

DETEKSI BUAH APEL SEGAR DAN BUSUK MENGGUNAKAN FRAMEWORK TENSORFLOW LITE DENGAN ARSITEKTUR MODEL EFFICIENTDET LITE 2

M. Rafli Julian¹

Program Studi Teknik Elektro Universitas Teknokrat Indonesia
Bandar Lampung, Indonesia
muhammadraflj@gmail.com¹

Abstrak

Buah apel adalah salah satu buah yang banyak di konsumsi oleh masyarakat Indonesia. Tidak semua daerah di Indonesia cocok untuk tanaman buah apel, tanaman buah apel akan tumbuh dan berproduksi baik pada lahan dengan ketinggian 700 - 1.200 meter dari permukaan laut (dpl), dengan suhu berkisar 16⁰- 25⁰ C. Ada tiga daerah penghasil apel terbesar yaitu pasuruan, malang dan kota batu.

Ketiga daerah ini menjadi penyuplai buah apel terbesar di berbagai daerah di Indonesia, untuk menjaga buah apel tetap segar sampai di tangan konsumen, setelah buah apel dipetik harus segera di distribusikan, namun proses sorting buah apel cukup memakan banyak waktu jika dilakukan secara manual, teknologi *deep learning* mampu mengatasi permasalahan ini. Penelitian ini bertujuan untuk menguji akurasi dari pendeteksian objek buah apel segar dan busuk menggunakan framework Tensorflow Lite dengan arsitektur model *EfficientDet Lite 2*, pendeteksian buah di uji melalui perangkat android secara real time guna menilai kinerja dari model pendeteksian. Hasil uji menunjukkan rata-rata akurasi pendeteksian pada buah apel segar menyentuh angka 91.02% dan 88.07% untuk pendeteksian buah apel busuk, hal ini menunjukkan bahwa model pendeteksi bekerja dengan cukup baik pada perangkat android.

Kata kunci: *Pendeteksi buah apel, Tensorflow Lite, EfficientDet Lite 2, Deep Learning*

1. PENDAHULUAN

Buah apel merupakan salah satu buah yang mengandung karbohidrat dan protein, banyak vitamin yang terkandung dalam buah apel diantaranya vitamin A, vitamin C, vitamin B2, vitamin B1 dan masih banyak lagi vitamin lainnya (Wijaya & Ridwan, 2019, p. 74).

Apel salah satu buah yang banyak di konsumsi oleh masyarakat, di Indonesia sendiri ada tiga daerah penghasil apel terbesar yaitu pasuruan, malang dan kota batu. Ketiga daerah ini menjadi penyuplai buah apel di berbagai daerah di Indonesia.

Buah apel sebelum di distribusikan harus melalui proses sorting terlebih dahulu agar hanya buah apel yang baik dan segar saja yang

di distribusikan, namun proses sorting akan memakan banyak waktu jika di lakukan secara manual sedangkan buah apel harus segera di distribusikan agar pada saat sampai di tangan konsumen buah apel masih dalam keadaan segar.

Dalam penelitian ini dibuat sebuah sistem pendeteksi kesegaran buah apel berdasarkan citra kulit buah apel menggunakan *deep learning*. Proses pelatihan model pendeteksi kesegaran buah apel dilakukan menggunakan *framework Tensorflow Lite*.

Adapun arsitektur model yang digunakan dalam penelitian ini adalah *EfficientDet Lite2*. Dengan menerapkan *deep learning* dalam pendeteksian buah apel segar dan buah apel

busuk, proses sorting buah apel akan lebih cepat dan efisien.

2. TINJAUAN PUSTAKA DAN TEORI

2.1. Tinjauan Pustaka

Novan Wijaya, dkk melakukan klasifikasi jenis buah apel dengan metode *K-Nearest Neighbors* mendapatkan akurasi pengujian sebesar 94% dan untuk apel jenis *grannsymith* mencapai akurasi sebesar 100% (Wijaya & Ridwan, 2019).

Kholiq Prajatama, dkk melakukan pendeteksian buah apel malang manalagi menggunakan *algoritma Naïve Bayes* hanya mendapatkan akurasi pendeteksian yang kurang memuaskan yaitu hanya sebesar 63% (Prajatama et al., 2019).

Julian Sahertian dan Ardi Sanjaya melakukan pendeteksian buah pada pohon dengan menerapkan metode SVM dan fitur tekstur mendapatkan rata-rata akurasi sebesar 76% dengan kesalahan prediksi lebih dari 10% yaitu 24% (Sahertian & Sanjaya, 2017).

Yunita Aulia Hasma dan Widya Silfianti melakukan pendeteksian jerawat dengan implementasi *deep learning* yang menerapkan metode *Faster Regional Convolutional Neural Network* mendapatkan hasil akurasi pendeteksian secara realtime sebesar 71,4% pada siang hari di dalam ruangan, 76,1% pada siang hari di luar ruangan dan sebesar 57,1% pada malam hari di dalam ruangan (Hasma & Silfianti, 2018).

2.2. Deep Learning

Deep Learning atau juga sering dikenal dengan istilah Pembelajaran Struktural Mendalam (*Deep Structured Learning*) adalah salah satu cabang dari ilmu pembelajaran mesin yang terdiri dari algoritma pemodelan abstraksi data tingkat tinggi menggunakan sekumpulan fungsi transformasi non-linear yang ditata berlapis-lapis dan mendalam (Aningtiyas et al., 2020, p.423).

Deep learning merupakan salah satu bidang dari *Machine Learning* yang menerapkan jaringan syaraf tiruan untuk

implementasi permasalahan dengan dataset yang besar (Wantania et al., 2020, p.103).

2.3. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) merupakan salah satu metode *Deep learning* yang dapat digunakan untuk mendeteksi dan mengenali sebuah objek pada sebuah gambar atau sebuah citra digital (Wantania et al., 2020, p.103).

Proses utama yang mendasari arsitektur jaringan CNN adalah *Convolution layer* yang mana merupakan bagian dari tahap pada arsitektur CNN yang berperan dalam melakukan operasi konvolusi pada *output* dari layer sebelumnya (Prasetya et al., 2021, p.38).

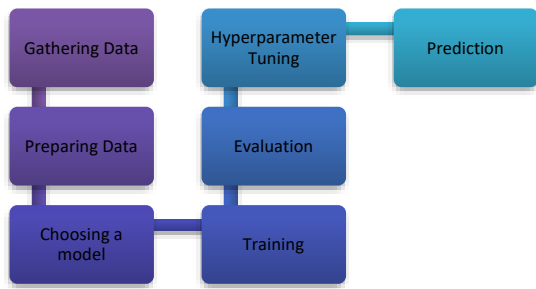
CNN merupakan algoritma yang paling banyak digunakan untuk implementasi deteksi objek (Rahma et al., 2021, p.213). salah satu alasannya adalah CNN didukung oleh *framework Tensorflow* buatan Google dan juga tingginya kecepatan dan akurasi yang dihasilkan.

2.4. TensorFlow Lite

TensorFlow Lite adalah *framework deep learning open source* untuk inferensi pada perangkat (*TensorFlow Developers*). *TensorFlow Lite* dirancang khusus untuk perangkat *mobile* dan IoT seperti Android, ios, Raspberry pi dan banyak lagi, Tensorflow Lite memiliki latensi yang rendah dan juga ukuran yg kecil sehingga sangat cocok untuk di implementasikan di perangkat mobile dan IoT.

3. METODE PENELITIAN

Dalam perancangan pendeteksi buah apel segar dan buah apel busuk digunakan bahasa pemrograman python dan java, data dilatih menggunakan *framework Tensorflow Lite* dengan arsitektur *model EfficientDet Lite2*. Penelitian ini dilaksanakan secara bertahap mengacu pada alur 7 Steps of *Machine Learning* yang dapat dilihat pada Gambar 1.



3.1. Gathering Data

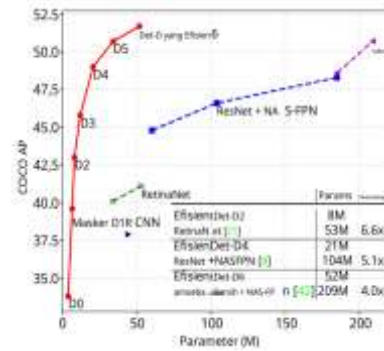
Tahap ini merupakan tahap pengumpulan data, tahap ini sangat penting karena kualitas dari data akan berpengaruh secara langsung terhadap model pendeteksian objek yang akan di buat. Pada penelitian ini terdapat 276 gambar yang terdiri dari buah apel segar dan buah apel busuk yang didapatkan dari Google.

3.2. Preparing Data

Tahap ini merupakan tahap persiapan data, setelah data gambar di kumpulkan selanjutnya di lakukan pelabelan pada gambar menggunakan tools yang ada pada website roboflow dan kemudian dilakukan ekspor dataset berupa gambar yang telah di labeli tadi dengan menerapkan data preprocessing dan augmentations dengan pembagian dataset yaitu sebesar 60% untuk data latih, 20% untuk data validasi dan 20% untuk data uji.

3.3. Choosing a Model

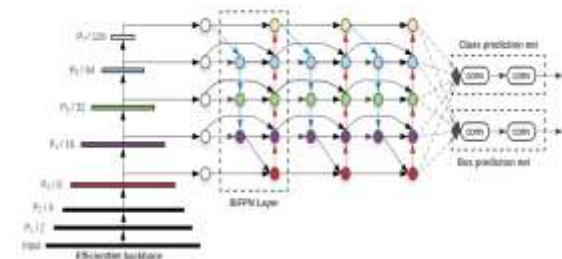
Langkah selanjutnya adalah pemilihan model, penelitian ini menggunakan arsitektur *model EfficientDet Lite 2*, model ini merupakan turunan dari model *EfficientDet* yang secara konsisten mencapai akurasi dan efisiensi yang lebih baik dari model-model sebelumnya (Tan et al., 2020, p.10788). *EfficientDet* memiliki delapan model yaitu D0-D7 yang mana akurasi dan kompleksitas waktu pada model meningkat seiring dengan ukuran model (Song et al., 2021). Perbandingan ukuran model dapat dilihat pada Gambar 2.



Gambar 2. Perbandingan Ukuran Model

Sumber: (Tan et al., 2020)

Selanjutnya pada Gambar 3 menunjukkan gambaran dari arsitektur *model EfficientDet*.



Gambar 3. Arsitektur *Model EfficientDet*

Sumber: (Tan et al., 2020)

Model EfficientDet menggunakan model *EfficientNet* sebagai jaringan tulang punggung, BiFPN sebagai jaringan fitur, dan jaringan prediksi kelas/kotak bersama. Lapisan BiFPN dan lapisan jaring kelas/kotak diulang beberapa kali berdasarkan batasan sumber daya yang berbeda (Tan et al., 2020, p.10785).

3.4. Training

Selanjutnya masuk ke proses pelatihan, pelatihan dilakukan di *Colab Notebooks* menggunakan *framework Tensorflow Lite*, *batch size* di atur ke 8 dan menggunakan 100 *epoch* untuk proses pelatihan, dari konfigurasi ini pemrosesan per *epoch* rata-rata memakan waktu sebanyak 24 detik, sehingga proses pelatihan model keseluruhan memakan waktu kurang lebih sekitar 40 menit.

3.5. Evaluation

Setelah pelatihan selesai, selanjutnya dilakukan evaluasi menggunakan data validasi yang telah dipisahkan sebelumnya untuk melihat apakah modelnya sudah bagus, pada tahap evaluasi ini model pendeteksi objek yang telah dilatih akan diuji menggunakan

data gambar yang belum pernah dilihat sebelumnya, di tahap ini kita dapat melihat kinerja model sebelum di terjunksan langsung di dunia nyata.

3.6. Hyperparameter Tuning

Hyperparameter Tuning dilakukan untuk lebih meningkatkan lagi akurasi dari model setelah dilakukan pelatihan dengan menyetel beberapa parameter, pada penelitian ini tidak di lakukan penyetelan ulang pada parameter karena akurasi yang di dapatkan usai pelatihan di rasa sudah cukup baik.

3.7. Prediction

Setelah melalui berbagai macam proses, tahap ini adalah inti dari semua proses yang telah di lalui, di mana nilai pembelajaran mesin di realisasikan untuk mendeteksi buah apel segar dan buah apel busuk, model yang telah selesai dilatih kemudian di *deploy* ke perangkat seluler android untuk secara nyata melihat akurasi dari prediksi model saat diterapkan pada perangkat dan agar dapat digunakan oleh masyarakat.

4. HASIL DAN PEMBAHASAN

Setelah melalui proses pelatihan dan penyetelan ulang parameter, maka di ambil model pendeteksi yang menghasilkan kinerja pendeteksian paling akurat, model ini kemudian di terapkan di sebuah aplikasi android agar dapat di uji coba pada perangkat seluler untuk menilai kinerja dari model yang telah di latih secara langsung, berikut adalah tangkapan layar dari model yang telah di terapkan pada aplikasi android dapat dilihat pada Gambar 4.



Gambar 4. Tangkapan Layar Aplikasi Android Pendeteksi Buah Apel Segar dan Buah Apel Busuk.

Uji coba pendeteksian di lakukan menggunakan 20 gambar apel segar dan 20 gambar apel busuk yang di dapatkan dari google, hasil pengujian dapat dilihat pada Tabel 1.

Apel Segar		
Terdeteksi Benar	Terdeteksi Salah	Rata-rata Akurasi
20	0	91.02%
Apel Busuk		
Terdeteksi Benar	Terdeteksi Salah	Rata-rata Akurasi
20	0	88.07%

Tabel 1. Tabel Hasil Pendeteksian berdasarkan data pada Tabel1. diketahui bahwa model mampu mendeteksi buah apel segar dan buah apel busuk dengan sangat baik yaitu dengan 0% kesalahan pendeteksian.

Pendeteksian terhadap buah apel segar mendapatkan Rata-rata tingkat kepercayaan pendeteksian sebesar 91.02% dengan hasil kepercayaan pendeteksian paling tinggi sebesar 95.31% dan paling rendah yaitu 87.11%. Hasil rata-rata tingkat kepercayaan pendeteksian pada buah apel busuk sedikit

lebih rendah dari hasil pendeteksian buah apel segar yaitu sebesar 88.07% dengan hasil kepercayaan pendeteksian tertinggi sebesar 95.14% dan paling rendah yaitu 75.39%.

5. KESIMPULAN DAN SARAN

Berdasarkan penelitian yang telah dilakukan tentang pendeteksian buah apel segar dan buah apel yang sudah tidak segar atau busuk, di dapatkan 0% kesalahan pada pendeteksian dengan rata-rata akurasi pendeteksian sebesar 91.02% untuk apel segar dan 88.07% untuk apel busuk.

Dengan dataset yang bisa terbilang sedikit yaitu hanya menggunakan 276 gambar dari Google dan hasil akurasi pada pendeteksian, dapat disimpulkan arsitektur *model EfficientDet Lite 2* yang di padukan oleh *framework Tensorflow Lite* mampu bekerja dengan baik. Untuk lebih meningkatkan lagi akurasi dan kinerja dari model, selanjutnya akan dilakukan pelatihan ulang model dengan dataset yang lebih banyak lagi dan dengan pelatihan yang lebih lama lagi untuk menghasilkan kinerja yang lebih optimal.

DAFTAR PUSTAKA

- Aningtiyas, P. R., Sumin, A., & Wirawan, S. (2020). Pembuatan Aplikasi Deteksi Objek Menggunakan TensorFlow Object Detection API dengan Memanfaatkan SSD MobileNet V2 Sebagai Model Pra-Terlatih. *jurnal ilmiah KOMPUTASI*, 19, 421-430.
- Hasma, Y. A., & Silfianti, W. (2018). Implementasi Deep Learning Menggunakan Framework Tensorflow dengan Metode Faster Regional Convolutional Neural Network untuk Pendeteksian Jerawat. *Jurnal Ilmiah Teknologi dan Rekayasa*, 23, 89-102.
- Prajatama, K., Nugroho, F. E., Sentosa, A. F., Fauziah, S., & Hartanto, A. D. (2019). Deteksi Kualitas Buah Apel Malang Manalagi Menggunakan Algoritma Naive Bayes. *Jurnal Sistem Informasi dan Teknologi Informasi*, 8, 32-38.
- Prasetya, A., Ihsanto, E., & Dani, A. W. (2021). Rancang Bangun Pendeteksi Wajah Bermasker Dan Tidak Bermasker Dalam Absensi Di Masa Pandemi COVID-19 Menggunakan Convolutional Neural Network. *Jurnal Teknologi Elektro*, 12, 37-44.
- Rahma, L., Syaputra, H., Mirza, A. H., & Purnamasari, S. D. (2021). Objek Deteksi Makanan Khas Palembang Menggunakan Algoritma YOLO (You Only Look Once). *Jurnal Nasional Ilmu Komputer*, 2, 213-232.
- Sahertian, J., & Sanjaya, A. (2017). Deteksi Buah Pada Pohon Menggunakan Metode SVM dan Fitur Tekstur. Yogyakarta.
- Song, S., Jing, J., Huang, Y., & Shi, M. (2021). EfficientDet for fabric defect detection based on edge computing. *Journal of Engineered Fibers and Fabrics*, 1-13.
- Tan, M., Pang, R., & Le, Q. V. (2020). EfficientDet: Scalable and Efficient Object Detection. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- TensorFlow Developers. (n.d.). *Deploy machine learning models on mobile and IoT devices*. Retrieved January 6, 2022, from [Tensorflow.org: https://www.tensorflow.org/lite](https://www.tensorflow.org/lite)
- Wantania, B. B., Sompie, S. R., & Kambey, F. D. (2020). Penerapan Pendeteksian Manusia Dan Objek Dalam Keranjang Belanja Pada Antrian Di Kasir. *Jurnal Teknik Informatika*, 15, 101-108.
- Wijaya, N., & Ridwan, A. (2019). Klasifikasi Jenis Buah Apel Dengan Metode K-Nearest Neighbors. *SISFOKOM*, VIII, 74-78.