

Penerapan Metode Convolutional Neural Network Untuk Clothing Image Recognition

Fuyi Gunawan Putri, Justinus Andjarwirawan, Anita Nathania Purbowo
Program Studi Informatika Fakultas Teknologi Industri Universitas Kristen Petra

Jl. Siwalankerto 121-131 Surabaya, 60236
Telp. (031) – 2983455, Fax. (031) - 8417658

E-mail : fuyigunawan@gmail.com, justin@petra.ac.id, anita.nathania@petra.ac.id

ABSTRAK

Objek *recognition* adalah suatu perkembangan dunia AI (*Artificial Intelligence*) yang digunakan untuk mengenali objek dalam citra digital. Berbagai macam pengenalan objek dapat dilakukan, salah satunya *Clothing Image Recognition*. Era industri global, kreativitas *fashion* akan terus diekspresikan dan berkembang dengan menciptakan berbagai macam banyak model dan jenis pakaian. Perkembangan ini, membuat orang kesulitan dalam mengenali variasi *fashion* yang baru, hal ini dikarenakan pakaian memiliki banyak jenis dan varian, misalnya untuk pakaian atas terdapat varian mulai dari *sweatshirts*, *kurtas*, *waistcoat* dan sebagainya. Hal ini membuat pengguna menjadi kesulitan untuk membedakan kategori dari jenis pakaian. Bahkan, seringkali pengguna tidak mengetahui atau lupa dengan nama suatu jenis pakaian yang ingin dicari dan lebih mengingat bentuknya.

Clothing image recognition menggunakan metode CNN (*Convolutional Neural Network*) dengan model VGG16. Penerapan CNN dengan model VGG16 akan dilakukan pada *python* dengan bantuan *library Keras* sebagai mempercepat proses riset pada *training data*. Untuk implementasi program ke perangkat *mobile* menggunakan Android Studio dengan bahasa pemrograman Kotlin, dengan bantuan *library TensorFlow Lite* sebagai pembelajaran *machine learning* pada perangkat *mobile*. Hasil skripsi ini menunjukkan akurasi 81.83% terhadap *training data* pada *epoch 1* dan 82%-94% terhadap *training data* pada *epoch 10* pada *clothing image recognition*.

Kata Kunci: *Convolutional Neural Network*, VGG16, *clothing image recognition*, *machine learning*, *artificial intelligence*.

ABSTRACT

Object recognition is a development of the world of AI (Artificial Intelligence) which is used to recognize objects in digital images. Various kinds of recognition can be done with Clothing Image Recognition. In the global industrial era, development of fashion type and variant makes it difficult for people to recognize, for example topwear has variants ranging from sweatshirts, kurtas, waistcoats, etc. This makes users hard to distinguish categories from types of clothing. In fact, users often forget or didn't know the name of a clothing type that they want to find because they just remember the shape.

Clothing image recognition uses the CNN (Convolutional Neural Network) method with the VGG16 model. The application of CNN with the VGG16 model will be carried out on Python and Keras as the library to speed up the research process on training data. Application of the program to mobile devices using Android Studio with the Kotlin programming language, with the help of the

Tensorflow Lite library as machine learning on mobile devices. The results of this thesis show an accuracy of 81.83% for training data in epoch 1 and 82%-94% for training data in epoch 10 on clothing image recognition.

Keywords: *Convolutional Neural Network*, VGG16, *clothing image recognition*, *machine learning*, *artificial intelligence*.

1. PENDAHULUAN

Era industri global, perkembangan *fashion* pada pakaian semakin meningkat. Dengan berkembangnya dunia *fashion*, membuat orang kesulitan dalam mengenali variasi *fashion* yang baru, hal ini dikarenakan pakaian memiliki banyak jenis dan varian, misalnya untuk pakaian atas saja terdapat varian mulai dari *sweatshirts*, *kurtas*, *waistcoat* dan masih banyak lagi. Sehingga, banyaknya jenis tersebut membuat pengguna menjadi kesulitan untuk membedakan detail dari jenis pakaian. Bahkan, seringkali pengguna tidak mengetahui atau lupa dengan nama suatu jenis pakaian yang ingin dicari dan lebih mengingat bentuknya. Permasalahan tersebut dapat dijawab dengan pengenalan objek (*image recognition*) salah satunya dengan menerapkan Metode *Convolutional Neural Network* untuk *Clothing Image Recognition*. Dimana mengenali dan mengidentifikasi nama dari suatu pakaian melalui gambar. Metode ini merupakan salah satu metode *Deep learning* yang dapat digunakan untuk mengenali dan mengklasifikasi sebuah objek pada sebuah citra digital [10].

Convolutional Neural Network (CNN) adalah pengembangan dari *Multilayer Perceptron* (MLP) yang termasuk dalam *neural network* bertipe *feedforward* (bukan berulang) [9]. Penelitian berjudul “*Clothing identification via deep learning: forensic applications*” berfokus pada mengidentifikasi orang berdasarkan visual yang diperoleh dari pakaian pengguna dengan menggunakan sistem DIGITS, NVIDIA's *deep learning* untuk mengklasifikasikan *dataset* pakaian. Hasil penelitian terkait menghasilkan akurasi pengklasifikasian gambar pakaian sekitar 75% dengan rata-rata prediksi sebesar 70%. Disimpulkan bahwa pengklasifikasian gambar mencapai tingkat keberhasilan lebih tinggi dari 70% dari set data pemberian label sebesar 70% dari gambar uji dengan benar. Dibandingkan dengan penelitian sejenis lainnya, penelitian terkait memiliki tingkat akurasi yang lebih rendah. Oleh karena itu, dilakukan penelitian ini dilakukan untuk melihat perbedaan akurasi saat mengenali gambar yang hanya berisi satu objek dan gambar dengan banyak objek didalamnya. Penelitian ini memberikan informasi mengenai jenis pakaian yang di *recognition*. Diharapkan dengan melakukan penelitian ini dapat menghasilkan akurasi yang lebih tinggi dengan menggunakan metode *Convolutional Neural Network*.

2. DASAR TEORI

2.1 Tinjauan Studi

Pada penelitian yang dilakukan oleh Jan Cychnerski, et.al [6] memanfaatkan metode *Convolutional Neural Network* dengan *Residual networks*, *SqueezeNet* dan *Single Multibox Detector (SSD)* pada tahun 2017 untuk membuktikan hasil akurasi pada proses klasifikasi akurasi *fashion* dan pakaian pada gambar *e-niaga* yang sulit. Hasil penelitian terkait mendapatkan rata tingkat kebenaran label sebesar 83% dan klasifikasi pakaian akurasi sebesar 70%. Penelitian selanjutnya dilakukan oleh Bedeli, et.al. [7] pada tahun 2018, dimana penelitian yang terkait dilakukan karena pengidentifikasian seseorang dengan *face recognition* sulit diterapkan karena bergantung pada kualitas gambar dan video. Oleh karena itu, penelitian terkait menggunakan pengenalan objek pakaian untuk mempermudah mengidentifikasi seseorang. Penelitian terkait menggunakan sistem DIGITS, NVIDIA's *deep learning* dalam pengklasifikasian *dataset* pakaian. Hasil penelitian mendapatkan akurasi sebesar 75% dengan rata-rata prediksi sebesar 70% untuk pengklasifikasian gambar pakaian dan disimpulkan tingkat keberhasilan lebih tinggi dari 70% dari set data pemberian label sebesar 70% dari gambar uji dengan benar. Penelitian terakhir, penelitian yang dilakukan oleh Shubathra, et.al [14] adalah penelitian untuk menangani masalah dalam mengidentifikasi gaya dan pengakuan gambar pakaian untuk *fashion item*, sehingga penelitian terkait melakukan penelitian dengan mengidentifikasi gambar pakaian dan pengklasifikasian *dataset* MNIST dengan menggunakan metode *multilayer perceptron*, *convolutional neural network* dan *extreme learning machine* pada tahun 2020. Hasil penelitian menghasilkan presisi gambar sebesar 97,5% dengan penggunaan *dataset Fashion-MNIST* dari artikel Zalando sebanyak 60.000 data *learning* dan 10.000 data *testing*. Pada penelitian ini dibuat suatu uji coba untuk mengenali jenis pakaian yang berdasarkan latar belakang masalah seperti pada penelitian terkait menggunakan CNN model VGG16. Dimana penggunaan CNN memiliki kelebihan untuk mengenali objek dalam gambar dengan akurasi lebih tinggi.

2.2 Tinjauan Pustaka

2.2.1 Jenis Pakaian

Pakaian merupakan suatu produk yang terbuat dari bahan tekstil atau garmen yang digunakan pada tubuh. Pakaian sendiri memiliki beberapa jenis (*sweater, dress, hoodies, t-shirt, flip-flops, shorts, skirt, swimsuit, tie, jeans, shoes, coat, high heels, suit, cap, socks, boots, polo shirt, shirt, scarf, hat, gloves, jacket, long coat, bra, sunglasses, leather jacket*) [15].

2.2.2 Convolutional Neural Network

Convolutional Neural Network merupakan metode yang terinspirasi dari *visual cortex* otak dan khusus untuk memproses suatu data dengan struktur *grid* atau bisa juga disebut sebagai metode jaringan syaraf tiruan. Dimana metode ini memiliki beberapa jenis *layer* yaitu *convolutional layer, subsampling layer, loss layer, dan fully connected layer* [16].

2.2.3 Object Detection

Object Detection menentukan keberadaan suatu objek dan ruang lingkungannya serta lokasi pada sebuah gambar [4]. Deteksi objek adalah proses menemukan *instance* objek dari kelas tertentu, seperti wajah, mobil, dan pohon, dalam gambar atau video [13]. Pada *object detection* dapat mengenali *Object Detection* ini

merupakan pendekatan dari Deep Learning yang didasarkan dari *Convolutional Neural Network. Object Detection* [3].

2.2.4 VGG16

VGG16 merupakan model CNN yang memanfaatkan *convolutional layer* dengan spesifikasi *convolutional filter* yang kecil (3×3) [8]. VGG16 bertujuan untuk mengidentifikasi gambar menjadi lebih akurat pada *image classification*.

2.2.5 Keras

Keras merupakan *library Machine Learning open source* berbasis Python, dikembangkan untuk membuat penerapan model pembelajaran yang mendalam secepat dan semudah mungkin untuk penelitian serta pengembangan yang dirilis berdasarkan lisensi MIT [5].

2.2.6 Tensorflow Lite

Tensorflow Lite adalah *library machine learning* yang dirancang khusus untuk perangkat *mobile*, yang memungkinkan mesin untuk belajar di perangkat dengan latensi rendah dan ukuran *binary* yang kecil [2].

3. ANALISA DAN DESAIN SISTEM

3.1 Analisa Data

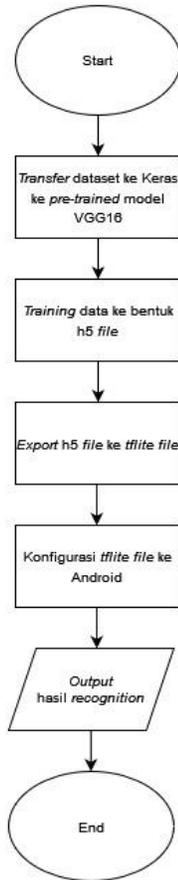
Analisa data dilakukan dengan pengambilan dataset, pembagian dataset serta pengelolaan dataset dan desain sistem dari penerapan metode *Convolutional Neural Network* untuk *Clothing Image Recognition*. Pada Analisa data hal pertama yang dilakukan dengan pengambilan dataset. Sebelumnya, dataset merupakan kumpulan data yang mencantumkan suatu nilai untuk setiap variable. Dataset yang digunakan pada skripsi ini berupa dataset gambar yang diambil dari beberapa sumber. Sumber pertama dataset pakaian yaitu diambil dari kaggle, yang disediakan oleh Param Aggarwal [1]. Dataset ini untuk setiap jenis dari kategorinya, data yang digunakan untuk masing-masing jenisnya berjumlah 2000 data. Pertama, dataset yang digunakan terdiri dari kategori *Topwear* yang terdiri dari jenis *Shirts, T-shirts, Tops, Sweatshirts, Kurtas, Rain jacket, Waistcoat*. Kedua, dataset pada kategori *One Piece* yang terdiri dari jenis *Dress dan Kurta Sets*. Ketiga, dataset kategori *Bottomwear* terdiri dari jenis *Jeans, Track pants, Capris, Shorts, Trousers*. Keempat, kategori *Footwear* yang terdiri dari jenis *Casual shoes, Formal shoes, Flats, Heels, Sandals, Flip flops, Sport shoes*. Dan yang terakhir ada kategori *Accessories* yang terdiri dari jenis *Socks, Watches, Sunglasses, Belts, HandBags, Caps*. Sumber kedua, diambil dari kaggle, yang disediakan oleh ololo [11]. Pada dataset ini hanya akan diambil jenis *Blazer* saja. Sumber ketiga, diambil dari *Pinterest* [12] yang diikuti dengan kata kunci jenis pakaian yang diperlukan untuk penambahan jenis pakaian dari dataset yang jumlah estimasinya belum mencukupi jumlah dari kriteria (2000 data). Pembagian dataset, pada dataset untuk VGG16 terdapat dua pembagian dataset yaitu dataset *testing* dan dataset *training*. Pada dataset keseluruhan yang berestimasi 2000 data, akan diambil 20% untuk data *testing* dari total data dan 80% untuk data *training* dari total data juga. Sehingga, estimasi untuk data *testing* sebesar 400 data dan data *training* sebesar 1600 data.

3.2 Desain Sistem Keseluruhan

3.2.1 Flowchart Sistem Program Keseluruhan

Dapat dilihat pada Gambar 1 dimulai dengan program python, mentransferkan dataset menggunakan Keras ke *pre-trained* model

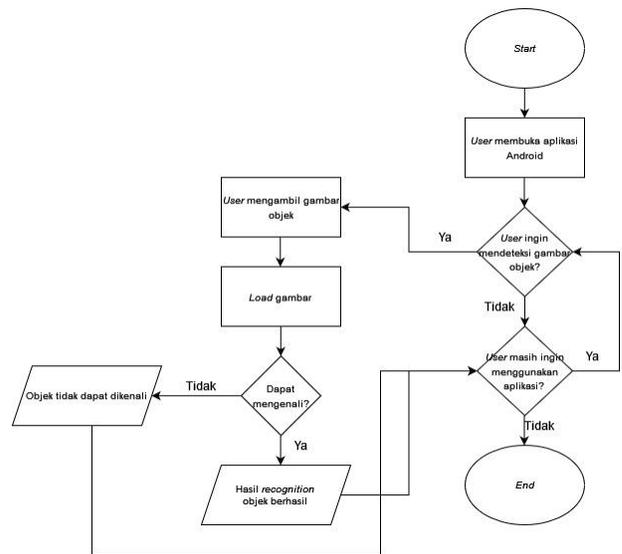
VGG16. Pentransferan dataset ini bertujuan agar dataset dari jenis pakaian dapat mengenali berdasarkan jenis pakaiannya dengan menggunakan metode CNN dengan model VGG16. Setelah dataset ditransfer ke *pre-trained* model VGG16, maka akan dilanjutkan dengan *training* data yang dimana hasil *training* data ini akan disimpan dalam bentuk *h5 file*. Hasil *h5 file*, kemudian akan di *export* dalam bentuk *tflite file* di python. Setelah itu dilakukan konfigurasi *tflite file* ke Android, *tflite file* dimasukkan ke Android secara manual. Tujuannya agar di Android dapat melakukan *recognition* objek dengan *pre-trained* model VGG16. Kemudian, dilanjutkan dengan *clothing image recognition*.



Gambar 1. Flowchart Sistem Program Keseluruhan

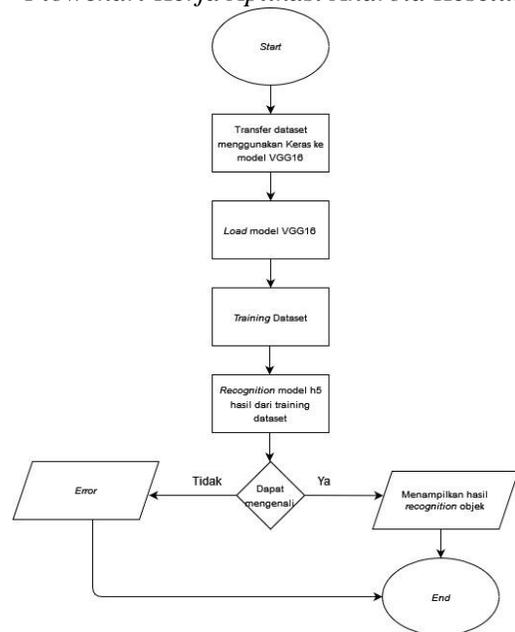
3.2.2 Flowchart Kerja Aplikasi Android Keseluruhan

Dapat dilihat pada Gambar 2 menjelaskan kerja aplikasi Android secara keseluruhan dapat dimulai dengan pengguna membuka aplikasi. Kemudian, pengguna dapat membuka kamera dan mengambil gambar objek dari pakaian yang diinginkan. Setelah itu, gambar objek akan di *load* pada Android untuk mengetahui apakah objek gambar dari pakaian dapat dikenali atau tidak. Jika berhasil, maka aplikasi akan memberikan *output* berupa label. Sedangkan, jika tidak maka pengguna dapat mengambil gambar objek lainnya atau keluar dari aplikasi.



Gambar 2. Flowchart Kerja Aplikasi Android Keseluruhan

3.2.3 Flowchart Kerja Aplikasi Android Keseluruhan

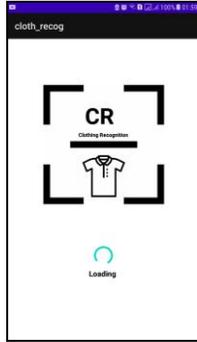


Gambar 3. Flowchart Kerja pada Python

Dapat dilihat pada Gambar 3, bahwa dataset akan ditransfer menggunakan Keras ke model VGG16 pada Python. Pentransferan dilakukan agar dataset dapat mengenali berdasarkan jenis pakaiannya dengan menggunakan metode *Convolutional Neural Network* dengan *pre-trained* model VGG16. Kemudian, *pre-trained* model VGG16 akan di *load* untuk menjalankan *training* data. Setelah proses *training dataset* selesai, akan menghasilkan *file* dalam bentuk *.h5*, *file* ini akan dipanggil ke dalam proses prediksi. Prediksi gambar dilakukan dengan memanggil nama *file* gambar yang akan di *recognition*, jika berhasil dikenali maka akan menampilkan hasil *recognition*, sedangkan jika tidak maka akan *error*, maksud dari *error* ini, jika nama *file* dari gambar yang dimasukkan tidak ada atau salah.

4. IMPLEMENTASI SISTEM

4.1 Implementasi Interface

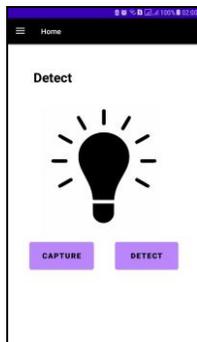


Gambar 4. *Splash Activity*



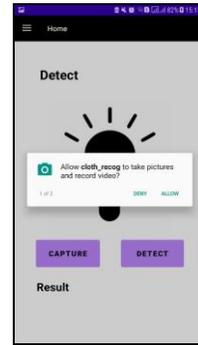
Gambar 5. *Navigation Drawer*

Gambar 4 *Splash activity* digunakan untuk memberikan fitur animasi bergerak pada tampilan *interface device*. *Splash Activity* akan muncul saat aplikasi pertama kali dibuka dan akan langsung dilanjutkan ke Halaman *Home*. Pada Gambar 5 *Navigation drawer* digunakan untuk berpindah dari satu *activity* ke *activity* lainnya. Pada *Navigation drawer* terdapat 3 *activity* yaitu *Home*, *Gallery* dan *About*.



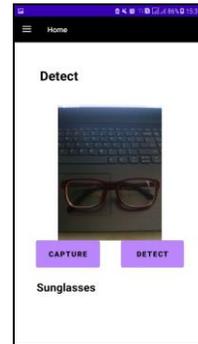
Gambar 6. *Home*

Dapat dilihat pada Gambar 6 merupakan tampilan *Home* saat pengguna telah memberikan akses izin kamera. Setelah pengaksesan diberikan maka pengguna dapat melanjutkan ke kamera untuk pengambilan objek gambar yang ingin dikenali. Lalu, pada Gambar 7 merupakan tampilan sebelumnya ketika aplikasi meminta izin hak akses kamera. Hak akses kamera ini digunakan untuk mendapatkan izin dari pengguna, jika hak akses diberikan maka pengguna dapat menggunakan kamera dan dapat melanjutkan *clothing image recognition*.



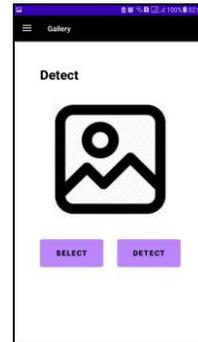
Gambar 7. *Tampilan Izin Hak Akses Kamera*

Setelah pengambilan gambar selesai, maka akan ditampilkan pada *device*. Lalu, pengguna dapat memprediksi gambar dengan *button detect* dan hasil akan berupa label seperti pada Gambar 8.



Gambar 8. *Pengambilan Gambar Selesai*

Halaman *Gallery* dapat dilihat pada Gambar 9, halaman ini untuk pengguna memilih gambar objek yang diinginkan melalui *gallery* pengguna. Selain itu pada aplikasi terdapat halaman *About* sebagai halaman informasi tentang *clothing image recognition*.



Gambar 9. *Halaman Gallery*

5. PENGUJIAN SISTEM

5.1 Pengujian Data dengan CNN Model VGG16 (Python)

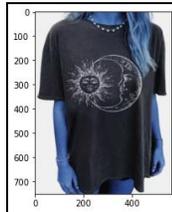
Tujuan pada tahap pengujian ini adalah untuk melihat hasil akurasi objek yang dapat dikenali oleh model VGG16. Semua pengujian ini dilakukan pada hasil *training epoch* 10 pada *pre-trained* model VGG16. Berikut dibawah ini merupakan gambar yang diambil dari *Pinterest* dan hasil pengujiannya dapat dilihat sebagai berikut :

Hasil *Recognition Model* : *Shirts*



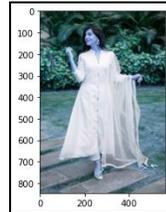
Gambar 10. Shirts

Hasil *Recognition Model* : *Tops*



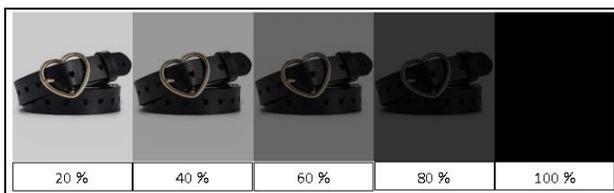
Gambar 11. T-shirts

Hasil *Prediksi Model* : *Kurtas_sets*



Gambar 12. Kurtas sets

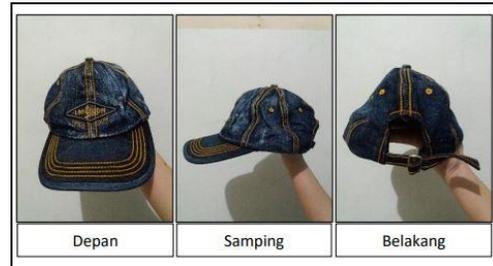
Hasil *recognition* yang dilakukan dapat dilihat pada Gambar 10, Gambar 11 dan Gambar 12. Dari gambar-gambar tersebut Gambar 10 dan Gambar 12 menghasilkan hasil *recognition* benar. Selain itu, bahwa ketika *recognition* dilakukan pada 1 objek maka hasil bisa terdapat kesalahan dalam memprediksi jenis objek. Contoh dapat dilihat pada Gambar 11. Pada contoh merupakan gambar *T-shirt*, namun hasil kinerja CNN *pre-trained* model VGG16 mengenali gambar tersebut sebagai jenis *Tops*. Hal ini dikarenakan pada *recognition* objek dapat menjadi bias atau penyimpangan ketika objek memiliki kemiripan jenis atau bentuk pada jenis objek lainnya. Selain itu, pengujian pada banyak objek yang berbeda, maka yang akan dikenali adalah gambar yang paling menonjol atau jelas. Dengan begitu objek lainnya tidak akan dikenali.



Gambar 13. Data dengan pencahayaan gelap

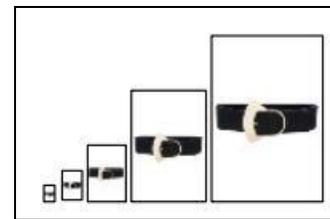
Pada Gambar 13, pada skala pencahayaan 20% dan 40%, semua objek masih dapat dikenali dengan jelas, yang berarti tidak mempengaruhi hasil *recognition*. Namun, pada skala pencahayaan ini, terdapat beberapa objek yang dikenali namun hasil *recognition* salah, dikarenakan objek memiliki kemiripan terhadap objek jenis lainnya. Skala 60% dan 80%, objek masih dapat dikenali, namun beberapa jenis menghasilkan hasil *recognition*

yang salah, pada skala pencahayaan yang lebih gelap membuat bentuk objek kurang jelas, sehingga model sulit untuk mengenali objek. Skala pencahayaan akhir yaitu 100%, semua objek tidak dapat dikenali, karena objek tidak terlihat pada saat kondisi gelap total. Hasil akurasi pada pengujian ini sebesar 69.05%.



Gambar 14. Data dengan sisi yang berbeda

Pada Gambar 14, gambar dari setiap sisi yang berbeda berpengaruh pada hasil *recognition*. Hal ini dikarenakan, setiap posisi yang berbeda bisa menampilkan kesamaan bentuk dari jenis objek yang lainnya. Pengujian data untuk setiap jenis masing-masing diuji sebanyak tiga data gambar yang berbeda. Pengujian data menggunakan tiga arah yang berbeda yaitu depan, samping, belakang. Hasil akurasi untuk data dengan sisi yang berbeda sebesar 63.1%, dengan jumlah *recognition* benar sebanyak 159 data dari 252 total data.



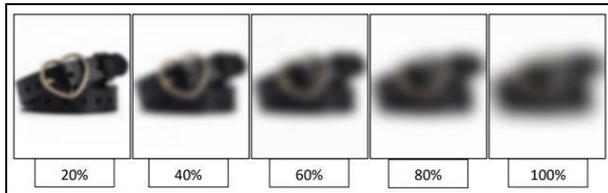
Gambar 15. Data dengan variasi ukuran yang berbeda

Pada Gambar 15, gambar dengan variasi ukuran yang berbeda mempengaruhi hasil *recognition*, dikarenakan untuk setiap ukuran bisa memberikan hasil yang berbeda-beda. Ukuran 5px dan 10px, model tidak dapat mengenali objek. Hal ini dikarenakan garis objek tidak terlihat dengan jelas. Ukuran 20px keatas, objek mulai dapat dikenali dan semakin jelas. Hasil akurasi pada data dengan ukuran yang berbeda sebesar 34,52%. Dapat disimpulkan bahwa semakin kecil ukuran objek, maka objek tidak dapat dikenali. Sebaliknya, semakin besar ukuran objek maka semakin mudah dikenali.



Gambar 16. Data terpotong

Pengujian data terpotong, masing-masing jenis diuji sebanyak satu data. Dapat dilihat salah satu contoh data terpotong pada Gambar 16. Hasil *recognition* sangat berpengaruh terhadap data terpotong. Hal ini dikarenakan, ketika data terpotong maka objek tidak memiliki bentuk yang spesifik untuk di kenali. Hasil akurasi terhadap data terpotong sebesar 16,96%.



Gambar 17. Data buram

Pengujian terhadap data blur atau buram dapat dilihat salah satu contoh data buram pada Gambar 17. Pengujian pada data ini diuji sebanyak tiga data untuk setiap jenis. Hasil akurasi terhadap data buram didapatkan sebesar 27.14%. Hal ini dapat disimpulkan bahwa data buram sangat mempengaruhi hasil *recognition*. Semakin tinggi tingkat keburaman suatu objek, maka model semakin susah mengenali objek dan sebaliknya.

Algoritma CNN diuji juga dengan data yang berbeda-beda untuk setiap jenis dataset yang ada. Masing-masing dataset diuji menggunakan 5 data dan hasil yang didapatkan dapat dilihat pada Tabel 1

Tabel 1 Hasil pengujian jenis terhadap 8 buah data

Jenis Dataset	True	False	Accuracy
Belts	8	0	100%
Blazer	8	0	100%
Capris	7	1	88%
Caps	7	1	88%
Casual_shoes	1	7	13%
Dress	6	2	75%
Flats	8	0	100%
Flipflops	6	2	75%
Formal_shoes	8	0	100%
Handbags	7	1	88%
Heels	7	1	88%
Jeans	5	3	63%
Kurtas	1	7	13%
Kurtas_sets	7	1	88%
Rain_jacket	8	0	100%
Sandals	4	4	50%
Shirts	7	1	88%
Shorts	8	0	100%
Socks	8	0	100%
Sport_shoes	7	1	88%
Sunglasses	8	0	100%
Sweatshirts	7	1	88%
Tops	5	3	63%
Track_pants	0	8	0%
Trousers	5	3	63%
T-shirts	0	8	0%
Waiscoats	8	0	100%
Watches	7	1	88%
Akurasi			75%

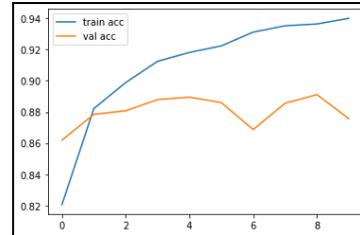
Dari hasil Tabel 1 di atas, dapat dilihat bahwa algoritma dapat mengenali dengan benar dan salah. Beberapa objek dapat mengenali dengan benar sebanyak 8 kali atau 100%, dikarenakan objek memiliki karakteristik yang tidak identik dengan objek lainnya, sehingga algoritma lebih mudah dalam mengenali. Sedangkan, penyebab objek gagal atau salah dikenali karena objek tidak jelas dan karakteristik objek tersebut mirip dengan

objek lainnya sehingga algoritma kesulitan dalam mengenali. Hasil akurasi untuk data yang dapat dikenali semua objeknya sebesar 100%, dan untuk hasil akurasi terkecil sebesar 0%.

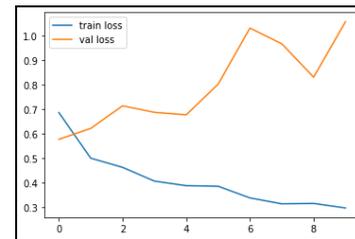
5.2 Pengujian Training Data pada Epoch

Pengujian *training data* pada *epoch* dilakukan pada 2 pengujian *epoch*. Pengujian pertama dilakukan pada *epoch* 1, menghasilkan akurasi 81.83% dan val akurasi sebesar 88.24%. Sedangkan pengujian pada *epoch* 10 menghasilkan akurasi sebesar 82%-94% dan untuk val akurasi menghasilkan akurasi 86%-87%.

Pengujian pada *epoch* 10 juga menghasilkan grafik *accuracy* dan grafik *loss* yang dapat dilihat pada Gambar 18 dan Gambar 19.



Gambar 18. Grafik Accuracy



Gambar 19. Grafik Loss

6. KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan implementasi dan pengujian sistem yang telah dilakukan pada *Convolutional Neural Network* model VGG16, maka dapat disimpulkan sebagai berikut :

- Akurasi CNN *pre-trained* model VGG16 pada *epoch* 1 menghasilkan akurasi sebesar 81.83% dan val akurasi sebesar 88.24%. Sedangkan pada pengujian *epoch* 10 menghasilkan akurasi dari 82%-94% dan val akurasi menghasilkan 86%-87%.
- Hasil kinerja menggunakan CNN *pre-trained* model VGG16 pada *recognition* gambar objek, sebagai berikut :
 - *Recognition* gambar pada 1 objek dapat menjadi bias atau penyimpangan ketika objek yang dikenali memiliki kemiripan jenis atau bentuk pakaian.
 - Algoritma dapat mengenali objek pada gambar yang memiliki banyak objek di dalamnya. Objek yang dapat dikenali adalah satu objek yang paling menonjol atau jelas.
 - Objek dengan pencahayaan berpengaruh dalam hasil *recognition*. Pada skala pencahayaan 40% dan seterusnya objek mulai sulit dikenali. Sehingga, dapat dikatakan bahwa semakin gelap objek maka objek dapat dikenali sebagai objek lain dan bahkan tidak dapat

dikenali. Akurasi pengujian pada data dengan pencahayaan dalam skripsi ini sebesar 69.05%.

- Data dengan sisi berbeda, dapat mempengaruhi hasil *recognition* dikarenakan setiap posisi yang berbeda bisa menampilkan kesamaan bentuk dari jenis objek yang lainnya. Akurasi pengujian data dengan sisi yang berbeda sebesar 63.1%.
- Data dengan variasi ukuran berbeda, berpengaruh pada hasil *recognition*. Pada skala ukuran mulai 40px ke bawah, objek mulai sulit dikenali. Dikarenakan semakin kecil objek maka objek dapat dikenali sebagai objek lain bahkan tidak dapat dikenali. Akurasi pengujian pada data ini sebesar 34.52%.
- Data terpotong sangat berpengaruh pada hasil *recognition*. Dikarenakan, ketika data terpotong objek tidak memiliki bentuk yang spesifik untuk di kenali. Akurasi pengujian terhadap data terpotong sebesar 16,96%.
- Data buram, berpengaruh pada hasil *recognition*. Pada skala 20% dan 40% objek dapat dikenali dengan benar dan dapat dikenali juga sebagai objek lain. Pada skala 60% keatas objek susah untuk dikenali bahkan tidak dapat dikenali. Semakin tinggi tingkat keburaman suatu objek, maka model semakin susah mengenali objek. Akurasi pengujian data ini sebesar 27.14%

6.2 Saran

Saran yang dapat diberikan dalam skripsi ini untuk pengembangan skripsi ini lebih lanjut, sebagai berikut :

- *Recognition* objek dapat langsung dilakukan menggunakan kamera tanpa harus menyimpan gambar terlebih dahulu (*Live camera*).
- Dataset dan pelabelan agar dapat lebih dispesifikkan agar algoritma lebih mudah dalam mengenali objek.
- *Clothing image recognition* dapat juga menggunakan atau diuji coba dengan metode lainnya.

7. DAFTAR PUSTAKA

- [1] Aggarwal, P. (2019). Fashion Product Images Dataset. Retrieved from kaggle: <https://www.kaggle.com/paramaggarwal/fashion-product-images-dataset>
- [2] Andrea, A. (2018, May 13). *Tensor Flow Lite Android*. Retrieved from <https://adiandrea.id/articles/2018-05/tensor-flow-lite-android>
- [3] Carnino, D. (2018, June 04). Clothing Detection for Fashion Recommendation. Retrieved from Valohai: <https://valohai.com/blog/clothes-detection-for-fashion-recommendation/>
- [4] Dewi, S. R. (2018). DEEP LEARNING OBJECT DETECTION PADA VIDEO MENGGUNAKAN TENSORFLOW DAN CONVOLUTION

AL NEURAL NETWORK. Retrieved from https://dspace.uui.ac.id/bitstream/handle/123456789/7762/14611242_Syarifah%20Rositita%20Dewi_Statistika.pdf?sequence=1&isAllowed=y

- [5] DOLab. (2020, Desember 15). *Belajar Machine Learning dengan Library Python Populer*. Retrieved from DOLab: <https://www.dqlab.id/belajar-machine-learning-dengan-library-python>
- [6] Jan Cychnerski, A. B. (2017). Clothes detection and classification using convolutional neural networks. doi:10.1109/ETFA.2017.8247638
- [7] Marianna Bedeli, Z. G. (2018). Clothing identification via deep learning: forensic. FORENSIC SCIENCES RESEARCH. doi:10.1080/20961790.2018.1526251
- [8] Muftah Afrizal Pangestu, H. B. (2018, Agustus). Analisis Performa dan Pengembangan Sistem Deteksi Ras Anjing pada Gambar dengan Menggunakan Pre-Trained CNN Model. Jurnal Teknik Informatika dan Sistem Informasi, Volume 4 Nomor 2, 337-344. doi:<http://dx.doi.org/10.28932/jutisi.v4i2.828>
- [9] Nugroho, P. A., Fenriana, I., & Rudy Arijanto, M. (2020). IMPLEMENTASI DEEP LEARNING MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK (CNN) PADA EKSPRESI MANUSIA. JURNAL ALGOR, VOL.2 NO.1, 12-21. Retrieved from <https://jurnal.buddhidharma.ac.id/index.php/algor/article/view/441/259>
- [10] Nurhikmat, T. (2018). IMPLEMENTASI DEEP LEARNING UNTUK IMAGCLASSIFICATION MENGGUNAKAN ALGORITMA CONVOLUTIONAL NEURAL NETWORK (CNN) PADA CITRA WAYANG GOLEK. Retrieved from Universitas Islam Indonesia: <https://dspace.uui.ac.id/handle/123456789/7843>
- [11] ololo. (2020, October). Clothing dataset (full, high resolution). Retrieved from kaggle: <https://www.kaggle.com/agrigorev/clothing-dataset-full>
- [12] Pinterest. (n.d.). Retrieved from Pinterest: <https://www.pinterest.nz/>
- [13] Salim, A. (2020, April 20). Object Detection (Case: Plat Detection). Retrieved from Medium: <https://medium.com/bisa-ai/object-detection-case-plat-detection-7cb5f53682ae>
- [14] S Shubathra, P. K. (2020). Clothing Image Recognition Based on Multiple Features Using Deep Neural Networks. 166-172. doi:10.1109/ICESC48915.2020.9155959
- [15] Teacher, T. E. (2019, May 16). Types of Clothing: Useful List of Clothing Names with the Picture. Retrieved from eslforum: https://eslforums.com/types-of-clothing/#Clothing_Names_with_Examples
- [16] Yulius Harjoseputro S.T., M. (2018). CONVOLUTIONAL NEURAL NETWORK (CNN) UNTUK PENGKLASIFIKASIAN AKSARA JAWA. Retrieved from <http://ejournal.uajy.ac.id/15485/1/TF04.15.897.pdf>