

TESIS

**IDENTIFIKASI BARANG BERTUMPUK PADA KASIR
SUPERMARKET DENGAN TEKNIK ANOTASI POLYGON
MENGUNAKAN VISI KOMPUTER**

*Identification of Overlapping Items at Supermarket Cashiers with
Polygon Annotation Technique Using Computer Vision*

La Ode Muh. Armadi. AM

D082202029



**PROGRAM PASCASARJANA
DEPARTEMEN TEKNIK INFORMATIKA
UNIVERSITAS HASANUDDIN
MAKASSAR**

2024



PENGAJUAN TESIS

IDENTIFIKASI BARANG BERTUMPUK PADA KASIR SUPERMARKET DENGAN TEKNIK ANOTASI POLYGON MENGUNAKAN VISI KOMPUTER

Tesis

Sebagai Salah Satu Syarat Untuk Mencapai Gelar Magister Program Studi Teknik
Informatika

Disusun dan diajukan oleh

La Ode Muh. Armadi. Am

D082202029

Kepada

**FAKULTAS TEKNIK
UNIVERSITAS HASANUDDIN**

GOWA

2024

iii



TESIS

IDENTIFIKASI BARANG BERTUMPUK PADA KASIR SUPERMARKET DENGAN TEKNIK ANOTASI POLYGON MENGUNAKAN VISI KOMPUTER

La Ode Muh. Armadi. AM
D082202029

Telah dipertahankan di hadapan Panitia Ujian Tesis yang dibentuk dalam rangka penyelesaian studi
pada Program Magister Teknik Informatika Fakultas Teknik
Universitas Hasanuddin
Pada tanggal 12 Februari 2024
dan dinyatakan telah memenuhi syarat kelulusan

Menyetujui,

Pembimbing Utama



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

Pembimbing Pendamping



Dr. Ir. Ingrid Nurtanio, M.T.
NIP. 19610813 198811 2 001

Dekan Fakultas Teknik
Universitas Hasanuddin



Prof. Dr.Eng. Ir. Muhammad Isran Ramli, M.T. IPM., ASEAN.Eng.
NIP. 19730926 200012 1 002

Ketua Program Studi
S2 Teknik Informatika



Dr. Ir. Zahir Zainuddin, M.Sc.
NIP. 19640427 198910 1 002



PERNYATAAN KEASLIAN TESIS DAN PELIMPAHAN HAK CIPTA

Yang bertanda tangan dibawah ini :

Nama : La Ode Muh. Armadi. AM

Nomor mahasiswa : D082202029

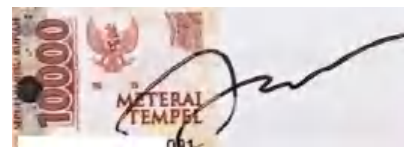
Program Studi : S2 Teknik Informatika

Dengan ini menyatakan bahwa, tesis berjudul “Identifikasi Barang Bertumpuk Pada Kasir Supermarket Dengan Teknik Anotasi Polygon Menggunakan Visi Komputer” adalah benar karya saya dengan arahan dari komisi pembimbing Prof. Dr. Ir. Indrabayu, S.T.,M.T.,M.Bus.Sys.,IPM.,ASEAN.Eng dan Dr. Ir. Ingrid Nurtanio, M.T. Karya ilmiah ini belum diajukan dan tidak sedang diajukan dalam bentuk apapun kepada perguruan tinggi mana pun. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan maupun tidak diterbitkan dari penulis lain telah disebutkan dalam teks dan dicantumkan kedalam Daftar Pustaka tesis ini. Sebagian isi tesis ini telah dipublikasi di Prosiding 2023 IEEE International Conference on Industry 4.0, Artificial Inteligence, and Communications Technology (IAICT), halaman 360-364, dan DOI: 10.1109/IAICT59002.2023.10205599 sebagai artikel yang berjudul “Snacks Detection Under Overlapped Conditions Using Computer Vision”

Dengan ini saya melimpahkan hak cipta dari karya tulis saya berupa tesis ini kepada Univeristas Hasanuddin

Gowa, Maret 2024

Yang Menyerahkan



La Ode Muh. Armadi. AM



KATA PENGANTAR

Alhamdulillah rabbil'alamin, segala puji bagi Allah Subhanahu Wa Ta'ala Yang Maha Sempurna, yang telah memberikan rahmat, hidayah dan pertolongan-Nya sehingga penulis dapat menyelesaikan tesis dengan judul “**Identifikasi Barang Bertumpuk Pada Kasir Supermarket Dengan Teknik Anotasi Polygon Menggunakan Visi Komputer**”. Tak lupa pula shalawat dan salam kepada Nabi Muhammad SAW yang telah menyinari dunia ini dengan keindahan ilmu dan akhlak yang diajarkan kepada seluruh umatnya.

Tesis ini disusun untuk memenuhi persyaratan untuk memperoleh gelar Magister Komputer (M.Kom.) pada Program Pascasarjana Departemen Teknik Informatika Universitas Hasanuddin Makassar. Tentunya penyelesaian tesis ini tidak terlepas dari dukungan dan bantuan dari semua pihak. Untuk itu, dengan penuh kerendahan hati penulis menyampaikan terima kasih setulus-tulusnya dan setinggi-tingginya kepada:

1. Ayahanda penulis Arman Mol, S.H. dan ibunda tercinta Alm. Nur Insani. yang telah memberikan dukungan materil, doa dan motivasi yang kuat kepada penulis, hingga penulis dapat menyelesaikan penelitian ini.
2. Bapak Dr. Ir. Zahir Zainuddin, M.Sc selaku Ketua Program Studi S2 Departemen Teknik Informatika, yang telah banyak mendukung dan membantu selama penulis menempuh pendidikan pascasarjana di Universitas Hasanuddin.



3. Bapak Prof. Dr. Indrabayu, S.T., M.T., M. Bus. Sys., IPM., ASEAN.Eng sebagai pembimbing pertama dan Ibu Dr. Ir. Ingrid Nurtanio, M.T. selaku dosen pembimbing kedua yang telah meluangkan waktunya kepada penulis untuk membimbing, memberikan masukan, memotivasi tiada hentihentinya hingga tahap penyelesaian tesis ini.
4. Bapak Dr. Ir. Zahir Zainuddin, M.Sc, Ibu Mukaramah Yusuf, B.Sc., M.Sc., Ph.D, dan Bapak Dr. Eng. Zulkifli Tahir, S.T., M.Sc selaku penguji yang memberikan masukan dan saran yang membangun selama proses penelitian berlangsung.
5. Karmila, S.I.Kom, selaku teman hidup dan teman yang tidak bisa digambarkan yang selalu menemani dalam suka dan duka dalam proses panjang penyelesaian tesis ini
6. Rekan–rekan Mahasiswa S2 Departemen Teknik Informatika angkatan 2020-2 yang selalu mendukung dalam proses penyelesaian tesis ini.

Penulis menyadari bahwa tesis masih jauh dari kata sempurna dan di dalam penyelesaiannya masih menemui kesulitan dan hambatan, sehingga penulis tetap mengharapkan saran dan kritik untuk pengembangan lebih lanjut, agar dapat memberikan manfaat yang banyak bagi semua pembaca.

Makassar, Februari 2024

Penulis



ABSTRAK

La Ode Muh. Armadi. AM, Identifikasi Barang Bertumpuk Pada Kasir Supermarket Dengan Teknik Anotasi Polygon Menggunakan Visi Komputer, (Dibimbing oleh Indrabayu dan Ingrid Nurtanio)

Pada masa kini, fenomena antrian telah menjadi komponen integral dalam kegiatan sehari-hari, salah satunya terdapat pada antrian kasir di supermarket. Salah satu kendala antrian di supermarket ini disebabkan oleh proses pemindaian kode *barcode* yang dilaksanakan satu per satu oleh *barcode scanner* di kasir. Selain itu, masalah lainnya adalah ketidakmampuan *barcode scanner* untuk mengenali *barcode* pada kemasan barang yang telah rusak atau pudar, sehingga membutuhkan entri manual oleh petugas kasir. Keadaan ini berimplikasi pada lamanya durasi transaksi. Penelitian ini bertujuan untuk membangun model yang mampu melakukan klasifikasi dan multideteksi terhadap objek yang saling tumpang tindih. Algoritma yang digunakan dalam penelitian ini untuk proses klasifikasi dan multideteksi adalah algoritma Mask-RCNN, dengan variasi *hyperparameter* pelatihan antara 50 hingga 250 epoch. Jumlah data yang digunakan dalam penelitian ini mencapai 635 *snack*, yang terbagi menjadi 508 data pelatihan dan 127 data validasi. Hasil dari penelitian ini diuji dalam 4 skenario dengan jarak pengambilan gambar sebesar 40 cm, 60 cm, 80 cm dan 100 cm dengan jumlah 2 hingga 20 *snack* pada masing-masing skenario. Skenario terbaik berada pada skenario 1 dengan jumlah 8 *snack* dengan jarak pengambilan 40 cm menghasilkan nilai *F1 score* sebesar 1.

Kata Kunci : Kasir, Mask R-CNN, Supermarket.



ABSTRACT

La Ode Muh. Armadi. AM. *Identification of Overlapping Items at Supermarket Cashiers with Polygon Annotation Technique Using Mask R-CNN Computer Vision. (Supervised by Indrabayu and Ingrid Nurtanio).*

Nowadays, the queuing phenomenon has become an integral component in daily activities, one of which is found in the cashier queue at the supermarket. One of the obstacles to queuing in supermarkets is caused by the process of scanning barcode codes that are carried out one by one by the barcode scanner at the cashier. In addition, another problem is the inability of the barcode scanner to recognize barcodes on the packaging of goods that have been damaged or faded, thus requiring manual entry by the cashier. This situation has implications for the duration of the transaction. This research aims to build a model that is able to perform classification and multidetection of overlapping objects. The algorithm used in this research for the classification and multidetection process is the Mask-RCNN algorithm, with variations in training hyperparameters between 50 to 250 epochs. The amount of data used in this study reached 635 snacks, which were divided into 508 training data and 127 validation data. The results of this study were tested in 4 scenarios with shooting distances of 40 cm, 60 cm, 80 cm and 100 cm with the number of 2 to 20 snacks in each scenario. The best scenario is in scenario 1 with the number of 8 snacks with a shooting distance of 40 cm resulting in an F1 score of 1.

Keywords: Cashier, Mask R-CNN, Supermarket.



DAFTAR ISI

PENGAJUAN TESIS.....	iii
PERSETUJUAN TESIS.....	iv
PERNYATAAN KEASLIAN TESIS DAN PELIMPAHAN HAK CIPTA	v
KATA PENGANTAR	vi
ABSTRAK	viii
ABSTRACT	ix
DAFTAR ISI	x
DAFTAR TABEL	xiii
DAFTAR GAMBAR	xvii
BAB I PENDAHULUAN	1
1.1. Latar Belakang Masalah.....	1
1.2. Rumusan Masalah	3
1.3. Tujuan Penelitian.....	3
1.4. Manfaat Penelitian.....	3
1.5. Batasan Masalah.....	4
BAB II LANDASAN TEORI	5
2.1 Landasan Teori	5
2.1.1 Pengolahan Citra (Image Processing).....	5
2.1.2. <i>Artificial Inteligence</i>	5
2.1.3. Augmentasi Data.....	7
2.1.4. <i>Computer Vision</i>	7
2.1.5. <i>Artificial Neural Netwok</i>	9
2.1.6. <i>Deep Learning</i>	10
2.1.7. <i>Convolutional Neural Network</i>	11
2.1.8. Mask R-CNN	12
2.1.9. ResNet.....	13



2.1.10.	Tensorflow.....	16
2.1.11.	Jupyter Notebook.....	18
2.1.12.	Evaluasi Sistem	19
2.2.	Metode Penyelesaian Masalah	21
2.2.2.	<i>State of The Art</i> Penelitian.....	21
2.1.	Target Hasil Penelitian	25
2.2.	Kerangka Pikir.....	25
BAB III LANGKAH PENELITIAN		26
3.1.	Tahapan Penelitian	26
3.1.1.	Tahap Studi Literatur	26
3.1.2.	Tahap Persiapan Data.....	27
3.1.3.	Tahap Pengolahan Data.....	27
3.1.4.	Tahap Perancangan Sistem	27
3.1.5.	Tahap Pengujian.....	28
3.1.6.	Tahap Evaluasi Sistem	28
3.1.7.	Pembuatan Laporan.....	28
3.2.	Waktu dan Lokasi Penelitian.....	28
3.3.	Instrumen Penelitian.....	29
1.	Perangkat Lunak.....	29
2.	Perangkat Keras.....	29
3.4.	Skenario Pengambilan Data	29
3.5.	Perancangan Sistem.....	30
3.5.1.	<i>Image Annotation</i>	31
3.5.2.	<i>Image Augmentation</i>	32
3.5.3.	<i>Split Data</i>	33
3.5.4.	Tahap <i>Training</i>	33
3.5.5.	Tahap <i>Validation</i>	42
3.5.6.	Tahap <i>Testing</i>	44
3.5.7.	POS System (Point of Sales System).....	45



3.6. Teknik Evaluasi Sistem	45
BAB IV HASI DAN PEMBAHASAN.....	46
4.1. Hasil Training.....	46
4.1.1. Pengujian Hyperparameter.....	46
4.1.2. Hasil Pengujian Skenario	49
4.1.3. Pengujian Sistem POS.....	106
BAB V KESIMPULAN DAN SARAN.....	109
5.1. Kesimpulan.....	109
5.2. Saran.....	109
DAFTAR PUSTAKA	110
LAMPIRAN.....	113



DAFTAR TABEL

Tabel 1. Tabel <i>State of The Art</i>	21
Tabel 2. Spesifikasi Hardware dan Software Pelatihan dan Pengujian	34
Tabel 3. Konfigurasi Algoritma	34
Tabel 4. Hasil Training <i>Mask R-CNN</i> Dengan ResNet50	47
Tabel 5. Hasil Training <i>Mask R-CNN</i> Dengan ResNet101	48
Tabel 6. Tabel Rincian Skenario Pengujian	49
Tabel 7. Pengujian Skenario 1	49
Tabel 8. Tabel Hasil <i>Confusion Matrix</i> Skenario 1 Pengujian 1.....	54
Tabel 9. Tabel Hasil Perhitungan Precision dan Recall Skenario 1 Pengujian 1..	55
Tabel 10. Tabel Hasil Perhitungan F1 Score Skenario 1 Pengujian 1	55
Tabel 11. Tabel Hasil <i>Confusion Matrix</i> Skenario 1 Pengujian 2.....	55
Tabel 12. Tabel Hasil Perhitungan Precision dan Recall Skenario 1 Pengujian 2	56
Tabel 13. Tabel Hasil Perhitungan F1 Score Skenario 1 Pengujian 2	56
Tabel 14. Tabel Hasil <i>Confusion Matrix</i> Skenario 1 Pengujian 3.....	56
Tabel 15. Tabel Hasil Perhitungan Precision dan Recall Skenario 1 Pengujian 3	56
Tabel 16. Tabel Hasil Perhitungan F1 Score Skenario 1 Pengujian 3	57
Tabel 17. Tabel Hasil <i>Confusion Matrix</i> Skenario 1 Pengujian 4.....	57
Tabel 18. Tabel Hasil Perhitungan Precision dan Recall Skenario Pengujian 4...	57
Tabel 19. Tabel Hasil Perhitungan F1 Score Skenario 1 Pengujian 4	58
Tabel 20. Tabel Hasil <i>Confusion Matrix</i> Skenario 1 Pengujian 5.....	58
Tabel 21. Tabel Hasil Perhitungan Precision dan Recall Skenario 1 Pengujian 5	58
Tabel 22. Tabel Hasil Perhitungan F1 Score Skenario 1 Pengujian 5	58
Tabel 23. Tabel Hasil <i>Confusion Matrix</i> Skenario 1 Pengujian 6.....	59
Tabel 24. Tabel Hasil Perhitungan Precision dan Recall Skenario 1 Pengujian 6	59
Tabel 25. Tabel Hasil Perhitungan F1 Score Skenario 1 Pengujian 6	59
Tabel 26. Tabel Hasil <i>Confusion Matrix</i> Skenario 1 Pengujian 7.....	59
Tabel 27. Tabel Hasil Perhitungan Precision dan Recall Skenario 1 Pengujian 7	60
Tabel 28. Tabel Hasil Perhitungan F1 Score Skenario 1 Pengujian 7	60
Tabel 29. Tabel Hasil <i>Confusion Matrix</i> Skenario 1 Pengujian 8.....	60
Tabel 30. Tabel Hasil Perhitungan Precision dan Recall Skenario 1 Pengujian 8	61
Tabel 31. Tabel Hasil Perhitungan F1 Score Skenario 1 Pengujian 5	61
Tabel 32. Tabel Hasil <i>Confusion Matrix</i> Skenario 1 Pengujian 9.....	61
Tabel 33. Tabel Hasil Perhitungan Precision dan Recall Skenario 1 Pengujian 9	61
Tabel 34. Tabel Hasil Perhitungan F1 Score Skenario 1 Pengujian 9	62
Tabel Hasil <i>Confusion Matrix</i> Skenario 1 Pengujian 10.....	62
Tabel Hasil Perhitungan Precision dan Recall Skenario 1 Pengujian 10	62



Tabel 37. Tabel Hasil Perhitungan F1 Score Skenario 1 Pengujian 10	62
Tabel 38. Pengujian Skenario 2	63
Tabel 39. Tabel Hasil <i>Confusion Matrix</i> Skenario 2 Pengujian 1.....	68
Tabel 40. Tabel Hasil Perhitungan Precision dan Recall Skenario 2 Pengujian 1	69
Tabel 41. Tabel Hasil Perhitungan F1 Score Skenario Pengujian 1	69
Tabel 42. Tabel Hasil <i>Confusion Matrix</i> Skenario 2 Pengujian 2.....	69
Tabel 43. Tabel Hasil Perhitungan Precision dan Recall Skenario 2 Pengujian 2	69
Tabel 44. Tabel Hasil Perhitungan F1 Score Skenario 2 Pengujian 2	70
Tabel 45. Tabel Hasil <i>Confusion Matrix</i> Skenario 2 Pengujian 3.....	70
Tabel 46. Tabel Hasil Perhitungan Precision dan Recall Skenario 2 Pengujian 3	70
Tabel 47. Tabel Hasil Perhitungan F1 Score Skenario 2Pengujian 3	71
Tabel 48. Tabel Hasil <i>Confusion Matrix</i> Skenario 2 Pengujian 4.....	71
Tabel 49. Tabel Hasil Perhitungan Precision dan Recall Skenario 2 Pengujian 4	71
Tabel 50. Tabel Hasil Perhitungan F1 Score Skenario 2 Pengujian 4	71
Tabel 51. Tabel Hasil <i>Confusion Matrix</i> Skenario 2 Pengujian 5.....	72
Tabel 52. Tabel Hasil Perhitungan Precision dan Recall Skenario 2 Pengujian 5	72
Tabel 53. Tabel Hasil Perhitungan F1 Score Skenario 2 Pengujian 5	72
Tabel 54. Tabel Hasil <i>Confusion Matrix</i> Skenario 2 Pengujian 6.....	72
Tabel 55. Tabel Hasil Perhitungan Precision dan Recall Skenario 2 Pengujian 6	73
Tabel 56. Tabel Hasil Perhitungan F1 Score Skenario 2 Pengujian 6	73
Tabel 57. Tabel Hasil <i>Confusion Matrix</i> Skenario 2 Pengujian 7.....	73
Tabel 58. Tabel Hasil Perhitungan Precision dan Recall Skenario 2 Pengujian 7	74
Tabel 59. Tabel Hasil Perhitungan F1 Score Skenario 2 Pengujian 7	74
Tabel 60. Tabel Hasil <i>Confusion Matrix</i> Skenario 2 Pengujian 8.....	74
Tabel 61. Tabel Hasil Perhitungan Precision dan Recall Skenario 2 Pengujian 8	74
Tabel 62. Tabel Hasil Perhitungan F1 Score Skenario 2 Pengujian 5	75
Tabel 63. Tabel Hasil <i>Confusion Matrix</i> Skenario 2 Pengujian 9.....	75
Tabel 64. Tabel Hasil Perhitungan Precision dan Recall Skenario 2 Pengujian 9	75
Tabel 65. Tabel Hasil Perhitungan F1 Score Skenario 2 Pengujian 9	76
Tabel 66. Tabel Hasil <i>Confusion Matrix</i> Skenario 2 Pengujian 10.....	76
Tabel 67. Tabel Hasil Perhitungan Precision dan Recall Skenario 2 Pengujian 10	76
Tabel 68. Tabel Hasil Perhitungan F1 Score Skenario 2 Pengujian 10	76
Tabel 69. Pengujian Skenario 3	77
Tabel 70. Tabel Hasil <i>Confusion Matrix</i> Skenario 3 Pengujian 1.....	82
Tabel 71. Tabel Hasil Perhitungan Precision dan Recall Skenario 3 Pengujian 1	83
Tabel Hasil Perhitungan F1 Score Skenario 3 Pengujian 1	83
Tabel Hasil <i>Confusion Matrix</i> Skenario 3 Pengujian 2.....	83



Tabel 74. Tabel Hasil Perhitungan Precision dan Recall Skenario 3 Pengujian 2	83
Tabel 75. Tabel Hasil Perhitungan F1 Score Pengujian 2	84
Tabel 76. Tabel Hasil <i>Confusion Matrix</i> Skenario 3 Pengujian 3.....	84
Tabel 77. Tabel Hasil Perhitungan Precision dan Recall Skenario 3 Pengujian 3	84
Tabel 78. Tabel Hasil Perhitungan F1 Score Skenario 3 Pengujian 3	85
Tabel 79. Tabel Hasil <i>Confusion Matrix</i> Skenario 3 Pengujian 4.....	85
Tabel 80. Tabel Hasil Perhitungan Precision dan Recall Skenario 3 Pengujian 4	85
Tabel 81. Tabel Hasil Perhitungan F1 Score Skenario 3 Pengujian 4	86
Tabel 82. Tabel Hasil <i>Confusion Matrix</i> Skenario 3 Pengujian 5.....	86
Tabel 83. Tabel Hasil Perhitungan Precision dan Recall Skenario 3 Pengujian 5	86
Tabel 84. Tabel Hasil Perhitungan F1 Score Skenario 3 Pengujian 5	86
Tabel 85. Tabel Hasil <i>Confusion Matrix</i> Skenario 3 Pengujian 6.....	87
Tabel 86. Tabel Hasil Perhitungan Precision dan Recall Skenario 3 Pengujian 6	87
Tabel 87. Tabel Hasil Perhitungan F1 Score Skenario 3 Pengujian 6	87
Tabel 88. Tabel Hasil <i>Confusion Matrix</i> Skenario 3 Pengujian 7.....	88
Tabel 89. Tabel Hasil Perhitungan Precision dan Recall Skenario 3 Pengujian 7	88
Tabel 90. Tabel Hasil Perhitungan F1 Score Skenario 3 Pengujian 7	88
Tabel 91. Tabel Hasil <i>Confusion Matrix</i> Skenario 3 Pengujian 8.....	88
Tabel 92. Tabel Hasil Perhitungan Precision dan Recall Skenario 3 Pengujian 8	89
Tabel 93. Tabel Hasil Perhitungan F1 Score Skenario 3 Pengujian 5	89
Tabel 94. Tabel Hasil <i>Confusion Matrix</i> Skenario 3 Pengujian 9.....	89
Tabel 95. Tabel Hasil Perhitungan Precision dan Recall Skenario 3 Pengujian 9	90
Tabel 96. Tabel Hasil Perhitungan F1 Score Skenario 3 Pengujian 9	90
Tabel 97. Tabel Hasil <i>Confusion Matrix</i> Skenario 3 Pengujian 10.....	90
Tabel 98. Tabel Hasil Perhitungan Precision dan Recall Skenario 3 Pengujian 10	91
.....	91
Tabel 99. Tabel Hasil Perhitungan F1 Score Skenario 3 Pengujian 10	91
Tabel 100. Tabel Pengujian Skenario 4	91
Tabel 101. Tabel Hasil <i>Confusion Matrix</i> Skenario 4 Pengujian 1.....	96
Tabel 102. Tabel Hasil Perhitungan Precision dan Recall Skenario 4 Pengujian 1	96
.....	96
Tabel 103. Tabel Hasil Perhitungan F1 Score Skenario 4 Pengujian 1	97
Tabel 104. Tabel Hasil <i>Confusion Matrix</i> Skenario 4 Pengujian 2.....	97
Tabel 105. Tabel Hasil Perhitungan Precision dan Recall Skenario 4 Pengujian 2	97
.....	97
Tabel 106. Tabel Hasil Perhitungan F1 Score Skenario 4 Pengujian 2	97
7. Tabel Hasil <i>Confusion Matrix</i> Skenario 4 Pengujian 3.....	98



Tabel 108. Tabel Hasil Perhitungan Precision dan Recall Skenario 4 Pengujian 3	98
Tabel 109. Tabel Hasil Perhitungan F1 Score Skenario 4 Pengujian 3	98
Tabel 110. Tabel Hasil <i>Confusion Matrix</i> Skenario 4 Pengujian 4.....	98
Tabel 111. Tabel Hasil Perhitungan Precision dan Recall Skenario 4 Pengujian 4	99
Tabel 112. Tabel Hasil Perhitungan F1 Score Skenario 4 Pengujian 4	99
Tabel 113. Tabel Hasil <i>Confusion Matrix</i> Skenario 4 Pengujian 5.....	99
Tabel 114. Tabel Hasil Perhitungan Precision dan Recall Skenario 4 Pengujian 5	100
Tabel 115. Tabel Hasil Perhitungan F1 Score Skenario 4 Pengujian 5	100
Tabel 116. Tabel Hasil <i>Confusion Matrix</i> Skenario 4 Pengujian 6.....	100
Tabel 117. Tabel Hasil Perhitungan Precision dan Recall Skenario 4 Pengujian 6	101
Tabel 118. Tabel Hasil Perhitungan F1 Score Skenario 4 Pengujian 6	101
Tabel 119. Tabel Hasil <i>Confusion Matrix</i> Skenario 4 Pengujian 7.....	101
Tabel 120. Tabel Hasil Perhitungan Precision dan Recall Skenario 4 Pengujian 7	101
Tabel 121. Tabel Hasil Perhitungan F1 Score Skenario 4 Pengujian 7	102
Tabel 122. Tabel Hasil <i>Confusion Matrix</i> Skenario 4 Pengujian 8.....	102
Tabel 123. Tabel Hasil Perhitungan Precision dan Recall Skenario 4 Pengujian 8	102
Tabel 124. Tabel Hasil Perhitungan F1 Score Skenario 4 Pengujian 5	103
Tabel 125. Tabel Hasil <i>Confusion Matrix</i> Skenario 4 Pengujian 9.....	103
Tabel 126. Tabel Hasil Perhitungan Precision dan Recall Skenario 4 Pengujian 9	103
Tabel 127. Tabel Hasil Perhitungan F1 Score Skenario 4 Pengujian 9	103
Tabel 128. Tabel Hasil <i>Confusion Matrix</i> Skenario 4 Pengujian 10.....	104
Tabel 129. Tabel Hasil Perhitungan Precision dan Recall Skenario 4 Pengujian 10	104
Tabel 130. Tabel Hasil Perhitungan F1 Score Skenario 4 Pengujian 10	104
Tabel 131. Tabel Rekapitulasi F1 Score Skenario 1 Hingga Skenario 4.....	105
Tabel 132. Evaluasi Kinerja Model Pada Sistem POS	108



DAFTAR GAMBAR

Gambar 1. Antrian Supermarket	1
Gambar 2. Hubungan AI, ML dan DL	6
Gambar 3. Cara Kerja ML dan DL (Elgendy, 2020).	7
Gambar 4. <i>Subfield in Computer Vision</i>	8
Gambar 5. Arsitektur CNN (Elgendy, 2020)	12
Gambar 6. Segmentasi Instance Mask R-CNN (He et al., 2017).....	12
Gambar 7. Arsitektur FCN (Shelhamer, Long dan Darrell, 2017).....	13
Gambar 8. Arsitektur Umum Mask R-CNN	13
Gambar 9. Arsitektur ResNet-50 (Mahmood et al., 2020).....	14
Gambar 10. Arsitektur ResNet-101 (Chen et al., 2021).....	15
Gambar 11. Perbedaan ResNet50 dan Resnet101	15
Gambar 12. <i>User Interface</i> IDE Jupyter Notebook	18
Gambar 13. Kerangka Pikir.....	26
Gambar 14. <i>Flowchart</i> Tahapan Penelitian	26
Gambar 15. Skenario Pengambilan Gambar	30
Gambar 16. Desain Alur Sistem.....	31
Gambar 17. Anotasi Gambar Menggunakan 2 Teknik Anotasi.....	32
Gambar 18. Hasil Proses Augmentasi.....	33
Gambar 19. Pseudocode Proses <i>Feature Extraction</i>	35
Gambar 20. Nilai <i>Pixel 3 Channel</i> Gambar Inputan	36
Gambar 21. Ilustrasi Proses <i>Convolution</i>	37
Gambar 22. Ilustrasi <i>Feature Map</i>	37
Gambar 23. Ilustrasi Proses <i>Max Pooling</i>	38
Gambar 24. Pseudocode Proses <i>Region Proposal Network</i>	39
Gambar 25. Ilustrasi Proses <i>Region Proposal Network</i>	40
Gambar 26. Pseudocode Proses <i>Fully Connected Layers</i>	41
Gambar 27. Ilustrasi Proses <i>FC Layers</i>	42
Gambar 28. Pseudocode Proses Validation.....	44
Gambar 29. Pseudocode Proses <i>Testing</i>	44
Gambar 30. Halaman Hasil Prediksi	107



BAB I PENDAHULUAN

1.1. Latar Belakang Masalah

Pada saat ini, fenomena antrian telah menjadi bagian integral dari aktivitas kehidupan sehari-hari, salah satunya adalah antrian pada kasir supermarket. Berbelanja di supermarket telah menjadi rutinitas bagi sebagian besar individu, dikarenakan kemudahan dalam menemukan beragam kebutuhan di satu lokasi. Meskipun berbelanja melalui platform *e-commerce* dapat menjadi alternatif solusi untuk mengatasi permasalahan antrian, namun kecenderungan konsumen untuk tetap memilih berbelanja di supermarket tidak dapat diabaikan. Salah satu alasan utamanya adalah ketidaksesuaian antara ekspektasi dan realitas produk yang diterima saat berbelanja secara *online*.



Gambar 1. Antrian Supermarket

Kendala antrian di supermarket salah satunya adalah disebabkan oleh proses pemindaian *barcode* yang dilakukan satu per satu *barcode scanner* pada kasir.



u, masalah lainnya adalah ketidakmampuan *barcode scanner* dalam li *barcode* pada kemasan barang yang sudah rusak atau pudar, sehingga akan entri manual oleh petugas kasir, dan cara pembacaan *barcode* pada

barang yang secara statis artinya masih harus mengarahkan barcode ke arah *barcode scanner*. Hal ini berdampak pada lamanya durasi transaksi. Untuk mengatasi permasalahan ini, penggunaan aplikasi berbasis teknologi pengolahan citra untuk pengenalan atau deteksi barang dapat menjadi solusi yang efisien, karena mampu memindai beberapa barang secara simultan dan melakukan deteksi dengan kecepatan tinggi. Namun, sebuah tantangan yang perlu diatasi dalam penelitian ini adalah implementasi algoritma yang efisien untuk mengenali barang-barang yang posisinya saling tumpang tindih.

Sebelumnya telah dilakukan beberapa penelitian tentang deteksi objek oleh beberapa peneliti dibidang *computer vision*. Scale Invariant Feature Transform (SIFT) untuk mendeteksi produk pada keranjang belanja. Penelitian tersebut mencapai tingkat akurasi rata-rata sebesar 48,89%, dengan tingkat akurasi tertinggi mencapai 100% untuk jumlah barang pada keranjang belanja sebanyak 2-4 barang pada jarak dekat sekitar 30-60 cm (Akbar et.al, 2018). Penelitian dengan judul *Object Detection in Shelf Images with YOLO* menggunakan algoritma YOLO V2 untuk mengenali produk rokok merek Marlboro, Kent, Camel, Parliament, Pall Mall, Monte Carlo, Winston, Lucky Strike, 2001, dan Lark pada rak supermarket. Hasil penelitian ini menunjukkan penurunan nilai loss sekitar 50% pada iterasi ke-900 (Melek, 2019). Penelitian dengan judul *Product Stock Management Using Computer Vision* menggunakan algoritma YOLO V2 untuk memantau ketersediaan produk pada rak supermarket, dan mencapai tingkat akurasi 97,61% pada hasil deteksi produk yang ditempatkan dengan benar, serta 76,67% pada hasil deteksi kesalahan penempatan produk (Majdi, 2020). Penelitian dengan judul *Optical Character Recognition for Autonomous Stores* penelitian menggunakan *algoritma Single Shoot Detector dan Bidirectional Gated Recurrent Unit (BGRU)* untuk mengenali karakter pada kemasan produk. Penelitian ini mencapai akurasi pada proses deteksi dan transkripsi sebesar 61,9% (Giorgieva, 2020). Penelitian dengan judul *Systematic Investigation on end-to-end Deep Recognition of Grocery in the Wild* menggunakan algoritma *Convolutional Neural Network* untuk mendeteksi produk pada rak supermarket dan mencapai akurasi



sebesar 95%, meningkat sebesar 15% dari penelitian sebelumnya dengan menggunakan dataset yang sama (Leo et,al, 2021).

Beberapa penelitian yang telah dilakukan memiliki kekurangan. Misalnya, pada penelitian yang dilakukan oleh Akbar, Melek, Majdi, dan Leo proses pendeteksian objek hanya bersifat statis, sehingga hanya dapat mendeteksi objek dalam satu posisi tertentu. Selanjutnya, pada penelitian yang dilakukan oleh Giorgieva, kekurangan terletak pada proses pendeteksian objek melalui pengenalan karakter pada kemasan produk, yang kurang efektif jika produk tersembunyi di dalam kemasan atau tertutup. Kontribusi dari penelitian ini adalah pada proses deteksi objek snack yang dapat mengenali objek dalam berbagai posisi dan bahkan dalam posisi yang bertumpuk.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah disampaikan sebelumnya, maka rumusan permasalahan pada penelitian ini yaitu Bagaimana membangun model untuk melakukan klasifikasi dan multideteksi untuk mengenali objek yang tumpang tindih?

1.3. Tujuan Penelitian

Adapun tujuan yang akan dicapai pada penelitian ini yaitu dapat membangun model untuk melakukan klasifikasi dan multideteksi untuk mengenali objek yang tumpang tindih.

1.4. Manfaat Penelitian

1. Membantu mempercepat transaksi pada kasir supermarket.
2. Menjadi referensi ilmiah bagi pada mahasiswa atau peneliti pada bidang penelitian serupa.



1.5. Batasan Masalah

1. Sistem ini tidak dapat mengenali objek dalam keadaan intensitas cahaya yang kurang.
2. Objek yang dideteksi tidak tertutup sepenuhnya.
3. Objek yang dideteksi adalah makanan ringan (*snack*)



BAB II

LANDASAN TEORI

2.1 Landasan Teori

2.1.1 Pengolahan Citra (Image Processing)

Pengolahan citra atau Image Processing adalah suatu sistem dimana proses dilakukan dengan masukan (input) berupa citra (image) dan hasilnya (output) juga berupacitra (image). Pada awalnya pengolahan citra ini dilakukan untuk memperbaiki kualitas citra, namun dengan berkembangnya dunia komputasi yang ditandai dengan semakin meningkatnya kapasitas dan kecepatan proses komputer, serta munculnya ilmu komputer yang memungkinkan manusia dapat mengambil informasi dari suatu citra maka image processing tidak dapat dilepaskan dengan bidang computer vision.

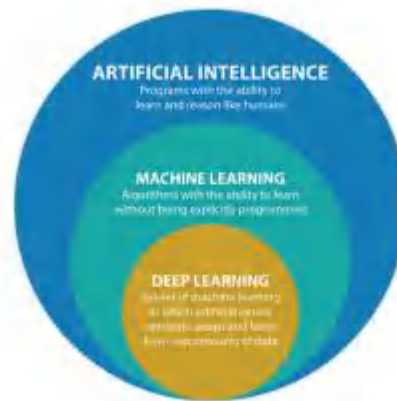
Umumnya citra digital berbentuk persegi panjang atau bujur sangkar (pada beberapa sistem pencitraan ada pula yang berbentuk segienam) yang memiliki lebar dan tinggi tertentu. Ukuran ini biasanya dinyatakan dalam banyaknya titik atau piksel sehingga ukuran citra selalu bernilai bulat. Setiap titik memiliki koordinat sesuai posisinya dalam citra. Koordinat ini biasanya dinyatakan dalam bilangan bulat positif, yang dapat dimulai dari 0 atau 1 tergantung pada sistem yang digunakan. Setiap titik juga memiliki nilai berupa angka digital yang merepresentasikan informasi yang diwakili oleh titik tersebut. Umumnya citra digital berbentuk persegi panjang atau bujur sangkar (pada beberapa sistem pencitraan ada pula yang berbentuk segienam) yang memiliki lebar dan tinggi tertentu. Ukuran ini biasanya dinyatakan dalam banyaknya titik atau piksel sehingga ukuran citra selalu bernilai bulat. Setiap titik memiliki koordinat sesuai posisinya dalam citra.



rtificial Intelligence

ificial Intelligence (AI), atau dalam bahasa Indonesia dikenal dengan kecerdasan Buatan adalah cabang ilmu komputer yang berfokus pada

pembuatan dan penerapan perangkat lunak dan algoritma yang dapat melakukan tugas-tugas yang biasanya memerlukan intervensi dan kecerdasan manusia. AI adalah topik yang mencakup bidang ilmu *Machine Learning* (ML) dan *Deep Learning* (DL). Berikut adalah skema hubungan antara AI ML dan DL dapat dilihat pada gambar 2.



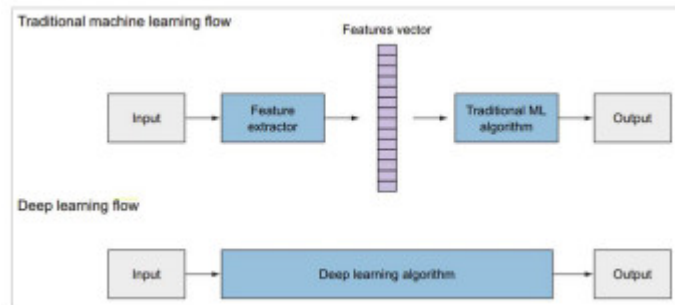
Gambar 2. Hubungan AI, ML dan DL

Machine Learning (ML) adalah bagian dari AI yang menggunakan metode statistik. Dengan metode tersebut memungkinkan mesin (komputer) untuk mengambil keputusan berdasarkan data. Algoritma machine learning dirancang agar dapat belajar dan kemampuannya meningkat seiring waktu ketika terdapat data baru, tanpa diprogram secara eksplisit (Seema Singh, 2018) (Priyanto Hidayatullah, 2021). Contoh penerapan machine learning adalah sebuah sistem yang dapat mendeteksi penyakit jantung berdasarkan rekaman elektrokardiogram (EKG).

Deep Learning (DL) adalah jenis dari machine learning yang terinspirasi oleh fungsi sel otak manusia yang disebut neuron. Dari inspirasi itulah terdapat konsep yang dinamakan *Neural network* (NN). Namun pada *Deep learning neural network*-nya lebih dari dua lapisan. Selain itu terdapat beberapa hal yang menjadi evolusi *neural network* pada *deep learning* lebih banyak neuron dari jaringan sebelumnya, cara yang lebih kompleks untuk menghubungkan lapisan/neuron, dan ekstraksi fitur otomatis (Vinet dan Zhedanov, 2011). Belum ada panduan yang betul-betul solid untuk menentukan jumlah hidden layer pada sebuah arsitektur model *deep learning* (Priyanto Hidayatullah, 2021). Terdapat perbedaan antara algoritma *machine learning* tradisional dengan algoritma *deep learning* pada *computer vision*. Pada



algoritma *machine learning* tradisional dilakukan ekstraksi fitur yang menghasilkan vektor fitur, setelah itu diklasifikasikan menggunakan algoritma machine learning. Sedangkan pada *deep learning*, ekstraksi fitur dan klasifikasi dilakukan oleh *neural network*. Ilustrasi cara kerja DL dan ML dapat dilihat pada Gambar 3.



Gambar 3. Cara Kerja ML dan DL (Elgendy, 2020).

2.1.3. Augmentasi Data

Augmentasi (augmentation) dalam konteks pengolahan citra merujuk pada teknik atau proses untuk membuat variasi atau modifikasi pada dataset gambar yang ada, Tujuannya adalah untuk meningkatkan jumlah dan keanekaragaman data pelatihan yang tersedia, sehingga memperkaya variasi pola dan fitur yang dapat dipelajari oleh model, Augmentasi digunakan untuk mengatasi masalah kurangnya data pelatihan yang mencukupi atau ketidakseimbangan dalam distribusi kelas (Sanjaya and Ayub, 2020).

Dengan melakukan augmentasi pada dataset gambar, dapat memperluas variasi kondisi pencahayaan, rotasi, pergeseran, zoom, refleksi, dan transformasi lainnya yang mungkin terjadi pada gambar dalam situasi dunia nyata, Hal ini membantu model untuk belajar secara lebih umum dan meningkatkan kemampuannya dalam mengenali objek cengkih dan puing (ranting dan daun) atau pola dalam berbagai kondisi (Sanjaya and Ayub, 2020).

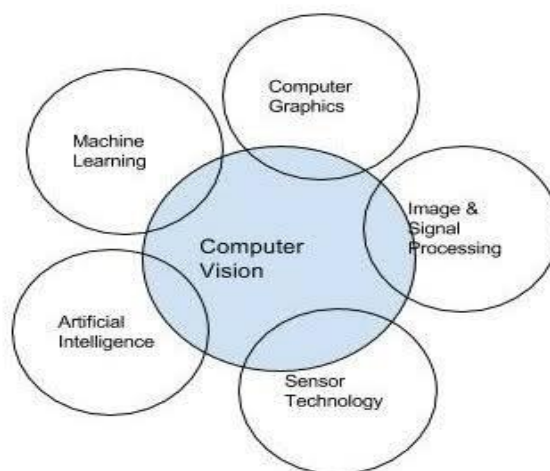


Computer Vision

Computer Vision adalah ilmu yang fokus membahas tentang kapabilitas untuk menerima, memproses, menganalisis, dan menyimpulkan masukan

berupa citra. Tujuan dari ilmu ini ialah menjadikan mesin agar bisa membuat keputusan rasional dan mengambil tindakan atau setidaknya membentuk kesimpulan berdasarkan masukan data visual yang diterima. Ballard dan Brown (1982) berpendapat bahwa vision komputer adalah ilmu yang membahas tentang konstruksi deskripsi eksplisit yang memiliki makna terkait dengan objek visual dari dunia nyata.

Sedangkan, Trucco dan Verri (1998) menyimpulkan bahwa bahasan dari vision komputer adalah tentang proses komputasi property visual tiga dimensi pada dunia nyata. Sebagai cabang ilmu, vision komputer tidak berdiri sendiri. Ada disiplin ilmu lain yang juga menggali hal beririsan dengan vision komputer, yaitu pengolahan citra dan pengenalan pola. Membuat komputer bisa melihat, memahami, dan memberikan kesimpulan yang tepat bukanlah hal yang mudah. Hal tersebut karena komputer melihat citra dengan cara yang berbeda dengan manusia. Seindah apapun citra yang di-input, komputer hanya melihat citra tersebut sebagai tumpukan kode-kode biner yang tersekat-sekat dalam matriks dua dimensi. Jangankan untuk menceritakan kondisi keseluruhan citra, untuk memahami bagian kecilnya saja sudah merupakan pekerjaan yang sangat sulit.



Gambar 4. *Subfield in Computer Vision*



Jika seorang guru melihat siswanya menengok ke belakang saat ujian, guru tersebut dapat dengan mudah mengatakan bahwa siswanya nakal dan siswa tersebut dapat langsung diberi hukuman. Namun, jika kita

ingin menyerahkan tugas pengawasan itu ke komputer, prosesnya tidak akan semudah itu. Citra yang didapat dari kamera harus diekstraksi dan ditransformasi sedemikian rupa sehingga bisa dihasilkan kesimpulan “siswa mencontek” atau kesimpulan lainnya. Hal tersebut dipersulit lagi dengan kondisi citra yang hanya terbatas pada ruang dua dimensi. Kondisi tersebut tentunya mempersulit penggalian informasi dari objek tertentu dalam citra tersebut, karena, objek tersebut sebenarnya berada pada lingkup tiga dimensi. Sedangkan, untuk mengubah citra menjadi berdimensi tiga bukanlah hal yang mudah.

2.1.5. *Artificial Neural Network*

Artificial Neural Network (ANN) atau Jaringan Syaraf Tiruan (JST) adalah sebuah model matematis yang terinspirasi oleh cara kerja sistem saraf biologi, terutama pada manusia dan hewan. Jaringan syaraf tiruan terdiri dari sejumlah besar elemen pemrosesan (neuron) yang terhubung dalam satu atau beberapa lapisan. Neuron-neuron tersebut menerima input dan menghasilkan output berdasarkan koneksi dan bobotnya masing-masing, sehingga mampu melakukan tugas seperti klasifikasi, pengenalan pola dan prediksi.

JST sangat berguna dalam bidang-bidang seperti pengenalan suara dan gambar, pengenalan teks, dan penerjemahan bahasa alami atau NLP (Natural Language Processing). JST juga digunakan dalam industri untuk memprediksi penjualan, mengoptimalkan rantai pasokan, dan meningkatkan kualitas produksi. (Ian, 2016). Artificial Neural Network (ANN) merupakan set algoritma yang bekerja seperti jaringan saraf otak manusia, dimana neuron saling terhubung satu dengan lainnya, bekerja untuk memproses informasi. Tujuan utama dari ANN adalah menjadikan komputer memiliki kemampuan kognitif seperti otak manusia, memiliki kemampuan problem solving dan dapat melakukan proses pembelajaran. (Laksamana, 2022).

ANN adalah sekumpulan neuron yang terorganisir dalam lapisan-lapisan

diantaranya :

Input Layer : lapisan yang membawa data masuk kedalam sistem untuk kemudian diproses pada layer selanjutnya.



- Hidden Layer : lapisan antara input layer dan output layer, dimana artificial neuron yang memiliki sekumpulan input pembobot 'weight' dan prosedur untuk menghasilkan output neuron melalui activation function.
- Output Layer : lapisan terakhir dari neuron yang menghasilkan output sistem.

2.1.6. *Deep Learning*

Deep learning adalah suatu cabang dari machine learning yang menggunakan algoritma neural network yang sangat kompleks dan terdiri dari banyak lapisan (deep). Tujuan dari deep learning adalah untuk mempelajari suatu representasi fitur dari data dengan menggunakan beberapa lapisan pemrosesan data secara bertahap. Dalam deep learning, setiap lapisan neural network terdiri dari sejumlah besar unit pengolahan sinyal (neuron) yang terhubung satu sama lain dalam pola tertentu.

Secara umum, deep learning digunakan untuk memecahkan masalah yang sangat kompleks dan memerlukan representasi data yang sangat abstrak, seperti pengenalan gambar dan suara, pengenalan bahasa alami, dan tugas-tugas lain yang terkait dengan data kompleks. Misalnya, dalam pengenalan gambar, deep learning dapat digunakan untuk memperbaiki kinerja pengenalan gambar dengan cara memperkenalkan banyak lapisan pemrosesan gambar yang semakin kompleks dan mampu mengenali fitur yang semakin abstrak. Sebagai hasilnya, deep learning mampu memecahkan masalah yang sebelumnya dianggap tidak mungkin dipecahkan oleh mesin. (Ian, 2016).

Deep learning juga memiliki beberapa keunggulan lain, seperti kemampuan untuk mengenali pola yang sangat kompleks dan abstrak, kemampuan untuk mengekstraksi fitur yang relevan secara otomatis, dan kemampuan untuk memperbaiki performa dengan data yang lebih banyak. Namun, deep learning juga memiliki beberapa kelemahan, seperti kebutuhan akan data yang sangat besar dan kemampuan komputasi yang sangat tinggi. (Yan, 2015).



Beberapa contoh aplikasi deep learning yang populer saat ini adalah pengenalan wajah seperti pada sistem keamanan atau pembuka kunci ponsel, klasifikasi objek seperti pada sistem keamanan atau kendaraan otonom, prediksi

harga saham dengan menggunakan data pasar yang akan dikenali pola-pola harga sahamnya dan memberikan prediksi yang akurat, pemrosesan bahasa alami seperti asisten virtual atau sistem terjemahan otomatis yang dilatih dengan menggunakan data teks, pengolahan citra medis, pengenalan suara, dan pembelajaran penguasaan permainan seperti permainan Go. Aplikasi lain dari deep learning meliputi bidang kesehatan, keuangan, dan teknologi informasi.

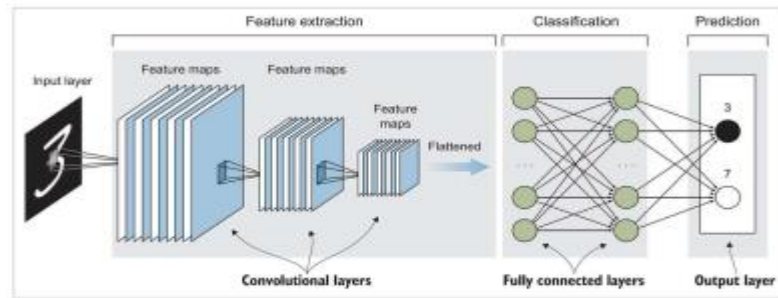
2.1.7. *Convolutional Neural Network*

Convolutional Neural Networks (CNN) merupakan bagian dari deep learning dan merupakan salah satu jenis arsitektur jaringan saraf yang secara luas diaplikasikan untuk tugas-tugas deteksi dan pengenalan gambar. Keunikan CNN terletak pada kemampuannya untuk mengidentifikasi dependensi spasial dan temporal dalam gambar, sehingga memungkinkan pemahaman yang lebih mendalam tentang konten gambar. Seperti halnya dengan *Artificial Neural Networks* (ANN), CNN juga terdiri dari beberapa komponen kunci seperti input, bobot (weight), bias, dan fungsi aktivasi. Namun, struktur CNN menawarkan fitur unik dalam bentuk arsitektur yang secara khusus dirancang untuk menangani data berdimensi tinggi, seperti gambar.

Arsitektur CNN umumnya dapat dibagi menjadi dua bagian utama: ekstraksi fitur dan klasifikasi. Bagian ekstraksi fitur terdiri dari lapisan konvolusional dan pooling yang bertujuan untuk mengidentifikasi dan memahami fitur penting dalam data. Setelah fitur-fitur ini diekstraksi, mereka kemudian diteruskan ke bagian klasifikasi, yang biasanya terdiri dari lapisan penuh (fully connected layers), untuk pengambilan keputusan. Secara keseluruhan, CNN menyediakan kerangka kerja yang kuat dan fleksibel untuk mengolah data berdimensi tinggi, dengan fitur-fitur uniknya yang memungkinkan pemahaman yang lebih baik tentang gambar dan memfasilitasi aplikasi lanjutan dalam bidang pengenalan pola dan analisis gambar.

Skema arsitektur CNN dapat dilihat pada gambar 5.





Gambar 5. Arsitektur CNN (Elgendy, 2020)

2.1.8. Mask R-CNN

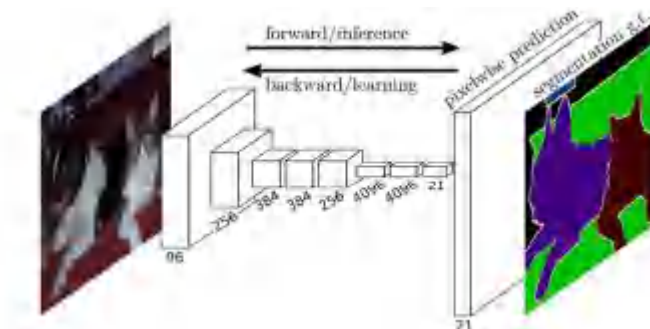
Mask R-CNN adalah perkembangan dari Faster R-CNN (algoritma deteksi objek dengan bounding box) Mask R-CNN ditujukan untuk menyelesaikan masalah segmentasi instan dalam deep learning atau visi komputer. Dengan kata lain, dapat memisahkan berbagai objek dalam gambar atau video (He et al., 2017). Gambar 6 merupakan hasil segmentasi instans dari algoritma Mask R-CNN.



Gambar 6. Segmentasi Instance Mask R-CNN (He et al., 2017)

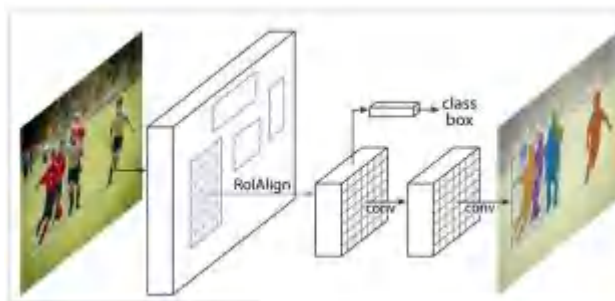
Mask-RCNN dapat dikatakan ekstensi dari Faster R-CNN dengan menambah satu cabang baru yaitu fully convolutional network (FCN) yang dimodifikasi, FCN digunakan sebagai segmentasi instan, atau dengan kata lain, untuk menghasilkan masker pixel untuk setiap objek yang terdeteksi dalam gambar. arsitektur original dari FCN dapat dilihat pada Gambar 7.





Gambar 7. Arsitektur FCN (Shelhamer, Long dan Darrell, 2017)

Dengan menambahkan FCN pada Mask R-CNN, Mask R-CNN tidak hanya dapat mengidentifikasi dan melokalisasi objek (Faster R-CNN), tetapi juga dapat menghasilkan segmen yang tepat dari objek tersebut, memungkinkan kita untuk membedakan antara instan objek yang berbeda dan memahami bentuk dan struktur objek dengan lebih baik. Arsitektur Mask R-CNN dapat dilihat pada Gambar 8.



Gambar 8. Arsitektur Umum Mask R-CNN

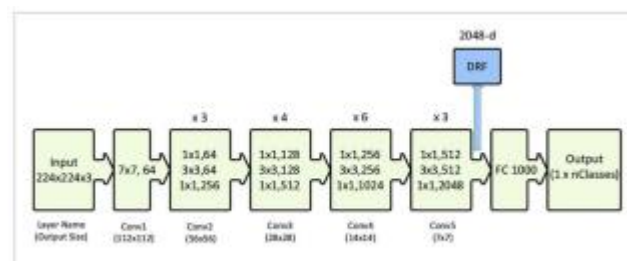
2.1.9. ResNet

ResNet atau Residual Networks adalah arsitektur Convolutional Neural Network (CNN) yang dikembangkan oleh Microsoft Research Asia. Salah satu masalah utama dalam penggunaan CNN adalah gradient vanishing dan exploding, yaitu ketika nilai gradien menjadi sangat kecil atau besar selama proses training, sehingga menyebabkan model tidak bisa belajar dengan baik. Untuk mengatasi masalah tersebut, ResNet memperkenalkan blok building baru bernama "residual" di mana input layer ditambahkan langsung ke output layer melalui shortcut connection. Hal ini memastikan bahwa informasi terjaga dan aliran gradien yang lancar selama proses training (He et al., 2016).



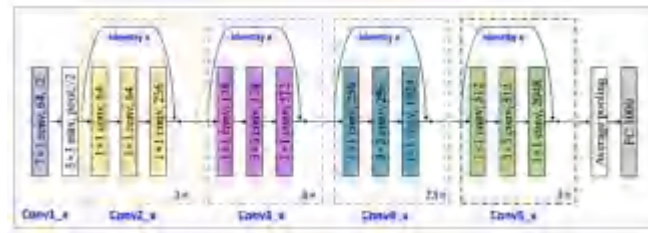
ResNet memiliki beberapa varian berdasarkan jumlah lapisannya, seperti ResNet18, ResNet-34, ResNet-50, ResNet-101 dan sebagainya. Semakin banyak lapisan pada arsitektur CNN biasanya akan memberikan kemampuan representasi fitur yang lebih kompleks namun juga dapat menimbulkan masalah seperti overfitting pada dataset training yang relatif kecil. ResNet-50 merupakan salah satu varian dari ResNet dimana jumlah layer-nya 50 layer. Model ini cukup populer di kalangan peneliti computer vision karena telah membuktikan diri mampu melakukan deteksi objek secara akurat serta efisien dalam waktu inferensi dibandingkan dengan menggunakan model lainnya. Sementara itu, ResNet101 memiliki jumlah layer yang lebih banyak daripada ResNet50 yakni mencapai 101 layers tetapi performa deteksinya tidak menjamin jauh lebih baik dibandingkan dengan menggunakan model ResNet50 namun tentunya akan memakan waktu komputasi inferensi yang sedikit lebih lama.

Kedua varian ini, baik ResNet50 maupun ResNet101, dapat digunakan sebagai backbone pada Mask R-CNN untuk ekstraksi fitur gambar secara efisien dan pengenalan objek pada gambar secara akurat. Keduanya juga memiliki kemampuan transfer learning yang baik sehingga model dapat belajar dari dataset besar sebelumnya dan meningkatkan performa deteksi objek pada Mask R-CNN. Berikut gambar arsitektur ResNet-50 dan ResNet-101 dapat dilihat pada Gambar 9 dan 10.



Gambar 9. Arsitektur ResNet-50 (Mahmood et al., 2020)





Gambar 10. Arsitektur ResNet-101 (Chen et al., 2021)

Perbedaan antara Layer ResNet50 dan ResNet101 untuk ekstraksi fitur dapat dilihat pada Gambar 11 (Malhotra et al., 2022).

Layer Name	ResNet50	Resnet101
Convolution 1	$7 \times 7, 64, \text{stride } 2$	
	$3 \times 3 \text{ Max pool, stride } 2$	
Convolution 2x	$\begin{bmatrix} 1 \times 1 & 64 \\ 2 \times 3 & 64 \\ 1 \times 1 & 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1 & 64 \\ 2 \times 3 & 64 \\ 1 \times 1 & 256 \end{bmatrix} \times 3$
Convolution 3x	$\begin{bmatrix} 1 \times 1 & 128 \\ 3 \times 3 & 128 \\ 1 \times 1 & 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1 & 128 \\ 3 \times 3 & 128 \\ 1 \times 1 & 512 \end{bmatrix} \times 4$
Convolution 4x	$\begin{bmatrix} 1 \times 1 & 256 \\ 3 \times 3 & 256 \\ 1 \times 1 & 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 & 256 \\ 3 \times 3 & 256 \\ 1 \times 1 & 1024 \end{bmatrix} \times 23$
Convolution 5x	$\begin{bmatrix} 1 \times 1 & 512 \\ 3 \times 3 & 512 \\ 1 \times 1 & 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1 & 512 \\ 3 \times 3 & 512 \\ 1 \times 1 & 2048 \end{bmatrix} \times 3$
Average pool, 1000-d FC, softmax		

Gambar 11. Perbedaan ResNet50 dan Resnet101

Berikut penjelasan dari gambar 11 :

1. Convolution 1: Gambar input melewati konvolusi 7×7 dengan 64 filter dan stride 2, mengurangi dimensi spasial gambar menjadi setengah.
2. Convolution 2x / 3x Max pool: Output dari Convolution 1 melewati layer Max Pooling 3×3 dengan stride 2, mengurangi dimensi spasial menjadi setengah lagi.
3. Conv2_x (3x): Blok pertama ResNet, Conv2_x, memiliki tiga layer:

- Konvolusi 1×1 dengan 64 filter.
- Konvolusi 3×3 dengan 64 filter.
- Konvolusi 1×1 dengan 256 filter.

Proses ini diulangi 3 kali baik di ResNet-50 maupun ResNet-101



4. Conv3_x (4x): Blok kedua ResNet, Conv3_x, memiliki tiga layer:

- Konvolusi 1x1 dengan 128 filter.
- Konvolusi 3x3 dengan 128 filter.
- Konvolusi 1x1 dengan 512 filter.

Proses ini diulangi 4 kali baik di ResNet-50 maupun ResNet-101.

5. Conv4_x (6x for ResNet-50, 23x for ResNet-101): Blok ketiga ResNet, Conv4_x, memiliki tiga layer:

- Konvolusi 1x1 dengan 256 filter.
- Konvolusi 3x3 dengan 256 filter.
- Konvolusi 1x1 dengan 1024 filter.

Proses ini diulangi 6 kali di ResNet-50, dan 23 kali di ResNet-101.

6. Conv5_x (3x): Blok keempat dan terakhir ResNet, Conv5_x, memiliki tiga layer: Konvolusi 1x1 dengan 512 filter.

- Konvolusi 3x3 dengan 512 filter.
- Konvolusi 1x1 dengan 2048 filter.

Proses ini diulangi 3 kali baik di ResNet-50 maupun ResNet-101.

2.1.10. Tensorflow

Tensorflow merupakan open source framework yang dapat digunakan untuk mengembangkan, melatih, dan menggunakan model deteksi objek. Sistem ini sudah banyak diterapkan pada berbagai produk Google antara lain pencarian image, deteksi wajah, dan plat nomor kendaraan pada google street, view, google assistant, way mo atau self driving car, dan lain-lain.

Tensorflow bekerja dengan computational untuk membuat model machine learning. Tensorflow menyediakan berbagai toolkit yang memungkinkan anda membuat model pada tingkat yang lebih rendah untuk membuat model dengan menentukan serangkaian matematis. Sebagai alternatif, anda dapat menggunakan

an tingkat yang lebih rendah untuk membuat model dengan menentukan
gan tingkat yang lebih tinggi (seperti tf.estimator) untuk menentukan
yang telah ditetapkan, seperti regresi linear atau neural network.



Framework Tensorflow digunakan pada proses pembuatan sistem deteksi objek agar memudahkan implementasi algoritma dan penggunaan bahasa pemrograman, terdapat juga GPU untuk mempercepat proses training. Tensorflow sebagai kerangka machine learning yang dapat digunakan untuk mengolah banyak data atau ingin mempelajari kecerdasan buatan (artificial intelligence) secara mendalam.

Tensorflow bekerja dengan cara membangun sebuah graph komputasi yang terdiri dari serangkaian operasi matematika yang saling terkait. Setiap operasi dalam graph tersebut dapat menerima satu atau lebih tensor sebagai input dan menghasilkan satu atau lebih tensor sebagai output. Tensor sendiri adalah sebuah array multidimensi yang digunakan untuk merepresentasikan data dalam model machine learning.

Tensorflow memungkinkan pengguna untuk membangun model machine learning dengan mudah dan cepat dengan menggunakan API yang intuitif. Pengguna dapat memilih antara menggunakan API high-level seperti Keras atau API low-level yang lebih fleksibel untuk membangun model sesuai dengan kebutuhan. Selain itu, Tensorflow juga menyediakan tools untuk memvisualisasikan graph komputasi, melakukan debugging, dan melakukan distribusi komputasi di multiple GPUs atau mesin. (Martin, 2016).

Berikut ini adalah beberapa kelebihan dari Tensorflow :

1. Kompatibilitas Platform yang luas : tensorflow dapat digunakan pada berbagai platform seperti desktop, server, mobile, dan edge devices. Hal ini membuat tensorflow sangat fleksibel dan dapat diintegrasikan ke dalam berbagai aplikasi.
2. Kecepatan dan Skalabilitas : tensorflow dirancang untuk dapat memproses data yang sangat besar dan kompleks dengan kecepatan yang tinggi. Hal ini menjadikan tensorflow sebagai salah satu framework tercepat yang tersedia saat ini.



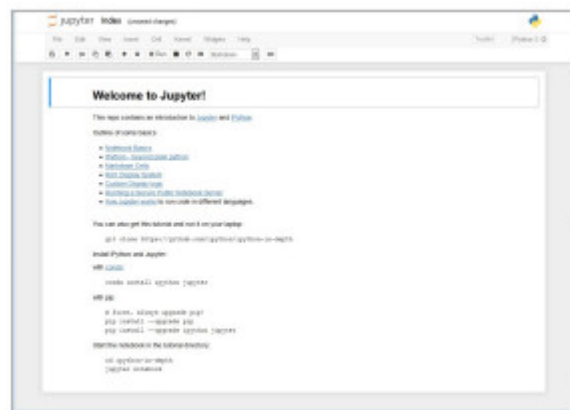
ngan komunitas yang luas : tensorflow memiliki komunitas yang besar aktif. Hal ini berarti bahwa ada banyak sumber daya dan dukungan yang

tersedia untuk para pengguna termasuk dokumentasi, tutorial, dan forum diskusi.

4. Memiliki banyak fitur : tensorflow menyediakan berbagai jenis layer dan model yang dapat digunakan untuk membangun berbagai jenis model machine learning, seperti model deep learning dan reinforcement learning. Tensorflow juga menyediakan berbagai algoritma optimasi yang dapat digunakan untuk meningkatkan performa model. (Martin, 2016).

2.1.11. Jupyter Notebook

Jupyter Notebook adalah sebuah IDE (Integrated Development Environment) yang digunakan untuk menulis, menjalankan, dan membagikan kode secara interaktif. IDE ini memungkinkan pengguna untuk menggabungkan teks, kode, visualisasi, dan dokumentasi dalam satu dokumen yang disebut sebagai notebook (Randles et al., 2017). Berikut dapat dilihat pada Gambar 23 menampilkan user interface dari Jupyter Notebook.



Gambar 12. User Interface IDE Jupyter Notebook

Jupyter Notebook merupakan aplikasi bawaan yang tersedia di platform Anaconda. Anaconda sendiri adalah sebuah platform yang bersifat open source yang digunakan untuk pengembangan dan pemrograman menggunakan bahasa



Optimized using
trial version
www.balesio.com

an R. Tujuan utama dari Anaconda adalah untuk menyediakan pengelolaan data yang mudah dan efisien di sistem pengguna (Anaconda Inc., 2021). Dengan menggunakan Anaconda, pengguna dapat menginstal dan mengelola paket Python dengan mudah, termasuk paket-paket yang sering digunakan

dalam pengembangan machine learning maupun deep learning. Anaconda menyediakan lingkungan virtual yang terisolasi yang memungkinkan pengguna untuk menginstal paket-paket Python yang spesifik dan menjaga kestabilan dan keseragaman lingkungan pengembangan mereka. Salah satu komponen utama dari Anaconda adalah Jupyter Notebook. Jupyter Notebook berfungsi sebagai antarmuka pengguna yang memungkinkan pengguna untuk membuat dan menjalankan kode Python secara interaktif melalui halaman web. Dalam Jupyter Notebook, pengguna dapat menulis dan menjalankan kode Python dalam sel-sel yang terpisah, serta menyertakan teks, visualisasi, dan dokumentasi dalam satu dokumen yang lengkap.

2.1.12. Evaluasi Sistem

Model Mask R-CNN (Region-based Convolutional Neural Networks) adalah model yang digunakan dalam tugas deteksi objek dan segmentasi instans dalam bidang visi komputer. Mask R-CNN adalah perkembangan dari model Faster R-CNN, dengan tambahan modul segmentasi masker yang memungkinkan model ini juga mendapatkan informasi piksel tingkat tinggi tentang bentuk objek yang terdeteksi.

a. Evaluasi Loss

Evaluasi model Mask R-CNN dapat dilihat dari nilai loss yang dicapai, semakin rendah nilai loss, semakin baik modelnya (Bhatti et al., 2020). Rumus loss function dapat ditulis sebagai berikut :

$$Loss = Lcls + Lbox + Lmask \quad (1)$$

Keterangan :

$Lcls$: Nilai loss dari hasil training berfokus pada klasifikasi yang diprediksi

$Lbox$: nilai loss regresi dari bounding boxe prediksi)

$Lmask$: nilai loss dari masker yang diprediksi



b. *Confusion Matrix*

Confusion Matrix adalah alat evaluasi visual yang digunakan dalam sistem klasifikasi. Confusion matrix ini berguna untuk mengukur seberapa baik model klasifikasi yang telah dibuat. *Confusion matrix* berukuran $n \times n$, dimana n merupakan jumlah kelas yang berbeda. *Confusion matrix* menentukan akurasi yang didapat dari nilai beberapa parameter, seperti *true positive* (TP), *false positive* (FP), *true negative* (TN), dan *false negative* (FN).

- a. TP adalah true positive, yaitu jumlah data positif yang terklasifikasi dengan benar oleh sistem.
- b. TN adalah true negative, yaitu jumlah data negative yang terklasifikasi dengan benar oleh sistem.
- c. FN adalah false negative, yaitu jumlah data negative namun terklasifikasi salah oleh sistem.
- d. FP adalah false positive, yaitu jumlah data positif namun terklasifikasi salah oleh sistem.

Berdasarkan nilai-nilai true positive (TP), false positive (FP), true negative (TN), dan false negative (FN), evaluasi klasifikasi dianalisis dari beberapa indikator yaitu diantaranya indikator, *precision*, *recall*, akurasi, dan *F1 score*. *Precision* merupakan merupakan rasio prediksi benar positif (TP) dibandingkan dengan keseluruhan hasil yang diprediksi positif. Berdasarkan *confusion matrix* yang digunakan rumus dari presisi pada penelitian ini dinyatakan pada persamaan berikut :

$$Precision = \left(\frac{TP}{TP + FP} \right) \quad (2)$$

Sedangkan recall atau rasio TP adalah ukuran untuk berapa banyak TP yang diprediksi dari semua positif dalam kumpulan data, kadang juga disebut kepekaan (*sensitivity*). Rumus recall pada penelitian ini dapat dituliskan seperti pada berikut :



$$Recall = \left(\frac{TP}{TP + FN} \right) \quad (3)$$

Akurasi merupakan rasio antara jumlah terprediksi benar dari semua data. Rumus akurasi pada penelitian ini dapat dituliskan seperti pada persamaan berikut :

$$Akurasi = \left(\frac{TP + FN}{TP + TN + FP + FN} \right) \quad (4)$$

F1 Score merupakan perbandingan rata-rata precision dan recall. Rumus F1 score dituliskan seperti pada persamaan berikut :

$$F1\ Score = \left(2 * \frac{Precision * Recall}{Precision + Recall} \right) \quad (5)$$

2.2. Metode Penyelesaian Masalah

2.2.2. State of The Art Penelitian

Tabel 1. Tabel *State of The Art*

No.	Judul Karya Ilmiah, Nama, Tahun Terbit dan Penerbit	Objek dan Permasalahan	Metode Penyelesaian	Kinerja
1.	Judul: Learning to Detect Partially Overlapping Instances Carlos Penulis : Arteta, et.al Tahun : 2013 Penerbit : IEEE	Objek: Citra orang Permasalahan: Bagaimana mendeteksi semua contoh kelas seperti orang pada sebuah citra	Menggunakan Yolo V4	Kinerja mampu mendeteksi kumpulan orang dalam kerumunan



No.	Judul Karya Ilmiah, Nama, Tahun Terbit dan Penerbit	Objek dan Permasalahan	Metode Penyelesaian	Kinerja
2.	Judul: Automatic Detection and Counting of Circular Shaped Overlapped Objects Using Circular Hough Transform and Contour Detection Penulis : Jianjun Ni, et.al Tahun : 2016 Penerbit : WCICA	Objek: Citra yang memiliki warna acak dari objek melingkar dan objek non lingkaran Permasalahan: Bagaimana menghitung objek lingkaran yang tumpang tindih	Menggunakan <i>Canny Edge, Circular Hough Transform</i>	Kinerja mampu mendeteksi semua objek yang berbentuk lingkaran seperti koin dan pena yang menempel pada koin
	Judul : Automatic Fruit Count On Coffee Branches Using Computer Vision Penulis : Ramos, et.al Tahun 2017 Penerbit : Elsevier	Objek : Biji Kopi Permasalahan : Bagaimana menghitung jumlah buah pada cabang kopi	Menggunakan Canny edge, naive bayes, K-nearest neighbor, Support vector machine	Kinerja Bayes : 83.57% KNN : 83.09% SVM : 88.02%
3.	Judul : Digital Image Processing Techniques for Object Detection From Complex Background Image Penulis : Husin, et.al	Objek : Buah Mangga Permasalahan : Bagaimana mendeteksi mangga pada pohon mangga	Menggunakan Circular hough transform (CHT)	Kinerja Accurate Object Detected (60%)



No.	Judul Karya Ilmiah, Nama, Tahun Terbit dan Penerbit	Objek dan Permasalahan	Metode Penyelesaian	Kinerja
4.	Judul : Automatic Dragon Fruit Counting Using Adaptive Thresholds For Image Segmentation And Shape Analysis Penulis : Tran, et.al [7] Tahun : 2017 Penerbit : NAFOSTED	Objek : Buah naga Permasalahan : Bagaimana melakukan perhitungan yang akurat jumlah buah naga di pohon	Menggunakan Metode Thresholding dan Otsu proses segmentasi	Kinerja mampu melakukan perhitungan buah naga dipohon
5.	Judul : Pengenalan Barang pada Kereta Belanja Menggunakan Metode Scale Invariant Feature Transform (SIFT) Penulis : Ronny, et.al Tahun : 2018 Penerbit : Jurnal Teknologi Informasi dan Komputer	Objek : Barang pada kereta belanja Permasalahan : Bagaimana mengambil citra barang di keranjang belanja untuk mengenali nama dan harga barang yang diambil dari database barang dengan menghitung total biaya daftar belanja	Menggunakan algoritma Scale Invariant Feature Transform	Kinerja Pada penelitian ini dapat mengenali barang dalam citra keranjang belanja dengan tingkat akurasi rata-rata 48,89% berdasarkan jumlah barang dan jarak pengambilan gambar. Tingkat akurasi 100% terhadap jumlah barang pada kereta belanja sebanyak 2-4 barang pada jarak dekat (30-60 cm), tingkat akurasi 46.67% dengan jumlah 5-7 barang pada jarak sedang (60-90 cm), tingkat akurasi 0% dengan jumlah 8-10 barang pada jarak jauh (lebih dari 90 cm). Penelitian ini hanya bisa mengenali



No.	Judul Karya Ilmiah, Nama, Tahun Terbit dan Penerbit	Objek dan Permasalahan	Metode Penyelesaian	Kinerja
				barang dalam bentuk objek datar
6.	Judul : A Novel Labelling Algorithm for Object Counting Penulis : Swaraj, et.al Tahun : 2012 Penerbit IEEE	Objek : Uang koin Permasalahan: Mengusulkan algoritma untuk menghitung uang koin yang tumpang tindih	Menggunakan algoritma Circular hough transform (CHT)	Kinerja Sistem yang dirancang mampu mengetahui dan menghitung uang koin yang tumpang tindih
7.	Judul : Image Segmentation of Multi-Shaped Overlapping Objects Penulis : Kumar, et.al Tahun : 2018 Penerbit : VISIGRAPP	Objek : Objek berbentuk cembung Permasalahan : Mengusulkan algoritma baru untuk mensegmentasi gambar yang berisi objek cembung dalam berbagai bentuk	Menggunakan Contour Region Detection (CRD), Remer Douglas Peucker (RDP),	Kinerja pada sistem yang dirancang berhasil mensegmentasi objek cembung dalam berbagai bentuk
8.	Judul : Separation Of Overlapping Object Segmentation Using Level Set With Automatic Initalization On Dental raph 	Objek : Dental panoramic radiograph Permasalahn : Bagaimana mensegmentasi citra gigi yang saling tumpang tindih	Menggunakan Region of Interest (ROI), Mendian Filtering, dan Contrast Limited Adequate Histogram Equalization(CLAH E), Hierarchical Cluster Analysis(HCA)	Kinerja HCA : 73% Missclassification Error (ME) : 16.41% Relative Foreground Area Error (RAE) : 52.14%

No.	Judul Karya Ilmiah, Nama, Tahun Terbit dan Penerbit	Objek dan Permasalahan	Metode Penyelesaian	Kinerja
	Tahun : 2020 Penerbit : Journal of Computer Science and Information			

2.1. Target Hasil Penelitian

Berdasarkan tabel dari *State of The Art*, target dari penelitian yang dilakukan adalah mengembangkan sebuah model deep learning yang dapat digunakan untuk melakukan klasifikasi dan multideteksi guna mengidentifikasi objek-objek yang tumpang tindih. Hasil dari penelitian ini diharapkan memiliki kontribusi akademis dengan potensi menjadi rujukan bagi peneliti-peneliti mendatang di dalam bidang ini.

2.2. Kerangka Pikir

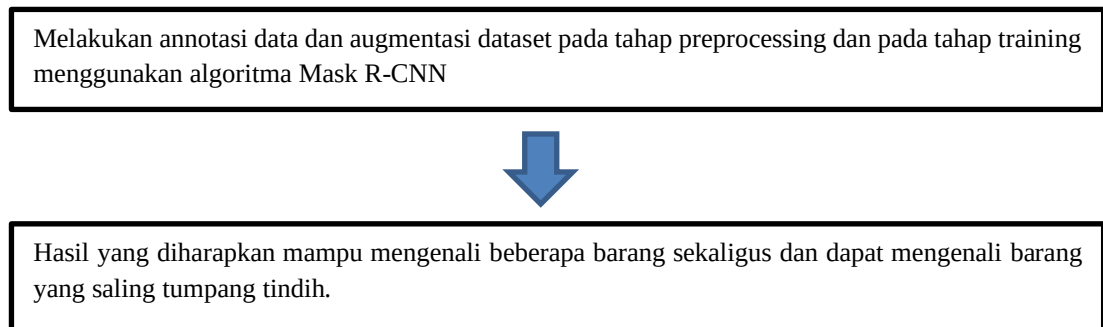
Kerangka pikir dapat dilihat pada gambar 13 yang menjelaskan mengenai alur penelitian yang akan dilakukan.

Antrian terjadi pada supermarket karena pada kasir mengharuskan setiap *barcode* barang dipindai satu per satu menggunakan *barcode reader* dan terkadang kemasan *barcode* barang sudah tidak dikenali lagi oleh *barcode reader* sehingga mengharuskan kasir untuk menginput manual kode *barcode* pada aplikasi kasir sehingga proses transaksi menjadi lama.



Hal ini dapat diatasi dengan menggunakan aplikasi pengenalan atau deteksi barang berbasis pengolahan citra yang dapat membantu mempercepat proses pemindaian pada kasir dengan i beberapa barang sekaligus serta mampu mendeteksi barang dengan cepat.





Gambar 13. Kerangka Pikir

