 12650021
Program Studi : Teknik Informatika
Fakultas : Sains dan Teknologi

Menyatakan bahwa skripsi dengan judul **"Perancangan Sistem Pengolahan Citra Digital untuk Menentukan Bobot Sapi Menggunakan Metode *Canny Edge Detection*"** tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Yogyakarta, 25 April 2016

Yang Menyatakan



Ahmad Mustafid
NIM. 12650021

KATA PENGANTAR

Puji Syukur kehadirat Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi dengan judul **”Perancangan Sistem Pengolahan Citra Digital untuk Menentukan Bobot Sapi Menggunakan Metode *Canny Edge Detection*”** sebagai salah satu syarat untuk mencapai gelar kesarjanaan pada program studi Teknik Informatika UIN Sunan Kalijaga Yogyakarta. *Shalawat* serta salam semoga tercurahkan kepada junjungan Nabi Muhammad SAW beserta seluruh keluarga dan sahabat beliau.

Penulis menyadari bahwa apa yang saya lakukan dalam penyusunan laporan tugas akhir ini masih terlalu jauh dari kata sempurna. Oleh karena itu, saya sangat mengharap kritik dan saran yang berguna dalam penyempurnaan sistem ini dimasa yang akan datang. Semoga apa yang telah saya lakukan ini dapat bermanfaat bagi pembaca.

Tak lupa penyusun juga mengucapkan terima kasih kepada pihak-pihak yang telah membantu dalam penyelesaian skripsi ini, baik secara langsung atau tidak langsung. Ucapan terima kasih penyusun sampaikan kepada:

1. Bapak Prof. Drs. H. Akh. Minhaji, M.A.,Ph.D., selaku Rektor UIN Sunan Kalijaga Yogyakarta.
2. Ibu Dr. Hj. Maizer Said Nahdi, M.Si., selaku Dekan Fakultas Sains dan Teknologi UIN Sunan Kalijaga Yogyakarta.
3. Bapak Sumarsono, S.T, M.Kom., selaku Ketua Program Studi Teknik Informatika UIN Sunan Kalijaga Yogyakarta.


4. Bapak Nurochman, M.Kom., selaku Sekertaris Program Studi Teknik Informatika UIN Sunan Kalijaga Yogyakarta.
5. Bapak Aulia Faqih Rifai, M.Kom., selaku Pembimbing Akademik selama masa kuliah.
6. Ibu Dr. Shofwatul 'Uyun, S.T., M.Kom., selaku Dosen Pembimbing yang telah membimbing, memberikan koreksi dan saran kepada penyusun sehingga skripsi ini dapat terselesaikan.
7. Seluruh Dosen Program Studi Teknik Informatika UIN Sunan Kalijaga Yogyakarta, terima kasih atas ilmu yang telah diberikan.
8. Gubernur Daerah Istimewa Yogyakarta yang telah memberikan izin untuk melakukan penelitian.
9. Walikota Yogyakarta yang telah memberikan izin untuk melakukan penelitian.
10. Pemerintah Kota (Pemkot) Yogyakarta
11. Kepala Dinas Perindagkoptan Kota Yogyakarta
12. Kepala Unit Pelaksana Teknis (UPT) Pelayanan Kehewananan Kota Yogyakarta
13. Pengelola Rumah Potong Hewan Giwangan Yogyakarta
14. Bapak Drh. Supriyanto selaku Dokter hewan di RPH Giwangan yang telah memberikan penjelasan, pengarahan serta membantu penelitian ini.
15. Ayahanda Ahmad Qodri, Ibunda Umi Saidah tercinta dan adek-adekku Muhammad Rofiqul Anam dan Isyfina Ziyantifani, penulis ucapkan terima kasih atas semua yang telah kalian berikan

16. Teman-teman seperjuangan angkatan 2012 Program Studi Teknik Informatika.

17. Kakak-kakak dan adik-adik angkatan yang sudah memberikan dukungan dan membantu dalam penyelesaian skripsi ini.

Semoga Allah SWT memberikan pahala yang setimpal atas segala dorongan, bantuan, dukungan, semangat dan keyakinan yang sudah diberikan kepada penulis untuk menyelesaikan skripsi ini. Amin.

Yogyakarta, April 2016



Penulis

HALAMAN PERSEMBAHAN

Alhamdulillah rabbil'alamin atas segala nikmat dan pertolongan-Nya.

Skripsi/Tugas Akhir ini kupersembahkan kepada :

1. Bapak Ahmad Qodri, Ibu Umi Saidah yang tercinta, terimakasih atas do'a, dukungan, semangat dan semua yang telah diberikan.
2. Adek-adekku Muhammad Rofiqul Anam dan Isyfina Ziyantifani, terimakasih atas dukungan dan semangatnya.
3. Wanita shalihah yang Allah Jalla Jalaluhu siapkan untukku kelak.
4. Seluruh keluarga besar Bani Siddiq Saryani dan Bani Isyhaq.
5. Seluruh Anggota Mabes Afha, Afif, Alfian, Amik, Fani, Faris, Fuad, Saipul, Weddy.
6. Seluruh Dosen Teknik Informatika Pak Sumarsono, Pak Nurrohhman, Pak Agus, Pak Bambang, Pak Aulia, Pak Didik, Pak Mustakim, Pak Agung, Pak Rahmat, Bu Uyun, Bu Ade, terimakasih atas ilmu yang telah diberikan, semoga bermanfaat dikemudian hari.
7. Teman-teman seperjuangan Winda, Fauzi, Pamuji, Miya, Puguh, Lina, Indah, Ica, Elva, Bains, Faizin, dan seluruh angkatan Teknik Informatika Reguler dan Mandiri 2012 yang tidak bisa disebut satu persatu. Adek dan kakak angkatan, terimakasih atas kebersamaan, semangat dan dukungannya.
8. Sahabat-sahabat aku Fifit, Mila, Tyas, Koles, Hendy, Rani, Budi, Devi, dan temen-temen yang lain yang tidak bisa saya sebutkan satu persatu.
9. Teman-teman KKN Aan, Mey, Ike, Farid, Dika, Hikmah, Ayu, Riyan, Samsiyah, Septi, Afif, Wahyu.
10. Teman-teman ITTC Pak Arif, Bu Ndari, mas Cahyo, mas Habibi, mas, erfan, mbak Estri, mbak ayu, mbak sasti, mbak amel, seluruh Instruktur dan Fasilitator ICT yang tidak bisa disebut satu per satu.
11. Teman-teman Silat Merpati Putih Mas Tiar, Mas Rahmat, Affan, Teguh, Rafdhal, Agung, Upik, Windy, dan temen temen yang lain yang tidak bisa disebut satu per satu.

HALAMAN MOTTO

فَبِأَيِّ آلَاءِ رَبِّكُمَا تُكَذِّبَانِ

Maka nikmat Tuhan kamu yang manakah yang kamu dustakan?

So which of the favors of your Lord would you deny?

(Q.S. Ar-Rahman : 55)

يَا رَبِّ كُلِّ عَالَمٍ عَلَىٰ مُحَمَّدٍ ، وَأُفْتَحْ مِنَ الْكَلْبِ كُلِّ مَغْلَقٍ

Ya Robbi, limpahkanlah rahmat kepada Nabi Muhammad (saw),

Bukalah segala kebaikan yang terkunci.

Dear Allah, give your blessings to Prophet Muhammad (PBUH),

Open up all the good things that was locked

كَذَّبُوا أَنْ عِطَّةَ اللَّهِ أَحْسَنُ لَنَا

Percayalah bahwa rencana-Nya adalah yang terbaik bagi kita

Believe that Allah's plan is the best for us

DAFTAR ISI

HALAMAN JUDUL.....	i
PENGESAHAN SKRIPSI/TUGAS AKHIR.....	ii
SURAT PERSETUJUAN SKRIPSI/TUGAS AKHIR.....	iii
PERNYATAAN KEASLIAN SKRIPSI.....	iv
KATA PENGANTAR.....	v
HALAMAN PERSEMBAHAN.....	viii
HALAMAN MOTTO.....	ix
DAFTAR ISI.....	x
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR.....	xv
DAFTAR LAMPIRAN.....	xvii
DAFTAR SINGKATAN.....	xviii
INTISARI.....	xix
ABSTRACT.....	xx
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	4
1.3. Batasan Masalah.....	4
1.4. Tujuan Penelitian.....	5
1.5. Manfaat Penelitian.....	5
1.6. Keaslian Penelitian.....	6
1.7. Sistematika Penulisan.....	6

BAB II	TINJAUAN PUSTAKA DAN LANDASAN TEORI.....	8
2.1.	Tinjauan Pustaka.....	8
2.2.	Landasan Teori.....	13
2.2.1.	Pengolahan Citra Digital.....	13
2.2.2.	<i>Edge Detection</i> (Pendeteksian Tepi).....	14
2.2.3.	Metode <i>Canny Edge Detection</i>	19
2.2.4.	<i>Sharp</i>	21
2.2.5.	<i>Median Blur</i>	21
2.2.6.	Titik Berat.....	22
2.2.7.	Android.....	24
2.2.8.	OpenCV.....	25
2.2.9.	Sapi.....	25
BAB III	METODE PENELITIAN.....	28
3.1.	Desain Penelitian.....	28
3.2.	Jenis Data.....	28
3.3.	Teknik Pengumpulan Data.....	29
3.4.	Metode Analisis Data.....	29
3.5.	Kebutuhan Sistem.....	30
3.6.	Alur Kerja Penelitian.....	31
BAB IV	HASIL DAN PEMBAHASAN.....	36
4.1.	Data Awal.....	37
4.2.	Perbandingan Rumus Penentuan Berat Badan.....	38
4.2.1.	Rumus Schoorl.....	38

4.2.2. Rumus Modifikasi.....	40
4.2.3. Perbandingan Penyimpangan Pengukuran Bobot Badan.....	41
4.3. Pengembangan Sistem.....	43
4.3.1. Perancangan Sistem.....	44
4.3.2. <i>Preprocessing</i>	46
4.3.3. Konversi Satuan.....	51
4.3.4. Perbandingan Algoritma.....	52
4.4. Hasil.....	71
BAB V PENUTUP.....	80
5.1. Kesimpulan.....	80
5.2. Saran.....	81
DAFTAR PUSTAKA.....	82

DAFTAR TABEL

Tabel 2.1. Ringkasan Referensi Penelitian.....	11
Tabel 2.2 Titik berat dari benda homogen yang simetris.....	23
Tabel 4.1 Table Data Awal.....	37
Tabel 4.2. Tabel Perhitungan Berat Badan dengan Rumus Schoorl.....	38
Tabel 4.3. Tabel Perhitungan Berat Badan dengan Rumus Modifikasi.....	40
Tabel 4.4. Persentase Perbandingan antara Pengukuran dengan menggunakan rumus Schoorl dan Modifikasi.....	42
Tabel 4.5 Tabel Skenario <i>Preprocessing</i> yang digunakan.....	46
Tabel 4.6 Tabel Perhitungan Algoritma A (Titik Tengah Gambar).....	54
Tabel 4.7 Tabel Perhitungan Algoritma B (Titik Berat).....	56
Tabel 4.8 Tabel Perhitungan Algoritma C (Titik Berat 10 Piksel).....	59
Tabel 4.9 Tabel Perhitungan Algoritma D (Titik Berat Keliling Lingkaran). 62	
Tabel 4.10 Tabel Perhitungan Algoritma E (Titik Berat Keliling Elips 1).....	65
Tabel 4.11 Tabel Perhitungan Algoritma F (Titik Berat Keliling Elips 2).....	68
Tabel 4.12 Tabel Perhitungan Berat Badan Algoritma A (Titik Tengah Gambar).....	71
Tabel 4.13 Tabel Perhitungan Berat Badan Algoritma B (Titik Berat).....	72
Tabel 4.14 Tabel Perhitungan Berat Badan Algoritma C (Titik Berat 10 Piksel).....	73
Tabel 4.15 Tabel Perhitungan Berat Badan Algoritma D (Titik Berat Keliling Lingkaran).....	74

Tabel 4.16 Tabel Perhitungan Berat Badan Algoritma E (Titik Berat Keliling Elips 1).....	75
Tabel 4.17 Tabel Perhitungan Berat Badan Algoritma F (Titik Berat Keliling Elips 2).....	76

DAFTAR GAMBAR

Gambar 2.1 Representasi Perubahan Gray Dari Tepi Obyek.....	15
Gambar 2.2 Gambar Proses Konvolusi.....	18
Gambar 2.3 Ilustrasi penggunaan median filter berukuran 3x3.....	22
Gambar 3.1 Proses Akuisisi Citra.....	32
Gambar 3.2 Alur Kerja Penelitian.....	35
Gambar 4.1 Tahapan Proses Sistem.....	36
Gambar 4.2 Halaman Awal.....	44
Gambar 4.3 Halaman Proses.....	45
Gambar 4.4 Halaman Perhitungan.....	46
Gambar 4.5 Diagram Teknik Evaluasi (<i>Diagram of evaluation technique</i>)..	47
Gambar 4.6 Contoh obyek gambar sapi dan <i>Ground Truth</i> (GT) nya.....	49
Gambar 4.7 Kurva Kinerja Skenario Deteksi Tepi (<i>Edge Detection</i>).....	50
Gambar 4.8 Grafik Hasil Perbandingan Deteksi Tepi (<i>Edge Detection</i>).....	51
Gambar 4.9 Langkah Algoritma A (Titik Tengah Gambar).....	53
Gambar 4.10 Langkah Algoritma B (Titik Berat).....	55
Gambar 4.11 Langkah Algoritma C (Titik Berat 10 Piksel).....	57
Gambar 4.12 Langkah Algoritma D (Titik Berat Keliling Lingkaran).....	60
Gambar 4.13 Langkah Algoritma E (Titik Berat Keliling Elips 1).....	63
Gambar 4.14 Langkah Algoritma F (Titik Berat Keliling Elips 2).....	66
Gambar 4.15 Diagram Perbandingan Nilai Sebaran Relatif Penyimpangan (<i>error</i>).....	70

Gambar 4.16 Diagram Perbandingan Nilai Sebaran Relatif Penyimpangan (<i>error</i>) Berat Badan.....	78
Gambar 4.17 Tampilan Hasil Sistem Pengolahan Citra.....	79

DAFTAR LAMPIRAN

Lampiran A Gambar Hasil Akuisisi.....	86
Lampiran B Gambar <i>Ground Truth</i> (GT).....	88
Lampiran C <i>Sourcecode</i> Skenario <i>Preprocessing</i>	90
Lampiran D <i>Sourcecode</i> Usulan Algoritma.....	93
<i>CURICULUM VITAE</i>	96

DAFTAR SINGKATAN

PSPK	: Pendataan Sapi Perah, Sapi Potong dan Kerbau
Kementan	: Kementerian Pertanian
BPS	: Badan Pusat Statistik
IDC	: <i>International Data Corporation</i>
OpenCV	: <i>Open Source Computer Vision library</i>
JST	: Jaringan Syaraf Tiruan
OS	: <i>Operation System</i> (Sistem Operasi)
BSD	: <i>Berkeley Software Distribution</i>
PB	: Panjang Badan
LD	: Lingkar Dada
RPH	: Rumah Potong Hewan
GT	: <i>Ground Truth</i>

PERANCANGAN SISTEM PENGOLAHAN CITRA DIGITAL UNTUK MENENTUKAN BOBOT SAPI MENGGUNAKAN METODE *CANNY* *EDGE DETECTION*

Ahmad Mustafid
NIM. 12650021

INTISARI

Ternak sapi memiliki nilai ekonomis tinggi serta penting di dalam kehidupan masyarakat. Penentuan harga sapi umumnya disepakati dengan tawar menawar antara penjual dan pembeli, bukan didasarkan pada bobot sapi yang dijual. Kenyataan dilapangan menunjukkan bahwa belum adanya alat untuk menghitung bobot sapi secara praktis dan akurat sehingga kebanyakan masih menggunakan perhitungan secara kasar maupun secara kira-kira untuk menghitung bobot sapi. Oleh sebab itu penelitian ini mencoba membuat sebuah Sistem Pengolahan Citra Digital menggunakan perangkat Android untuk menentukan bobot sapi secara praktis dan akurat dengan menggunakan metode *Canny Edge Detection*.

Tahap awal peneliti mencari rumus yang paling akurat untuk menentukan berat badan/bobot sapi, kemudian melakukan proses perancangan sistem. Setelah rancangan sudah dibuat baru kita melakukan *preprocessing* dengan bantuan OpenCV untuk menentukan deteksi tepi mana yang terbaik. Langkah berikutnya melakukan konversi satuan dan mengusulkan beberapa algoritma untuk menemukan panjang badan serta lingkaran dada sehingga bisa menghitung berat badan/bobot dari sapi yang ada dalam citra/gambar.

Hasil analisis penentuan bobot sapi menggunakan rumus Schoorl dan rumus Modifikasi memiliki nilai penyimpangan bobot badan sebesar 16,87% untuk rumus Schoorl dan nilai penyimpangan bobot badan sebesar 10,58 % untuk rumus Modifikasi. Hasil analisis *preprocessing* menunjukkan bahwa Skenario 3 (Median Blur dan Canny) menunjukkan hasil yang terbaik. Hasil perhitungan citra tidak berbeda secara signifikan yaitu dengan faktor ketelitian secara statistis dengan nilai sebaran relatif penyimpangan (*error*) sebesar 8,15% untuk panjang badan sedangkan nilai sebaran relatif penyimpangan (*error*) sebesar 4,10% untuk lingkaran dada. Aplikasi pengolahan citra digital yang dibangun dapat mengetahui berat badan/bobot sapi dengan nilai sebaran relatif penyimpangan (*error*) sebesar 8,97% terhadap rumus Modifikasi.

Kata Kunci : *Edge Detection, Canny, Median Blur, OpenCV, Android, Sapi, Bobot Sapi, Pengolahan Citra Digital.*

DIGITAL IMAGE PROCESSING SYSTEM DESIGN FOR DETERMINING THE WEIGHT OF COW USING CANNY EDGE DETECTION METHOD

Ahmad Mustafid
NIM. 12650021

ABSTRACT

Cow cattle has a high economic value and its important in the life of society. Generally, determining the price of cow agreed by negotiation between the seller and the buyer, not based on the weight of cow is sold. In the fact, it seemed that there has no tool to calculate the weight of cow in practical and accurate, therefore they still use the calculation manually and approximately for finding the weight of cow. In other that this research tries to create a digital image processing system using android device for determining the weight of a cow in practical and accurate using Canny edge detection method.

The first way, the researcher is looking for the most accurate formula to determine the weight of cow, and then making the system design process. After the design has been made, we perform the processing with the help of OpenCV to determine where the best edge detection is. In next step is performing unit conversion and proposing some algorithms to find the body length and chest girth, therefore can be calculated the weight of cow in the image.

The result of analysis determining the weight of cow using Schoorl and Modification formula has the error/deviation values of weight of 16.87 % for Schoorl formula and 10.58 % for Modification formula. The result of analysis preprocessing showed that scenario 3 (Median blur and canny) get the best result. The calculation result, image does not have difference significantly, it has the accuracy factor statistically with error of 8.15 % for the body length and 4.10% for chest girth. Digital image processing application that was built can determine the weight of cow or the heavy cow with the value relative deviation (error) of 8.97 % toward Modification formula.

Keywords : Edge Detection, Canny, Median Blur, OpenCV, Android, Cow, Weight Of Cow, Digital Image Processing.

BAB I

PENDAHULUAN

1.1. Latar Belakang

Idul Adha merupakan salah satu hari besar bagi umat Islam. Umat Islam pada saat Idul Adha dianjurkan untuk berkorban dengan melakukan pemotongan hewan yaitu sapi, kambing, onta, dan lain-lain. Daging hewan kurban akan dibagi sesuai dengan syariat Islam kepada *Shohibul Qurban* (Orang yang berkorban) maupun ke *Mustahiq* (Orang yang menerima daging kurban). Umumnya hewan kurban dibeli dari peternak sapi, kambing, dan lain-lain.

Ternak sapi memiliki nilai ekonomis tinggi serta penting didalam kehidupan masyarakat. Berdasarkan hasil akhir PSPK2011 (Kementan - BPS, 2011) populasi sapi (sapi potong dan sapi perah) di Indonesia sebanyak 15,4 juta ekor. Pertumbuhan populasi sapi selama 2003–2011 mencapai 5,33 persen per tahun atau rata-rata pertambahan 655,5 ribu ekor setiap tahunnya. Ternak sapi merupakan penghasil sumber bahan makanan bagi masyarakat yaitu berupa susu dan daging potong yang biasa digunakan oleh masyarakat untuk bahan makanan sehari-hari. Ternak sapi banyak dijual oleh peternak di pasar hewan.

Ternak Sapi pada umumnya dijual berdasarkan perkiraan kasar dalam menentukan bobot sapi. Penentuan harganya umumnya disepakati lewat tawar-menawar antara penjual dan pembeli, bukan didasarkan pada bobot sapi yang dijual. Masalah yang biasanya terjadi yaitu ketika tidak adanya alat untuk menghitung bobot sapi di pasar hewan sehingga dapat menyulitkan penjual

maupun pembeli untuk menentukan bobot serta harga ternak yang akan dijual. Kenyataan dilapangan saat ini menunjukkan bahwa masih belum adanya alat untuk menghitung bobot sapi secara praktis dan akurat sehingga kebanyakan masih menggunakan perhitungan secara kasar maupun secara kira kira untuk menghitung bobot sapi.

Untuk mendapatkan cara yang lebih praktis, pada bidang teknologi dapat membantu memberikan solusi atas permasalahan tersebut. Untuk mengetahui ukuran fisik baik itu panjang badan maupun lingkaran dada yang biasa digunakan untuk menghitung berat badan kita dapat menggunakan kamera dengan cara mengambil gambar sapi yang ingin diketahui berat badannya dan selanjutnya melakukan proses pengolahan citra digital.

Pengolahan citra digital (*Image Processing*) adalah suatu proses yang digunakan untuk mengolah citra atau gambar untuk mendapatkan citra yang lebih bagus menggunakan perangkat sistem komputer. Terdapat beberapa cara untuk mengolah gambar agar mendapatkan informasi dari gambar supaya bisa digunakan dalam hal ini kita bisa menggunakan cara atau langkah deteksi tepi (*edge detection*) untuk mendapatkan tepi dari suatu citra yang kita ambil.

Edge Detection adalah proses dalam pengolahan citra digital untuk mengetahui tepi dari citra atau obyek didalam citra sehingga kita dapat mengambil informasi yang berguna dari citra tersebut. Terdapat beberapa metode dalam pendeteksian tepi yaitu metode Sobel, Prewitt, Canny, dll. Menurut Joshi dan Koju (2012) Metode deteksi tepi Canny merupakan metode yang terbaik untuk mendeteksi tepi.

Metode Canny Merupakan salah satu metode untuk mendeteksi tepi obyek. Untuk mengetahui panjang badan serta lingkaran dada obyek sapi kita terlebih dahulu mendeteksi tepi dari sapi tersebut dalam citra, sehingga memudahkan kita dalam mengambil data panjang maupun lingkaran dada obyek sapi yang berada dalam citra tersebut. Untuk melakukan hal tersebut perlu adanya sistem pengolahan citra digital yang mendukung dalam pemrosesan, pengambilan data serta perhitungan untuk menentukan berat badan sapi.

Sistem pengolahan citra digital merupakan suatu sistem yang dapat mengambil citra, memproses citra serta menampilkan informasi yang didapatkan dari suatu citra yang sudah diambil. Sistem ini bisa dibangun di komputer maupun perangkat *smartphone*.

Android merupakan sistem operasi *mobile* yang biasa digunakan dalam *smartphone*. Android merupakan sistem operasi yang mendominasi pasar serta yang paling banyak dipakai saat ini dengan nilai 82,8% dari pangsa pasar ponsel (IDC: Smartphone OS market share, 2016). Selain dipakai oleh kebanyakan orang Android juga memiliki kemampuan kamera yang bisa digunakan untuk mendapatkan citra gambar suatu obyek dalam hal ini ternak sapi secara mudah.

Penelitian ini akan mencoba membuat sebuah alat untuk menentukan bobot sapi secara praktis dan akurat pada perangkat Android dan selanjutnya dibuat sebuah Sistem Pengolahan Citra Digital dengan menggunakan metode *Canny Edge Detection* dalam pemrosesan didalamnya sehingga dapat menentukan bobot sapi secara praktis dan akurat.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang sudah dijelaskan sebelumnya, permasalahan yang dapat diangkat yaitu :

1. Bagaimana cara mengimplementasikan pengolahan citra digital pada perangkat android untuk menghitung bobot sapi dengan metode *canny edge detection*?
2. Ada berapa cara/rumus untuk menentukan bobot sapi dan seberapa besar tingkat akurasi rumus tersebut ?
3. Bagaimana cara untuk menentukan *preprocessing* yang tepat untuk menentukan deteksi tepi gambar yang cocok dalam penelitian ini ?
4. Bagaimana cara untuk menentukan panjang badan dan lingkar dada dari obyek sapi pada citra yang diambil ?
5. Seberapa tingkat akurasi penentuan bobot sapi dari aplikasi yang dibangun ?

1.3. Batasan Masalah

Hal-hal yang akan dilakukan dalam penelitian ini akan dibatasi pada beberapa batasan masalah, yaitu :

1. Pengambilan citra/gambar sapi dilakukan dengan jarak 1,5 m
2. Aplikasi yang akan dikembangkan menggunakan sistem operasi android
3. Obyek yang diteliti adalah sapi dengan jenis LIMOUSIN (Diamond Limousine), SIMMENTAL (Metal), PO (Peranakan Ongole), FH (Friesian Holstein/Fries Holland)

1.4. Tujuan Penelitian

Tujuan dari penelitian ini adalah

1. Mengetahui cara mengimplementasikan pengolahan citra digital pada perangkat android untuk menghitung bobot sapi dengan metode *canny edge detection*.
2. Mengetahui cara/rumus untuk menentukan bobot sapi dan mengetahui seberapa besar tingkat akurasi rumus tersebut.
3. Mengetahui cara untuk menentukan *preprocessing* yang tepat untuk menentukan deteksi tepi gambar yang cocok dalam penelitian.
4. Mengetahui cara untuk menentukan panjang badan dan lingkaran dada dari obyek sapi pada citra yang diambil.
5. Menghitung seberapa tingkat akurasi penentuan bobot sapi dari aplikasi yang dibangun.

1.5. Manfaat Penelitian

Manfaat penelitian yang diharapkan yaitu agar dapat menemukan metode yang paling tepat untuk menghitung bobot/berat badan dari ternak sapi. Kedepannya diharapkan hasil dari penelitian ini berupa aplikasi *mobile* android yang bisa digunakan untuk menghitung bobot sapi.

Hasil dari penelitian ini adalah berupa aplikasi yang nantinya bisa menghitung bobot/berat badan sapi sebelum sapi tersebut disembelih ataupun dipilih saat membeli sapi di pasar hewan ternak.

1.6. Keaslian Penelitian

Penelitian tentang pengolahan citra digital untuk menentukan bobot sapi menggunakan metode *canny edge detection* sejauh pengetahuan penulis belum pernah dilakukan sebelumnya. Model penelitian tentang pengolahan citra digital pada obyek sapi pernah dilakukan sebelumnya tetapi perbedaannya terdapat pada metode yang digunakan, proses perhitungan, aplikasi yang digunakan, output hasil dan data-data lainnya.

1.7. Sistematika Penulisan

Laporan penelitian tugas akhir ini disusun secara sistematis dan dibagi dalam beberapa bagian bab. Penulisan laporan tugas akhir ini memiliki urutan yang dimulai dari BAB I sampai BAB V

BAB I. PENDAHULUAN

Bagian ini menerangkan tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, keaslian penelitian dan sistematika penulisan

BAB II. TINJAUAN PUSTAKA DAN LANDASAN TEORI

Bagian ini berisi tentang teori-teori yang digunakan dalam penelitian ini. Terdiri dari teori pengolahan citra digital, *edge detection* (pendeteksian tepi), *canny edge detection*, *sharp*, *median blur*, titik berat, android, OpenCV, dan penjelasan mengenai obyek sapi.

BAB III. METODE PENELITIAN

Bagian ini berisi tentang uraian rinci tentang alat dan bahan penelitian serta memberikan penjelasan mengenai detail langkah-langkah yang dilakukan

untuk mencapai tujuan dan simpulan akhir penelitian.

BAB IV. HASIL DAN PEMBAHASAN

Pada bab ini memuat hasil dari penelitian dan pembahasan penelitian yang telah dilakukan.

BAB V. PENUTUP

Bagian ini berisi tentang kesimpulan dan saran-saran untuk penelitian selanjutnya.

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan hasil penelitian dan analisis sebelumnya, maka dapat ditarik kesimpulan sebagai berikut.

1. Aplikasi pengolahan citra digital yang dibangun dapat mengetahui bobot sapi menggunakan metode *Canny Edge Detection* dengan menggunakan variabel panjang badan dan lingkaran dada pada objek sapi.

2. Hasil analisis penentuan bobot sapi menggunakan rumus Schoorl dan rumus Modifikasi memiliki nilai penyimpangan bobot badan sebesar 16,87% untuk rumus Schoorl dan nilai penyimpangan bobot badan sebesar 10,58 % untuk rumus Modifikasi.

3. Hasil analisis *preprocessing* untuk beberapa skenario menunjukkan bahwa Skenario 3 (Median Blur dan Canny) menunjukkan hasil yang terbaik daripada skenario-skenario yang lain.

4. Hasil analisis perbandingan algoritma untuk menentukan panjang badan dan lingkaran dada hasil perhitungan citra tidak berbeda secara signifikan yaitu dengan faktor ketelitian secara statistik dengan nilai sebaran relatif penyimpangan (*error*) sebesar 8,15% untuk panjang badan sedangkan nilai sebaran relatif penyimpangan (*error*) sebesar 4,10% untuk lingkaran dada.

5. Hasil analisis penentuan berat badan/bobot sapi dari hasil perhitungan citra memiliki faktor ketelitian secara statistik dengan nilai sebaran relatif penyimpangan (*error*) sebesar 8,97% terhadap rumus Modifikasi.

5.2. Saran

Untuk pengembangan lebih lanjut sistem ini, dapat memperbaiki rumus untuk menentukan berat badan sapi yang digunakan sehingga meminimalisasi penyimpangan yang tinggi untuk hasil perhitungan aplikasi.

Mencoba metode *preprocessing* yang lain misalnya menggunakan segmentasi untuk mendapatkan hasil obyek yang lebih jelas dari gambar yang diambil. Sistem yang dikembangkan dalam penelitian ini masih terbatas untuk menghitung berat badan sapi. Untuk pengembangan sistem ini juga dapat dilakukan dengan menambah obyek yang dapat dihitung seperti kerbau, kambing dan domba.

DAFTAR PUSTAKA

- Abidin, Z. (2002) *Penggemukan Sapi Potong*. Jakarta: AgroMedia Pustaka.
- Ahmad, U. (2005) *Pengolahan Citra Digital & Teknik Pemrogramannya*. Yogyakarta: Graha Ilmu.
- Astuti, W. (2014) *Prototype Aplikasi Deteksi Buah Apel, Jeruk dan Pisang Berbasis Pengolahan Citra dengan Segmentasi Bentuk dan Warna Citra*. Yogyakarta: Universitas Gadjah Mada.
- Beeran Kutty, S., Saaidin, S., Megat Yunus, P.N.A. and Abu Hassan, S. (2014) 'Evaluation of canny and sobel operator for logo edge detection', *2014 International Symposium on Technology Management and Emerging Technologies*. Institute of Electrical & Electronics Engineers (IEEE).
- Bowyer, K., Kranenburg, C. and Dougherty, S. (1999) 'Edge Detector Evaluation Using Empirical ROC Curves', *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*. Institute of Electrical & Electronics Engineers (IEEE).
- Canny, J. (1986) 'A Computational Approach to Edge Detection', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6), pp. 679–698. doi: 10.1109/tpami.1986.4767851.
- Ensminger, M.E. and Olentine, C.G. (1980) *Feeds and Nutrition Complete*. West Sierra Avenue Clovis California: The Ensminger Publishing Company.
- Firmansyah, S. (2013) *Implementasi Pengolahan Citra Digital Sebagai Pengukur Nilai Resistor pada Sistem Pemindai Resistor Berbasis Android*. Yogyakarta: Universitas Gadjah Mada.

- Fowles, G.R. and Cassiday, G.L. (2004) *Analytical mechanics*. 7th edn. United States: Thomson Brooks/Cole.
- Hidayattullah, A. (2013) *Identifikasi Tingkat Kematangan Buah Tomat (Lycopersicon Esculentum Mill) Menggunakan Metode Pengolahan Citra dan Jaringan Saraf Tiruan*. Yogyakarta: Universitas Gadjah Mada.
- IDC: *Smartphone OS market share* (2016) Available at: <http://www.idc.com/prodserv/smartphone-os-market-share.jsp> (Diakses pada tanggal: 12 April 2016).
- Jain, R.C., Kasturi, R. and Schunck, B.G. (1995) *Machine vision*. New York: McGraw Hill Higher Education.
- Joshi, S.R. and Koju, R. (2012) 'Study and comparison of edge detection algorithms', *2012 Third Asian Himalayas International Conference on Internet*. Institute of Electrical & Electronics Engineers (IEEE).
- Kapur, S. and Thakkar, N. (2015) *Mastering OpenCV Android application programming*. United Kingdom: Packt Publishing.
- Kementan - BPS (2011) *Rilis Hasil Akhir PSPK2011*. Kementerian Pertanian - Badan Pusat Statistik.
- Lasfeto, D.B., Susanto, A. dan Agus, A. (2012) 'Aplikasi Pengolahan Citra untuk Estimasi Bobot Badan Ternak Sapi', *Buletin Peternakan*, 32(3), pp. 167–176. doi: 10.21059/buletinpeternak.v32i3.1254.
- Munir, R. (2004) *Pengolahan Citra Digital Dengan Menggunakan Pendekatan Algoritmik*. Bandung: Informatika.
- Murugan, M.V. and Jeyanthi, P. (2014) 'Content based image retrieval using color and texture feature extraction in Android', *International Conference*

- on Information Communication and Embedded Systems (ICICES2014)*.
Institute of Electrical & Electronics Engineers (IEEE).
- Nixon, M.S. and Aguado, A.S. (2002) *Feature extraction and image processing*.
Oxford: Newnes (an imprint of Butterworth-Heinemann Ltd).
- OpenCV* (2015) Available at: <http://opencv.org/> (Diakses pada tanggal: 12 April 2016).
- Prewitt, J.M.S. and Mendelsohn, M.L. (1966) 'THE ANALYSIS OF CELL IMAGES*', *Annals of the New York Academy of Sciences*, 128(3), pp. 1035 – 1053. doi: 10.1111/j.1749-6632.1965.tb11715.x.
- Purnomo, M.H. dan Muntasa, A. (2010) *Konsep Pengolahan Citra Digital dan Ekstraksi Fitur*. Yogyakarta: Graha Ilmu.
- Rajput, R.K. (1988) *A Textbook of Applied Mechanics*. India: Laxmi Publications.
- Roberts, L.G. (1963) 'Machine Perception of Three-Dimensional Solids',
Cambridge: Massachusetts Institute of Technology.
- Sobel, I.E. (1970) *Camera Models and Machine Perception*. Stanford: Computer Science Dept., Stanford University.
- Valverde, F.L., Guil, N., Munoz, J., Nishikawa, R. and Doi, K. (no date) 'An Evaluation Criterion for Edge Detection Techniques in Noisy Images', *Proceedings 2001 International Conference on Image Processing (Cat. No.01CH37205)*. Institute of Electrical & Electronics Engineers (IEEE).
- Wahid, A., Ahmad, K., Tyagi, G. and Rizvi, M.A. (2014) 'Anti-theft Cloud Apps for Android Operating System', *2014 International Conference on*

Computational Intelligence and Communication Networks. Institute of Electrical & Electronics Engineers (IEEE).

Widjanarko, D. (2014) *Program Aplikasi Untuk Identifikasi Tingkat Kematangan Buah Pisang Mas (Musa Paradisiaca l) Berdasarkan Parameter Citra dengan Teknologi Pengolahan Citra Digital*. Yogyakarta: Universitas Gadjah Mada.

Wijaya, A.A. dan Prayudi, Y. (2010) 'Implementasi Visi Komputer dan Segmentasi Citra untuk Klasifikasi Bobot Telur Ayam Ras', in *Seminar Nasional Aplikasi Teknologi Informasi (SNATI)*. Available at: <http://journal.uui.ac.id/index.php/Snati/article/view/1949> .

LAMPIRAN A
GAMBAR HASIL AKUISISI

Gambar Sapi dari proses akuisisi citra





Sapi 5 (Kode N)



Sapi 6 (Kode O)



Sapi 7 (Kode P)



Sapi 8 (Kode Q)



Sapi 9 (Kode R)

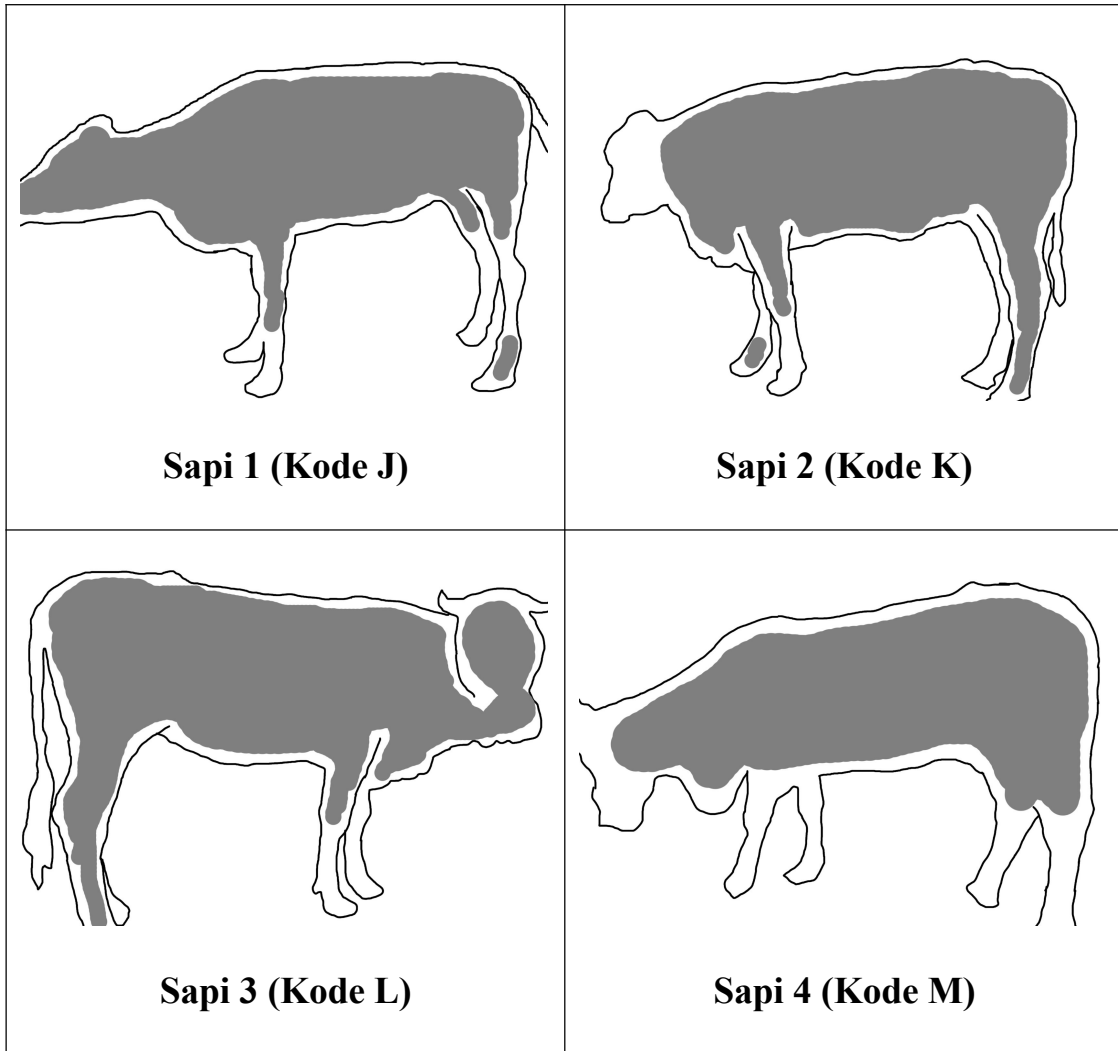


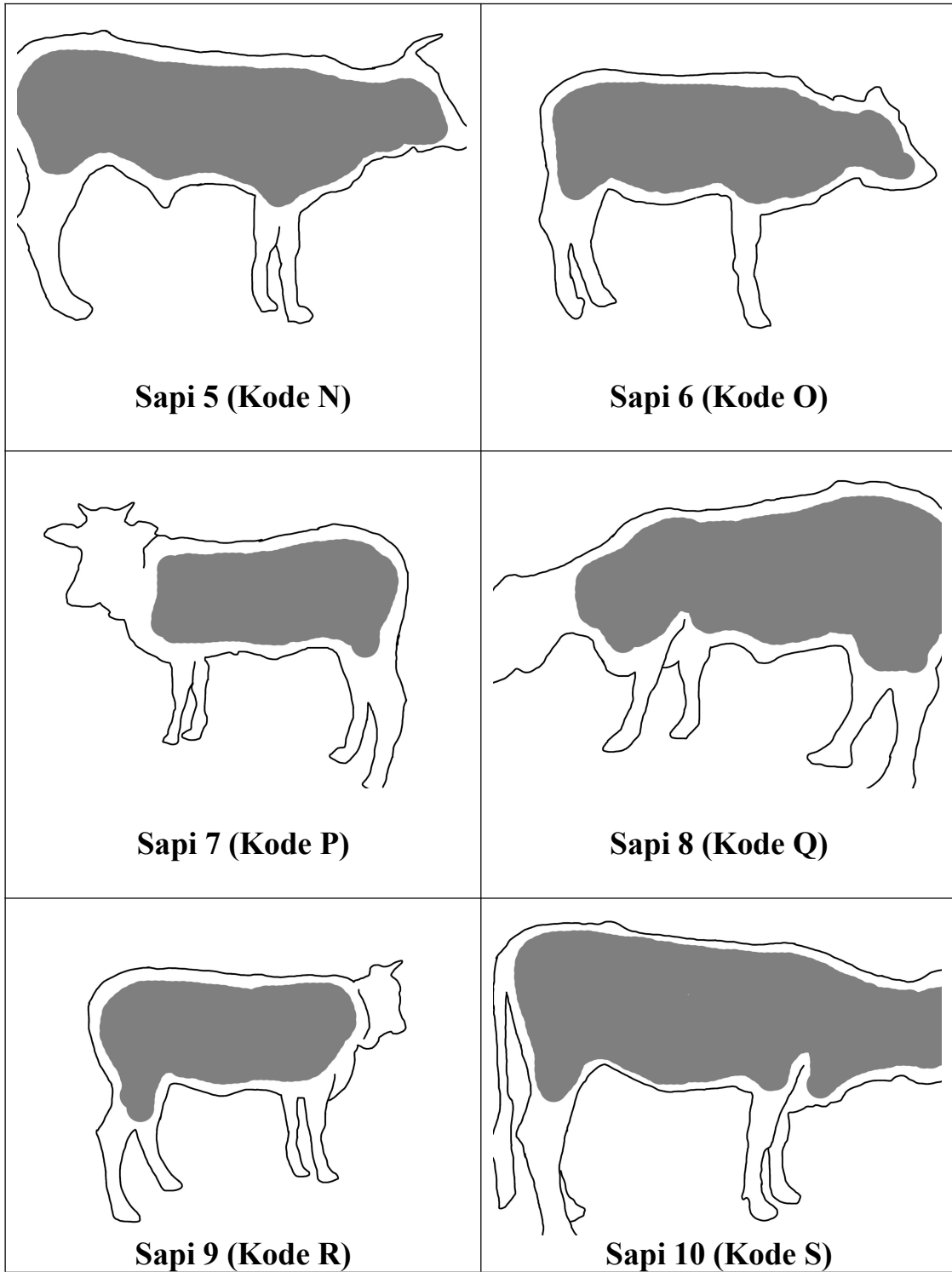
Sapi 10 (Kode S)

LAMPIRAN B

GAMBAR *GROUND TRUTH* (GT)

Gambar *Ground Truth* (GT) yang digunakan untuk evaluasi





LAMPIRAN C

SOURCECODE SKENARIO PREPROCESSING

a. Skenario 1 (Canny)

```
//===== Proses 1 canny =====  
  
public void Canny() {  
    if (OpenCVLoader.initDebug()) {  
  
        Mat originalMat = new Mat(bitmap.getWidth(), bitmap.getHeight(), CvType.CV_8UC1);  
        Mat grayMat = new Mat(bitmap.getWidth(), bitmap.getHeight(), CvType.CV_8UC1);  
        Mat cannyEdges = new Mat(bitmap.getWidth(), bitmap.getHeight(), CvType.CV_8UC1);  
  
        // Convert Bitmap to Mat  
        Utils.bitmapToMat(bitmap, originalMat);  
        // Process  
        Imgproc.cvtColor(originalMat, grayMat, Imgproc.COLOR_BGR2GRAY);  
        Imgproc.Canny(grayMat, cannyEdges, 10, 100, 3, true);  
  
        // Convert Mat to Bitmap  
        Bitmap tmp = Bitmap.createBitmap(cannyEdges.cols(), cannyEdges.rows(), Bitmap.Config.ARGB_8888);  
        Utils.matToBitmap(cannyEdges, tmp);  
  
        bitmap = tmp;  
        ivG.setImageBitmap(tmp);  
    }  
}
```

b. Skenario 2 (Canny + Median Blur)

```
//===== Proses 2 canny + median blur =====  
  
public void CannyMedian() {  
    if (OpenCVLoader.initDebug()) {  
  
        Mat originalMat = new Mat(bitmap.getWidth(), bitmap.getHeight(), CvType.CV_8UC1);  
        Mat grayMat = new Mat(bitmap.getWidth(), bitmap.getHeight(), CvType.CV_8UC1);  
        Mat cannyEdges = new Mat(bitmap.getWidth(), bitmap.getHeight(), CvType.CV_8UC1);  
  
        // Convert Bitmap to Mat  
        Utils.bitmapToMat(bitmap, originalMat);  
        // Process  
        Imgproc.cvtColor(originalMat, grayMat, Imgproc.COLOR_BGR2GRAY);  
        Imgproc.Canny(grayMat, cannyEdges, 10, 100, 3, true);  
        Imgproc.medianBlur(cannyEdges, cannyEdges, 3);  
  
        // Convert Mat to Bitmap  
        Bitmap tmp = Bitmap.createBitmap(cannyEdges.cols(), cannyEdges.rows(), Bitmap.Config.ARGB_8888);  
        Utils.matToBitmap(cannyEdges, tmp);  
  
        bitmap = tmp;  
        ivG.setImageBitmap(tmp);  
    }  
}
```

c. Skenario 3 (Median Blur + Canny)

```
//===== Proses 3 median blur + canny =====
public void MedianCanny() {
    if (OpenCVLoader.initDebug()) {

        Mat originalMat = new Mat(bitmap.getWidth(), bitmap.getHeight(), CvType.CV_8UC1);
        Mat grayMat = new Mat(bitmap.getWidth(), bitmap.getHeight(), CvType.CV_8UC1);
        Mat cannyEdges = new Mat(bitmap.getWidth(), bitmap.getHeight(), CvType.CV_8UC1);

        // Convert Bitmap to Mat
        Utils.bitmapToMat(bitmap, originalMat);
        // Process
        Imgproc.medianBlur(originalMat, originalMat, 3);
        Imgproc.cvtColor(originalMat, grayMat, Imgproc.COLOR_BGR2GRAY);
        Imgproc.Canny(grayMat, cannyEdges, 10, 100, 3, true);

        // Convert Mat to Bitmap
        Bitmap tmp = Bitmap.createBitmap(cannyEdges.cols(), cannyEdges.rows(), Bitmap.Config.ARGB_8888);
        Utils.matToBitmap(cannyEdges, tmp);

        bitmap = tmp;
        ivG.setImageBitmap(tmp);
    }
}
}
```

d. Skenario 4 (Canny + Sharp)

```
//===== Proses 4 canny + sharpen =====
public void CannySharpen() {
    if (OpenCVLoader.initDebug()) {

        Mat originalMat = new Mat(bitmap.getWidth(), bitmap.getHeight(), CvType.CV_8UC1);
        Mat grayMat = new Mat(bitmap.getWidth(), bitmap.getHeight(), CvType.CV_8UC1);
        Mat cannyEdges = new Mat(bitmap.getWidth(), bitmap.getHeight(), CvType.CV_8UC1);
        Mat result = new Mat(bitmap.getWidth(), bitmap.getHeight(), CvType.CV_8UC1);
        Mat kernel = new Mat(3, 3, CvType.CV_16SC1);
        kernel.put(0, 0, 0, -1, 0, -1, 5, -1, 0, -1, 0);

        // Convert Bitmap to Mat
        Utils.bitmapToMat(bitmap, originalMat);
        // Process
        Imgproc.cvtColor(originalMat, grayMat, Imgproc.COLOR_BGR2GRAY);
        Imgproc.Canny(grayMat, cannyEdges, 10, 100, 3, true);
        Imgproc.filter2D(cannyEdges, result, result.depth(), kernel);

        // Convert Mat to Bitmap
        Bitmap tmp = Bitmap.createBitmap(result.cols(), result.rows(), Bitmap.Config.ARGB_8888);
        Utils.matToBitmap(result, tmp);

        bitmap = tmp;
        ivG.setImageBitmap(tmp);
    }
}
}
```

e. Skenario 5 (Sharp + Canny)

```

//===== Proses 5 sharpen + canny=====
public void SharpenCanny() {
    if (OpenCVLoader.initDebug()) {
        Mat originalMat = new Mat(bitmap.getWidth(), bitmap.getHeight(), CvType.CV_8UC1);
        Mat grayMat = new Mat(bitmap.getWidth(), bitmap.getHeight(), CvType.CV_8UC1);
        Mat cannyEdges = new Mat(bitmap.getWidth(), bitmap.getHeight(), CvType.CV_8UC1);
        Mat kernel = new Mat(3, 3, CvType.CV_16SC1);
        kernel.put(0, 0, 0, -1, 0, -1, 5, -1, 0, -1, 0);

        // Convert Bitmap to Mat
        Utils.bitmapToMat(bitmap, originalMat);
        // Process
        Imgproc.filter2D(originalMat, originalMat, originalMat.depth(), kernel);
        Imgproc.cvtColor(originalMat, grayMat, Imgproc.COLOR_BGR2GRAY);
        Imgproc.Canny(grayMat, cannyEdges, 10, 100, 3, true);

        // Convert Mat to Bitmap
        Bitmap tmp = Bitmap.createBitmap(cannyEdges.cols(), cannyEdges.rows(), Bitmap.Config.ARGB_8888);
        Utils.matToBitmap(cannyEdges, tmp);

        bitmap = tmp;
        ivG.setImageBitmap(tmp);
    }
}

```


LAMPIRAN D

SOURCECODE USULAN ALGORITMA

a. Algoritma A (Titik Tengah Gambar)

```
//===== Hitung 1 Algoritma A =====  
  
public void Hitung() {  
    int x = 0, y = 0;  
  
    // get image size  
    int width = bitmap.getWidth();  
    int height = bitmap.getHeight();  
  
    //showDetail(bitmap, width, height);  
  
    // get center image  
    int xCenter = width / 2;  
    int yCenter = height / 2;  
  
    Log.e(TAG, "center x =" + xCenter + " | center y =" + yCenter);  
  
    x = calculatePixelX(xCenter, yCenter, width);  
    y = calculatePixelY(xCenter, yCenter, height);  
  
    conversionPixeltoCm(x, y);  
    sendValue();  
  
}
```

b. Algoritma B (Titik Berat)

```
//===== Hitung 2 Algoritma B =====  
  
public void Hitung2() {  
    int x = 0, y = 0;  
  
    // get image size  
    int width = bitmap.getWidth();  
    int height = bitmap.getHeight();  
  
    // get center image  
    int xCenter = width / 2;  
    int yCenter = height / 2;  
  
    Log.e(TAG, "center x =" + xCenter + " | center y =" + yCenter);  
  
    x = calculatePixelX(xCenter, yCenter, width);  
    y = calculatePixelY(xCenter, yCenter, height);  
  
    // get new center  
    xCenter = x / 2;  
    yCenter = y / 2;  
  
    Log.e(TAG, "new center x =" + xCenter + " | new center y =" + yCenter);  
  
    x = calculatePixelX(xCenter, yCenter, width);  
    y = calculatePixelY(xCenter, yCenter, height);  
  
    conversionPixeltoCm(x, y);  
    sendValue();  
  
}
```

c. Algoritma C (Titik Berat 10 Piksel)

```

//===== Hitung 3 Algoritma C =====
public void Hitung3() {
    int x = 0, y = 0;

    // get image size
    int width = bitmap.getWidth();
    int height = bitmap.getHeight();

    // get center image
    int xCenter = width / 2;
    int yCenter = height / 2;
    Log.e(TAG, "center x = " + xCenter + " | center y = " + yCenter);

    x = calculatePixelX(xCenter, yCenter, width);
    y = calculatePixelY(xCenter, yCenter, height);

    // get new center
    xCenter = x / 2;
    yCenter = y / 2;
    Log.e(TAG, "new center x = " + xCenter + " | new center y = " + yCenter);

    // search long line use 10 pixel (5 pixel up, 5 pixel down) and 10 pixel (5 pixel left, 5 pixel right)
    for (int i = 0; i <= 10; i++) {
        int xNew, yNew, xExtended, yExtended;
        yExtended = yCenter - 5 + i;
        xNew = calculatePixelX(xCenter, yExtended, width);

        xExtended = xCenter - 5 + i;
        yNew = calculatePixelY(xExtended, yCenter, height);

        Log.e(TAG, "xNew = " + xNew + " | x = " + x);
        if (xNew > x) {
            Log.i(TAG, "change value x from " + x + " to " + xNew);
            x = xNew; // change value x
        }

        Log.e(TAG, "yNew = " + yNew + " | y = " + y);
        if (yNew > y) {
            Log.i(TAG, "change value y from " + y + " to " + yNew);
            y = yNew; // change value y
        }
    }

    conversionPixeltoCm(x, y);
    sendValue();
}

```

d. Algoritma D (Titik Berat Keliling Lingkaran)

```

//===== Hitung 4 Algoritma D =====
public void Hitung4() {
    int x = 0, y = 0;

    // get image size
    int width = bitmap.getWidth();
    int height = bitmap.getHeight();

    // get center image
    int xCenter = width / 2;
    int yCenter = height / 2;
    Log.e(TAG, "center x = " + xCenter + " | center y = " + yCenter);

    x = calculatePixelX(xCenter, yCenter, width);
    y = calculatePixelY(xCenter, yCenter, height);

    // get new center
    xCenter = x / 2;
    yCenter = y / 2;
    Log.e(TAG, "new center x = " + xCenter + " | new center y = " + yCenter);

    // search long line use 10 pixel (5 pixel up, 5 pixel down) and 10 pixel (5 pixel left, 5 pixel right)
    for (int i = 0; i <= 10; i++) {
        int xNew, yNew, xExtended, yExtended;
        yExtended = yCenter - 5 + i;
        xNew = calculatePixelX(xCenter, yExtended, width);

        xExtended = xCenter - 5 + i;
        yNew = calculatePixelY(xExtended, yCenter, height);

        Log.e(TAG, "xNew = " + xNew + " | x = " + x);
        if (xNew > x) {
            Log.i(TAG, "change value x from " + x + " to " + xNew);
            x = xNew; // change value x
        }

        Log.e(TAG, "yNew = " + yNew + " | y = " + y);
        if (yNew > y) {
            Log.i(TAG, "change value y from " + y + " to " + yNew);
            y = yNew; // change value y
        }
    }

    conversionPixeltoCmUpdate1(x, y);
    sendValue();
}

```

e. Algoritma E (Titik Berat Keliling Elips 1)

```

//===== Hitung 5 Algoritma E =====
public void Hitung5() {
    int x = 0, y = 0;

    // get image size
    int width = bitmap.getWidth();
    int height = bitmap.getHeight();

    // get center image
    int xCenter = width / 2;
    int yCenter = height / 2;
    Log.e(TAG, "center x =" + xCenter + " | center y =" + yCenter);

    x = calculatePixelX(xCenter, yCenter, width);
    y = calculatePixelY(xCenter, yCenter, height);

    // get new center
    xCenter = x / 2;
    yCenter = y / 2;
    Log.e(TAG, "new center x =" + xCenter + " | new center y =" + yCenter);

    // search long line use 10 pixel (5 pixel up, 5 pixel down) and 10 pixel (5 pixel left, 5 pixel right)
    for (int i = 0; i <= 10; i++) {
        int xNew, yNew, xExtended, yExtended;
        yExtended = yCenter - 5 + i;
        xNew = calculatePixelX(xCenter, yExtended, width);

        xExtended = xCenter - 5 + i;
        yNew = calculatePixelY(xExtended, yCenter, height);

        Log.e(TAG, "xNew = "+xNew+" | x = "+x);
        if (xNew > x) {
            Log.i(TAG, "change value x from " + x + " to " + xNew);
            x = xNew; // change value x
        }

        Log.e(TAG, "yNew = "+yNew+" | y = "+y);
        if (yNew > y) {
            Log.i(TAG, "change value y from " + y + " to " + yNew);
            y = yNew; // change value y
        }
    }

    conversionPixeltoCmUpdate2(x, y);
    sendValue();
}

```

f. Algoritma F (Titik Berat Keliling Elips 2)

```

//===== Hitung 6 Algoritma F =====
public void Hitung6() {
    int x = 0, y = 0;

    // get image size
    int width = bitmap.getWidth();
    int height = bitmap.getHeight();

    // get center image
    int xCenter = width / 2;
    int yCenter = height / 2;
    Log.e(TAG, "center x =" + xCenter + " | center y =" + yCenter);

    x = calculatePixelX(xCenter, yCenter, width);
    y = calculatePixelY(xCenter, yCenter, height);

    // get new center
    xCenter = x / 2;
    yCenter = y / 2;
    Log.e(TAG, "new center x =" + xCenter + " | new center y =" + yCenter);

    // search long line use 10 pixel (5 pixel up, 5 pixel down) and 10 pixel (5 pixel left, 5 pixel right)
    for (int i = 0; i <= 10; i++) {
        int xNew, yNew, xExtended, yExtended;
        yExtended = yCenter - 5 + i;
        xNew = calculatePixelX(xCenter, yExtended, width);

        xExtended = xCenter - 5 + i;
        yNew = calculatePixelY(xExtended, yCenter, height);

        Log.e(TAG, "xNew = "+xNew+" | x = "+x);
        if (xNew > x) {
            Log.i(TAG, "change value x from " + x + " to " + xNew);
            x = xNew; // change value x
        }

        Log.e(TAG, "yNew = "+yNew+" | y = "+y);
        if (yNew > y) {
            Log.i(TAG, "change value y from " + y + " to " + yNew);
            y = yNew; // change value y
        }
    }

    conversionPixeltoCmUpdate3(x, y);
    sendValue();
}

```

CURICULUM VITAE

Nama : Ahmad Mustafid

Tempat, tanggal lahir : Jepara, 20 Agustus 1994

Jenis kelamin : Laki-laki

Status : Belum Menikah

Agama : Islam

Kewarganegaraan : Indonesia

Golongan Darah : B

Hobi : *Travelling, Internet, Gaming, Reading.*

E-mail : ahmad.mstfd@gmail.com

Riwayat Pendidikan :

2000 – 2006 : SDN 2 Brantak Sekarjati

2006 – 2009 : SMPN 1 Welahan

2009 – 2012 : SMAN 1 Pecangaan

2012 – 2016 : S1 Teknik Informatika

UIN Sunan Kalijaga Yogyakarta

