

Evaluasi Kinerja Baseline Model YOLOv4 Pre-trained COCO pada Dataset MinneApple

¹Irfan Sriyono Putro*, ²Billy Ibrahim Hasbi, ³Purwantoro

^{1,3}Informatika, Universitas Singaperbangsa Karawang, kab. Karawang

²Sistem Informasi, Universitas Singaperbangsa Karawang, kab. Karawang

E-mail: ¹irfan.sriyono@cs.unsika.ac.id, ²billy.ibrahim@staff.unsika.ac.id,
³purwantoro.masbro@staff.unsika.ac.id

ABSTRAK

Model deep learning YOLOv4 pre-trained menawarkan potensi besar untuk otomatisasi di bidang pertanian presisi. Namun, kinerja baseline ("out-of-the-box") dari varian-varianannya (Tiny dan Full) pada domain spesifik seperti penghitungan buah belum terdokumentasi dengan baik. Penelitian ini menyajikan evaluasi kinerja baseline dari model YOLOv4-Tiny dan YOLOv4-Full pre-trained COCO. Studi ini menggunakan metode benchmark kuantitatif pada 100 test set dari dataset MinneApple, yang diimplementasikan pada platform MATLAB dengan Computer Vision Toolbox™. Parameter kinerja yang dievaluasi adalah Mean Absolute Error (MAE) untuk akurasi penghitungan dan Frames Per Second (FPS) untuk kecepatan. Hasil menunjukkan trade-off (pertukaran) kinerja yang jelas: YOLOv4-Full (MAE 21,57) terbukti 40% lebih akurat daripada YOLOv4-Tiny (MAE 36,07). Sebaliknya, YOLOv4-Tiny terbukti 24 kali lebih cepat (12,70 FPS) dibandingkan YOLOv4-Full (0,53 FPS). Temuan penting lainnya adalah tingginya nilai MAE pada kedua model, yang mengindikasikan adanya kesenjangan kinerja (performance gap) signifikan. Studi ini menyediakan data baseline krusial yang menegaskan bahwa fine-tuning adalah langkah esensial untuk aplikasi praktis di domain spesifik.

Kata kunci : *YOLOv4, Model Pre-trained, Kinerja Baseline, MinneApple*

ABSTRACT

Pre-trained YOLOv4 deep learning models offer significant potential for automation in precision agriculture. However, the "out-of-the-box" (baseline) performance of its variants (Tiny and Full) on specific domains, such as fruit counting, is not well-documented. This study presents a baseline performance evaluation of the pre-trained COCO YOLOv4-Tiny and YOLOv4-Full models. The study uses a quantitative benchmark method on 100 test images from the MinneApple dataset, implemented on the MATLAB platform using the Computer Vision Toolbox™. The primary evaluated performance are Mean Absolute Error (MAE) for counting accuracy and Frames Per Second (FPS) for speed. The results show a clear performance trade-off: YOLOv4-Full (MAE 21.57) was 40% more accurate than YOLOv4-Tiny (MAE 36.07). Conversely, YOLOv4-Tiny proved to be 24 times faster (12.70 FPS) than YOLOv4-Full (0.53 FPS). Another critical finding is the high MAE for both models, indicating a significant performance gap. This study provides crucial baseline data, confirming that fine-tuning is an essential step for practical applications in specific domains.

Keyword : *YOLOv4, Pre-trained Models, Baseline Performance, MinneApple*

1. PENDAHULUAN

Precision agriculture memanfaatkan kecerdasan buatan (AI) dan teknologi digital untuk meningkatkan efisiensi dan profitabilitas di era Industri 4.0 (Bhatti et al., 2023). Penerapan teknologi memiliki tujuan untuk memperbaiki produktivitas pertanian dan ketahanan pangan melalui pemantauan digital berbasis citra satelit dan data terkini (Wijaya et al., 2023). Estimasi hasil panen apel penting untuk mendukung perencanaan logistik dan pengelolaan sumber daya yang efektif. Namun, metode manual yang umum digunakan memiliki keterbatasan waktu, biaya, dan akurasi (Bhatti et al., 2023).

Pemanfaatan metode deep learning dan computer vision telah menjadi solusi otomatis yang menjanjikan dalam estimasi hasil panen (Xiao et al., 2023). Salah satu dataset besar yang menjadi standar dalam bidang deteksi objek adalah COCO (*Common Objects in Context*) yang menyediakan berbagai kelas objek termasuk apel (Lin et al., 2014). Banyak penelitian menggunakan teknik transfer learning dengan memanfaatkan model yang sudah dilatih pada dataset COCO untuk diaplikasikan ke domain baru, termasuk pertanian. Dalam konteks ini, arsitektur YOLO yang populer digunakan dengan model pre-trained pada COCO telah menunjukkan performa yang baik dalam berbagai tugas deteksi objek, sehingga sering dijadikan pilihan utama dalam transfer learning (Bochkovskiy et al., 2020; Redmon et al., 2016).

Saat ini, YOLO sudah mengalami perkembangan hingga versi 12 ke atas, namun demikian penelitian ini memilih YOLOv4 sebagai fokus. Hal ini dikarenakan YOLOv4 merupakan versi stabil yang telah memiliki dokumentasi lengkap, performa teruji secara luas, serta kompatibel dengan berbagai perangkat

yang umum digunakan dalam aplikasi pertanian.

Terdapat dua varian utama pada YOLOv4, yaitu YOLOv4-Full yang menggunakan backbone CSPDarknet53 untuk hasil akurasi tinggi namun kompleks secara komputasi, serta YOLOv4-Tiny yang merupakan versi sederhana dan ringan untuk kebutuhan kecepatan pada perangkat edge (Bochkovskiy et al., 2020).

Dataset COCO memiliki kelas apel, dimana kebanyakan apel dalam dataset tersebut merupakan apel yang berada di atas meja. Hal ini berbeda dengan apel hasil panen pertanian yang masih menempel di pohon dimana sering terhalang daun dan bertumpuk rapat. Oleh karena itu, hal ini menimbulkan pertanyaan apakah model YOLO yang sudah dilakukan training terhadap dataset COCO bisa langsung diaplikasikan pada dataset apel hasil pertanian yang masih menempel pada pohon. Hal ini juga diperkuat dengan fakta bahwa kinerja transfer learning langsung tanpa proses fine-tuning pada domain baru belum banyak didokumentasikan (Xiao et al., 2023).

Di sisi lain, terdapat dilema mengenai *trade-off* antara akurasi dan kecepatan antara YOLOv4-Full dan YOLOv4-Tiny saat diterapkan secara langsung pada dataset pertanian apel. Penelitian ini bertujuan untuk mengisi kesenjangan tersebut dengan melakukan benchmark dengan mengukur dan menganalisis kinerja baseline kedua model *pre-trained* tersebut dalam hal akurasi dan kecepatan pada dataset MinneApple tanpa melakukan *fine-tuning* atau yang biasa disebut dengan *zero-shot transfer*. MinneApple sendiri merupakan dataset buah apel yang masih berada di pohon (Hani et al., 2020). Hasil penelitian diharapkan dapat menjadi acuan bagi peneliti maupun praktisi pertanian dalam

memilih model yang tepat sesuai kebutuhan perangkat dan aplikasi.

2. LANDASAN TEORI

2.1 Deteksi Objek pada Pertanian

Metode tradisional untuk deteksi buah dalam bidang pertanian, yang mengandalkan fitur warna dan tepi, memiliki keterbatasan dalam kondisi lapangan yang bervariasi, khususnya terhadap perubahan pencahayaan dan oklusi oleh daun atau cabang pohon (Xiao et al., 2023). Deep learning telah terbukti mengatasi masalah ini, dengan model seperti R-CNN yang berhasil mendeteksi buah tomat, mangga, anggur, dan apel dengan tingkat akurasi tinggi dalam kondisi lingkungan yang kompleks (Girshick et al., 2014; Xiao et al., 2023). Metode deep learning juga telah diterapkan pada buah lain seperti jeruk (Chen et al., 2020) dan mangga (Chen et al., 2020) yang menunjukkan keberhasilan teknologi ini dalam bidang pertanian.

2.2 YOLO

YOLO (You Only Look Once) mengubah paradigma deteksi objek dari pendekatan *region purpose* menjadi regresi langsung, menjadikannya sangat cepat dan efisien untuk pemrosesan real-time (Redmon et al., 2016). Sejak YOLOv1 hingga YOLOv3, perkembangan fokus pada peningkatan akurasi tanpa mengorbankan kecepatan. YOLOv4, yang menggunakan backbone CSPDarknet53, berhasil menyeimbangkan kecepatan dan akurasi lebih baik dengan optimisasi terbaru pada arsitektur dan teknik augmentasi data (Bochkovskiy et al., 2020). YOLOv4-Tiny adalah versi ringkas yang memprioritaskan efisiensi komputasi untuk perangkat edge dengan sumber daya terbatas, meskipun dengan kompromi pada kapasitas belajar dan akurasi (Bochkovskiy et al., 2020).

2.3 Akurasi dan Kecepatan

Dalam penerapan deep learning, terdapat trade-off mendasar antara kedalaman dan kompleksitas model yang meningkatkan akurasi, dengan kebutuhan komputasi yang lebih tinggi (FLOPs), sehingga memengaruhi kecepatan inferensi (Mittal, 2024). Model seperti YOLOv4-Full memiliki FLOPs yang tinggi dan kapasitas fitur yang lebih baik, menghasilkan hasil deteksi yang akurat tapi lambat. Sebaliknya, YOLOv4-Tiny mengurangi FLOPs untuk kecepatan tinggi dengan mengorbankan kompleksitas fitur dan tingkat akurasi.

2.4 Research Gap

Sebagian besar penelitian yang ada menitikberatkan pada pelatihan dan fine-tuning model YOLO pada dataset spesifik pertanian, sering kali tanpa membandingkan langsung kinerja model pre-trained COCO secara zero-shot transfer (Xiao et al., 2023). Studi benchmark yang mengukur performa out-of-the-box model pre-trained tersebut penting untuk menilai kemampuan generalisasi model, serta memberikan gambaran trade-off baseline antara YOLOv4-Full dan Tiny pada domain apel pertanian yang memiliki tantangan oklusi dan kepadatan objek (Bhatti et al., 2023). Penelitian ini menjawab gap literatur tersebut dengan pengujian komprehensif terhadap kedua varian model pada dataset MinneApple secara zero-shot transfer, memberikan data kuantitatif yang dapat menjadi dasar pemilihan model sesuai kebutuhan aplikasi.

3. METODOLOGI

Penelitian ini dirancang menggunakan pendekatan metode campuran yang menggabungkan studi eksperimental kuantitatif dengan analisis kualitatif visual. Tahap kuantitatif berfokus pada pengumpulan data benchmark (MAE, FPS) secara sistematis pada seluruh test

set. Tahap kualitatif kemudian digunakan untuk menganalisis sampel citra secara mendalam guna membantu menjelaskan mengapa temuan kuantitatif tersebut terjadi.

Metodologi penelitian dibagi menjadi lima tahap utama: persiapan dataset, konfigurasi lingkungan eksperimen, arsitektur model, skenario pengujian, dan parameter evaluasi kinerja.

3.1 Dataset

Dataset yang digunakan dalam penelitian ini berasal dari MinneApple (Hani et al., 2020), sebuah dataset publik yang aslinya diperkenalkan oleh Hani et al. untuk benchmarking deteksi buah apel. Penelitian ini berfokus hanya pada 100 gambar test set yang tersedia dari dataset tersebut. Sesuai dengan yang disediakan oleh dataset, penelitian ini menggunakan format anotasi ground truth PASCAL VOC (file .xml). Ketersediaan spesifik dan rincian pembagian dari subset data yang digunakan dalam studi ini dijelaskan lebih lanjut di bagian "Ketersediaan Data".

3.2 Konfigurasi Perangkat dan Alat

Penelitian ini diimplementasikan dengan menggunakan perangkat lunak *Matlab 2025a*. Secara spesifik pemrosesan data dan eksekusi model menggunakan *yolov4ObjectDetector* dari *Computer Vision Toolbox™* untuk memuat dan menjalankan model *pre-trained*.

Penelitian ini dijalankan menggunakan Laptop Dynabook Portege X30W-K dengan konfigurasi processor intel core i5 1340P, RAM 16gb, dengan sistem operasi Windows 11. Secara keseluruhan, semua proses berjalan hanya menggunakan CPU tanpa menggunakan *Graphics Processing Unit* (GPU).

3.3 Arsitektur Model

Penelitian ini membandingkan dua arsitektur model yaitu YOLOv4 *pre-*

trained pada dataset COCO. Model *pre-trained* COCO dipilih karena merupakan *baseline* yang umum digunakan dalam penelitian *transfer learning*. Selain itu, dataset COCO telah memiliki kelas 'apple', sehingga memungkinkan evaluasi transfer learning zero-shot.

Model YOLO yang digunakan pada penelitian ini adalah YOLOv4-Tiny dan YOLOv4-Full. YOLOv4-Tiny (*tiny-yolov4-coco*) merupakan varian ringan yang dioptimalkan untuk inferensi berkecepatan tinggi, sedangkan YOLOv4-Full (*csp-darknet53-coco*) merupakan varian penuh yang dioptimalkan untuk akurasi deteksi yang lebih baik.

3.4 Skenario Pengujian

Skenario pengujian dirancang sesuai pendekatan *mixed methods*. Tahap pengujian kuantitatif melibatkan evaluasi seluruh test set menggunakan kedua model (tanpa *fine-tuning*) untuk mengukur kinerja baseline "out-of-the-box".

Untuk memastikan perbandingan berjalan secara adil dan valid, *hyperparameter* utama, *Confidence Threshold*, ditetapkan pada nilai konstan 0.2 untuk kedua model. Tahap pengujian kualitatif kemudian dilakukan dengan memilih satu citra uji representatif. Citra ini dianalisis menggunakan *ground truth* dan kemudian diproses oleh kedua model untuk dianalisis secara visual.

3.5 Parameter Evaluasi Kinerja

Kinerja model dievaluasi menggunakan dua pendekatan, yaitu evaluasi kuantitatif dan kualitatif.

Pendekatan evaluasi kuantitatif berfokus pada dua aspek utama: akurasi penghitungan dan kecepatan komputasi. Aspek akurasi penghitungan diukur menggunakan MAE (**Mean Absolute Error**) yang mengukur rata-rata

kesalahan absolut antara jumlah apel hasil *ground truth* (y_i) dan hasil diprediksi ($\hat{y}_i$), sesuai Persamaan (1).

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (1)$$

Selain itu, digunakan pula **RMSE (Root Mean Square Error)** yang merupakan akar dari rata-rata kuadrat *error* sesuai Persamaan (2).

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n}} \quad (2)$$

Aspek kecepatan komputasi diukur menggunakan **Avg. Inference Time (ms)**, yaitu rata-rata waktu (dalam milidetik) yang dibutuhkan model untuk memproses satu gambar, dan **Avg. FPS (Frames Per Second)**, yaitu rata-rata jumlah gambar yang dapat diproses per detik ($1000/\text{Avg. Inference Time (ms)}$).

Pendekatan evaluasi kualitatif dilakukan melalui analisis visual komparatif, yaitu perbandingan hasil deteksi dari kedua model terhadap *ground truth* untuk mengidentifikasi pola kegagalan.

4. HASIL DAN PEMBAHASAN

Eksperimen komparatif dijalankan secara otomatis pada seluruh test set. Hasil kinerja baseline kuantitatif dari kedua model pre-trained dirangkum dalam Tabel 1, sementara analisis kualitatif visual disajikan pada Gambar 1 dan Gambar 2.

4.1 Analisis *trade-off* Kuantitatif

Data pada Tabel 1 menunjukkan *trade-off* yang sangat jelas antara dua arsitektur pre-trained tersebut:

Tabel 1. Hasil Benchmark Kinerja YOLOv4-Tiny vs YOLOv4-Full (Pre-trained COCO, Threshold = 0.2)

Parameter	YOLOv4-Tiny	YOLOv4-Full
MAE	36,07	21,57
RMSE	45,073	32,223
Avg. Inference Time (ms)	78,765	1881,1
Avg. FPS	12,696	0,532

Model YOLOv4-Full (MAE 21,57) secara signifikan 40% lebih akurat daripada model YOLOv4-Tiny (MAE 36,07). Arsitektur CSPDarknet53 yang lebih kompleks terbukti mampu menghasilkan baseline akurasi yang lebih baik, bahkan tanpa *fine-tuning*.

Model YOLOv4-Tiny (12,70 FPS) terbukti 24 kali lebih cepat ($12,70 / 0,53$) daripada model YOLOv4-Full (0,53 FPS). Kecepatan model Tiny memungkinkannya beroperasi pada kecepatan mendekati *real-time*, sementara model Full sama sekali tidak praktis untuk aplikasi langsung di lapangan dikarenakan membutuhkan waktu pemrosesan sekitar 1,9 detik per gambar.

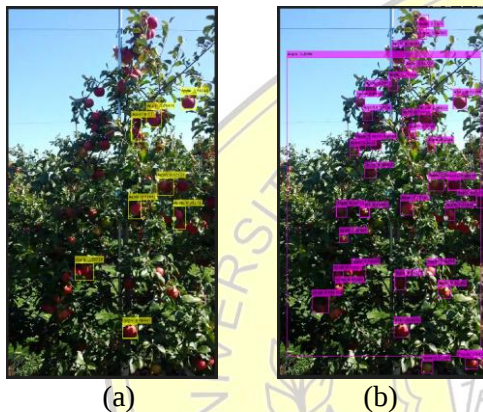
4.2 Analisis Kinerja Kualitatif Visual

Untuk memahami mengapa terdapat perbedaan MAE yang signifikan, analisis visual kualitatif dilakukan pada sampel citra uji yang representatif.

Gambar 1 menyajikan contoh citra uji dari dataset MinneApple beserta anotasi *ground truth*. Citra ini dipilih karena mewakili tantangan umum di lapangan, yaitu kepadatan objek yang tinggi dan tingkat oklusi yaitu terhalang oleh daun yang signifikan. Pada citra spesifik yang dianalisis di Gambar 1, jumlah apel *ground truth* yang merupakan jumlah hitungan sebenarnya adalah 87 buah.



Gambar 1. Citra Uji Representatif dan Anotasi Ground Truth.



Gambar 2. Perbandingan deteksi kualitatif (Threshold 0.2): (a) Hasil YOLOv4-Tiny; (b) Hasil YOLOv4-Full

Analisis visual pada Gambar 2 mengkonfirmasi data kuantitatif dari Tabel 1. Pada Gambar 2(a), terlihat bahwa model YOLOv4-Tiny hanya mampu memprediksi 8 apel dari 87 apel asli. Model ini gagal mendeteksi sejumlah besar apel yang valid (false negative tinggi), terutama apel yang: (1) mengalami oklusi yaitu terhalang sebagian oleh daun atau apel lain, (2) berada dalam kerumunan padat, dan (3) berukuran kecil atau berada jauh di latar belakang.

Sebaliknya, pada Gambar 2(b), model YOLOv4-Full mampu memprediksi 37 apel. Meskipun masih jauh dari nilai asli, namun nilai ini lebih mendekati nilai aslinya dibandingkan dengan YOLOv4-Tiny. Hal ini juga menjelaskan mengapa

nilai MAE-nya (21,57) jauh lebih rendah dibandingkan model Tiny (36,07).

4.3 Diskusi dan Implikasi

Temuan paling penting dari studi benchmark ini bukanlah sekadar *trade-off* antara akurasi dan kecepatan, melainkan tingginya nilai MAE (kesalahan) pada kedua model (21,57 dan 36,07). Meskipun model pre-trained COCO memiliki kelas 'apple', kinerjanya dalam menghitung apel di dataset MinneApple masih jauh dari optimal.

Analisis kuantitatif (Tabel 1) dan kualitatif (Gambar 1 dan Gambar 2) ini mengindikasikan adanya kesenjangan kinerja yang signifikan antara domain generik (COCO) dan domain spesifik (MinneApple). Pengetahuan model tentang "apel COCO" terbukti tidak dapat ditransfer secara efektif ke dalam dataset MinneApple yang padat, tumpang tindih, dan teroklusi.

Temuan ini secara kuantitatif membuktikan bahwa model pre-trained generik tidak dapat diandalkan untuk tugas penghitungan spesifik "out-of-the-box". Data baseline ini (MAE 21,57) justru menggarisbawahi pentingnya *fine-tuning* pada dataset domain spesifik untuk meningkatkan nilai akurasi.

5. KESIMPULAN

Penelitian ini telah berhasil menyajikan studi benchmark yang mengukur kinerja baseline "out-of-the-box" dari model YOLOv4-Tiny dan YOLOv4-Full pre-trained COCO pada studi kasus penghitungan apel. Hasil membuktikan adanya *trade-off* kinerja yang jelas: model YOLOv4-Full 40% lebih akurat (MAE 21,57 vs 36,07) namun 24 kali lebih lambat (0,53 FPS vs 12,70 FPS) dibandingkan YOLOv4-Tiny.

Kontribusi utama dari penelitian ini adalah penyajian data baseline kuantitatif

yang membuktikan adanya kesenjangan kinerja (performance gap) yang signifikan. Tingginya nilai MAE pada kedua model menunjukkan bahwa transfer learning *zero-shot* dari model COCO tidak mencukupi untuk domain pertanian yang kompleks. Temuan ini menegaskan bahwa *fine-tuning* pada dataset spesifik domain adalah langkah yang mutlak diperlukan sebelum implementasi praktis.

KETERSEDIAAN DATA

Dataset subset MinneApple (Hani et al., 2020) yang digunakan dalam penelitian ini tersedia secara publik di repositori Roboflow Universe pada tautan:

<https://universe.roboflow.com/dissertation-bqltf/minneapple-qo6jh> (diakses pada 1 Oktober 2025). Platform ini menyediakan dataset dalam dua versi, yaitu dataset original (1001 gambar) dan dataset *augmented* (1702 gambar). Versi original pada platform ini telah dibagi oleh uploader menjadi 701 gambar training, 200 gambar validasi, dan 100 gambar test. Penelitian ini hanya menggunakan 100 gambar test set tersebut untuk evaluasi baseline.

6. UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Fakultas Ilmu Komputer, Universitas Singaperbangsa Karawang, atas dukungan institusional dan fasilitas yang telah diberikan selama proses penelitian ini, sehingga penelitian ini dapat diselesaikan dengan baik.

DAFTAR PUSTAKA

Bhatti, U. A., Masud, M., Bazai, S. U., & Tang, H. (2023). Editorial: Investigating AI-based smart precision agriculture techniques. *Frontiers in Plant Science*, 14(July), 1–3.
<https://doi.org/10.3389/fpls.2023.1237783>

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). *YOLOv4: Optimal Speed and Accuracy of Object Detection*.

<http://arxiv.org/abs/2004.10934>

Chen, W., Lu, S., Liu, B., Li, G., & Qian, T. (2020). Detecting citrus in orchard environment by using improved YOLOv4. *Scientific Programming*, 2020.
<https://doi.org/10.1155/2020/8859237>

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 580–587.
<https://doi.org/10.1109/CVPR.2014.81>

Hani, N., Roy, P., & Isler, V. (2020). MinneApple: A Benchmark Dataset for Apple Detection and Segmentation. *IEEE Robotics and Automation Letters*, 5(2), 852–858.
<https://doi.org/10.1109/LRA.2020.2965061>

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). LNCS 8693 - Microsoft COCO: Common Objects in Context. *Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland*, 740–755.

Mittal, P. (2024). A comprehensive survey of deep learning-based lightweight object detection models for edge devices. In *Artificial Intelligence Review* (Vol. 57, Issue 9). Springer Netherlands.
<https://doi.org/10.1007/s10462-024-10877-1>

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Computer Society Conference on*

Computer Vision and Pattern Recognition, 2016-Decem, 779–788.
<https://doi.org/10.1109/CVPR.2016.91>

Wijaya, B. S., Munir, R., & Utama, N. P. (2023). Curating Multimodal Satellite Imagery for Precision Agriculture Datasets with Google Earth Engine. *Proceedings of The International Conference on Data Science and Official Statistics*, 2023(1), 368–381.
<https://doi.org/10.34123/icdsos.v2023i1.399>

Xiao, F., Wang, H., Xu, Y., & Zhang, R. (2023). Fruit Detection and Recognition Based on Deep Learning for Automatic Harvesting: An Overview and Review. *Agronomy*, 13(6).
<https://doi.org/10.3390/agronomy13061625>

