DAFTAR PUSTAKA

Agkland, B. D., & Weste, N. H. (1981). The edge flag algorithm — A fill method for raster scan displays. *IEEE Transactions on Computers*, *C–30*(1), 41–48. https://doi.org/10.1109/TC.1981.6312155

Alice Zheng. (2015). *Evaluating Machine Learning Models* (Shannon Cutt, Ed.). O'Reilly Media.

Amit, Y., Felzenszwalb, P., & Girshick, R. (2020). *Object Detection* (pp. 1–9). https://doi.org/10.1007/978-3-030-03243-2_660-1

Antonius Hintono. (2022). *Ilmu Pengetahuan Telur*. UNDIP Press Semarang.

Aragua, A., & Mabayo, V. İ. (2018). A cost-effective approach for chicken egg weight estimation through computer vision. *International Journal of Agriculture Environment and Food Sciences*, *2*(3), 82–87. https://doi.org/10.31015/jaefs.18013

Arnita, Faridawaty Marpaung, Fitrahuda Aulia, Nita Suryani, & Rinjani Cyra Nabila. (2022). *Computer Vision dan Pengolahan Citra Digital* (Ahmad Baharuddin Surya, Ed.). Pustaka Aksara.

Badan Pusat Statistik. (2022). *Peternakan Dalam Angka 2022*. Badan Pusat Statistik.

Basak, J. K., Paudel, B., Kim, N. E., Deb, N. C., Kaushalya Madhavi, B. G., & Kim, H. T. (2022). Non-Destructive Estimation of Fruit Weight of Strawberry Using Machine Learning Models. *Agronomy*, *12*(10), 2487. https://doi.org/10.3390/agronomy12102487

Bataineh, B. (2023). Brightness and Contrast Enhancement Method for Color Images Via Pairing Adaptive Gamma Correction and Histogram Equalization. *International Journal of Advanced Computer Science and Applications*, *14*. https://doi.org/10.14569/IJACSA.2023.0140314

Chantal D. Larose, & Daniel T. Larose. (2015). *Data Mining and Predictive Analytics* (2nd ed.). Wiley.

Chicco, D., Warrens, M., & Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ Computer Science*, *7*, e623. https://doi.org/10.7717/peerj-cs.623

Dhanar Intan Surya Saputra, Tu Bagus Pranata, & Sitaresmi Wahyu Handani. (2016). Prototype Aplikasi Pengolah CItra Invert Sebagai Media Pengolah Klise Foto. *Conference on Information Technology, Information System and Electrical Engineering* , 85–89.

Dong, Y., & Pan, W. (2023). A Survey on Compression Domain Image and Video Data Processing and Analysis Techniques. *Information*, *14*, 184. https://doi.org/10.3390/info14030184

Douglas C. Montgomery, Elizabeth A. Peck, & G. Geoffrey Vining. (2021). *Introduction To Linear Regression Analysis* (6th ed.). John Wiley & Sons, Inc.

F., I. K. U. M. S., DEWI, G. A. M. K., & WIRAPARTHA, M. (2021). Internal Organs of Isa Brown Chickens Aged 104 Weeks Fed with Calcium That Was Derived From Seashells in Their Ration. *Majalah Ilmiah Peternakan*, *24*(2), 96–101. https://doi.org/10.24843/MIP.2021.v24.i02.p08

Fisher, R. B., Breckon, T. P., Dawson-Howe, K., Fitzgibbon, A., Robertson, C., Trucco, E., & Williams, C. K. I. (2013). *Dictionary of Computer Vision and Image Processing* (2nd ed.). Wiley Publishing.

Fisher, R., Perkins, S., Walker, A., & Wolfart, E. (1996). Hypermedia Image Processing Reference (HIPR). *Artificial Intelligence - AI*.

Gibson, J. D., & Bovik, A. (2000). *Handbook of Image and Video Processing* (1st ed.). Academic Press, Inc.

González, R. C., & Woods, R. E. (2008). *Digital image processing, 3rd Edition*. Pearson Education. https://www.worldcat.org/oclc/241057034

Hendrix genetics. (2024). *Isa Brown*. https://layinghens.hendrix-genetics.com/en/market/indonesia/ISA/

Irianto, S. (2014). *PENGOLAHAN CITRA DIGITAL*.

Jain, A. K. (1989). *Fundamentals of digital image processing*. Prentice-Hall, Inc.

Jonathan Sarwono. (2018). *Metode Penelitian Kuantitatif dan Kualitatif* (2nd ed.). Suluh Media.

Jude Chukwura Obi. (2023). A comparative study of several classification metrics and their performances on data. *World Journal of Advanced Engineering Technology and Sciences*, *8*(1), 308–314. https://doi.org/10.30574/wjaets.2023.8.1.0054

Kang, B. (2007). A Review on Image & Video Processing. *International Journal of Multimedia and Ubiquitous Engineering*, *2*.

Krig, S. (2016). *Computer Vision Metrics*. Springer International Publishing. https://doi.org/10.1007/978-3-319-33762-3

Kuhn, M., & Johnson, K. (2013). *Applied Predictive Modeling*. Springer.

Kurtini, T., Nova, K., & Septinova, D. (2011). *Produksi Ternak Unggas*. Universitas Lampung.

Nabillah, I., & Ranggadara, I. (2020). Mean Absolute Percentage Error untuk Evaluasi Hasil Prediksi Komoditas Laut. *JOINS (Journal of Information System)*, *5*, 250–255. https://doi.org/10.33633/joins.v5i2.3900

Niu, Z., & Li, H. (2019). Research and analysis of threshold segmentation algorithms in image processing. *Journal of Physics: Conference Series*, *1237*(2), 22122. https://doi.org/10.1088/1742-6596/1237/2/022122

Nova, I., Kurtini, T., & Wanniatie, V. (2014). Pengaruh Lama Penyimpanan Terhadap Kualitas Internal Telur Ayam Ras Pada Fase Produksi Pertama. *Jurnal Ilmiah Peternakan Terpadu*, *2*(2). https://doi.org/10.23960/jipt.v2i2.478

Nurwantoro, & Sri Mulyani. (2003). *Dasar Teknologi Hasil Ternak*. Fakultas Peternakan Universitas Diponegoro.

Nyalala, I., Okinda, C., Nyalala, L., Makange, N., Chao, Q., Chao, L., Yousaf, K., & Chen, K. (2019). Tomato volume and mass estimation using computer vision and machine learning algorithms: Cherry tomato model. *Journal of Food Engineering*, *263*, 288–298. https://doi.org/https://doi.org/10.1016/j.jfoodeng.2019.07.012

OpenCV. (2024). *Miscellaneous Image Transformations*. https://docs.opencv.org/4.x/d7/d1b/group__imgproc__misc.html#gaf1f55a048f8a45bc3383586e80b1f0d0

Ortiz, J. H., Marroquin, W. G., & Cifuentes, L. Z. (2020). Industry 4.0: Current Status and Future Trends. In J. H. Ortiz (Ed.), *Industry 4.0*. IntechOpen. https://doi.org/10.5772/intechopen.90396

Padmavathi, K., & Thangadurai, K. (2016). Implementation of RGB and Grayscale Images in Plant Leaves Disease Detection – Comparative Study. *Indian*

Journal of Science and Technology, *9*. https://doi.org/10.17485/ijst/2016/v9i6/77739

Ranjay Khrisna. (2017). *Computer Vision: Foundations and Applications*. Stanford University.

Schober, P., Boer, C., & Schwarte, L. (2018). Correlation Coefficients: Appropriate Use and Interpretation. *Anesthesia & Analgesia*, *126*, 1. https://doi.org/10.1213/ANE.0000000000002864

Sholihin, M. (2019). Implementasi K-Nearest Neighbor Untuk Klasifikasi Grade Telur Ayam Ras Berdasarkan Ukuran. *Inform : Jurnal Ilmiah Bidang Teknologi Informasi Dan Komunikasi*, *4*. https://doi.org/10.25139/inform.v4i2.1760

Singh, N., & Bhandari, A. (2020). Image Contrast Enhancement with Brightness Preservation Using an Optimal Gamma and Logarithmic Approach. *IET Image Processing*, *14*. https://doi.org/10.1049/iet-ipr.2019.0921

SNI (Standar Nasional Indonesia). (2023). *SNI 3926:2023*. Badan Standar Nasional.

Sri Ratna Sulistiyanti, FX Arianto Setyawan, & Muhamad Komarudin. (2016). *Pengolahan Citra Dasar dan Contoh Penerapannya* (1st ed.). Teknosain.

Sudaryani, T. (2006). *Kualitas Telur*. Penebar Swadaya.

Szeliski, R. (2010). *Computer Vision: Algorithms and Applications* (1st ed.). Springer-Verlag.

Tay, S., Te Chuan, L., Aziati, A., & Ahmad, A. N. A. (2018). An Overview of Industry 4.0: Definition, Components, and Government Initiatives. *Journal of Advanced Research in Dynamical and Control Systems*, *10*, 14.

Tekalp, A., & Murat, A. (1995). Digital Video Processing. In *Journal of Computational and Applied Mathematics - J COMPUT APPL MATH* (Vol. 66, p. 526).

Vujovic, Ž. Đ. (2021). Classification Model Evaluation Metrics. *International Journal of Advanced Computer Science and Applications*, *12*(6). https://doi.org/10.14569/IJACSA.2021.0120670

weblab.deusto.es. (2023). *Parts of the Egg*. https://weblab.deusto.es/olarex/cd/UD/Incubator_EN_final/parts_of_the__egg.html

William K. Pratt. (2001). *Digital Image Processing* (third). John Wiley & Sons, Inc. https://doi.org/10.1002/0471221325.ch17

Yang, X., Bist, R. B., Subedi, S., & Chai, L. (2023). A Computer Vision-Based Automatic System for Egg Grading and Defect Detection. *Animals*, *13*(14), 2354. https://doi.org/10.3390/ani13142354

Young, I., Gerbrands, J., Van Vliet, L., Bibliotheek, C., Haag, D., Theodore, Y., Jacob, G., Vliet, V., & Jozef, L. (2004). *Fundamentals Of Image Processing*.

Yusuf, M., Kusumanto, R., Oktarina, Y., Dewi, T., & Risma, P. (2018). BLOB Analysis for Fruit Recognition and Detection. *Computer Engineering and Applications Journal*, *7*, 25–36. https://doi.org/10.18495/comengapp.v7i1.237

Zinzendoff Okwonu, F., Laro Asaju, B., & Irimisose Arunaye, F. (2020). Breakdown Analysis of Pearson Correlation Coefficient and Robust Correlation Methods. *IOP Conference Series: Materials Science and Engineering*, *917*(1), 012065. https://doi.org/10.1088/1757-899X/917/1/012065

Lampiran 1 Hasil pengambilan data berat aktual telur

Telur	**Berat** ($gram$)
1	44
2	48
3	37
4	40
5	48
6	48
7	45
8	46
9	48
10	48
11	46
12	48
13	41
14	42
15	49
16	42
17	48
18	46
19	44
20	41
21	48
22	41
23	43
24	50
25	40
26	49
27	44
28	44
29	47
30	43
31	55
32	56
33	52
34	52
35	55
36	56
37	57
38	57
39	59
40	53
41	58
42	56
43	55
44	55
45	57
46	54

Telur	**Berat** ($\boldsymbol{gram}$)
47	56
48	55
49	55
50	52
51	53
52	51
53	54
54	51
55	50
56	51
57	52
58	50
59	51
60	51
61	64
62	63
63	57
64	61
65	59
66	58
67	60
68	65
69	58
70	61
71	58
72	61
73	60
74	59
75	65
76	69
77	61
78	63
79	61
80	65
81	66
82	60
83	60
84	54
85	58
86	64
87	63
88	58
89	55
90	57

Lampiran 2 Hasil ekstraksi fitur

Area	*Centroid X*	*Centroid Y*	*Min Radius Distance*	*Max Radius Distance*	*Highest Eigen Value*	*Lowest Eigen Value*	*Eigen Value Ratio*
2795	39,30161002	34,80930233	0,356839188	36,96094685	749324,5358	516543,5651	0,689345591
2805	36,82495544	31,66702317	0,376183687	39,57204647	772545,2942	508080,7586	0,657671158
2528	38,99011076	32,14833861	0,148667883	38,3350892	573129,8201	451648,3057	0,788038399
2728	40,09127566	35,75403226	0,262357344	36,66289412	665312,0065	527538,2215	0,792918535
2823	42,16861495	35,30145236	0,345404869	42,49778305	792718,9723	508687,231	0,641699327
2995	41,321202	33,86277129	0,349288487	42,95100053	915735,8133	556773,7894	0,608007005
2847	36,08710924	34,33895328	0,349967639	34,04327755	729457,8497	570636,4573	0,78227475
2808	43,85897436	32,22863248	0,268628073	46,74452688	804996,1537	490821,2181	0,609718712
2942	43,41332427	33,95445275	0,41582629	44,78167715	854014,4233	555604,8711	0,650580196
3078	43,38434048	31,16699155	0,419051052	50,16159886	932324,4573	610468,0343	0,654780672
2827	38,3463035	36,46126636	0,576795259	34,85083948	827047,348	490201,1295	0,592712292
2985	45,5561139	34,04455611	0,446116705	47,75302204	922486,6403	549887,2847	0,59609241
2640	36,8969697	30,44924242	0,460905629	38,86053553	691666,5222	445256,6521	0,643744692
2916	39,93347051	34,80486968	0,206160164	40,86832689	814123,5273	563407,5368	0,692041831
3163	40,06860575	39,13215302	0,148899866	36,09786697	982561,1575	645491,7151	0,65694813
2771	41,20750632	29,01768315	0,208258408	49,33034486	779962,4513	479505,3661	0,614780064
2762	40,48370746	30,80376539	0,521997058	43,07531165	737592,5379	501186,8698	0,679490158
3167	38,42311336	42,45184717	0,619024055	37,99189153	1066082,724	598082,7105	0,561009664
2880	39,57881944	37,72673611	0,502061962	34,89896106	809351,5284	538596,5209	0,665466737
2934	36,24369461	38,58111793	0,4846124 8	35,16994018	866936,583	541658,3693	0,624795838
2747	39,28431016	37,19657809	0,345651861	34,13076418	757890,1098	476698,6918	0,628981281
3111	33,31822565	42,34233365	0,467396936	45,64800277	999091,3094	594716,0618	0,595256966
2699	33,23267877	37,85698407	0,27311713	37,00000664	801750,8508	420889,8228	0,524963363
2898	31,27501725	38,35024155	0,445312957	43,24120915	853500,4142	523902,9019	0,613828527
2825	33,67292035	43,08389381	0,337667389	43,66100751	804416,965	501837,9306	0,623852992
3071	33,06740475	41,17095409	0,18376262	45,33059828	1032880,284	545926,012	0,528547229
2751	41,17593602	34,84223919	0,236309032	37,46859885	823682,3612	440716,0176	0,535055791

Area	*Centroid X*	*Centroid Y*	*Min Radius Distance*	*Max Radius Distance*	*Highest Eigen Value*	*Lowest Eigen Value*	*Eigen Value Ratio*
2822	40,18001417	38,60311836	0,435798282	34,44556277	826010,7238	486427,3215	0,588887417
3131	32,83296072	42,42702012	0,458528414	45,20705863	959836,5424	634929,1702	0,661497184
3181	35,58975165	43,921094	0,417767718	44,33490538	1077360,928	602025,892	0,558796849
3409	38,11146964	40,66471106	0,353332923	41,05243016	1229194,381	696533,0249	0,566658159
3288	37,68734793	39,61618005	0,495044514	40,84521323	1163064,894	637527,3187	0,548144237
3188	35,23055207	47,15715182	0,279017834	50,17034405	1059423,149	617578,662	0,582938614
3330	38,01441441	42,74774775	0,252663757	40,32906159	1124989,558	692775,8578	0,615806478
3372	35,04863582	44,87158956	0,13731236	48,01328645	1141165,857	717653,5652	0,62887753
3340	31,47754491	44,09700599	0,487297961	53,16102844	1300723,534	606630,3521	0,466379162
3613	34,76861334	46,32189316	0,396427793	49,6544489	1289223,125	838048,0738	0,650041143
3267	34,70982553	47,33547597	0,443559863	50,97628672	1243916,957	580204,2763	0,466433288
3736	46,07253747	35,42826552	0,434365105	49,60239901	1428981,052	864787,0655	0,605177419
3141	40,07513531	39,15377268	0,171147167	34,94391287	952262,4105	647454,5854	0,679911943
3593	34,0044531	46,20985249	0,209899733	50,23971011	1270824,511	831637,1888	0,654407577
3852	39,24195223	47,35280374	0,427798271	48,37466532	1449852,353	963769,6868	0,664736436
3471	38,72486315	39,1019879	0,293431111	38,26843752	1260630,051	729844,0894	0,578951841
3564	42,7996633	39,81425365	0,273196817	38,32000209	1243456,113	822821,8824	0,661721691
3623	37,62655258	44,25752139	0,453630072	43,82251151	1398601,435	781429,0229	0,558721737
3248	44,9408867	32,97998768	0,062408934	51,09492232	1031051,023	685011,3264	0,664381598
3514	34,88303927	43,62122937	0,396417713	47,55866944	1345553,987	719011,7986	0,534361167
3581	36,67774365	42,82211673	0,368091858	43,71172541	1330098,983	783898,821	0,589353748
3777	35,48742388	43,53402171	0,6743276	45,32327172	1336587,271	965248,2603	0,722173764
3103	33,39252336	44,50145021	0,634528554	48,2921384	1002153,41	586452,2404	0,585192082
2964	34,38900135	39,41396761	0,568059181	41,11437938	933261,8394	523989,7038	0,561460548
3121	34,66356937	45,10829862	0,353431975	47,36215046	997676,5019	602647,641	0,604051153
3512	33,21326879	43,83627563	0,2688666	50,43613285	1266686,01	761548,1102	0,601213011
2930	31,21877133	44,34266212	0,406544242	49,60919719	909806,3912	514114,3436	0,565081042
3387	38,19309123	42,36669619	0,414427702	41,46651681	1194303,881	698042,401	0,584476373

Area	*Centroid X*	*Centroid Y*	*Min Radius Distance*	*Max Radius Distance*	*Highest Eigen Value*	*Lowest Eigen Value*	*Eigen Value Ratio*
3189	35,41831295	42,24521794	0,484889226	46,86961689	1124254,419	583771,7912	0,519252388
3110	33,72700965	40,88424437	0,296518294	43,54574577	1013156,824	585356,7351	0,57775531
3350	34,71850746	41,52955224	0,548232747	43,56019295	1077778,725	740651,4016	0,687201727
3140	32,26528662	43,32515924	0,419649284	49,55968856	1067402,451	577230,578	0,540780638
3385	34,94800591	41,77784343	0,228159875	45,2754485	1172369,136	710094,6513	0,605692038
3994	40,62769154	48,69929895	0,478575714	49,44604188	1713799,371	941933,8646	0,549617348
4406	38,95869269	44,29187472	0,294783215	47,40855658	1674073,352	1429347,779	0,853814307
3322	35,66646598	40,9397953	0,338924099	43,88571441	1213249,423	636486,9801	0,524613462
3532	33,9306342	46,55520951	0,450166848	54,3432001	1424072,206	693492,0339	0,48697814
3385	37,89512555	46,57577548	0,436995529	49,25368994	1325023,18	628177,4028	0,474087859
3222	34,05090006	45,16263191	0,170411129	50,64268387	1057463,276	645719,157	0,610630337
3495	39,11072961	43,71330472	0,307335696	45,17790337	1332632,738	710566,1405	0,533204776
3770	39,0535809	48,36100796	0,364962544	50,4859309	1500469,572	854691,2725	0,569615864
3848	34,86876299	42,31262994	0,339058446	46,56114608	1477654,773	941178,8587	0,636940966
3540	38,67683616	44,34124294	0,469980437	46,28748876	1379688,345	721227,7339	0,522746848
3335	34,97511244	42,93103448	0,073318709	47,30515606	1222767,853	641157,2191	0,524349097
3586	38,6469604	45,91299498	0,363602574	49,95531772	1510460,389	698377,5176	0,462360697
3843	36,38199323	46,47957325	0,613114454	54,31715687	1731202,521	799485,8593	0,461809551
3914	40,55288707	42,68421053	0,547387397	41,78661015	1554210,863	957646,3732	0,61616245
3752	40,59275053	42,06609808	0,41257858	42,51794638	1486203,426	846707,9041	0,569711985
4023	37,02411136	50,66418096	0,33668351	58,69687724	1868942,661	887727,3088	0,474989055
3975	35,7363522	47,37031447	0,454579988	56,02449466	1768365,222	895321,3718	0,506298903
3899	32,59656322	49,27032572	0,485630757	62,71956566	1781204,134	823451,3361	0,462300373
3621	36,95719415	44,94145264	0,072526787	50,61466988	1413244,81	770733,1433	0,545364213
3857	36,08970703	50,15348717	0,17777981	59,98775889	1670411,391	840966,7061	0,503448857
3786	36,82857898	47,22768093	0,284997848	52,24600348	1472606,264	884687,223	0,600762909
3838	39,34392913	47,09119333	0,355813813	48,18076498	1567436,689	879163,4066	0,560892451
3433	37,76551121	49,83046898	0,289354036	53,78719878	1389674,773	634232,7967	0,456389372

Area	***Centroid X***	***Centroid Y***	***Min Radius Distance***	***Max Radius Distance***	***Highest Eigen Value***	***Lowest Eigen Value***	***Eigen Value Ratio***
3325	31,50887218	44,39699248	0,631513709	53,85549599	1143973,043	677541,9148	0,592270875
3788	38,35718057	43,96937698	0,358490905	47,75919892	1616290,007	807204,1749	0,499417908
3783	36,17393603	47,57097542	0,462942584	54,72367023	1657088,353	783157,89	0,472610823
3908	34,68961105	44,56780962	0,532099447	53,62740708	1660070,076	890167,4528	0,536222817
4153	37,10931856	45,23669636	0,260721532	50,26760629	1631523,989	1157054,708	0,709186451
3471	33,24056468	48,0613656	0,248268205	56,76306461	1239743,9	742050,1573	0,598551166
3392	34,078125	48,23024764	0,243140889	55,11798699	1197846,733	700096,7404	0,584462704

Lampiran 3 *Source Code Testing*

```
import cv2
import os
from skimage import transform
import numpy as np

# Path folder
input_file = r'C:\Users\HP\Documents\SKRIPSI\Citra\FINAL\14. Testing video\Testing
FIX\10cm_60rpm.mp4'
output_file = r'C:\Users\HP\Documents\SKRIPSI\Citra\FINAL\14. Testing video\Testing
FIX\output_10cm_60rpm.mp4'

original_size = (108, 192)

# Menentukan nilai alpha dan beta
alpha = 1.1
beta = 20

# Menentukan lower_bound dan upper_bound untuk peregangan kontras
lower_bound = 40
upper_bound = 275

# Menentukan threshold value
threshold_value = 215

# Menentukan treshold magnitudo
threshold_magnitudo = 50

# Margin crop
margin = 5

# Membaca video
cap = cv2.VideoCapture(input_file)
frame_width = int(cap.get(cv2.CAP_PROP_FRAME_WIDTH))
frame_height = int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT))

# Membuat video writer untuk menyimpan hasil pengolahan
fourcc = cv2.VideoWriter_fourcc(*'mp4v')
out = cv2.VideoWriter(output_file, fourcc, 30, (192, 108))

# Menghitung eigen values
def calculate_eigen_values(M):
    cov_mat = np.array([[M['mu20'], M['mu11']], [M['mu11'], M['mu02']]])
    eigen_values = np.linalg.eigvalsh(cov_mat)
    return eigen_values

# Variabel update frame
calculated = False
calculated_frames = 0

# Loop untuk setiap frame dalam video
while cap.isOpened():
    ret, frame_input = cap.read()
    if not ret:
        break

    # Resize 10%
    frame = transform.resize(frame_input, (int(frame_height * 0.1), int(frame_width * 0.1)),
anti_aliasing=True)
    frame_resize = (frame * 255).astype('uint8')

    # Peningkatan kontras dan kecerahan
    frame_kontras_kecerahan = cv2.convertScaleAbs(frame_resize, alpha=alpha, beta=beta)

    # Peregangan kontras
    frame_peregangan = cv2.normalize( frame_kontras_kecerahan, None, alpha=0, beta=255,
norm_type=cv2.NORM_MINMAX)

    # RGB ke Gray
    frame_gray = cv2.cvtColor(frame_peregangan, cv2.COLOR_RGB2GRAY)

    # Segmentasi
    _, frame_biner = cv2.threshold(frame_gray, threshold_value, 255, cv2.THRESH_BINARY)

    # Invers
    frame_invers = 255-frame_biner
```

```
    # Deteksi objek
    kontur,_ = cv2.findContours(frame_invers, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
    if kontur and cv2.contourArea(max(kontur, key=cv2.contourArea)) >= 2000:
        kontur_terbesar = max(kontur, key=cv2.contourArea)
        x, y, w, h = cv2.boundingRect(kontur_terbesar)

        # Gambar bounding box
        cv2.rectangle(frame_resize, (x, y), (x + w, y + h), (0, 255, 0), 1)

        # Crop objek
        x_start = max(x-margin, 0)
        y_start = max(y-margin, 0)
        x_end = min(x + w + margin, frame_invers.shape[1])
        y_end = min(y + h + margin, frame_invers.shape[0])
        frame_crop = frame_invers[y_start:y_end, x_start:x_end]

        # Deteksi tepi
        sobelx = cv2.Sobel(frame_crop, cv2.CV_64F, 1, 0, ksize = 3)
        sobely = cv2.Sobel(frame_crop, cv2.CV_64F, 0, 1, ksize = 3)
        magnitude = np.sqrt(sobelx**2 + sobely**2)
        magnitude = np.uint8(255*magnitude/np.max(magnitude))
        _, frame_treshold_magnitude = cv2.threshold(magnitude, threshold_magnitudo, 255,
cv2.THRESH_BINARY)

        # Hole filling
        _, frame_binary = cv2.threshold(frame_treshold_magnitude, 127, 255, cv2.THRESH_BINARY)
        frame_flood_filled = frame_binary.copy()
        mask = np.zeros((frame_flood_filled.shape[0] + 2, frame_flood_filled.shape[1] + 2),
np.uint8)
        cv2.floodFill(frame_flood_filled, mask, (0,0), 255)
        frame_invert = cv2.bitwise_not(frame_flood_filled)
        frame_filling = cv2.bitwise_or(frame_binary, frame_invert)
        frame_filter = cv2.medianBlur(frame_filling, 11)
        _, frame_biner = cv2.threshold(frame_filter, 127, 1, cv2.THRESH_BINARY)

        # Cek frame
        if calculated == True:
            cv2.putText(frame_resize, f'{prediksi}, {egg_size}', (x, y - 10),
cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 0, 0), 1)
            calculated_frames += 1
            if calculated_frames >= 8:
                calculated = False
                calculated_frames = 0

        # Perhitungan berat
        if y_start >= 15 and not calculated:
            moments = cv2.moments(frame_biner)
            eigen_values = calculate_eigen_values(moments)
            highest_eigen_value = np.max(eigen_values)
            prediksi = int(0.0000209547 * highest_eigen_value + 28.7338098999)
            print(highest_eigen_value)
            calculated = True
            calculated_frames = 1
            if calculated == True:
                egg_size = ""
            if prediksi < 50:
                egg_size = "Kecil"
            elif prediksi >= 50 and prediksi <= 60:
                egg_size = "Sedang"
            else:
                egg_size = "Besar"
            cv2.putText(frame_resize, f'{prediksi}, {egg_size}', (x, y - 10),
cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 0, 0), 1)

    # Menyimpan frames
    out.write(frame_resize)

    # Menampilkan frame
    cv2.imshow('Frame', frame_resize)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

# Menutup video writer dan frame
cap.release()
out.release()
cv2.destroyAllWindows()
```

Lampiran 4 Perhitungan RMSE

Telur ke-n	**Hasil Estimasi ($\hat{y}$)**	**Berat Aktual (y)**	$(\hat{y} - y)^2$
1	48	47	1
2	44	48	16
3	37	41	16
4	40	43	9
5	46	46	0
6	42	45	9
7	49	51	4
8	50	49	1
9	52	49	9
10	53	51	4
11	56	52	16
12	57	54	9
13	54	56	4
14	51	51	0
15	60	58	4
16	65	64	1
17	61	58	9
18	61	60	1
19	59	56	9
20	69	67	4
21	58	61	9
22	58	58	0
23	52	52	0
24	52	53	1
25	52	51	1
Jumlah			137

Dari hasil penjumlahan ini diperoleh RMSE sebagai berikut

$$\text{RMSE} = \sqrt{\frac{1}{25}(137)}$$

$$\text{RMSE} \approx 2{,}3409$$

Lampiran 5 Perhitungan MAPE

Telur ke-n	**Hasil Estimasi ($\hat{y}$)**	**Berat Aktual (y)**	$\left\|\frac{y-\hat{y}}{y}\right\|$
1	48	47	0,020833333
2	44	48	0,090909091
3	37	41	0,108108108
4	40	43	0,075
5	46	46	0
6	42	45	0,071428571
7	49	51	0,040816327
8	50	49	0,02
9	52	49	0,057692308
10	53	51	0,037735849
11	56	52	0,071428571
12	57	54	0,052631579
13	54	56	0,037037037
14	51	51	0
15	60	58	0,033333333
16	65	64	0,015384615
17	61	58	0,049180328
18	61	60	0,016393443
19	59	56	0,050847458
20	69	67	0,028985507
21	58	61	0,051724138
22	58	58	0
23	52	52	0
24	52	53	0,019230769
25	52	51	0,019230769
Jumlah			0,967931135

Dari hasil penjumlahan ini diperoleh MAPE sebagai berikut

$$\text{MAPE} = \frac{1}{25}(0{,}967931135)$$

$$\text{MAPE} \approx 0{,}038717245 \text{ atau } 3{,}8717\%$$

Lampiran 6 Perhitungan rata-rata RMSE dan MAPE

Skenario Seleksi *Frame*	**Percobaan *Random State***	**RMSE**	**MAPE**
Semua *frame* pertelur	1	3,3634	5,3328%
	2	4,5449	6,8968%
	3	3,8568	5,2523%
	4	3,9370	5,9237%
	5	3,7417	6,2916%

$$\text{Rata-rata RMSE} = \frac{3{,}3634 + 4{,}5449 + 3{,}8568 + 3{,}9370 + 3{,}7417}{5}$$
$$= \frac{19{,}4438}{5}$$
$$= 3{,}8887$$

$$\text{Rata-rata MAPE} = \frac{5{,}3328 + 6{,}8968 + 5{,}2523 + 5{,}9237 + 6{,}2916}{5}$$
$$= \frac{29{,}6972}{5}$$
$$= 5{,}9394\%$$

Lampiran 7 Lembar perbaikan skripsi

LEMBAR PERBAIKAN SKRIPSI

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

OLEH:

AGUNAWAN ALI NUR
D121201081

Skripsi ini telah dipertahankan pada Ujian Akhir Sarjana tanggal 7 Juni 2024.

Telah dilakukan perbaikan penulisan dan isi skripsi berdasarkan usulan dari penguji dan pembimbing skripsi.

Persetujuan perbaikan oleh tim penguji:

	Nama	Tanda Tangan
Ketua	Prof. Dr. Ir. Indrabayu, S.T.,M.T., M.Bus.Sys., IPM, ASEAN. Eng.	
Sekretaris	Elly Warni, S.T., M.T.	
Anggota	Novy Nur R.A Mokobombang,ST.,Ms.TM. Ph.D	
	A. Ais Prayogi Alimuddin, S.T., M.Eng	

Persetujuan Perbaikan oleh pembimbing:

Pembimbing	Nama	Tanda Tangan
I	Prof. Dr. Ir. Indrabayu, S.T.,M.T., M.Bus.Sys., IPM, ASEAN. Eng.	
II	Elly Warni, S.T., M.T.	