

# Identifikasi Foto Fashion Dengan Menggunakan *Convolutional Neural Network (CNN)*

## *Identify Fashion Images Using Convolutional Neural Network (CNN)*

<sup>1</sup>Green Arther Sandag, <sup>2</sup>Jacqueline Waworundeng

Universitas Klabat, Jln. Arnold Mononutu, Airmadidi – Minahasa Utara

<sup>1,2</sup>Program Studi Informatika Fakultas Ilmu Komputer

e-mail: <sup>1</sup>greensandag@unklab.ac.id, <sup>2</sup>jacqueline.morlav@unklab.ac.id

### **Abstrak**

*Perkembangan teknologi sekarang ini berdampak pada banyak hal, salah satunya ialah pada bidang fashion. Penggunaan Artificial Intelligence dan juga deep learning dapat dimanfaatkan dalam bidang fashion, salah satu contohnya adalah pengenalan objek clothing. Pada penelitian ini, peneliti mengidentifikasi mode pakaian dengan menggunakan metode Convolutional Neural Network (CNN), dan library Tensorflow, serta menggunakan Fashion MNIST dataset untuk menguji kemampuan CNN model. Hasil yang didapatkan saat pengujian dengan menggunakan berbagai convolutional layer sekuensial yang kompleks, didapati dua hasil yang sedikit berbeda. Pengujian pada model pertama, terjadi overfitting, sehingga menghasilkan akurasi sebesar 91%. Pada pengujian kedua, dengan penambahan Dropout layers, menghasilkan akurasi yang lebih baik, yaitu sebesar 93%. Melihat dari hasil yang didapatkan, penggunaan CNN dalam mengidentifikasi mode pakaian cukup sesuai karena dapat mencapai akurasi hingga 93%.*

**Kata kunci** — Deep Learning, Pengenalan objek , Convolutional Neural Network (CNN), Tensorflow, Fashion MNIST

### **Abstract**

*The development of technology nowadays impacts on many things, one of which is the fashion industry. In the area of fashion, the application of Artificial Intelligence and deep learning can also be used, such as the implementation of classification of clothing items. In this paper, using the Convolutional Neural Network (CNN) framework and the tensorflow library, we proposed a model for identifying clothing fashions and used the Fashion-MNIST dataset to test the ability of the CNN model. The results obtained when testing using various complex sequential convolutional layers, found two slightly different results. On the first model, overfitting occurs, resulting in an accuracy of 91%. In the second tester, with the addition of the Dropout layer resulted in better accuracy, namely by 93%. Judging from the results obtained, the use of CNN in clothing mode is quite suitable because it can achieve up to 93% accuracy.*

**Keywords** — Deep Learning, Object Classification , Convolutional Neural Network (CNN), Tensorflow, Fashion MNIST

## 1. PENDAHULUAN

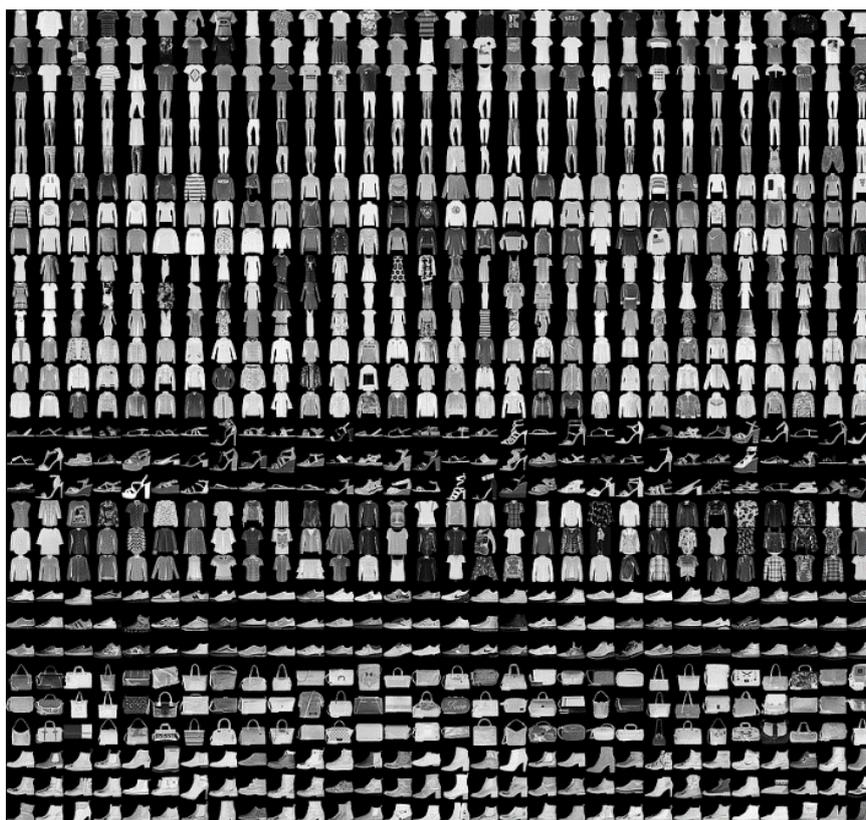
Perkembangan zaman yang melekat dengan kemajuan teknologi berdampak pada berbagai bidang industri, salah satunya adalah industri *fashion*. Peranan *Artificial Intelligence* (AI) dan *deep learning* akhir-akhir ini sudah mulai dapat dirasakan manfaatnya. Salah satu contohnya adalah pengenalan objek *clothing*. Variasi jenis brand yang beragam membuat algoritma pengenalan objek dalam industri *fashion* mulai dikembangkan dengan gencar, dengan berbagai tahapan yang dilakukan mulai dari pengenalan, hingga klasifikasi objek. Bidang *fashion* merupakan salah satu bidang yang menggunakan gambar yang cukup banyak [1]. Ada banyak ciri khas tertentu dalam mengkategorikan setiap *fashion* yang ada, sehingga membutuhkan klasifikasi untuk dapat mengidentifikasi dengan baik. Dengan mengklasifikasi gambar dapat membantu industri *fashion* untuk digunakan dalam klasifikasi pencarian kategori dan jenis dari berbagai *fashion*, serta dapat membantu untuk pemasok dan konsumen yang tertarik untuk membuat busana mereka sendiri lewat data gambar *fashion* yang tersedia secara online [2], [3]. Dalam *computer vision*, pengenalan objek atau klasifikasi objek adalah salah satu aplikasi yang populer untuk mengklasifikasikan gambar. Tujuan dari klasifikasi objek adalah untuk mengekstrak fitur dari gambar dan kemudian akan diklasifikasikan ke dalam setiap kategori kelas yang berbeda dengan menggunakan salah satu metode klasifikasi [4]. Salah satu permasalahan mendasar yang ada pada *computer vision* adalah akurasi untuk mengklasifikasikan objek atau gambar. Klasifikasi objek adalah bagian penting dari *computer vision*, seperti pengambilan gambar dan mobil tanpa pengemudi, dan lain sebagainya. Dalam Algoritma komputer, mengidentifikasi entitas sebuah gambar adalah hal yang menantang, karena akan mengidentifikasi sesuai akurasi manusia [5].

Pembangunan dataset yang besar dilakukan guna untuk mengenali setiap objek gambar dalam *fashion*, dan dituntut untuk mengklasifikasikannya dengan benar [6]. Salah satu metode *deep learning* yang dinilai cukup sukses dalam pengenalan objek, adalah *Convolutional Neural Network* (CNN) [7]. Metode CNN akan lebih jelas lagi dibahas pada bagian-bagian berikutnya. Untuk penelitian tentang klasifikasi *Fashion-MNIST* ini, peneliti menggunakan *Convolutional Neural Network* (CNN). Metode *Convolutional Neural Network* (CNN) yang dapat mengklasifikasikan objek berdasarkan karakteristiknya. Oleh karena itu, penerapan metode yang dijalankan dapat dianalisis berdasarkan karakteristik tertentu dari objek yang akan dideteksi atau dikenali. CNN adalah salah satu jaringan neural yang paling populer dan sering digunakan. CNN memiliki beberapa lapisan untuk menangkap fitur gambar yaitu, *convolutional layer*, *pooling layer*, dan *fully connected layer*. Tujuan dari lapisan ini selain untuk menangkap fitur gambar, juga digunakan untuk mempertahankan aspek terpenting dengan mengurangi peta fitur. CNN memberikan akurasi yang lebih baik dan ada peningkatan kinerja sistem pada aplikasi-aplikasi seperti klasifikasi gambar atau *computer vision* [8].

Kecerdasan buatan yang semakin hari semakin berkembang, membuat suatu masalah-masalah yang awalnya sulit untuk dipecahkan oleh manusia, namun setelah adanya kecerdasan buatan masalah dapat dengan mudah diselesaikan. Untuk menyelesaikan masalah menggunakan kecerdasan buatan, biasanya digunakan konsep *deep learning*. *Deep learning* merupakan konsep yang digunakan dalam kecerdasan buatan dan *machine learning*. *Deep learning* juga merupakan pengembangan dari *Neural Network Multiple Layer* untuk mendeteksi suatu objek. Perbedaan *deep learning* dan teknik *machine learning* sangat berbeda, dari teknik *deep learning* akan secara otomatis melakukan representasi dari data gambar, dan video. *Deep learning*, telah membuat terobosan besar dalam mengklasifikasi gambar menggunakan *Computer Vision*. *Deep learning* sudah banyak digunakan dan sudah mencapai hasil yang selalu baik [2].

*Fashion-MNIST* merupakan dataset yang diperoleh dari kumpulan gambar produk *fashion*. Sebelumnya, pada tahun 1998 *Fashion-MNIST* sudah memiliki beberapa kumpulan data namun berupa numerik MNIST yang ditulis dengan tangan yang diperkenalkan oleh LeCun et al, dan itu yang menjadi kumpulan data terbesar yang akan digunakan sebagai kumpulan data uji dalam *deep learning* [9]. Dataset *Fashion-MNIST* biasanya memiliki jumlah yang sama, namun akan ada jenis klasifikasi yang berbeda, karena klasifikasi dari *Fashion-MNIST* yaitu terdiri dari *T-shirt*, *Trouser*, *Pullover*, *Coat*, *Dress*, *Sandals*, *Shirt*, *Sneaker*, *Bag*, dan *Boots* [10]. Dataset

MNIST telah menjadi cara untuk melakukan pengujian yang paling banyak digunakan hingga saat ini khusus untuk pembelajaran deep learning. MNIST lebih sering digunakan ketimbang CIFAR dan ImageNet, karena MNIST dapat lebih cepat untuk membuat prototipe algoritma. Contohnya Tensorflow yang menyediakan fungsi dan contoh dalam menggunakan MNIST. Untuk penelitian tentang klasifikasi Fashion-MNIST ini, peneliti menggunakan Convolutional Neural Network (CNN). Saat ini, Implementasi dan proses build dari dataset Fashion-MNIST menggunakan Tensorflow. Tensorflow merupakan *software open source* yang biasa digunakan dalam pengembangan machine learning yang berisikan berbagai library dengan bahasa pemrograman python, untuk menunjang proses komputasi [11].

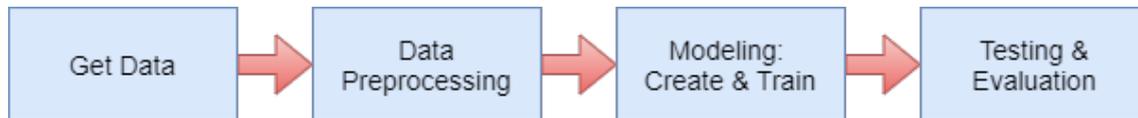


Gambar 1 Fashion-MNIST Dataset

Pada penelitian [8], menggunakan 5 arsitektur CNN dalam penyelesaian masalah klasifikasi gambar pada Dataset MNIST dan Fashion-MNIST, dan hasil penelitian yang mereka temukan bahwa setiap model CNN dapat memberikan akurasi hingga 99%. Arsitektur 3 (3 lapisan konvolusional dan 2 connected layer) memberikan hasil pengujian yang lebih baik dalam set data Fashion-MNIST, karena hasil pengujian training accuracy dan testing accuracy mencapai 93,09% dan 93,56%. Pada penelitian [5], menggabungkan 2 model CNN bersama dengan Batch Normalization dan Skip Connections dan mendapatkan akurasi sebesar 92,54%. Menurut mereka, penggunaan Batch Normalization dan Skip Connections dapat meningkatkan akurasi secara keseluruhan dan bisa mengurangi waktu pelatihan secara signifikan. Pada penelitian lain tentang klasifikasi Garmen dari Fashion MNIST Dataset, diketahui bahwa penelitian tersebut menggunakan LeNet-5 Architecture pada Fashion MNIST Dataset dan mendapatkan akurasi lebih dari 98% dan menurut penulis, LeNet-5 Architecture memberikan kinerja yang lebih tinggi dibandingkan dengan model yang sudah ada [2]. Tujuan dari penelitian ini adalah menggunakan *Convolutional Neural Networks* (CNN) dan library Tensorflow, untuk mengidentifikasi mode pakaian.

## 2. METODE PENELITIAN

Metode penelitian yang digunakan dalam penelitian ini adalah menggunakan arsitektur *Convolutional Neural Network* (CNN) dan library Tensorflow untuk membuat model klasifikasi gambar pada aplikasi untuk *generate* data berbasis *cloud*, yaitu Google Collaboratory. Alur diagram untuk metodologi penelitian dapat dilihat pada Gambar 2.



Gambar 2 Tahapan Metodologi Penelitian

### 2.1 Dataset

Fashion-MNIST merupakan kumpulan data (dataset) gambar dari artikel atau penelitian yang dibuat oleh tim Zalando, yang bertujuan untuk mengganti dataset MNIST yang asli dalam rangka membandingkan algoritma Machine Learning, serta sebagai tolak ukur dalam verifikasi algoritma yang dibuat. Fashion-MNIST diambil berdasarkan gambar setiap produk mode pakaian Zalando, dimana kumpulan gambar tersebut menunjukkan berbagai sisi produk untuk memperjelas tampilan. Visualisasi gambar terdiri dari berbagai tahap diantaranya dari proses *converting*, hingga melakukan proses *re-converting*, tahap-tahap ini akan lebih lanjut dibahas di bagian *modeling process* [9].

Proses *build* Fashion-MNIST menggunakan total 70.000 sampel yang terbagi menjadi 60.000 sampel *train*, dan 10.000 sampel *test*, serta jumlah kolom masing-masing 785 kolom. Dari 70.000 total sampel yang ada, terdapat distribusi yang merata menjadi 10 kelas. Setiap kumpulan gambar yang merepresentasikan label jenis produk menghasilkan kode siluet yang akan digunakan untuk melabeli setiap kelas dataset (dapat dilihat pada Gambar 3). Kode siluet tersebut hanya menghasilkan satu kode untuk setiap label produk. Dimensi gambar yang digunakan dalam dataset ini adalah 28x28, dengan deskripsi setiap kelas dan ukuran masing-masing seperti yang terlihat pada Tabel 1.

Label	Description	Examples
0	T-Shirt/Top	
1	Trouser	
2	Pullover	
3	Dress	
4	Coat	
5	Sandals	
6	Shirt	
7	Sneaker	
8	Bag	
9	Ankle boots	

Gambar 3 Sampel yang digunakan beserta kelas Fashion-MNIST [10]

Tabel 1 Kelas dan ukuran dataset Fashion-MNIST [9]

Nama	Deskripsi	Contoh	Ukuran
train-images-idx3-ubyte.gz	Training set images	60.000	25 MBytes
train-labels-idx1-ubyte.gz	Training set labels	60.000	140 Bytes
t10k-images-idx3-ubyte.gz	Test set images	10.000	4.2 MBytes
t10k-labels-idx1-ubyte.gz	Test set labels	10.000	92 Bytes

## 2.2 Data Preprocessing

Pada fase ini, disediakan data untuk proses training dalam pemrosesan pada gambar seperti normalisasi nilai piksel, mengubah ukuran gambar dan lain-lain [12]. Training set akan menerima 6000 contoh yang dipilih dari setiap kelas secara acak. Gambar dan label akan disimpan dengan format file yang sama dengan kumpulan data MNIST, yang dibuat khusus untuk menyimpan vektor dan matriks multidimensi. Setelah mengekstrak gambar, kami membagi jalur dataset training dan kumpulan validasi data. Kami menetapkan rasio 80% untuk dataset training dan 20% validasi data. Ini mungkin menunjukkan peningkatan dalam set pengujian karena model dan akurasi yang lebih tinggi. Dataset yang dihasilkan bersifat publik sehingga dapat digunakan untuk keperluan masa depan dalam pengerjaan Fashion-MNIST dataset [13].

## 2.3 Convolutional Neural Network

*Convolutional Neural Network* (CNN) adalah algoritma *Deep Learning* yang dapat menangkap *input image* atau gambar, menetapkan signifikansi (bobot yang dapat dipelajari) ke dalam aspek atau objek yang berbeda pada gambar, dan memisahkannya satu sama lain [14]. Struktur pada *convolutional network* (ConvNet) mirip dengan pola komunikasi neuron otak manusia, sebagaimana juga yang dibahas dalam penelitian [8], yaitu pengertian *Artificial neural network* adalah sistem komputer yang secara samar-samar menyerupai cara kerja otak manusia.

Secara umum, terdapat tiga jenis lapisan di CNN: *convolutional layer*, *pooling layer*, dan *fully connected layer*. Semua lapisan ini melakukan tugas yang berbeda pada input data. Pada *convolutional layer*, filter diterapkan untuk mengekstrak fitur. *Pooling layer* melakukan penggabungan maksimal atau penggabungan rata-rata, yang mengekstrak nilai maksimum di wilayah filter atau nilai rata-rata di wilayah filter. Sedangkan *fully connected layer* mengumpulkan informasi dari fitur map dan menghasilkan klasifikasi akhir [8]. Berdasarkan penelitian ini, CNN model yang digunakan menggunakan images dari dataset Fashion-MNIST sebagai input, lalu kemudian diproses pada *convolutional* dan *pooling layers*, sebelum akhirnya menghasilkan output pada *fully connected layer*.

## 2.4 Create and Train the Model

Struktur Convolutional Neural Network yang digunakan pada penelitian ini, dapat dilihat pada Gambar 4. *Pooling layers* digunakan untuk mengurangi dimensi fitur maps, khususnya untuk *width and height*, dan menjaga kedalaman (*depth*). Max pooling menghasilkan nilai maximum dalam porsi gambar yang tercakup oleh filter. Max pooling lebih sesuai dalam mengekstraksi fitur dominan dan oleh sebab itu dianggap lebih baik.

Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 26, 26, 32)	320
max_pooling2d_2 (MaxPooling2)	(None, 13, 13, 32)	0
conv2d_4 (Conv2D)	(None, 11, 11, 64)	18496
max_pooling2d_3 (MaxPooling2)	(None, 5, 5, 64)	0
conv2d_5 (Conv2D)	(None, 3, 3, 128)	73856
flatten_1 (Flatten)	(None, 1152)	0
dense_2 (Dense)	(None, 128)	147584
dense_3 (Dense)	(None, 10)	1290
Total params: 241,546		
Trainable params: 241,546		
Non-trainable params: 0		

Gambar 4 Struktur Conv CNN

Pada proses *build the model*, yang terdiri atas *convolution blocks* dan *max pool layer*, untuk menghindari terjadinya *overfitting* yang dapat mengakibatkan prediksi yang buruk pada dataset, maka digunakanlah teknik *Dropout layer*. *Dropout* sendiri merupakan teknik pada CNN yang dapat mengatasi permasalahan *interdependency* antar neuron [15]. Langkah selanjutnya adalah proses pada 'Flatten' layer, yang fungsinya adalah untuk *flatten* input, tanpa parameter, sehingga tidak akan mempengaruhi *batch size* pada proses *training dataset*.

### 2.5 Evaluation

Pada tahapan akhir klasifikasi pada semua class, untuk mendapatkan evaluasi dari pengujian model, maka *Precision*, *Recall*, serta *F1-Score* didefinisikan untuk mengetahui Fashion-MNIST dataset beserta dengan kelas-kelas nya pada saat proses *validation accuracy* dan *loss*. Algoritma yang digunakan untuk menghitung akurasi baik untuk *Precision*, *Recall*, maupun *F1-Score*, dinyatakan sebagai berikut:

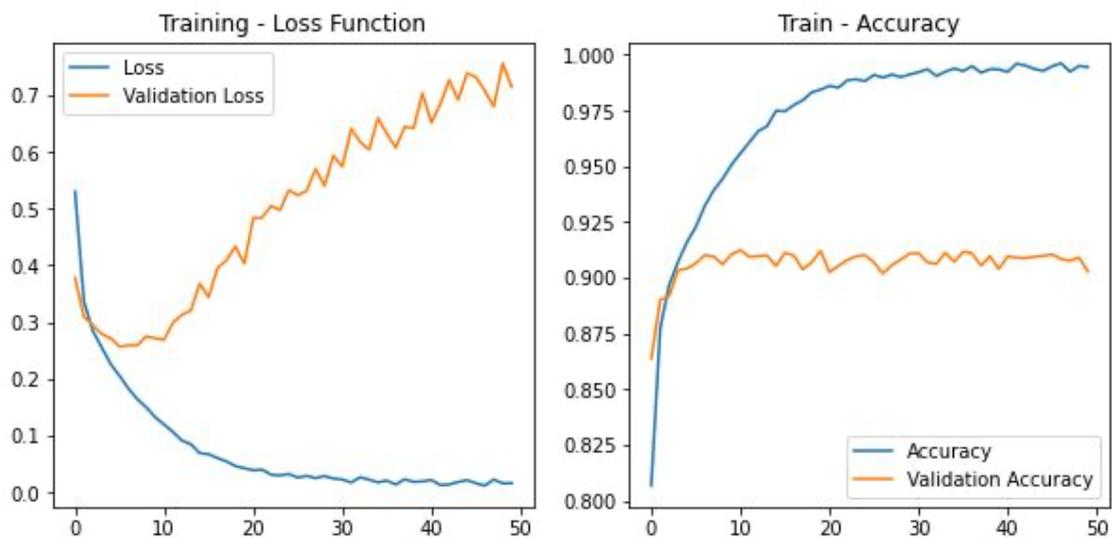
$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

$$F1Score = \frac{2 * Precision * Recall}{Precision + Recall}$$

### 3. HASIL DAN PEMBAHASAN

Pada proses *modeling*, dijalankan dengan menggunakan *training set*, serta menggunakan *validation set* (subset dari *original training set*) untuk validasi model. Setelah proses *validation sequence model*, dengan 50 epochs, maka didapati hasil untuk akurasi mencapai 0,91 atau 91%. Setelah mengevaluasi akurasi model berdasarkan nilai prediksi untuk test set, *validation accuracy* tidak mengalami peningkatan setelah beberapa epoch, sedangkan *validation loss* justru meningkat (seperti yang terlihat pada gambar 5). Berdasarkan hal ini, maka dapat diasumsikan bahwa model tersebut *overfitted*, sehingga perlu menambahkan beberapa *Dropout layers* ke dalam model.



Gambar 5 *Validation Accuracy and Loss without Dropout layer*



Gambar 6 *Validation Accuracy and Loss with Dropout layer*

Dengan menambahkan *Dropout layers* pada model, maka dapat membantu untuk menghindari terjadinya *overfitting*. Setelah itu model tersebut di train kembali, lalu kemudian dijalankan sama persis seperti sebelumnya dengan 50 epochs. Hasil train menunjukkan bahwa *validation accuracy* dan *loss* terlihat lebih baik dan konsisten (seperti yang terlihat di gambar 6) dengan akurasi tes mencapai 0,93 atau 93%. Berdasarkan tabel 2, Class dengan akurasi *precision* terbaik adalah Class 1, Class 5, Class 8, Class 9, dan Class 7. Untuk akurasi yang terburuk yaitu Class 6. Kemudian untuk class dengan *recall* tertinggi yaitu Class 8, Class 5, dan yang terkecil yaitu Class 6 dan Class 4. Class dengan *f1-score* tertinggi adalah Class 1, Class 5, dan Class 8. Sedangkan yang terendah yaitu Class 6, diikuti oleh Class 4, dan Class 2.

Tabel 2 Precision, Recall, dan F1 Score All Classes

Class	Precision	Recall	F1-Score
Class 0 (T-Shirt/Top)	0.86	0.88	0.87
Class 1 (Trousers)	0.99	0.99	0.99
Class 2 (Pullover)	0.91	0.86	0.88
Class 3 (Dress)	0.94	0.93	0.93
Class 4 (Coat)	0.87	0.92	0.90
Class 5 (Sandals)	0.99	0.97	0.98
Class 6 (Shirt)	0.79	0.78	0.78
Class 7 (Sneakers)	0.95	0.98	0.96
Class 8 (Bags)	0.98	0.98	0.98
Class 9 (Ankle Boots)	0.97	0.97	0.97

#### 4. KESIMPULAN

Penggunaan *Convolutional Neural Network* (CNN) model pada klasifikasi Fashion-MNIST dataset pada penelitian ini, menggunakan Tensorflow, serta diimplementasikan proses *run model* dengan memanfaatkan Google Collaboratory. *Sequential* model yang kompleks dengan berbagai convolution layers dan 50 epochs untuk training, didapati bahwa akurasi untuk prediksi test yaitu 0,91 atau 91%. Setelah dievaluasi Kembali validasi akurasi dan loss, didapati kesimpulan bahwa model tersebut *overfitting*, yang menyebabkan akurasinya tidak maksimal. Setelah di train kembali dengan menambahkan *Dropout layers*, dan dengan jumlah epochs yang sama yaitu 50, terjadi peningkatan hasil pada model baru dengan akurasi prediksi test 0,93 atau 93%. Hanya terdapat beberapa kelas saja yang tidak terklasifikasi dengan baik, khususnya Class 6 (Shirt), dan Class 2 (Pullover). Algoritma pada model CNN yang menggunakan layer yang berlapis-lapis, memungkinkan tercapainya akurasi yang baik, bahkan hanya dengan menambahkan satu layer saja, dapat meningkatkan akurasi sebesar 2%.

#### 5. SARAN

Perkembangan teknologi yang pesat terhadap industri-industri, tidak terkecuali industri *fashion*, berpotensi memicu industri tersebut untuk beradaptasi dengan memunculkan inovasi-inovasi yang baru agar dapat bersaing dalam proses adaptasi tersebut. Baik untuk mengklasifikasikan, mengidentifikasi suatu objek atau produk, bahkan untuk memperoleh data historis sekalipun yang semakin hari semakin besar (big data), memerlukan suatu model yang

dapat memuat serta kompatibel sesuai keinginan. Penggunaan *Convolutional Neural Network* (CNN) sudah diterapkan secara baik dalam industri fashion, baik dalam klasifikasi pakaian, pengambilan pakaian, dan pemberian label secara otomatis. Berdasarkan penelitian yang telah dilakukan, diharapkan untuk penelitian-penelitian selanjutnya, dapat membuat atau menambah model yang akan diuji, yang mampu memuat jumlah data yang lebih besar. Tidak menutup kemungkinan untuk model yang baru di implementasikan pada dataset yang sama, tetapi dengan pencapaian akurasi yang lebih tinggi, serta mengembangkan pengenalan objek dengan fokus untuk dapat mendeteksi dan mengenali motif dari objek fashion.

#### DAFTAR PUSTAKA

- [1] Y. Seo and K.-S. Shin, "Hierarchical Convolutional Neural Networks for Fashion Image Classification," *Expert Systems With Applications*, vol. 116, pp. 328–339, Feb. 2019, doi: 10.1016/j.eswa.2018.09.022.
- [2] M. Kayed, A. Anter, and H. Mohamed, "Classification of Garments from Fashion MNIST Dataset Using CNN LeNet-5 Architecture," p. 6.
- [3] Y. Zhang, "Evaluation of CNN Models with Fashion MNIST Data," p. 42, 2019.
- [4] K. V. Greeshma and K. Sreekumar, "Fashion-MNIST Classification Based on HOG Feature Descriptor Using SVM," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 8, no. 5, pp. 960–962, 2019.
- [5] M. S. Bhatnagar, M. D. Ghosal, and D. M. H. Kolekar, "Classification of Fashion Article Images Using Convolutional Neural Networks," p. 6, 2017.
- [6] L. Donati, E. Iotti, G. Mordonini, and A. Prati, "Fashion Product Classification through Deep Learning and Computer Vision," p. 22, 2019.
- [7] M. Xu, "Segmentation of lung parenchyma in CT images using CNN trained with the clustering algorithm generated dataset," p. 21, 2019.
- [8] S. S. Kadam, A. C. Adamuthe, and A. B. Patil, "CNN Model for Image Classification on MNIST and Fashion-MNIST Dataset," *Journal of Scientific Research*, vol. 64, no. 2, p. 11, 2020.
- [9] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms," *arXiv:1708.07747 [cs, stat]*, Sep. 2017, Accessed: Nov. 18, 2020. [Online]. Available: <http://arxiv.org/abs/1708.07747>.
- [10] S. Priyowidodo, "KLASIFIKASI GAMBAR DATASET FASHION-MNIST MENGGUNAKAN DEEP CONVOLUTIONAL NEURAL NETWORK," vol. 7, no. 1, p. 5, 2019.
- [11] A. F. M. Agarap, "Deep Learning using Rectified Linear Units (ReLU)," p. 8, Mar. 2018.
- [12] A. Prajapati, A. Kaushik, P. Gupta, S. Sharma, and S. Jain, "Fashion Product Image Classification Using Neural Network," vol. 3, no. 4, p. 3.
- [13] C. Geier, "Training on test data: Removing near duplicates in Fashion-MNIST," *arXiv:1906.08255 [cs, stat]*, Jun. 2019, Accessed: Nov. 29, 2020. [Online]. Available: <http://arxiv.org/abs/1906.08255>.

- 
- [14] S. Shubathra, P. Kalaivaani, and S. Santhoshkumar, "Clothing Image Recognition Based on Multiple Features Using Deep Neural Networks," p. 7, 2020.
  - [15] F. K. Celsia, G. A. Sandag, "Implementasi Deep Learning pada Pengenalan Angka dalam Sign Language," Jurnal Ilmiah Sistem Informasi dan Teknik informatika (SISFOTENIKA). vol 11, no. 2. p 124-126. Juli. 2021.