**JTRISTE**, Vol.13, No.2, Oktober 2026, pp. 10~17
ISSN: **2355-3677 | e-ISSN : 2460-8548**

# Identifikasi Barang Secara Otomatis Pada E-Thrift Warshop Menggunakan Deep Learning

**Christofer Reynaldy Wijaya[1], Mohammad Fajar[2*], Baizul Zaman[3]**
[1,2,3]Teknik Informatika, STMIK Kharisma
e-mail: [1]christofer_21@kharisma.ac.id, [2]fajar@kharisma.ac.id, [3]baizul@kharisma.ac.id

***Abstrak***

*Penelitian ini bertujuan untuk mengimplementasikan teknologi deep learning dalam pengenalan barang pada aplikasi e-thrift Warshop, yang merupakan platform jual beli barang bekas. Masalah utama yang dihadapi adalah identifikasi barang yang sering kali subjektif dan tidak akurat dalam proses manual. Dengan menggunakan teknologi machine learning berbasis Convolutional Neural Networks (CNN) dan MobileNet V1, penelitian ini mencoba mengatasi tantangan tersebut. Metode yang digunakan meliputi pengunggahan gambar oleh pengguna, pemrosesan gambar oleh model MobileNet yang di-host di jsDelivr, dan penyimpanan hasil klasifikasi di Airtable untuk ditampilkan di situs web Warshop. Hasil penelitian menunjukkan bahwa model yang diterapkan berhasil mencapai akurasi sebesar 75.5%, presisi 100%, dan recall 77.5%.Kesimpulan dari penelitian ini adalah bahwa teknologi deep learning dapat meningkatkan keakuratan dalam pengenalan barang, meskipun ada kebutuhan untukmemperluas dataset dan memperbarui model secara berkala. Fitur feedback dari pengguna juga disarankan untuk meningkatkan akurasi dan kepuasan pengguna.*

***Kata kunci:*** *Deep Learning, MobileNet, e-Thrift, TensorFlow, CNN*

***Abstract***

*This study aimed to implement deep learning technology for item recognition within the e-thrift Warshop application, a platform for buying and selling used items. The primary issue was the subjective and often inaccurate identification of items through manual processes. The research utilised machine learning technologies, specifically Convolutional Neural Networks (CNN) and MobileNet V1, to address this challenge. The methodology involved users uploading images, processing these images using the MobileNet model hosted on jsDelivr, and storing classification results in Airtable for display on the Warshop website. The results demonstrate that the model achieves an accuracy of 75.5%, with a precision of 100% and a recall of 77.5%. The conclusion is that deep learning technology effectively enhances item recognition accuracy. However, it is essential to expand the dataset and regularly update the model. Additionally, incorporating user feedback features is recommended to further improve accuracy and user satisfaction*

***Keywords:*** *Deep Learning, MobileNet, e-Thrift, TensorFlow, CNN*

## 1. Pendahuluan

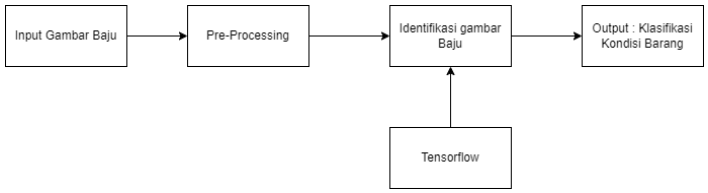
Aplikasi e-thrift Warshop merupakan platform berbasis web yang memfasilitasi jual beli barang bekas dan koleksi unik. Aplikasi ini telah menjadi pilihan favorit bagi konsumen yang mencari barang-barang dengan harga terjangkau. Namun, salah satu tantangan utama yang dihadapi oleh penjual dan pembeli adalah pengenalan terhadap barang yang ditawarkan. Dalam proses manual, identifikasi barang sering kali subjektif karena faktor-faktor visual dan deskripsi yang tidak selalu akurat. Oleh karena itu, dibutuhkan teknologi yang dapat secara otomatis mengenali barang-barang berdasarkan ciri-ciri fisiknya.

[*] Corresponding author : Mohammad Fajar (fajar@kharisma.ac.id)

***Diterima :*** *Agustus, 2025* ***Disetujui :*** *Oktober, 2025* ***Dipublikasikan :*** *Oktober, 2026*

Untuk mengatasi permasalahan ini, teknologi machine learning dapat digunakan untuk mengenali barang secara otomatis dan akurat. Deep learning, yang merupakan cabang dari machine learning, memberikan kemampuan yang lebih canggih untuk mempelajari pola dan karakteristik dari berbagai barang melalui model neural networks[1]. Salah satu pendekatan yang umum digunakan adalah penggunaan framework TensorFlow, yang memanfaatkan Convolutional Neural Networks (CNN) untuk klasifikasi gambar[2]

TensorFlow adalah kerangka kerja komputasi yang digunakan untuk membangun model pembelajaran mesin. Bersama dengan berbagai toolkitnya, TensorFlow memungkinkan Anda membuat model pada tingkat abstraksi yang Anda suka dan memungkinkan grafik berjalan pada berbagai platform hardware, seperti CPU, GPU, dan TPU[3]. Tensorflow, yang pertama kali dikembangkan oleh Tim Brain Google pada tahun 2015, banyak digunakan oleh perusahaan besar untuk mengembangkan aplikasi AI seperti penyematan kata, klasifikasi gambar, dan pengembangan chatbot[4].




Gambar 1. Arsitektur Tensorflow(Feraco, Bonfitto, Amati, & Tonoli, 2022)

Beberapa karakteristik utama tensorflow adalah sebagai berikut[5]:

1. Definisi, optimalisasi, dan pemrosesan perhitungan dilakukan secara efisien dalam bentuk ekspresi matematis berisi array multidimensi bentuk, juga dikenal sebagai tensorflow;
2. Karakteristik dan sifat pembelajaran mesin pemrograman yang mendukung perhitunganjaringan saraf dalam. Tensorflow menawarkan informasi yang digunakan dalam proses melakukan tugas perhitungan ekspresi matematika. GPU juga transparan dan memiliki fitur optimalisasi dan otomatisasi.
3. Unit Pemrosesan Grafis (GPU) yang transparan. Ini juga memiliki kemampuan untuk optimalisasi dan otomatisasi manajemen memori sesuai dengan jumlah data yang digunakan. Tensorflow dapat menulis kode dan bekerja dengan baik dengan unit pemrosesan pusatatau GPU. Lebih khusus lagi, dia dapat mengetahui bagian mana dari perhitungan yang harus dan harus diteruskan ke GPU.
4. Akurasi yang baik dihasilkan dari kumpulan data yang besar dan skalabilitas pemrosesan data yang tinggi di setiap aspeknya.

CNN adalah perangkat multilapis yang memiliki setiap neuron terhubung ke neuron di lapisan berikutnya. Namun, CNN memiliki kemampuan untuk menemukan pola hierarkis dalam data dan mengumpulkan piksel yang lebih kompleks dari piksel yang lebih kecil dan lebih sederhana. Akibatnya, CNN sangat baik dalam hal keterhubungan dan kompleksitas piksel [6].CNN pada dasarnya merupakan arsitektur ANN dengan fitur unik, dapat dilatih, dan terdiri dari beberapa tahapan yang terdiri dari masukan (input) dan keluaran (output). Array terdiri dari masukan dan keluaran pada setiap tahapan, yang menghasilkan feature map[7]. Salah satu metode machine learning yang dapat digunakan untuk klasifikasi gambar objek adalah Convolutional Neural Network (CNN). Hal ini karena CNN dapat meniru kemampuan manusia dalam sistem pengenalan gambar pada visual cortex manusia, sehingga dapat mengolah gambar[8].

MobileNet adalah model berbasis arsitektur sederhana yang membangun deep neural networks [9]dengan menggunakan pergeseran kedalaman yang dapat dipecahkan. Pergeseran kedalaman terdiri dari dua lapisan, di mana pergeseran kedalaman digunakan untuk menerapkan filter pada gambar input, dan pergeseran kedalaman digunakan untuk menggabungkan hasil dari pergeseran kedalaman sehingga mengurangi beban komputasi[10].

Dengan menggunakan teknologi ini, aplikasi e-thrift Warshop dapat memberikan pengalaman berbelanja yang lebih transparan dan andal[11].sekaligus meningkatkan kepercayaan antara penjual dan pembeli. Pengenalan mesin terhadap barang memungkinkan sistem untuk secara otomatis mengklasifikasikan jenis barang, sehingga memberikan informasi yang lebih objektif dan akurat kepada pengguna. Dalam penelitian ini, penggunaan TensorFlow, CNN, dan MobileNet v1 akan dieksplorasi untuk mendukung pengembangan sistem yang mampu mengenali barang-barang pada platform e-thrift Warshop.

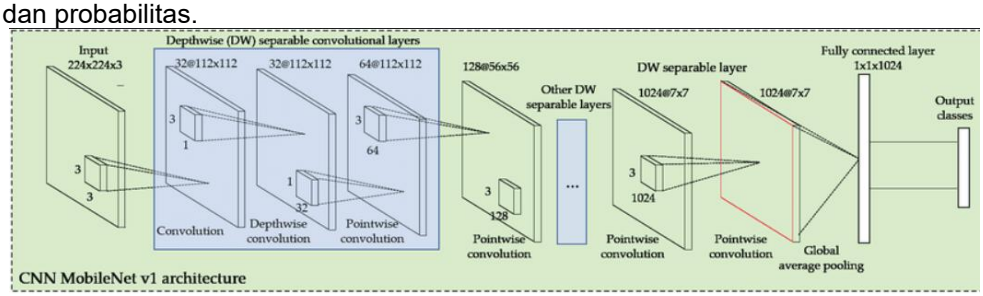
## 2. Metode Penelitian

Pada penelitian ini, metode pengumpulan data yang digunakan adalah observasi. Ini berarti mengumpulkan data dalam bentuk gambar dengan ekstensi jpeg dan jpg dari akun media sosial penjual thrift yang bekerja sama dengan aplikasi WarShop. Gambar berikut menunjukkan jenis barang yang akan diklasifikasikan. Kelas Unknown terdiri dari gambar yang berada diluar kaos, celana, dan Sepatu

Tabel 1 Pembagian data uji

| Kategori | Jumlah data Uji |
|---|---|
| Kaos | 10 |
| Celana | 10 |
| Sepatu | 10 |

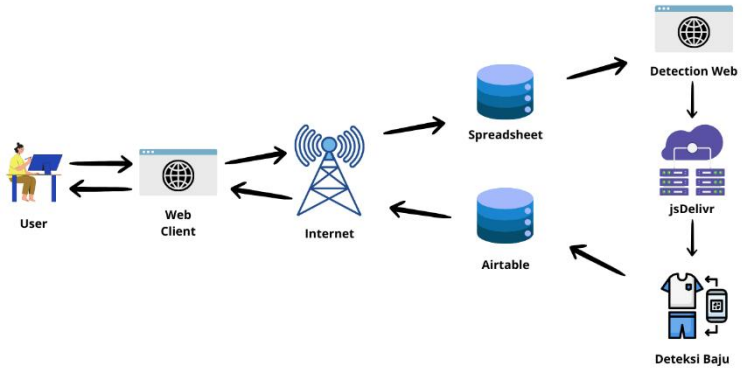
Terdapat beberapa tahapan dalam metode yang diajukan. Tahap pertama adalah pengumpulan data uji, yaitu gambar kaos, celana, dan sepatu. Pada tahap ini, pre-processing tidak digunakan karena penelitian memanfaatkan dataset yang sudah ada. Selanjutnya, setelah gambar data uji diinput, klasifikasi gambar akan dilakukan menggunakan metode MobileNet V1. Setelah gambar melalui proses MobileNet V1, akan dihasilkan output berupa prediksi dan probabilitas.




Gambar 2. Arsitektur MobileNet V1(Feraco, Bonfitto, Amati, & Tonoli, 2022)

## 3. Hasil Dan Pembahasan

### 3.1. Arsitektur Sistem




Gambar 3. Arsitektur Sistem

Arsitektur sistem klasifikasi barang dalam aplikasi Warshop dimulai dengan pengguna yang mengakses situs web Warshop untuk melakukan penjualan barang. Lalu Pengguna mengunggah gambar produk yang ingin diklasifikasi. Setelah gambar diunggah, gambar tersebut secara otomatis disimpan ke dalam spreadsheet sebagai penyimpanan awal. Gambar dari spreadsheet kemudian diakses oleh web pendeteksian, yang memanfaatkan model MobileNet yang sudah dilatih menggunakan dataset yang sudah ada dan di-hosting di jsDelivr. Proses pendeteksian dijalankan oleh web pendeteksian untuk mengevaluasi jenis produk berdasarkan gambar yang telah diunggah. Skor hasil klasifikasi kemudian disimpan dalam Airtable, yang berfungsi sebagai database untuk menyimpan informasi skor tersebut. Skor yang disimpan ini akhirnya ditampilkan kembali di situs web Warshop, memungkinkan pengguna untuk melihat hasil klasifikasi secara langsung. Sistem ini menggabungkan beberapa komponen web dan basis data untuk menyediakan layanan deteksi yang efisien dan terintegrasi bagi pengguna.




Gambar 4. Klasifikasi Data Uji

### 3.2. Pengujian Sistem

Pengujian Sistem ini dilakukan dengan metode confusion matrix yang digunakan untuk menguji akurasi, error, recall, dan precision. Tujuannya adalah untuk memahami sejauh mana model mampu mengklasifikasikan data dengan benar secara keseluruhan, mengidentifikasi area di mana kesalahan paling sering terjadi, dan memberikan gambaran tentang kinerja model dalam menemukan kasus positif dan negatif. Berikut adalah rumus yang digunakan untuk menghitung akurasi, precision,dan recall menggunakan confusion matrix.

Tabel 2. Tabel Akurasi

| | Actual Positif | Actual Negative |
|---|---|---|
| Predicted Positif | TP | FP |
| Predicted Negative | FN | TN |

$$Akurasi = \frac{TP}{Jumlah\ Data} \times 100\%$$

$$Precision = \frac{TP}{TP + FP} \times 100\%$$

$$Recall = \frac{TP}{TP + FN} \times 100\%$$

Keterangan :

1. TP adalah ketika model benar-benar benar mengklasifikasikan instance positif.
2. FP adalah ketika model salah memprediksi positif.
3. FN adalah ketika model salah memprediksi negative
4. TN adalah ketika model benar-benar benar mengklasifikasikan instance negatif.

Pengujian dilakukan terhadap gambar yang sudah disiapkan yaitu Kaos, celana, dan Sepatu masing masing 10 gambar.

Tabel 3 Tabel Hasil Pengujian

| Prediksi / Aktual | Kaos | Celana | Sepatu | Unknown |
|---|---|---|---|---|
| Kaos | 7 | 0 | 0 | 0 |
| Celana | 0 | 8 | 0 | 0 |
| Sepatu | 0 | 0 | 6 | 0 |
| Unknown | 3 | 2 | 4 | 10 |

Tabel 4 Tabel Hasil Pengujian

| Prediksi / Aktual | Kaos | Celana | Sepatu | Unknown |
|---|---|---|---|---|
| TP | 7 | 8 | 6 | 10 |
| FP | 0 | 0 | 0 | 0 |
| FN | 3 | 2 | 4 | 0 |

Pengujian Akurasi adalah pengujian seberapa sering model membuat prediksi yang benar. Ini adalah rasio dari jumlah prediksi benar (baik positif maupun negatif) terhadap total jumlah prediksi.

$$Akurasi = \frac{TP}{Jumlah\ Data} \times 100\%$$

$$Akurasi = \frac{31}{40} \times 100\%$$

$$Akurasi = 75\%$$

Pengujian Presisi dilakukan untuk mengukur seberapa banyak dari prediksi positif yang benar-benar positif. Ini adalah rasio dari jumlah prediksi positif yang benar terhadap total jumlah prediksi positif yang dibuat oleh model.

$$Presisi = \frac{TP}{TP + FP}$$

disetiap itemnya dengan hasil sebagai berikut :

Tabel 5 Tabel Presisi

| Prediksi / Aktual | Kaos | Celana | Sepatu | Unknown |
|---|---|---|---|---|
| TP | 7 | 8 | 6 | 10 |
| FP | 0 | 0 | 0 | 0 |
| Presisi | 1 | 1 | 1 | 1 |

$$Presisi\ Keseluruhan = \frac{Total\ \ hasil\ Presisi}{Jumlah\ Presisi} \times 100\%$$

$$Presisi\ Keseluruhan = \frac{4}{4} \times 100\%$$

$$Presisi\ Keseluruhan = 100\%$$

Pengujian Recall ini dilakukan mengukur seberapa baik model menemukan semua contoh positif. Ini adalah rasio dari jumlah prediksi positif yang benar terhadap total jumlah kasus positif yang sebenarnya.

$$Recall = \frac{TP}{TP + FN}$$

disetiap itemnya dengan hasil sebagai berikut :

Tabel 6 Tabel Recall

| Prediksi / Aktual | Kaos | Celana | Sepatu | Unknown |
|---|---|---|---|---|
| TP | 7 | 8 | 6 | 10 |
| FN | 3 | 2 | 4 | 0 |
| Recall | 0,7 | 0,8 | 0,6 | 1 |

$$Recall\ Keseluruhan\ = \frac{Total\ \ hasil\ Presisi}{Jumlah\ Presisi} \times 100\%$$

$$Recall\ Keseluruhan\ = \frac{3{,}1}{4} \times 100\%$$

$$Recall\ Keseluruhan\ = 77{,}5\%$$

## 4. Kesimpulan

Penelitian ini menunjukkan bahwa implementasi deep learning untuk pengenalan barang pada aplikasi e-thrift Warshop mencapai akurasi 75.5%, dengan presisi 100% dan recall 77.5% menggunakan dataset MobileNet yang sudah ada. Hasil ini menandakan bahwa model deep learning cukup efektif dalam mengidentifikasi barang, dengan presisi yang sangat baik dan recall yang cukup memadai. Namun, untuk meningkatkan akurasi secara keseluruhan, disarankan agar dataset diperluas dengan mencakup lebih banyak variasi barang dan kondisi yang berbeda. Penambahan data yang beragam akan membantu model dalam memahami dan mengklasifikasikan barang dengan lebih akurat. Untuk pengembang aplikasi, penting untuk terus memperbarui dan melatih model dengan data terbaru serta melakukan evaluasi rutin. Selain itu, tim pengelola aplikasi e-thrift harus mempertimbangkan implementasi fitur feedback dari pengguna untuk memperbaiki model sesuai kebutuhan pengguna, yang dapat meningkatkan akurasi dan kepuasan pengguna aplikasi.

## Daftar Pustaka

[1] R. Darma Nurfita and G. Ariyanto, “Implementasi Deep Learning Berbasis Tensorflow Untuk Pengenalan Sidik Jari.” [Online]. Available: http://bias.csr.unibo.it/fvc2004/databases.asp

[2] Y. A. Hasma and W. Silfianti, “Implementasi Deep Learning Menggunakan Framework Tensorflow Dengan Metode Faster Regional Convolutional Neural Network Untuk Pendeteksian Jerawat,” *Jurnal Ilmiah Teknologi dan Rekayasa*, vol. 23, no. 2, pp. 89–102, 2018, doi: 10.35760/tr.2018.v23i2.2459.

[3] N. Hikmatia A.E and M. Ihsan Zul, “Aplikasi Penerjemah Bahasa Isyarat Indonesia menjadi Suara berbasis Android menggunakan Tensorflow,” *Jurnal Komputer Terapan*, no. Vol. 7 No. 1 (2021), pp. 74–83, Jun. 2021, doi: 10.35143/jkt.v7i1.4629.

[4] T. Patrícia, P. Padilha, and L. E. Alves De Lucena, “A Systematic Review About Use of TensorFlow for Image Classification and Word Embedding in the Brazilian Context.” [Online]. Available: https://scholar.google.com/

[5] Chairil Anwar, “Deteksi Objek Berbasis Web Menggunakan Tensorflow Js dan Coco Dataset pada Framework React Js,” *JurnalNasionalKomputasi dan Teknologi Informasi*, vol. 5, pp. 1088–1015, Dec. 2022.

[6] E. Rasywir, R. Sinaga, Y. Pratama, U. Dinamika, and B. Jambi, “Analisis dan Implementasi Diagnosis Penyakit Sawit dengan Metode Convolutional Neural Network (CNN),” vol. 22, no. 2, 2020, doi: 10.31294/p.v21i2.

[7] N. Nufus *et al.*, “Sistem Pendeteksi Pejalan Kaki Di Lingkungan Terbatas Berbasis SSD MobileNet V2 Dengan Menggunakan Gambar 360° Ternormalisasi,” *Prosiding Seminar Nasional Sains Teknologi dan Inovasi Indonesia (SENASTINDO)*, vol. 3, pp. 123–134, Dec. 2021, doi: 10.54706/senastindo.v3.2021.123.

[8] A. Hayyu Gustsa and G. Setyo Permadi, “Sistem Deteksi Bahasa Isyarat Secara Realtime Dengan Tensorflow Object Detection dan Python Menggunakan Metode Convolutional Neural Network.”

[9] Z. Syahputra, “Penerapan SSD-Mobilenet Dalam Identifikasi Jenis Buah Apel,” *Indonesian Journal of Education And Computer Science*, vol. 1, no. 1, 2023.

[10] A. G. Howard *et al.*, “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.”

[11] Febriyanto Muhammad, “Legalitas Jual Beli Pakaian Bekas Impor (Thrift) Secara Online Di Aplikasi Tiktok,” Universitas Islam Negeri, Jakarta, 2023