

SKRIPSI

**IMPLEMENTASI REGRESI LINEAR BERGANDA
UNTUK ESTIMASI DAN KLASIFIKASI BERAT TELUR
MENGUNAKAN *VIDEO PROCESSING***

Disusun dan diajukan oleh:

**AGUNAWAN ALI NUR
D121 20 1081**



**PROGRAM STUDI SARJANA TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS HASANUDDIN
GOWA
2024**

LEMBAR PENGESAHAN SKRIPSI

IMPLEMENTASI REGRESI LINEAR BERGANDA UNTUK ESTIMASI DAN KLASIFIKASI BERAT TELUR MENGUNAKAN *VIDEO PROCESSING*

Disusun dan diajukan oleh

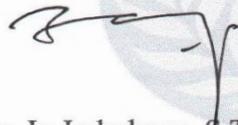
Agunawan Ali Nur
D121 20 1081

Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka Penyelesaian Studi Program Sarjana Program Studi Teknik Informatika Fakultas Teknik Universitas Hasanuddin Pada tanggal 7 Juni 2024 dan dinyatakan telah memenuhi syarat kelulusan

Menyetujui,

Pembimbing Utama,

Pembimbing Pendamping,

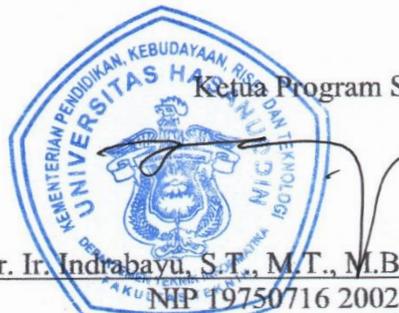


Prof. Dr. Ir. Indrabayu, S.T., M.T.,
M.Bus.Sys., IPM., ASEAN.Eng.
NIP 19750716 200212 1 004



Elly Warni, S.T., M.T.
NIP 19820216 200812 2 0001

Ketua Program Studi,



Prof. Dr. Ir. Indrabayu, S.T., M.T., M.Bus.Sys., IPM., ASEAN.Eng.
NIP 19750716 200212 1 004

PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini ;
Nama : Agunawan Ali Nur
NIM : D121 20 1081
Program Studi : Teknik Informatika
Jenjang : S1

Menyatakan dengan ini bahwa karya tulisan saya berjudul

Implementasi Regresi Linear Berganda untuk Estimasi dan Klasifikasi Berat Telur
Menggunakan *Video Processing*

Adalah karya tulisan saya sendiri dan bukan merupakan pengambilan alihan tulisan orang lain dan bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri.

Semua informasi yang ditulis dalam skripsi yang berasal dari penulis lain telah diberi penghargaan, yakni dengan mengutip sumber dan tahun penerbitannya. Oleh karena itu semua tulisan dalam skripsi ini sepenuhnya menjadi tanggung jawab penulis. Apabila ada pihak manapun yang merasa ada kesamaan judul dan atau hasil temuan dalam skripsi ini, maka penulis siap untuk diklarifikasi dan mempertanggungjawabkan segala resiko.

Segala data dan informasi yang diperoleh selama proses pembuatan skripsi, yang akan dipublikasi oleh Penulis di masa depan harus mendapat persetujuan dari Dosen Pembimbing.

Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan isi skripsi ini hasil karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Makassar, 13 Juni 2024

Yang Menyatakan



Agunawan Ali Nur

ABSTRAK

AGUNAWAN ALI NUR. *Implementasi Regresi Linear Berganda untuk Estimasi dan Klasifikasi Berat Telur Menggunakan Video Processing* (dibimbing oleh Indrabayu dan Elly Warni)

Industri 4.0 mendorong transformasi dalam proses produksi melalui digitalisasi dan otomatisasi dengan kecerdasan buatan. Sektor peternakan telur ayam di Indonesia mengalami peningkatan produksi setiap tahunnya. Sehingga kualitas telur perlu dijaga sebelum didistribusikan ke pasar. Saat ini klasifikasi berat telur dilakukan secara manual. Metode ini memerlukan tenaga kerja besar, hasil yang tidak akurat dan tidak konsisten. Oleh karena itu, diperlukan sistem terstandarisasi dalam melakukan estimasi dan klasifikasi telur.

Tujuan penelitian ini adalah mengimplementasikan Regresi Linear Berganda dalam mengestimasi berat telur ayam dan menganalisis kinerja model estimasi berat telur menggunakan *video processing*.

Penelitian ini menggunakan data video dengan berat bervariasi yang direkam dari atas *conveyor*. Tahap *training* menggunakan 90 telur dan untuk *testing* menggunakan 25 telur. Hasil video ini diekstrak menjadi *frame* citra dan diseleksi. Metode *preprocessing* yang digunakan berupa *resize*, peningkatan kontras dan kecerahan, peregangan kontras, segmentasi, deteksi objek, deteksi tepi, *hole filling* dan penghilangan derau. Fitur yang digunakan sebagai variabel Regresi Linear Berganda adalah *area*, *radius distance* dan *principal axes*. Metode validasi dan *testing* model menggunakan metrik evaluasi *Mean Absolute Percentage Error* (MAPE), *Root Mean Squared Error* (RMSE), dan akurasi.

Berdasarkan hasil penelitian diperoleh skenario terbaik yaitu menggunakan metode seleksi 1 *frame* pertelur, *resizing* citra sebesar 10% dari citra asli, rasio *split data* 90% : 10% untuk data latih dan validasi, menggunakan fitur *Highest Eigen Value* dan pengambilan data dengan kecepatan *conveyor* 60 RPM dan jarak kamera 10 cm. Estimasi berat telur pada *training* menghasilkan RMSE sebesar 2,1344 dan MAPE sebesar 3,1734%. Hasil *testing* dengan data video menghasilkan RMSE sebesar 2,3409, MAPE sebesar 3,8717% dan klasifikasi menunjukkan akurasi sebesar 76%.

Kata Kunci: Telur, Regresi Linear Berganda, estimasi, *video processing*

ABSTRACT

AGUNAWAN ALI NUR. *Implementation of Multiple Linear Regression for Estimation and Classification of Egg Weight Using Video Processing* (supervised by Indrabayu and Elly Warni)

Industry 4.0 is driving transformation in production processes through digitalization and automation with artificial intelligence. The chicken egg farming sector in Indonesia has seen an increase in production annually. Therefore, it is crucial to maintain egg quality before distribution to the market. Currently, egg weight classification is done manually. This method requires a large workforce, and the results are inaccurate and inconsistent. Hence, a standardized system is needed for estimating and classifying eggs.

The aim of this study is to implement Multiple Linear Regression to estimate the weight of chicken eggs and to analyze the performance of the egg weight estimation model using video processing. This study utilizes video data with varying weights recorded from above a conveyor. The training phase used 90 eggs, and for testing, 25 eggs were used. The videos were extracted into image frames and selected. The preprocessing methods used include resizing, enhancing contrast and brightness, contrast stretching, segmentation, object detection, edge detection, hole filling, and noise removal. The features used as variables in Multiple Linear Regression are area, radius distance, and principal axes. Model validation and testing methods use evaluation metrics such as Mean Absolute Percentage Error (MAPE), Root Mean Squared Error (RMSE), and accuracy.

Based on the research findings, the best scenario involved selecting 1 frame per egg, resizing the image to 10% of the original size, splitting data at a ratio of 90%:10% for training and validation, using the Highest Eigen Value feature, and collecting data at a conveyor speed of 60 RPM and a camera distance of 10 cm. The weight estimation during training yielded an RMSE of 2.1344 and a MAPE of 3.1734%. Testing with video data resulted in an RMSE of 2.3409, a MAPE of 3.8717%, and the classification showed an accuracy of 76%.

Keywords: Egg, Multiple Linear Regression, estimation, video processing

DAFTAR ISI

LEMBAR PENGESAHAN SKRIPSI.....	i
PERNYATAAN KEASLIAN.....	ii
ABSTRAK	iii
ABSTRACT	iv
DAFTAR ISI.....	v
DAFTAR GAMBAR	vi
DAFTAR TABEL.....	viii
DAFTAR SINGKATAN DAN ARTI SIMBOL	ix
DAFTAR LAMPIRAN.....	x
KATA PENGANTAR	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian	3
1.4 Manfaat Penelitian	3
1.5 Ruang Lingkup Penelitian.....	4
BAB II TINJAUAN PUSTAKA.....	5
2.1 Telur.....	5
2.2 Visi Komputer.....	7
2.3 Citra.....	8
2.4 Pengolahan Citra Digital	12
2.5 Pengolahan Video Digital	14
2.6 <i>Pre-processing</i>	15
2.7 Ekstraksi Fitur	25
2.8 <i>Pearson Correlation</i>	30
2.9 Estimasi.....	31
2.10 <i>Linear Regression</i>	32
2.11 Metrik Evaluasi	34
BAB III METODE PENELITIAN.....	38
3.1 Waktu dan Lokasi Penelitian	38
3.2 Benda Uji dan Alat.....	38
3.3 Tahapan Penelitian.....	39
3.4 Teknik Pengambilan Data	41
3.5 Perancangan dan Implementasi Sistem.....	43
BAB IV HASIL DAN PEMBAHASAN	72
4.1 Hasil <i>Data Preparation</i>	72
4.2 Hasil <i>Preprocessing</i>	74
4.3 Hasil <i>Training</i>	81
4.4 Hasil <i>Testing</i>	104
BAB V KESIMPULAN DAN SARAN.....	110
5.1 Kesimpulan	110
5.2 Saran.....	110
DAFTAR PUSTAKA	111

DAFTAR GAMBAR

Gambar 1 Metode klasifikasi telur berdasarkan ukuran secara konvensional	2
Gambar 2 Struktur dan bagian-bagian telur	5
Gambar 3 Bentuk telur	6
Gambar 4 Perbedaan letak titik (0,0)	10
Gambar 5 Contoh citra biner	10
Gambar 6 Contoh citra <i>grayscale</i>	11
Gambar 7 Contoh citra berwarna	12
Gambar 8 Tahapan pengolahan citra	13
Gambar 9 Ilustrasi berbagai tipe operasi pengolahan citra	14
Gambar 10 Ilustrasi ketetangaan piksel	16
Gambar 11 Contoh segmentasi	18
Gambar 12 Contoh <i>bounding box</i>	20
Gambar 13 Tabel kebenaran logika NOT	21
Gambar 14 Contoh deteksi tepi	22
Gambar 15 Tabel kebenaran logika OR	24
Gambar 16 Contoh penghilangan derau	25
Gambar 17 Lokasi penelitian	38
Gambar 18 Tahapan penelitian	39
Gambar 19 Persiapan pengambilan data	41
Gambar 20 Orientasi telur pada pengambilan data	42
Gambar 21 Pengambilan data telur pada <i>conveyor</i>	42
Gambar 22 Alur Perancangan Sistem	43
Gambar 23 Contoh hasil seleksi <i>frame</i>	44
Gambar 24 Ilustrasi proses <i>resize</i> pada citra	45
Gambar 25 Contoh peningkatan kontras dan kecerahan	45
Gambar 26 Contoh peregangan kontras	46
Gambar 27 Contoh konversi RGB ke <i>grayscale</i>	48
Gambar 28 Contoh <i>thresholding</i>	49
Gambar 29 Contoh <i>invert</i>	49
Gambar 30 <i>Flowchart</i> proses deteksi objek	50
Gambar 31 Lubang pada objek	50
Gambar 32 Ilustrasi filter Sobel dua arah	51
Gambar 33 <i>Threshold</i> magnitudo gradien	52
Gambar 34 Iterasi pertama <i>flood filling</i>	53
Gambar 35 <i>Flood filling</i> pada titik (0, 1)	54
Gambar 36 <i>Flood filling</i> pada titik (1, 0)	54
Gambar 37 Hasil operasi <i>flood filling</i>	55
Gambar 38 Contoh operasi OR	55
Gambar 39 Derau pada objek	56
Gambar 40 Contoh proses <i>median filter</i>	56
Gambar 41 Contoh citra dan matriksnya	58
Gambar 42 Nilai x , y dan intensitas citra	58
Gambar 43 Memilih <i>board</i> dan <i>port</i>	69
Gambar 44 <i>Source code</i> kecepatan <i>conveyor</i>	69
Gambar 45 <i>Heatmap</i> korelasi <i>Pearson</i>	84

Gambar 46 Rata-rata RMSE skenario seleksi <i>frame</i>	87
Gambar 47 Rata-rata MAPE skenario seleksi <i>frame</i>	88
Gambar 48 Rata-rata RMSE skenario <i>resizing</i> data	91
Gambar 49 Rata-rata MAPE skenario <i>resizing</i> data	91
Gambar 50 Rata-rata RMSE skenario <i>split data</i>	94
Gambar 51 Rata-rata MAPE skenario <i>split data</i>	95
Gambar 52 Rata-rata RMSE skenario jumlah fitur.....	98
Gambar 53 Rata-rata MAPE skenario jumlah fitur.....	99
Gambar 54 Rata-rata RMSE skenario pengambilan data	102
Gambar 55 Rata-rata MAPE skenario pengambilan data	103

DAFTAR TABEL

Tabel 1. Klasifikasi bobot telur ayam	7
Tabel 2. Arti nilai koefisien korelasi.....	30
Tabel 3. Data untuk Regresi Linear Berganda	33
Tabel 4. <i>Range</i> nilai MAPE	36
Tabel 5. Ilustrasi <i>confusion matrix</i>	36
Tabel 6. Skenario pengambilan data	42
Tabel 7. Rasio <i>split data</i>	57
Tabel 8. Tabel fitur area dan berat aktual telur	62
Tabel 9. Contoh data untuk algoritma estimasi.....	64
Tabel 10. Contoh data validasi model.....	66
Tabel 11. Hasil perhitungan persamaan regresi	66
Tabel 12. Contoh perhitungan RMSE.....	67
Tabel 13. Contoh perhitungan MAPE.....	68
Tabel 14. Hasil pengambilan data video	72
Tabel 15. Data berat aktual telur	73
Tabel 16. Hasil ekstraksi <i>frame</i>	73
Tabel 17. Hasil <i>resizing</i>	74
Tabel 18. Hasil peningkatan kontras dan kecerahan telur ke-45	75
Tabel 19. Hasil peningkatan kontras dan kecerahan pada telur ke-49.....	75
Tabel 20. Peregangan kontras pada telur ke-45	76
Tabel 21. Peregangan kontras pada telur ke-49	77
Tabel 22. Hasil segmentasi citra	78
Tabel 23. Hasil deteksi tepi	78
Tabel 24. Hasil deteksi tepi.....	79
Tabel 25. Hasil hole filling.....	80
Tabel 26. Hasil penghilangan derau.....	81
Tabel 27. Hasil ekstraksi fitur	82
Tabel 28. Hasil korelasi <i>Pearson</i>	83
Tabel 29. Skenario jumlah fitur	85
Tabel 30. Hasil skenario seleksi <i>frame</i>	86
Tabel 31. Hasil skenario <i>resize</i>	89
Tabel 32. Hasil skenario <i>split data</i>	93
Tabel 33. Hasil skenario jumlah fitur.....	96
Tabel 34. Hasil skenario pengambilan data	100
Tabel 35. Hasil <i>testing</i>	104
Tabel 36. <i>Confusion matrix</i> hasil testing.....	107
Tabel 37. Pembagian kelompok data	108
Tabel 38. <i>Confusion matrix</i> kelompok A.....	108
Tabel 39. <i>Confusion matrix</i> kelompok B.....	109

DAFTAR SINGKATAN DAN ARTI SIMBOL

Lambang/Singkatan	Arti dan Keterangan
S_x	Turunan pertama arah horizontal
S_y	Turunan pertama arah vertikal
G_m	Magnitudo Gradien
\bar{x}	<i>Centroid</i> pada koordinat horizontal
\bar{y}	<i>Centroid</i> pada koordinat vertikal
d_E	<i>Euclidean Distance</i>
$M(m, n)$	<i>Spatial moments</i> dari orde (m, n)
$U(m, n)$	<i>Spatial central moments</i> dari orde (m, n)
λ_M	<i>Major axis</i> atau <i>highest eigen value</i>
λ_N	<i>Minor axis</i> atau <i>lowest eigen value</i>
R_A	Rasio <i>minor axis</i> terhadap <i>major axis</i> atau <i>eigenvalue ratio</i>
r	Nilai koefisien korelasi
α	Parameter <i>gain</i> atau pengatur kontras
β	Parameter <i>bias</i> atau pengatur kecerahan
t	Ambang batas
$\hat{\beta}$	<i>Least-squares estimator</i>
RMSE	<i>Root Mean Squared Error</i>
MAPE	<i>Mean Absolute Percentage Error</i>
FPS	<i>Frame Per Second</i>
RPM	<i>Revolutions Per Minute</i>
TP	<i>True positive</i>
TN	<i>True negative</i>
FP	<i>False positive</i>
FN	<i>False negative</i>

DAFTAR LAMPIRAN

Lampiran 1 Hasil pengambilan data berat aktual telur	116
Lampiran 2 Hasil ekstraksi fitur.....	118
Lampiran 3 <i>Source code testing</i>	122
Lampiran 4 Perhitungan RMSE	11624
Lampiran 5 Perhitungan MAPE.....	125
Lampiran 6 Perhitungan rata-rata RMSE dan MAPE.....	126
Lampiran 7 Lembar perbaikan skripsi	127

KATA PENGANTAR

Alhamdulillah, puji syukur kehadirat Allah SWT atas berkat rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan tugas akhir ini yang berjudul “Implementasi Regresi Linear Berganda untuk Estimasi dan Klasifikasi Berat Telur Menggunakan Video Processing” sebagai salah satu persyaratan yang harus dipenuhi dalam menyelesaikan jenjang Strata-1 pada Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin. Sholawat serta salam kepada nabi Muhammad SAW yang telah menunjukkan dan mengajarkan akhlak mulia sehingga didapatkan kenyamanan dan keramahan dalam berhubungan dengan orang di sekitar.

Dengan segala kerendahan hati, 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 ini. Sehingga, pada kesempatan ini penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Allah SWT. Atas semua karunia serta pertolongan-Nya yang tiada batas, yang telah diberikan kepada penulis di setiap langkah dalam penelitian hingga penulisan laporan ini.
2. Kedua orang tua penulis, Bapak Muhammad Ali. R dan Ibu Sitti Nurbayah yang selalu mendoakan untuk kebaikan penulis, selalu memberikan kasih sayang, cinta, dukungan dan motivasi. Terima kasih Bapak dan Ibu yang selalu sabar dan semangat tiada henti dalam menghadapi dan mendidik penulis.
3. Bapak Prof. Dr. Ir. Indrabayu, S.T., M.T., M.Bus.Sys., IPM., ASEAN.Eng., selaku pembimbing 1 yang senantiasa meluangkan waktu, tenaga dan pikiran serta perhatian yang luar biasa untuk membimbing penulis dalam proses penyelesaian tugas akhir ini.
4. Ibu Elly Warni, S.T., M.T., selaku pembimbing 2 yang senantiasa meluangkan waktu, tenaga dan pikiran dalam proses penyelesaian tugas akhir ini.
5. Segenap Dosen dan Staf Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin yang telah memberikan ilmu dan pengetahuan baru, serta bantuan kepada penulis selama menuntut masa perkuliahan.
6. Segenap keluarga AIMP Research Group Universitas Hasanuddin, terutama Kak Fadilah Ramadhani, Kak Ahmad Reza Syahbana, Kak Clara Diva dan Kak Herlina Anwar yang telah memberikan banyak bantuan selama penelitian dan teman diskusi terkait progres penyusunan tugas akhir.
7. Teman-Teman REZOLVER yang banyak membantu selama kuliah dan dalam proses menyelesaikan tugas akhir ini.

8. Keluarga besar RESPECT yang telah membantu dari SMA hingga sekarang ini.
9. Teman-Teman KKNT UH Gelombang 110 Posko 2 Kelurahan Kiru-Kiru yang menjadi bagian dari perjalanan studi penulis.
10. Serta pihak-pihak lain yang tidak sempat disebutkan dan tanpa sadar telah membantu penulis dalam menyelesaikan tugas akhir.

Dengan rasa syukur dan kerendahan hati, penulis memberikan rasa hormat yang tak terhingga, semoga Allah SWT. membalas semua kebaikan dari semua pihak yang telah banyak membantu penulis dalam menyelesaikan tugas akhir ini. Penulis menyadari tugas akhir ini masih banyak kekurangan dan jauh dari kata sempurna. Oleh karena itu, penulis mengharapkan segala masukan dan saran yang membangun sehingga tugas akhir ini dapat memberi manfaat bagi penulis dan pembaca. Akhir kata, semoga tugas akhir ini dapat dijadikan sebagai sumber ilmu pengetahuan dan bermanfaat bagi penulis dan pembaca pada umumnya.

Makassar, Juni 2024

Penulis

BAB I PENDAHULUAN

1.1 Latar Belakang

Industri 4.0 mendorong perubahan menyeluruh di setiap bagian perusahaan termasuk proses produksi melalui digitalisasi dan otomatisasi. Konsep industri 4.0 akan menjadikan proses produksi menjadi lebih efisien dan lebih murah (Tay dkk., 2018). Revolusi industri yang akan datang dapat dicirikan oleh integrasi antara internet dan proses produksi, dengan bantuan sensor-sensor dan kecerdasan buatan yang diterapkan pada mesin. Kecerdasan buatan untuk industri 4.0 adalah penggunaan program komputer untuk melakukan tugas-tugas kompleks pada *smart industry* (Ortiz dkk., 2020).

Salah satu industri yang terus berkembang pesat di Indonesia adalah sektor peternakan. Menurut data dari Badan Pusat Statistik (BPS) pada tahun 2022, produksi telur ayam ras petelur di Indonesia meningkat sebesar 14,92% dibandingkan dengan tahun sebelumnya. Hal ini sejalan dengan permintaan akan telur ayam ras yang mengalami peningkatan sebesar 4,81% (Badan Pusat Statistik, 2022). Dengan potensi yang begitu besar, perhatian terhadap mutu telur merupakan hal yang sangat penting. Penimbangan telur merupakan salah satu tahapan penting dalam pengolahan telur. Karena berat telur memengaruhi nilai dan penggunaan telur dalam berbagai aplikasi, terutama dalam industri makanan dan pengolahan (Yang dkk., 2023). Telur yang sesuai dengan standar akan lebih mudah diterima di pasar lokal dan internasional. Sehingga pemeriksa dan *grader* telur harus memeriksa dengan cermat sebelum didistribusikan ke pasar (Aragua & Mabayo, 2018).

Saat ini, klasifikasi berat telur dilakukan secara manual dengan cara memasukkan telur ke dalam lubang yang sudah diberi ukuran seperti pada Gambar 1. Metode ini tidak efektif karena sulit diterapkan dalam industri skala besar dan membutuhkan waktu lama (Sholihin, 2019). Metode lain yang diterapkan adalah pengamatan secara visual dan penimbangan akumulasi satu rak. Hal ini terjadi di CV Clever Bone yang merupakan peternakan telur ayam ras petelur terbesar di Kabupaten Bone. Peternakan ini memiliki 2 kandang dengan total luas 50 Are atau

5000 m². Populasi ayam peternakan ini terdiri dari *Isa Brown* sebanyak 5000 ekor yang produktif dengan produksi harian mencapai 160 rak telur. Berdasarkan wawancara dengan pekerja, proses klasifikasi telur dilakukan pada hari yang sama setelah telur dikumpulkan dari kandang. Klasifikasi dilakukan dengan cara memegang telur satu persatu dan pengamatan secara visual lalu menempatkannya pada kelompok rak tertentu. Setelah satu rak terkumpul, dilakukan penimbangan berat dan pergantian telur hingga mendapatkan akumulasi berat yang diinginkan. Proses sortasi ini dilakukan dengan durasi rata-rata 3 jam perhari. Metode ini tidak efektif karena memerlukan tenaga kerja besar dan hasil sortir yang tidak akurat.



Gambar 1 Metode klasifikasi telur berdasarkan ukuran secara konvensional
Sumber: (shopee.co.id, 2024)

Untuk meningkatkan efisiensi dan meminimalisir potensi kesalahan dalam proses klasifikasi, diperlukan sistem terstandarisasi yang dapat melakukan estimasi dan klasifikasi telur berdasarkan berat secara otomatis dalam jumlah besar. Klasifikasi berat telur berpedoman pada Badan Standarisasi Nasional (BSN) dengan nomor SNI 3926:2023 bobot telur ayam ras dan ayam kampung di golongan menjadi 3, yaitu: kecil (< 50g), sedang (50g – 60g), dan besar (> 60g).

Algoritma Regresi Linear merupakan algoritma yang sering digunakan dalam proses estimasi karena kesederhanaan model dalam memprediksi nilai variabel tak bebas berdasarkan nilai variabel bebas. Dari penelitian yang dilakukan oleh Basak dkk. (2022) terkait estimasi berat buah stroberi dengan membandingkan algoritma Regresi Linear dan *Support Vector Regression* (SVR), didapatkan hasil testing algoritma Regresi Linear lebih unggul dengan $R^2 = 0,896$ dan $RMSE = 1,504$, sedangkan algoritma SVR menghasilkan nilai $R^2 = 0,856$ dan $RMSE = 1,239$. Penelitian ini menggunakan fitur panjang, diameter dan berat buah stroberi untuk menentukan batas luasan piksel setiap kelas. Penelitian ini menjadi dasar pengimplementasian algoritma Regresi Linear untuk estimasi berat telur.

Penelitian terkait untuk estimasi berat telur telah dilakukan oleh Yang et al. (2023) terkait sistem pemilahan telur otomatis dan deteksi cacat pada telur dengan menggunakan *Convolutional Neural Network* (CNN) dan *Random Forest*, didapatkan hasil $R^2 = 0,96\%$ untuk model estimasi. Penelitian ini menggunakan fitur panjang *major axis* dan *minor axis* sebagai inputan algoritma. Namun, penelitian ini hanya menggunakan data berupa citra gambar sebagai data input.

Berdasarkan hal tersebut penulis mengajukan judul “Implementasi Regresi Linear Berganda untuk Estimasi dan Klasifikasi Berat Telur Menggunakan *Video Processing*”, untuk membangun sebuah sistem yang dapat melakukan estimasi berat telur ayam dengan *computer vision* dan algoritma Regresi Linear Berganda. Hasil estimasi berat akan dilakukan klasifikasi kelas yaitu kecil, sedang dan besar.

1.2 Rumusan Masalah

Rumusan masalah dari penelitian ini adalah sebagai berikut:

1. Bagaimana implementasi algoritma Regresi Linear Berganda dalam mengestimasi berat telur ayam ras?
2. Bagaimana hasil kinerja model estimasi berat telur ayam ras menggunakan *video processing*?

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah sebagai berikut:

1. Mengimplementasikan algoritma Regresi Linear Berganda dalam mengestimasi berat telur ayam ras.
2. Menganalisis kinerja model estimasi berat telur ayam ras menggunakan *video processing*.

1.4 Manfaat Penelitian

Manfaat dari penelitian ini adalah sebagai berikut:

1. Akademisi:

Diharapkan dapat menjadi referensi bagi akademisi maupun peneliti berikutnya dalam melakukan penelitian yang terkait dengan tema yang penulis teliti.

2. Industri Peternakan:
 - a. Sebagai inovasi yang dapat meningkatkan kualitas dan konsistensi klasifikasi berat telur.
 - b. Dapat meningkatkan penggunaan teknologi *video processing* pada bidang peternakan.

1.5 Ruang Lingkup Penelitian

Pada penelitian ini akan dibatasi ruang lingkup pembahasannya yaitu:

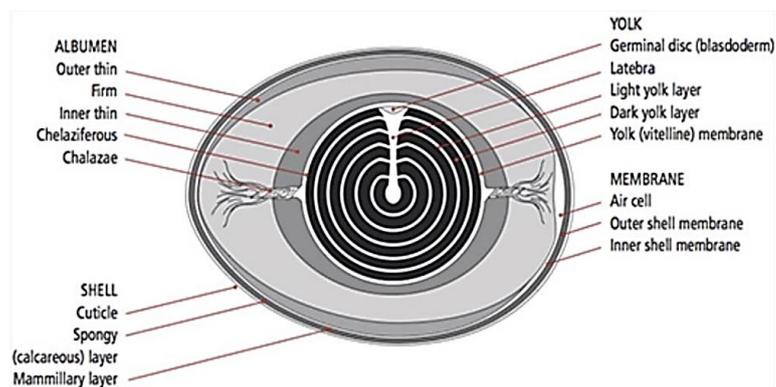
1. Telur ayam yang digunakan adalah telur ayam ras petelur *Isa Brown*.
2. Mengestimasi berat telur berdasarkan fitur bentuk.
3. Telur hasil estimasi diklasifikasi dalam 3 kelas yaitu kecil ($< 50g$), sedang ($50g - 60g$), dan besar ($> 60g$).
4. Pengambilan data dan pengujian dilakukan pada telur dengan masa penyimpanan kurang dari 1 hari.

BAB II TINJAUAN PUSTAKA

2.1 Telur

Telur merupakan hasil sekresi organ reproduksi ternak unggas yang berguna untuk meneruskan kehidupan/perkembangbiakan. Dalam perkembangannya, telur yang pada awalnya merupakan sel reproduktif, oleh manusia telah dimanfaatkan sebagai bahan pangan (Nurwantoro & Sri Mulyani, 2003). Telur sebagai bahan pangan serba guna kaya akan nilai gizi yang terkandung di dalamnya. Protein telur berkualitas tinggi mengandung semua asam amino esensial bagi pertumbuhan dan pemeliharaan jaringan tubuh manusia. Telur kaya akan asam lemak esensial, sumber vitamin A, D, B₁, *riboflavin*, mineral besi (Fe) dan *fosfor* (P) (Antonius Hintono, 2022). Beberapa jenis telur yang dapat dikonsumsi diantaranya telur ayam (lokal, ras), telur bebek, telur burung puyuh, dan lain-lain.

Telur semua spesies burung mempunyai struktur yang sama, namun berbeda dalam proporsi bagian-bagiannya. Telur terdiri atas 4 bagian utama yaitu *yolk* (kuning telur), *albumen* (putih telur), *shell membrane* (membran cangkang) dan *shell* (cangkang) (Antonius Hintono, 2022). Struktur telur dan bagian-bagiannya ditunjukkan pada Gambar 2.

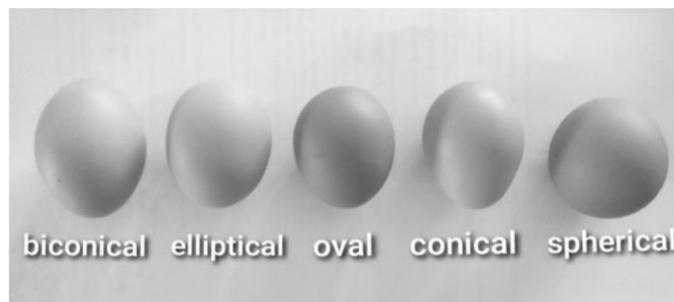


Gambar 2 Struktur dan bagian-bagian telur

Sumber: (weblab.deusto.es, 2023)

Bentuk telur dapat dipengaruhi oleh faktor genetik dan umur induk. Induk yang baru mulai bertelur menghasilkan bentuk telur yang cenderung runcing dan memanjang, sedangkan induk yang semakin tua menghasilkan telur yang semakin ke arah bulat (Nurwantoro & Sri Mulyani, 2003). Kualitas telur yang terbaik berada

pada saat ditelurkan, semakin lama penyimpanan mengakibatkan penurunan kualitas telur. Semakin lama waktu penyimpanan akan mengakibatkan terjadinya penguapan cairan dan gas dalam telur semakin banyak (Sudaryani, 2006). Penurunan berat telur merupakan salah satu perubahan yang nyata selama penyimpanan dan berkorelasi linier terhadap waktu dibawah kondisi lingkungan yang konstan (Kurtini dkk., 2011). Penyimpanan telur selama 5 hari, 10 hari dan 15 hari menyebabkan penurunan berat sebesar 0,90%, 1,87% dan 3,09% pada masing-masing lama masa penyimpanan (Nova dkk., 2014).



Gambar 3 Bentuk telur
Sumber: (Antonius Hintono, 2022)

Berdasarkan bentuknya, telur dibedakan menjadi 5 macam, seperti ditunjukkan pada Gambar 3, yaitu:

1. *Biconical*, adalah telur yang kedua ujungnya runcing seperti kerucut.
2. *Conical*, adalah yang salah satu ujungnya runcing seperti kerucut.
3. *Elliptical*, adalah bentuk telur yang menyerupai elip.
4. *Oval*, adalah bentuk telur yang menyerupai oval.
5. *Spherical*, adalah bentuk telur yang hampir bulat (Nurwantoro & Sri Mulyani, 2003).

Standar mutu telur ayam kampung dan ras perlu diterapkan dalam pemasaran telur terutama untuk memudahkan konsumen dalam menentukan pilihan sehingga dapat memberikan kepuasan dan kepastian mutu untuk konsumen. Ukuran telur sering diwujudkan dalam berat, yang secara jelas ditentukan oleh berat kolektif dari bagian-bagian penyusunnya. Berdasarkan SNI 3926:2023 bobot telur ayam ras dan ayam kampung digolongkan menjadi 3 pada Tabel 1.

Tabel 1. Klasifikasi bobot telur ayam

Bobot Telur (gram)	Ukuran
< 50	Kecil
50 – 60	Sedang
> 60	Besar

Sumber: (SNI, 2023)

Ayam *Isa Brown* merupakan strain ayam ras yang diciptakan di Inggris pada tahun 1972. Ayam ini diciptakan untuk memenuhi keunggulan standar yang diinginkan pada konsumen yang meliputi faktor produktivitas telur dan bobot telur tinggi, konversi ransum rendah, daya hidup tinggi, dan masa bertelur panjang (F. dkk., 2021). Ayam petelur *Isa Brown* memiliki periode bertelur 18 hingga 100 minggu dengan kemampuan hidup (*liveability*) sebesar 93,2%. Berat telur dari jenis ini memiliki rata-rata berat 63 gram. Hal ini menjadikan *Isa Brown* diakui sebagai ayam petelur terbaik di Indonesia dan luar negeri. (Hendrix genetics, 2024).

2.2 Visi Komputer

Visi komputer dapat didefinisikan sebagai sebuah bidang ilmiah yang menggali informasi dari citra digital (Arnita dkk., 2022). Langkah awal untuk menyelesaikan masalah visi komputer adalah dengan memahami bagaimana penglihatan manusia bekerja, dan kemudian menerapkan pengetahuan ini pada komputer (Ranjay Khrisna, 2017). Visi komputer menduplikasikan mata manusia yang baik dalam melakukan penangkapan atau perekaman citra, analisa citra, ataupun dalam pemahaman citra untuk diterapkan di dalam sebuah mesin atau komputer. Aplikasi visi komputer telah berkembang luas dalam beberapa tahun terakhir seperti pengawasan dan pemantauan cerdas (*smart surveillance*), kesehatan dan kedokteran, olahraga dan rekreasi, robotika, *drone*, mobil *self-driving*, dan lain sebagainya (Arnita dkk., 2022).

2.3 Citra

Citra adalah suatu cahaya pada bidang dua dimensi. Ditinjau dari sudut pandang matematis, citra merupakan fungsi kontinu dari intensitas cahaya pada bidang dua dimensi. Sumber cahaya menerangi objek, objek memantulkan kembali sebagian dari berkas cahaya tersebut. Pantulan cahaya ini ditangkap oleh alat-alat optik, misalnya mata pada manusia, kamera, *scanner* dan sebagainya, sehingga bayangan objek yang disebut citra tersebut terekam (Irianto, 2014). Secara umum, citra terbagi menjadi dua yaitu:

2.3.1 Citra analog

Citra analog adalah jenis citra yang memiliki sifat kontinu, seperti gambar yang ditampilkan pada layar televisi, foto yang diproduksi oleh sinar-X, atau foto yang dicetak pada kertas foto. Selain itu, citra analog juga dapat berupa lukisan pemandangan alam, hasil *CT scan*, gambar yang direkam pada pita kaset, dan banyak lagi. Salah satu ciri utama dari citra analog yaitu tidak bisa direpresentasikan secara langsung di dalam komputer. Untuk dapat memproses citra analog menggunakan komputer, perlu dilakukan proses konversi analog ke digital terlebih dahulu. Citra analog biasanya dihasilkan oleh perangkat atau alat-alat analog, seperti *webcam*, *CT scan*, sensor *rontgen* untuk *thorax*, sensor gelombang pendek pada sistem radar, dan sensor *ultrasound* pada sistem USG. Proses konversi dari citra analog ke citra digital memungkinkan citra tersebut untuk diolah lebih lanjut menggunakan perangkat lunak atau aplikasi komputer (Dhanar Intan Surya Saputra dkk., 2016).

2.3.2 Citra digital

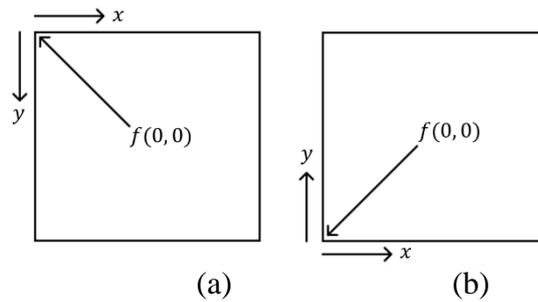
Citra digital adalah kumpulan titik yang dinamakan piksel (*pixel* atau *picture element*). Setiap piksel digambarkan sebagai satu kotak kecil yang memiliki koordinat posisi. Citra merupakan fungsi dua dimensi $f(x, y)$ yang didapatkan dari peralatan sensor yang menyimpan ciri citra pada semua titik elemen (x, y) . Sebuah citra digital a $[m, n]$ dapat dijelaskan sebagai ruang diskrit berdimensi-dua yang berasal dari sebuah citra analog $a(x, y)$ menjadi

citra kontinu 2D melalui proses digitalisasi (Arnita dkk., 2022). Secara garis besar proses digitalisasi citra terbagi menjadi dua proses yaitu pencuplikan (*sampling*) dan kuantisasi (*quantization*) (Sri Ratna Sulistiyanti dkk., 2016). *Sampling* adalah proses mengubah sinyal ruang kontinu menjadi sinyal ruang diskrit. Sedangkan kuantisasi adalah proses mengubah sebuah nilai kontinu citra yang memiliki rentang kontinu menjadi sebuah citra bernilai diskrit dengan rentang diskrit (Gibson & Bovik, 2000).

Citra digital dihasilkan oleh berbagai perangkat fisik, termasuk kamera *still* dan video, perangkat sinar-X, mikroskop *electron*, radar, dan *ultrasound* (Kang, 2007). Jika citra 2D kontinyu $a(x, y)$ dibagi menjadi N baris dan M kolom, maka perpotongan baris dan kolom disebut sebagai piksel. Nilai yang diberikan ke integer koordinat $[m, n]$ dengan $\{m = 0, 1, 2, \dots, M - 1\}$ dan $\{n = 0, 1, 2, \dots, N - 1\}$ adalah $[m, n]$ (Arnita dkk., 2022). Secara matematis citra digital dapat diungkapkan dalam bentuk matrik seperti ditunjukkan pada persamaan 1 (Sri Ratna Sulistiyanti dkk., 2016).

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0, M - 1) \\ f(1,0) & f(0,0) & \dots & f(1, M - 1) \\ \vdots & \vdots & & \vdots \\ f(N - 1,0) & f(N - 1,1) & \dots & f(N - 1, M - 1) \end{bmatrix} \quad (1)$$

Terdapat perbedaan cara menyatakan posisi titik antara koordinat pada citra dengan koordinat grafik (kartesian). Gambar 4 menunjukkan perbedaan letak titik (0,0) antara koordinat pada citra dengan koordinat kartesian. Kesepakatan peletakan titik asal ini merujuk pada cara penulisan matriks pada pemrograman komputer yang berbeda dengan peletakan pada koordinat grafik. Oleh karena citra digital berbentuk matriks maka pemanipulasiannya juga menggunakan kaidah matriks (Sri Ratna Sulistiyanti dkk., 2016).

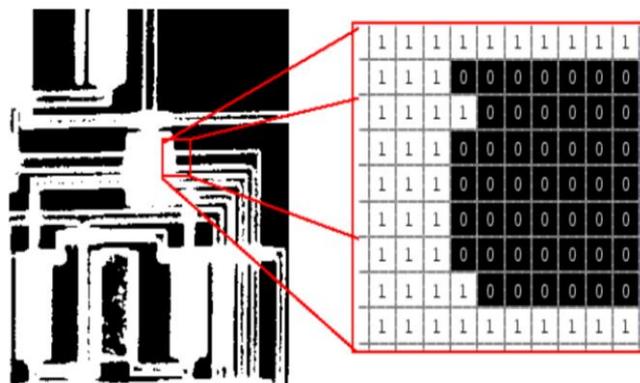


Gambar 4 Perbedaan letak titik (0,0): (a) koordinat pada citra; (b) koordinat kartesian

Sumber: (Sri Ratna Sulistiyanti dkk., 2016)

2.3.2.1 Citra biner

Citra biner adalah citra yang berisi piksel yang berwarna hitam (0) atau putih (1) (Ranjay Khrisna, 2017). Citra biner juga dikenal dengan istilah *binary bilevel image*. Piksel dalam citra biner hanya dapat mengasumsikan dua kemungkinan nilai, 0 atau 1. Dimana nilai 0 menyatakan warna hitam dan nilai 1 menyatakan warna putih. Citra biner dinilai paling ringan karena menggunakan penyimpanan dan juga persyaratan pemrosesan yang paling sedikit. Contoh citra biner adalah citra garis, teks tercetak pada halaman putih, atau siluet. Citra-citra ini berisi informasi yang cukup tentang objek dalam citra dan dapat dikenali dengan mudah. Dalam visi komputer, citra biner biasa digunakan untuk pengenalan objek, dan pelacakan. Citra *grayscale* dapat diubah menjadi citra biner dengan proses *thresholding* dengan nilai ambang batas tertentu (Arnita dkk., 2022). Contoh citra biner ditunjukkan pada Gambar 5.

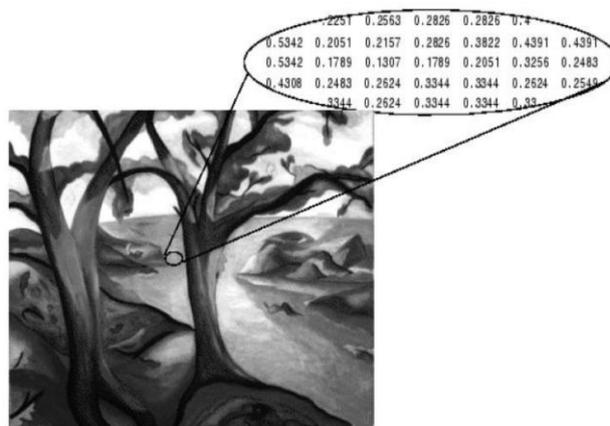


Gambar 5 Contoh citra biner

Sumber: (Irianto, 2014)

2.3.2.2 Citra *grayscale*

Citra *grayscale* adalah citra yang memiliki nilai piksel berkisar antara 0 (hitam) hingga 255 (putih) (Ranjay Khrisna, 2017). Citra ini memiliki gradasi hitam dan putih hingga menghasilkan warna abu. Biasa juga dikenal dengan istilah *gray level*. Pada jenis citra ini piksel bergantung dengan intensitasnya. Kelas *array* dalam citra keabuan bisa terdiri dari *single*, *double*, *uint8*, maupun *int16* (Arnita dkk., 2022). Contoh citra *grayscale* ditunjukkan pada Gambar 6.



Gambar 6 Contoh citra *grayscale*
Sumber: (Irianto, 2014)

2.3.2.3 Citra berwarna

Citra berwarna adalah citra yang memiliki beberapa saluran warna. Setiap citra berwarna dapat direpresentasikan dalam berbagai model warna (misalnya, RGB, LAB, HSV). Sebagai contoh, sebuah citra dalam model RGB terdiri dari saluran merah, hijau, dan biru. Setiap piksel dalam sebuah saluran memiliki nilai intensitas yang berkisar dari 0 hingga 255. Rentang warna ini bergantung pada pilihan model warna. Citra berwarna biasanya direpresentasikan dengan tensor 3D (Lebar \times Panjang \times 3), dimana 3 saluran tersebut dapat mewakili model warna seperti RGB (*Red Green Blue*), LAB (*Lightness A B*) dan HSV (*Hue Saturation Value*) (Ranjay Khrisna, 2017).

Citra RGB merupakan jenis citra yang menyajikan warna dalam bentuk komponen R (*red*/merah), G (*green*/hijau), dan B (*blue*/biru) yang jika digabungkan akan membentuk suatu susunan warna yang luas. Setiap komponen warna menggunakan 8 bit (nilainya berkisar antara 0 hingga 255). Oleh karena itu, kemungkinan warna yang bisa disajikan mencapai $255 \times 255 \times 255$ atau 16.581.375 warna. Satu jenis warna, dapat dimisalkan sebagai sebuah vektor di ruang 3 dimensi yang biasanya dipakai dalam matematika. Koordinatnya dinyatakan dalam bentuk x , y dan z . Misalkan sebuah vektor dituliskan sebagai $r = (x, y, z)$, maka untuk setiap citra, komponen-komponen tersebut digantikan oleh komponen RGB. Dapat dituliskan sebagai berikut untuk citra berwarna = RGB (30, 75, 255). Putih = RGB (255, 255, 255), sedangkan untuk hitam = RGB (0, 0, 0) (Arnita dkk., 2022). Contoh citra RGB ditunjukkan pada Gambar 7.



Gambar 7 Contoh citra berwarna
Sumber: (Irianto, 2014)

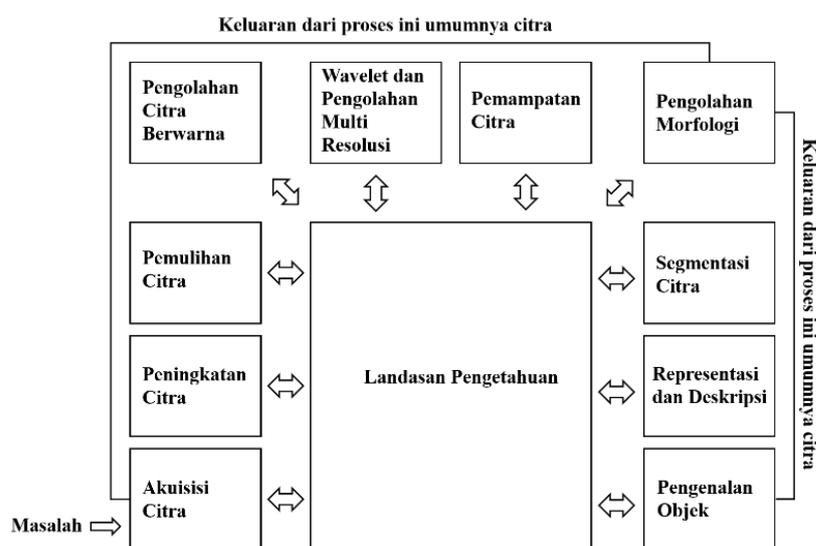
2.4 Pengolahan Citra Digital

Arti pengolahan menurut Kamus Besar Bahasa Indonesia (KBBI) adalah suatu cara atau proses mengolah sesuatu supaya menjadi lain atau menjadi lebih sempurna. Pengolahan citra adalah suatu proses dengan masukan citra dan menghasilkan keluaran berupa citra seperti yang dikehendaki. Poin terpenting dalam pengolahan citra adalah manipulasi dan analisis sebuah citra dilakukan

dengan bantuan komputer. Secara garis besar pengolahan citra dikelompokkan menjadi 2 jenis kegiatan:

1. Memperbaiki kualitas citra agar lebih mudah diinterpretasikan
2. Mengolah informasi pada citra untuk keperluan yang diinginkan biasanya untuk mengenali pola suatu objek

Secara umum tahapan pengolahan citra digital meliputi akuisisi citra, peningkatan kualitas citra, segmentasi citra, representasi dan uraian, pengenalan dan interpretasi (Sri Ratna Sulistiyanti dkk., 2016). Gambar 8 menunjukkan diagram blok tahapan pengolahan citra menurut (González & Woods, 2008).



Gambar 8 Tahapan pengolahan citra

Sumber: (González & Woods, 2008)

Berdasarkan proses manipulasi elemen citra maka pengolahan citra dapat dikelompokkan menjadi tiga tingkat operasi, yaitu:

1. Operasi tingkat titik

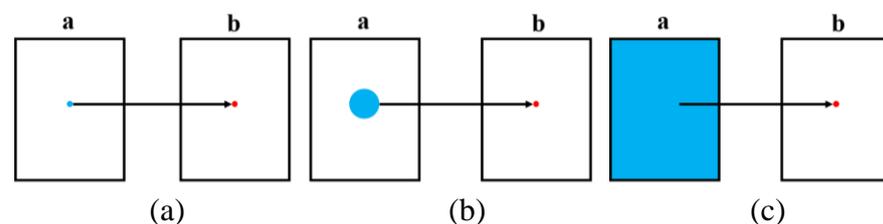
Operasi tingkat titik adalah operasi pengolahan citra yang dilakukan pada piksel tunggal pada sebuah citra. Pada operasi ini, piksel pada lokasi yang dikehendaki dimodifikasi dengan operasi yang ditentukan dan hasilnya berupa nilai piksel baru ditempatkan pada lokasi yang bersesuaian di dalam citra yang baru. Operasi ini dilakukan untuk keseluruhan piksel di dalam citra (Sri Ratna Sulistiyanti dkk., 2016).

2. Operasi tingkat lokal

Operasi tingkat lokal adalah operasi pengolahan citra yang dilakukan pada piksel yang dikehendaki dan dipengaruhi oleh intensitas piksel tetangganya. Pada operasi ini, piksel yang dikehendaki dimodifikasi menggunakan operasi dimana tetangga piksel tersebut memberikan pengaruh terhadap nilai yang dihasilkan dari operasi. Nilai baru ini ditempatkan pada lokasi yang bersesuaian di dalam citra yang baru dan dilakukan untuk keseluruhan piksel pada citra. Contoh aplikasi dari operasi ini adalah pelembutan dan penajaman citra (Sri Ratna Sulistiyanti dkk., 2016).

3. Operasi tingkat global

Operasi tingkat global adalah operasi pengolahan citra yang dilakukan pada piksel untuk menghasilkan nilai baru pada lokasi yang bersesuaian dan penentuan nilai baru ini dipengaruhi oleh nilai intensitas seluruh piksel pada citra. Contoh aplikasi operasi ini adalah operasi perataan histogram (Sri Ratna Sulistiyanti dkk., 2016). Ilustrasi berbagai tipe operasi pengolahan citra ditunjukkan pada Gambar 9.



Gambar 9 Ilustrasi berbagai tipe operasi pengolahan citra: (a) operasi tingkat titik; (b) operasi tingkat lokal; (c) operasi tingkat global

Sumber: (Young dkk., 2004)

2.5 Pengolahan Video Digital

Video adalah bentuk informasi visual yang mencakup citra diam dan citra yang berubah seiring waktu. Citra diam adalah distribusi intensitas spasial yang tetap, sementara citra yang berubah seiring waktu adalah citra dengan pola intensitas spasial-temporal yang berubah seiring waktu. Video dapat disebut juga sebagai urutan citra, karena citra yang berubah seiring waktu direpresentasikan sebagai urutan citra diam yang diambil pada interval waktu tertentu. Saat ini video telah dikembangkan dari rekaman, penyimpanan dan transmisi dalam bentuk

analog ke bentuk digital untuk komunikasi video interaktif dan layanan yang memerlukan pemrosesan video digital (Tekalp & Murat, 1995).

Video digital terdiri dari piksel, dengan banyak piksel dalam satu *frame* dan beberapa *frame* per detik yang biasanya 60 *Frame Per Second* (FPS). Jumlah piksel menentukan kompleksitas perangkat keras yang diperlukan untuk manipulasi. Setiap piksel memiliki warna unik yang dibuat dari kombinasi warna primer merah, biru, dan hijau, dengan nilai setiap komponen dijelaskan secara presisi oleh nilai piksel yang direpresentasikan dalam bit. Jumlah bit yang lebih tinggi memberikan representasi yang lebih akurat. Misalnya, menggunakan delapan bit untuk setiap komponen menghasilkan 256 nilai yang mungkin untuk merah, biru, dan hijau, yang dikenal sebagai kedalaman warna 8 bit (ichael Parker & Suhel Dhanani, 2012).

Pengolahan video digital merujuk pada manipulasi *bistream* video digital. Semua aplikasi dari video digital saat ini memerlukan pemrosesan digital untuk kompresi data. Beberapa aplikasi dari pemrosesan ini seperti analisis gerakan, konversi standar, peningkatan dan restorasi untuk mendapatkan citra yang berkualitas lebih baik atau mengekstrak beberapa informasi tertentu (Tekalp & Murat, 1995).

2.6 Pre-processing

Tahap *pre-processing* dikenal juga sebagai tahap peningkatan citra. Tujuan tahap ini adalah untuk meningkatkan keberhasilan dalam proses pengolahan citra tahap berikutnya, serta memperjelas kemunculan ciri-ciri yang terdapat dalam sebuah citra. Beberapa tahap yang umum dilakukan seperti peningkatan kecerahan, pengubahan kontras, penghilangan *noise*, transformasi geometri dan penentuan bagian citra yang dibutuhkan untuk diolah lebih lanjut. Secara matematis, peningkatan kualitas citra merupakan proses mengubah citra $f(x, y)$ menjadi $f'(x, y)$ sehingga ciri-ciri yang ada pada $f(x, y)$ lebih menonjol. Proses pada domain spasial dinyatakan secara matematis pada persamaan 2 (Sri Ratna Sulistiyanti dkk., 2016).

$$g(x, y) = T[f(x, y)] \quad (2)$$

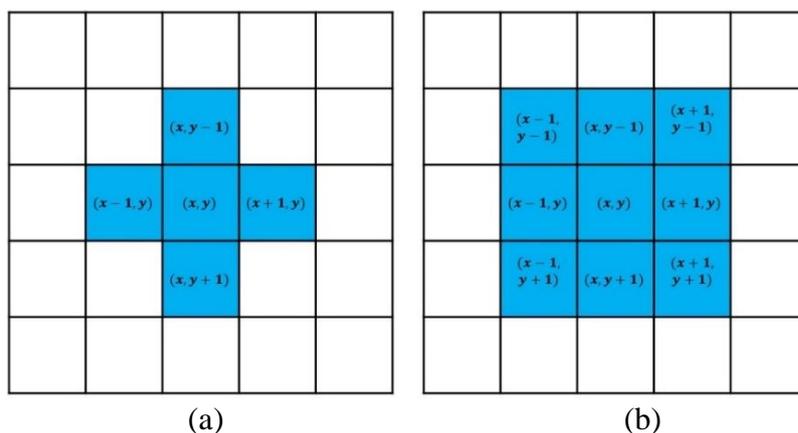
dimana,

T = operator

$f(x, y)$ = citra masukan

$g(x, y)$ = citra keluaran

Tahap *pre-processing* erat kaitannya dengan operasi ketetanggaan piksel. Operasi ketetanggaan piksel adalah operasi pengolahan citra untuk mendapatkan nilai suatu piksel yang melibatkan nilai piksel-piksel di sekitarnya. Hal ini didasarkan pada kenyataan bahwa setiap piksel pada umumnya tidak berdiri sendiri, melainkan terkait dengan piksel tetangga. Karena merupakan bagian suatu objek tertentu di dalam citra. Ketetanggaan piksel yang umum digunakan adalah 4-ketetanggaan dan 8-ketetanggaan. Pada 4-ketetanggaan yang merupakan tetangga terdekat piksel P yaitu $\{(x, y - 1), (x + 1, y), (x - 1, y), (x, y + 1)\}$. Sedangkan pada 8-ketetanggaan, tetangga piksel P yaitu piksel-piksel yang berada disekitar P dengan totalnya sebanyak 8 buah. Sebagai *filter* atau tapis, operasi ketetanggaan piksel berfungsi untuk menyaring atau paling tidak mengurangi gangguan atau penyimpangan pada citra (Arnita dkk., 2022). Gambar 10 menunjukkan ilustrasi ketetanggaan piksel.



Gambar 10 Ilustrasi ketetanggaan piksel: (a) 4-tetangga; (b) 8-tetangga
Sumber: (Irianto, 2014)

2.6.1 Image Resizing

Image resizing berarti mengubah ukuran citra dari ukuran yang diberikan menjadi ukuran yang berbeda. Secara umum, *resizing* termasuk memperbesar dan memperkecil citra. Memperbesar citra di domain spasial biasanya dilakukan dengan menggunakan algoritma interpolasi, seperti *spline*, *bicubic*, *bilinear*, dan lain-lain. Memperkecil citra dapat dilakukan menggunakan teknik serupa, atau dengan hanya mengambil sampel piksel pada interval yang tetap pada citra asli (Dong & Pan, 2023).

2.6.2 Peningkatan kontras dan kecerahan

Metode peningkatan kontras dan kecerahan banyak diimplementasikan dalam pemrosesan citra. Biasanya, citra yang diperoleh memiliki kontras rendah dan pencahayaan yang buruk. Hal ini menjadi tantangan bagi persepsi manusia, serta aplikasi pemrosesan citra dan visi komputer. Oleh karena itu, kualitas citra harus ditingkatkan sebelum digunakan agar lebih mudah dianalisis oleh mesin. Hal ini bertujuan untuk menekankan fitur-fitur yang diperlukan dalam citra dan video (Bataineh, 2023; Singh & Bhandari, 2020). Rumus untuk meningkatkan kontras dan kecerahan pada citra ditunjukkan pada persamaan 3 (Szeliski, 2010).

$$g(i, j) = \alpha \times f(i, j) + \beta \quad (3)$$

dimana,

- $f(i, j)$ = intensitas titik (i, j) pada citra input
- $g(i, j)$ = intensitas titik (i, j) pada citra output
- α = parameter *gain* atau pengatur kontras
- β = parameter *bias* atau pengatur kecerahan

2.6.3 Peregangan kontras

Citra dengan kontras rendah sering terjadi akibat kondisi pencahayaan yang buruk atau tidak merata, atau disebabkan oleh nonlinieritas atau rentang dinamis kecil dari sensor pengambil citra (Jain, 1989). Peregangan kontras atau sering disebut normalisasi adalah teknik penyempurnaan citra yang

sederhana untuk meningkatkan kontras dalam sebuah citra dengan memperpanjang rentang nilai intensitas yang terkandung di dalamnya untuk mencakup rentang nilai yang diinginkan. Rumus untuk melakukan peregangan kontras ditunjukkan pada persamaan 4 (R. Fisher dkk., 1996).

$$P_{out} = (P_{in} - c) \left(\frac{b - a}{d - c} \right) + a \quad (4)$$

dimana,

- P_{in} = nilai intensitas pada setiap piksel citra asal
- P_{out} = nilai intensitas pada piksel setelah peregangan kontras
- a = batas terendah untuk setiap piksel setelah normalisasi
- b = batas tertinggi untuk setiap piksel setelah normalisasi
- c = nilai piksel terendah dalam citra
- d = nilai piksel tertinggi dalam citra

2.6.4 Segmentasi citra

Segmentasi citra melibatkan pembagian atau pemisahan citra menjadi wilayah-wilayah dengan atribut yang serupa (William K. Pratt, 2001). Dapat diartikan juga sebagai proses pemberian label ke setiap piksel dalam citra sedemikian rupa sehingga piksel dengan label yang sama memiliki karakteristik tertentu. Tujuan dari segmentasi adalah untuk menyederhanakan dan/atau mengubah representasi suatu citra menjadi sesuatu yang lebih bermakna dan lebih mudah untuk dianalisis. (Ranjay Khrisna, 2017). Contoh aplikasi segmentasi adalah untuk memisahkan antara objek dengan latar belakang (Sri Ratna Sulistiyanti dkk., 2016). Gambar 11 menunjukkan contoh citra sebelum dan setelah dilakukan segmentasi.



Gambar 11 Contoh segmentasi: (a) citra original; (b) citra setelah segmentasi
Sumber: (Niu & Li, 2019)

Salah satu metode dalam segmentasi citra adalah sistem ambang batas (*thresholding*). Metode ini menetapkan nilai piksel menjadi 255 (mewakili putih atau daerah terang) jika nilainya lebih besar atau sama dengan ambang batas t , atau menjadi 0 (mewakili hitam atau daerah gelap) jika di bawah ambang batas. Secara matematis, fungsi untuk setiap piksel $g(m, n)$ ditunjukkan pada persamaan 5. Metode ini membagi piksel-piksel dalam sebuah citra menjadi klasifikasi biner dari daerah-daerah terang dan daerah-daerah gelap, tergantung apakah $f(m, n)$ lebih besar atau sama dengan t (Ranjay Khrisna, 2017). Warna objek (biasanya putih) disebut sebagai warna *foreground*. Bagian lainnya (biasanya hitam) disebut sebagai warna *background*. Namun, tergantung pada gambar yang akan dilakukan *threshold*, polaritas ini dapat terbalik, objek ditampilkan dengan nilai 0 dan *background* dengan nilai 1 (R. Fisher dkk., 1996).

$$g(m, n) = \begin{cases} 255 & f(m, n) \geq t \\ 0 & \text{sebaliknya} \end{cases} \quad (5)$$

dimana,

$$\begin{aligned} f(m, n) &= \text{intensitas citra } \textit{grayscale} \text{ pada titik } (m, n) \\ g(m, n) &= \text{intensitas citra biner pada titik } (m, n) \\ t &= \text{ambang batas} \end{aligned}$$

Untuk mengubah citra berwarna (RGB) menjadi citra biner menggunakan *thresholding*, langkah pertama adalah mengonversi citra menjadi citra *grayscale*. Setelah itu, citra *grayscale* diubah menjadi citra biner menggunakan teknik *thresholding*. Persamaan 6 menunjukkan formula yang digunakan dalam proses konversi dari RGB ke *Grayscale* (Padmavathi & Thangadurai, 2016).

$$f(x) = 0,2989 \times R + 0,5870 \times G + 0,1140 \times B \quad (6)$$

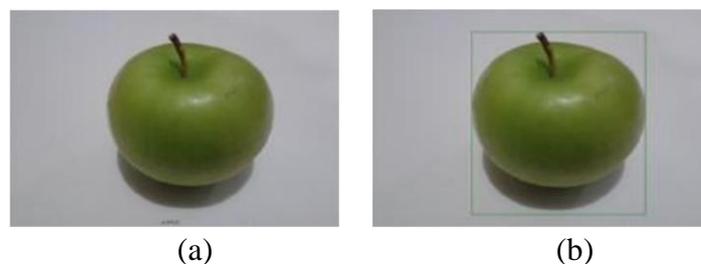
dimana,

$$\begin{aligned} f(x) &= \text{intensitas citra hasil konversi} \\ R &= \text{nilai intensitas citra pada saluran } \textit{red} \end{aligned}$$

G = nilai intensitas citra pada saluran *green*
 R = nilai intensitas citra pada saluran *blue*

2.6.5 Deteksi objek

Deteksi objek melibatkan deteksi instansi dari objek dari satu atau beberapa kelas dalam sebuah citra. Tujuan dari deteksi objek adalah untuk mendeteksi semua instansi objek dari satu atau beberapa kelas yang diketahui, seperti manusia, mobil, atau wajah dalam sebuah citra (Amit dkk., 2020). Salah satu metode dalam mendeteksi objek adalah *blob detection*. Analisis *blob* merupakan teknik dasar dalam pengolahan citra yang bertujuan untuk menganalisis karakteristik khusus dalam sebuah objek. Metode ini cocok untuk membedakan dan mengekstrak bentuk dan ukuran objek yang dideteksi dengan lingkungannya. Sebuah *blob* terdiri dari sekelompok piksel yang terhubung dan menentukan piksel tetangga dan bukan tetangga. Jumlah *blob* adalah jumlah piksel dalam *blob* untuk memilih ukuran blob dan menggambar kotak untuk menunjukkan *blob* yang dideteksi. *Bounding box* dari *blob* didefinisikan sebagai kotak terkecil dalam *blob* dari sebuah citra. *Bounding box* ini dikenal juga sebagai ROI (*Region of Interest*) (Yusuf dkk., 2018). Gambar 12 menunjukkan contoh citra hasil *bounding box*.



Gambar 12 Contoh *bounding box*: (a) citra original; (b) citra dengan *bounding box*
 Sumber: (Yusuf dkk., 2018)

Deteksi *blob* bekerja dengan mendeteksi *blob* putih (objek) pada *background* hitam. Sehingga apabila citra inputan berupa objek hitam, harus dikonversi terlebih dahulu menjadi berwarna putih dengan menggunakan metode *invert*.

A	Q
0	1
1	0

Gambar 13 Tabel kebenaran logika NOT
Sumber: (R. Fisher dkk., 1996)

Logika NOT atau *invert* adalah operator yang mengambil citra biner atau citra *grayscale* sebagai input dan menghasilkan citra negatif yaitu area gelap dalam citra input menjadi terang dan area terang menjadi gelap. Untuk menghasilkan citra negatif dari citra biner, dapat menggunakan operator logika NOT yang ditunjukkan pada Gambar 13 (R. Fisher dkk., 1996). Setiap piksel pada citra masukan yang memiliki logika 1 akan bernilai logika 0. Begitupun sebaliknya, piksel yang memiliki logika 0 akan bernilai 1. Persamaan 7 menunjukkan rumus untuk melakukan operasi *invert* (Fisher dkk., 1996).

$$Q(i, j) = 255 - P(i, j) \quad (7)$$

dimana,

$P(i, j)$ = intensitas citra input pada baris i dan kolom j

R = intensitas citra output pada baris i dan kolom j

2.6.6 Deteksi tepi

Deteksi tepi merupakan teknik yang sering digunakan pada bidang pengolahan citra dan juga *computer vision*, terutama pada bidang *feature detection* dan *feature extraction*. Deteksi tepi adalah suatu proses yang menghasilkan tepi-tepi dari objek pada citra yang bertujuan untuk memperbaiki detail dari citra yang kabur, yang terjadi karena *error* atau adanya efek dari proses akuisisi citra. Suatu titik (x, y) dikatakan sebagai tepi dari suatu citra bila titik tersebut mempunyai perbedaan yang tinggi dengan tetangganya. Dalam hal ini, batas objek berupa bagian tepi objek dalam suatu citra digital, tepi atau *edge* adalah sebuah kurva yang mengikuti yang berubah

dengan cepat berdasarkan intensitasnya. Contoh deteksi tepi ditunjukkan pada Gambar 14 (Arnita dkk., 2022).



Gambar 14 Contoh deteksi tepi
Sumber: (Arnita dkk., 2022)

Salah satu metode deteksi tepi adalah operator Sobel. Metode ini paling banyak digunakan sebagai deteksi tepi karena kesederhanaan dan keampuannya. Kelebihan dari metode ini adalah kemampuan untuk mengurangi derau sebelum melakukan perhitungan deteksi tepi (Arnita dkk., 2022). Metode ini mendeteksi magnitudo gradien dan arah untuk tepi deteksi. Pendeteksian tepi dapat dibuat lebih selektif dengan menerapkan *threshold* dari hasil citra gradien. Metode dasar dalam menggunakan operator Sobel metode ini sebagai berikut (Krig, 2016).

1. Melakukan *Sobel filter* dua arah yaitu sumbu x dan y menggunakan pendekatan kernel turunan dasar seperti 3×3 , dengan nilai yang ditunjukkan pada persamaan 8 dan 9.

$$S_x = \cdot \begin{bmatrix} -1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (8)$$

$$S_y = \cdot \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (9)$$

dimana,

S_x = turunan pertama arah horizontal

S_y = turunan pertama arah vertikal

2. Menghitung magnitudo gradien yang ditunjukkan pada persamaan 10.

$$G_m = \sqrt{S_y^2 + S_x^2} \quad (10)$$

dimana,

G_m = magnitudo gradien

3. Melakukan *threshold* nilai magnitudo gradien.

2.6.7 Hole Filling

Sebuah *hole* atau lubang dapat didefenisikan sebagai sebuah wilayah *background* yang dikelilingi oleh batas yang terhubung dari piksel *foreground* (González & Woods, 2008). Untuk melakukan pengisian pada sebuah lubang dapat menggunakan gabungan beberapa metode berikut.

1. Menerapkan algoritma *Flood Fill*

Algoritma *fill* bekerja dengan metode *scanline*. Algoritma ini mengisi sepanjang baris pemindaian sebelum berpindah pada koordinat vertikal y . Inti dari proses ini adalah bahwa *fill* akan mengisi semua piksel ketetangga-4 dengan titik awal pada baris pemindaian yang pertama. Kemudian, algoritma akan memeriksa baris pemindaian di atas dan di bawah untuk menemukan piksel ketetangga-4 dengan segmen baris pemindaian yang baru saja diisi dan berwarna sama seperti piksel titik awal sebelum diisi. Titik yang terhubung ditempatkan pada *stack*. Ketika pemindaian selesai, sebuah titik diambil dari *stack* yang kemudian menjadi titik awal yang baru (Agkland & Weste, 1981). Penggunaan *flood fill* pada *library* OpenCV akan mengisi komponen yang terhubung dimulai dari titik awal dengan warna yang ditentukan. Koneksi ditentukan oleh kedekatan warna/kecerahan piksel tetangga (OpenCV, 2024).

2. Melakukan operasi *Invert*

3. Melakukan operasi OR

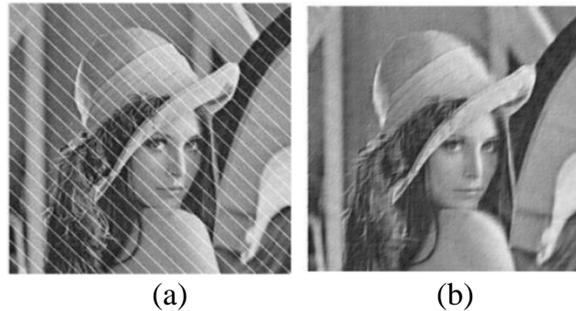
Operator OR merupakan sebuah operator logika yang didefinisikan pada variabel biner. Operator ini mengambil dua citra biner, I_1 dan I_2 , sebagai input dan mengembalikan citra I_3 dimana setiap piksel adalah 0 jika baik I_1 dan I_2 adalah 0, dan 1 jika tidak (R. B. Fisher dkk., 2013). Gambar 15 menunjukkan tabel kebenaran logika OR.

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

Gambar 15 Tabel kebenaran logika OR
Sumber: (R. Fisher dkk., 1996)

2.6.8 Penghilangan Derau

Dalam pengolahan citra digital, derau atau *noise* merupakan komponen yang tidak dikehendaki dan seringkali mengganggu kualitas citra. Untuk membersihkan citra dari derau, seringkali diperlukan proses penghalusan yang bertujuan untuk mengurangi atau menghilangkan visibilitas derau tanpa merusak detail-detail penting dalam citra. Salah satu cara umum untuk menghilangkan derau adalah dengan menggunakan proses *filtering*. *Filtering* adalah proses standar yang digunakan untuk menghapus derau dari citra sambil mempertahankan detail yang relevan. Dalam proses ini, nilai setiap piksel diganti dengan rata-rata nilai piksel di sekitarnya, yang membantu menciptakan citra yang lebih halus dan bebas dari gangguan derau (Arnita dkk., 2022). Gambar 16 menunjukkan contoh citra sebelum dan setelah dilakukan penghilangan derau.



Gambar 16 Contoh penghilangan derau: (a) citra dengan derau; (b) citra setelah penghilangan derau

Sumber: (Arnita dkk., 2022)

Median filter adalah teknik pemrosesan citra yang digunakan untuk mengurangi derau dalam citra. *Median filter* bekerja dengan mengambil nilai tengah dari nilai-nilai piksel tetangga. Proses ini dilakukan dengan mengurutkan nilai piksel di sekitar piksel target dari yang terkecil hingga terbesar dan kemudian memilih nilai yang berada di posisi tengah sebagai nilai baru untuk piksel target tersebut. Teknik ini efektif dalam mempertahankan detail di dalam citra sekaligus mengurangi derau, karena tidak begitu sensitif terhadap nilai-nilai ekstrem yang sering kali merupakan hasil dari derau (R. Fisher dkk., 1996).

2.7 Ekstraksi Fitur

Fitur citra adalah karakteristik atau atribut primitif yang membedakan dari suatu citra. Beberapa fitur bersifat alami atau didefinisikan oleh penampilan visual suatu citra, dapat pula berupa fitur buatan yang muncul dari manipulasi khusus terhadap suatu citra (William K. Pratt, 2001). Ekstraksi fitur adalah bagian dari proses pengurangan dimensi, dimana satu set awal data mentah dibagi dan direduksi menjadi grup yang lebih mudah dikelola sehingga ketika ingin mengolahnya akan lebih mudah. Karakteristik terpenting dari kumpulan data besar ini adalah memiliki sejumlah besar variabel. Variabel-variabel ini membutuhkan banyak sumber daya komputasi untuk diproses. Jadi ekstraksi fitur membantu untuk mendapatkan fitur terbaik dari kumpulan data besar tersebut dengan memilih dan menggabungkan variabel ke dalam fitur, sehingga secara efektif mengurangi jumlah data. Fitur-fitur ini mudah diproses, tetapi masih mampu menggambarkan kumpulan data aktual dengan akurasi dan orisinalitas (Arnita dkk., 2022).

2.7.1 Area

Sebuah citra biner diskrit yang berisi satu atau lebih objek, dimana $f(j, k) = 1$ jika sebuah piksel adalah bagian dari objek dan $f(j, k) = 0$ untuk semua piksel bukan objek atau latar belakang, maka luas setiap objek dalam citra adalah jumlah piksel dalam objek tersebut dimana $f(j, k) = 1$. Sebagai contoh, untuk sebuah persegi 2×2 piksel, luas objeknya adalah 4. Sebuah objek yang terdiri dari tiga piksel yang terhubung secara diagonal, luas objeknya adalah 3 (William K. Pratt, 2001). Untuk mendapatkan nilai *area*, dapat menggunakan *image spatial moment* pada orde $(0, 0)$ yang ditunjukkan pada persamaan 11.

$$M(m, n) = \sum_{x=1}^X \sum_{y=1}^Y (x)^m (y)^n F(x, y) \quad (11)$$

dimana,

- $M(m, n)$ = *spatial moments* dari orde (m, n)
- x = indeks iterasi baris
- y = indeks iterasi kolom
- m = orde momen dalam arah horizontal (baris)
- n = orde momen dalam arah vertikal (kolom)
- X = jumlah total baris citra
- Y = jumlah total kolom citra
- $F(x, y)$ = intensitas piksel pada lokasi (j, k) pada citra

2.7.2 Radius Distance

Distance adalah fungsi bernilai riil yang mengukur jarak antara dua titik citra (j_1, k_1) dan (j_2, k_2) . *Radius distance* adalah jarak antara titik *centroid* dan sebuah piksel region dari objek. Dari perhitungan ini dapat diperoleh fitur *minimum radius* dan *maximum radius*. *Minimum radius* adalah jarak minimum antara titik *centroid* dan sebuah piksel region. Sebaliknya, maksimum radius adalah jarak maksimum antara titik *centroid* dan sebuah piksel region. *Centroid* atau *center of gravity* adalah titik keseimbangan dari

fungsi citra $F(j, k)$, dimana sisi kiri dan kanan dari \bar{x}_k dan sisi atas dan bawah dari \bar{y}_j adalah sama. Tahapan untuk mendapatkan jarak tersebut sebagai berikut:

1. Menghitung *spatial moments* citra seperti ditunjukkan pada persamaan 11.
2. Melakukan perhitungan titik *centroid* \bar{x} dan \bar{y} seperti ditunjukkan pada persamaan 12 dan 13.

$$\bar{x} = \frac{M(1,0)}{M(0,0)} \quad (12)$$

$$\bar{y} = \frac{M(0,1)}{M(0,0)} \quad (13)$$

dimana,

$M(m, n)$ = *spatial moments* dari orde (m, n)

\bar{x} = *centroid* pada koordinat horizontal

\bar{y} = *centroid* pada koordinat vertikal

3. Melakukan perhitungan jarak antara *centroid* dengan piksel region menggunakan *euclidean distance* yang ditunjukkan pada persamaan 14.

$$d_E = [(j_1 - j_2)^2 + (k_1 - k_2)^2]^{1/2} \quad (14)$$

dimana,

d_E = *euclidean distance*

(j_1, k_1) = koordinat titik A pada citra

(j_2, k_2) = koordinat titik B pada citra

2.7.3 Principal Axes

Principal axes dari sebuah objek dapat didefinisikan sebagai segmen-segmen yang saling berpotongan secara tegak lurus di *centroid* objek dan mewakili arah dengan *zero-cross correlation*. *Major axis* atau *highest eigen value* adalah garis terpanjang yang memotong objek, sementara *minor axis* atau *lowest eigen value* adalah garis terpendek yang memotong objek.

Tahapan untuk memperoleh *eigen value* menurut (William K. Pratt, 2001) sebagai berikut:

1. Melakukan perhitungan *spatial central moments* yang ditunjukkan pada persamaan 15.

$$U(m, n) = \sum_{x=1}^X \sum_{y=1}^Y (x - \bar{x})^m (y - \bar{y})^n F(x, y) \quad (15)$$

2. Melakukan perhitungan *row-column cross moment of inertia* dan *three second-order moments of inertia* yang dapat dibuat matriks *the moment of inertia covariance* yang ditunjukkan pada persamaan 16

$$U = \begin{bmatrix} U(2, 0) & U(1, 1) \\ U(1, 1) & U(0, 2) \end{bmatrix} \quad (16)$$

3. Melakukan perhitungan untuk memperoleh *eigenvalues* dan *eigenvectors* menggunakan *singular-value decomposition* dari *covariance matrix* pada persamaan 17

$$E^T U E = \Lambda \quad (17)$$

Dimana kolom pada persamaan 18 adalah *eigenvectors* dari matriks U.

$$E = \begin{bmatrix} e_{11} & e_{12} \\ e_{21} & e_{22} \end{bmatrix} \quad (18)$$

Dan matriks Λ pada persamaan 19 adalah *eigenvalues* dari matriks U.

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \quad (19)$$

λ_1 dan λ_2 dapat diperoleh menggunakan persamaan 20 dan 21.

$$\lambda_1 = \frac{1}{2} [U(2,0) + U(0,2)] + \frac{1}{2} [U(2,0)^2 + U(0,2)^2 - 2U(2,0)U(0,2) + 4U(1,1)^2]^{1/2} \quad (20)$$

$$\lambda_2 = \frac{1}{2} [U(2,0) + U(0,2)] - \frac{1}{2} [U(2,0)^2 + U(0,2)^2 - 2U(2,0)U(0,2) + 4U(1,1)^2]^{1/2} \quad (21)$$

4. Melakukan perhitungan untuk memperoleh *major axis* atau *highest eigen value* dan *minor axis* atau *lowest eigen value* menggunakan persamaan 22 dan 23.

$$\lambda_M = \text{MAX}\{\lambda_1, \lambda_2\} \quad (22)$$

$$\lambda_N = \text{MIN}\{\lambda_1, \lambda_2\} \quad (23)$$

Selain itu, dapat diperoleh fitur rasio antara *minor axis* terhadap *major axis* atau *eigenvalue ratio* pada persamaan 24.

$$R_A = \frac{\lambda_N}{\lambda_M} \quad (24)$$

dimana,

$U(m, n)$ = *spatial central moments* dari orde (m, n)

x = indeks iterasi baris

y = indeks iterasi kolom

m = orde momen dalam arah horizontal (baris)

n = orde momen dalam arah vertikal (kolom)

X = jumlah total baris citra

Y = jumlah total kolom citra

$F(x, y)$ = intensitas piksel pada lokasi (j, k) pada citra

\bar{x} = *centroid* pada koordinat horizontal

\bar{y} = *centroid* pada koordinat vertikal

U	= three second-order moments of inertia
E	= eigenvectors dari U
Λ	= eigenvalues dari U
λ_M	= major axis atau highest eigen value
λ_N	= minor axis atau lowest eigen value
R_A	= rasio minor axis terhadap major axis atau eigenvalue ratio

2.8 Pearson Correlation

Korelasi adalah ukuran dari asosiasi monotonik antara 2 variabel. Hubungan monotonik antara 2 variabel adalah hubungan dimana saat nilai 1 variabel meningkat, demikian pula nilai variabel lainnya atau saat nilai 1 variabel meningkat, nilai variabel lainnya menurun. Dalam data yang berkorelasi, perubahan dalam magnitude 1 variabel berhubungan dengan perubahan dalam magnitude variabel lainnya, baik dalam arah yang sama atau berlawanan (Schober dkk., 2018).

Pearson Product Moment Correlation Coefficient adalah suatu metode yang menggambarkan hubungan antara 2 variabel. Titik-titik data 2 variabel menentukan arah atau tanda nilai koefisien korelasi (r) (Zinzendoff Okwonu dkk., 2020). Korelasi yang sempurna adalah -1 atau $+1$ artinya semua titik data berada tepat pada garis lurus (Schober dkk., 2018). Korelasi dapat positif, yang artinya searah jika variabel pertama besar, maka variabel kedua semakin besar juga. Korelasi negatif, yang artinya berlawanan arah jika variabel pertama besar, maka variabel kedua semakin mengecil (Jonathan Sarwono, 2018). Arti nilai koefisien korelasi ditunjukkan pada Tabel 2.

Tabel 2. Arti nilai koefisien korelasi

Nilai Koefisien Korelasi (r)	Kekuatan Korelasi
< 0,20	Hubungan dapat dianggap tidak ada
0,20 – 0,40	Hubungan ada tetapi rendah
0,40 – 0,70	Hubungan cukup
0,70 – 0,90	Hubungan tinggi
> 0,90 – 1,00	Hubungan sangat tinggi

Sumber: (Jonathan Sarwono, 2018)

Adapun rumus untuk memperoleh nilai *Pearson Correlation* ditunjukkan pada persamaan 25.

$$r = \frac{N \sum xy - (\sum x)(\sum y)}{\sqrt{[N \sum x^2 - (\sum x)^2] [N \sum y^2 - (\sum y)^2]}} \quad (25)$$

dimana,

r	= nilai koefisien korelasi
N	= jumlah pasangan data
x	= variabel bebas
y	= variabel terikat
$\sum x$	= jumlah setiap nilai variabel x
$\sum y$	= Jumlah setiap nilai variabel y
$\sum xy$	= Jumlah hasil kali nilai variabel x dan variabel y
$\sum x^2$	= Jumlah hasil kuadrat nilai variabel x
$\sum y^2$	= Jumlah hasil kuadrat nilai variabel y

2.9 Estimasi

Estimasi adalah proses mendekati nilai variabel target numerik menggunakan serangkaian variabel *predictor*. Model ini dibangun menggunakan *record* yang lengkap, yang memberikan nilai variabel target, serta prediktor. Kemudian, estimasi nilai variabel target pada data baru dibuat berdasarkan nilai prediktor. Bidang analisis statistik menyediakan beberapa metode estimasi seperti regresi linear sederhana dan regresi berganda (Chantal D. Larose & Daniel T. Larose, 2015).

Sebagai contoh, memperkirakan tekanan darah sistolik pasien di rumah sakit, berdasarkan usia pasien, jenis kelamin, indeks massa tubuh, dan tingkat natrium darah. Hubungan antara tekanan darah sistolik dan variabel prediktor dalam proses pelatihan akan memberikan model estimasi. Kemudian dapat diterapkan model tersebut pada kasus-kasus baru (Chantal D. Larose & Daniel T. Larose, 2015). Saat ini, metode estimasi banyak diterapkan pada *computer vision* seperti penelitian yang dilakukan oleh (Nyalala dkk., 2019) dalam mengestimasi massa dan volume tomat berbasis model regresi.

2.10 Linear Regression

Regression analysis adalah teknik statistik untuk menginvestigasi dan memodelkan hubungan antara beberapa variabel. Aplikasi dari regresi banyak terjadi di hampir setiap bidang, termasuk teknik, ilmu fisika dan kimia, ekonomi, manajemen, ilmu kehidupan dan biologi, serta ilmu sosial. Bahkan, analisis regresi mungkin merupakan teknik statistik yang paling banyak digunakan.

Multiple linear regression atau regresi linear berganda adalah persamaan yang menjelaskan hubungan satu variabel tak bebas y , dengan dua atau lebih variabel bebas x_1, x_2, \dots, x_k . Tujuan dari persamaan ini untuk memprediksi nilai variabel tak bebas y apabila nilai-nilai variabel bebasnya (x_1, x_2, \dots, x_k) diketahui (Douglas C. Montgomery dkk., 2021). Persamaan regresi linear berganda secara matematis ditunjukkan pada persamaan 26.

$$y = \beta_0 + x_1\beta_1 + x_2\beta_2 + \dots + x_k\beta_k + \varepsilon \quad (26)$$

dimana,

y	= variabel terikat atau respons
β_0	= konstanta
x_1, x_2, \dots, x_k	= variabel bebas atau regresor
$\beta_0, \beta_1, \dots, \beta_k$	= nilai koefisien regresi
ε	= <i>error</i>

Metode *least squares* dapat digunakan untuk mengestimasi koefisien regresi. Anggap bahwa $n > k$ observasi tersedia, dan y_i menyatakan respons yang diamati dan x_{ij} menyatakan observasi ke- i atau level dari regresor x_j . Data akan tampak pada Tabel 3. Diasumsikan istilah *error* ε dalam model memiliki $E(\varepsilon) = 0$, $Var(\varepsilon) = \sigma^2$, dan bahwa kesalahannya tidak berkorelasi (Douglas C. Montgomery dkk., 2021).

Secara umum, y adalah vektor $n \times 1$ dari observasi, X adalah matriks $n \times p$ dari level variabel regressor dimana $p = k + 1$, β adalah vektor $p \times 1$ dari koefisien regresi, dan e adalah vektor $n \times 1$ dari *random error*. Persamaan 29 menunjukkan persamaan *least-squares normal*.

$$X'X\hat{\beta} = X'y \quad (29)$$

dimana,

$$\hat{\beta} = \text{least-squares estimator}$$

Untuk mendapatkan *least-squares estimator* dari persamaan 29, digunakan persamaan 30

$$\hat{\beta} = (X'X)^{-1}X'y \quad (30)$$

Model regresi yang disesuaikan sesuai dengan tingkat variabel regresi $x' = [1, x_1, x_2, \dots, x_k]$ ditunjukkan pada persamaan 31

$$\hat{y} = x'\hat{\beta} = \hat{\beta}_0 + \sum_{j=1}^k \hat{\beta}_j x_j \quad (31)$$

2.11 Metrik Evaluasi

Sebuah model divalidasi dengan mengevaluasi kinerja prediksinya. Idealnya, evaluasi ini harus mewakili bagaimana model akan berperilaku saat diimplementasikan dalam pengaturan kehidupan nyata (Alice Zheng, 2015). Evaluasi untuk tugas estimasi dapat menggunakan RMSE, MAPE, sedangkan untuk tugas klasifikasi dapat menggunakan akurasi.

2.11.1 RMSE

Metode paling umum untuk menggambarkan kemampuan prediktif suatu model adalah dengan menggunakan *Root Mean Squared Error* (RMSE). Metrik ini merupakan fungsi dari residu model, yang merupakan nilai-nilai yang diamati dikurangi dengan prediksi model. Nilai biasanya diinterpretasikan sebagai jarak rata-rata antara residu dan nol atau sebagai

jarak rata-rata antara nilai-nilai yang diamati dan prediksi model (Kuhn & Johnson, 2013). Nilai 0 menunjukkan nilai terbaik, sedangkan nilai terburuk adalah $+\infty$. Persamaan untuk menghitung RMSE ditunjukkan pada persamaan 32 (Chicco dkk., 2021).

$$\text{RMSE} = \sqrt{\frac{1}{m} \sum_{i=1}^m (X_i - Y_i)^2} \quad (32)$$

dimana,

m = jumlah data

X_i = nilai prediksi

Y_i = nilai aktual

2.11.2 MAPE

MAPE merupakan metrik lainnya yang memiliki interpretasi sangat intuitif dalam hal kesalahan relatif. Penggunaan metrik ini disarankan pada tugas di mana lebih penting untuk peka terhadap variasi relatif daripada variasi absolut. Nilai 0 menunjukkan nilai terbaik, sedangkan nilai terburuk adalah $+\infty$. Persamaan untuk menghitung MAPE ditunjukkan pada persamaan 33 (Chicco dkk., 2021).

$$\text{MAPE} = \frac{1}{m} \sum_{i=1}^m \left| \frac{Y_i - X_i}{Y_i} \right| \quad (33)$$

dimana,

m = jumlah data

X_i = nilai prediksi

Y_i = nilai aktual

Dari perhitungan MAPE dapat diartikan bahwa hasil pengurangan antara nilai aktual dan prediksi yang telah dimutlakkan, kemudian di bagi dengan nilai aktual, kemudian dilakukan penjumlahan terhadap hasil-hasil tersebut. Semakin rendah nilai MAPE, kemampuan dari model yang digunakan dapat dikatakan baik. Tabel 4 menunjukkan *range* nilai yang dapat dijadikan bahan pengukuran mengenai kemampuan dari suatu model (Nabillah & Ranggadara, 2020).

Tabel 4. *Range* nilai MAPE

Range MAPE	Keterangan
< 10 %	Model sangat baik
10 – 20 %	Model baik
20 – 50 %	Model layak
> 50 %	Model buruk

Sumber: (Nabillah & Ranggadara, 2020)

2.11.3 Akurasi

Tingkat akurasi dapat didefinisikan sebagai jumlah klasifikasi yang benar pada keseluruhan set pengujian. *Confusion matrix* adalah tabel $n \times n$ yang berisi informasi tentang prediksi model klasifikasi dibandingkan dengan observasi aktual berdasarkan data. Tabel 5 menunjukkan ilustrasi *confusion matrix*. Untuk tabel 2×2 , *confusion matrix* berisi informasi tentang *true positive* (TP), *true negative* (TN), *false positive* (FP), dan *false negative* (FN) (Jude Chukwura Obi, 2023). TP adalah kondisi ketika hasil yang positif diprediksi dan apa yang terjadi adalah sama. FP adalah kondisi ketika hasil yang positif diprediksi, tetapi yang terjadi adalah hasil negatif. FN adalah kondisi ketika hasil yang negatif diprediksi dan apa yang terjadi adalah hasil positif. TN adalah kondisi ketika hasil yang negatif diprediksi dan apa yang terjadi adalah sama (Vujovic, 2021).

Tabel 5. Ilustrasi *confusion matrix*

		Nilai Aktual	
		P	N
Nilai Prediksi	P	TP (<i>True Positive</i>)	FP (<i>False Positive</i>)
	N	FN (<i>False Negative</i>)	TN (<i>True Negative</i>)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} = \frac{TP + TN}{P + N} \quad (34)$$

dimana,

TP = *true positive*

TN = *true negative*

FP = *false positive*

FN = *false negative*

$P + N$ = total dataset