

Pengembangan Model Klasifikasi Produk Furnitur Sebagai *Visual Search* Menggunakan Algoritma *Convolutional* *Neural Network*

Akbar Ihsanul Ahadin^{1*}, Fida Maisa Hana², Agung Prihandono³, Imam Prayogo Pujiono⁴

^{1,2,3}Universitas Muhammadiyah Kudus, ⁴UIN K.H. Abdurrahman Wahid Pekalongan
¹32021110004 @std.umku.ac.id, ²fidamaisa@umkudus.ac.id, ³agungdono@umkudus.ac.id,
⁴imam.prayogopujiono@uingusdur.ac.id

Diterima : 30 Agustus 2024
Disetujui : 30 September 2024

Abstract— Dalam perkembangan teknologi yang pesat, kebutuhan terkait sistem pencarian visual yang efisien untuk produk furnitur semakin meningkat. Penelitian ini mengeksplorasi pengembangan model deep learning untuk pencarian dan klasifikasi produk furnitur berbasis gambar menggunakan framework Keras dan merancang model Convolutional Neural Network (CNN) yang mampu mengenali dan mengklasifikasikan gambar furnitur dari berbagai sudut pandang. Dataset gambar furnitur digunakan untuk melatih model, dengan hasil yang menunjukkan akurasi tinggi dalam klasifikasi multi-kelas. Model ini dioptimalkan untuk meningkatkan kinerja dan akurasi dalam pengenalan visual untuk memberikan solusi inovatif yang dapat diterapkan pada sistem pencarian berbasis gambar di industri retail dan e-commerce. Image search engine dalam penelitian ini menggunakan arsitektur VGG-16, sebuah metode convolutional neural network yang telah melalui proses pelatihan. Berdasarkan penelitian yang telah dilakukan, bisa disimpulkan bahwa implementasi CNN dengan library Keras memberikan hasil yang memuaskan dengan akurasi kecocokan data mencapai 97,78%.

Keywords — Convolutional Neural Network, Deep Learning, Deteksi Objek, VGG 16.

I. PENDAHULUAN

Teknologi cepat berkembang dalam memudahkan kegiatan manusia sehari-sehari. Kemudahan dalam memperoleh informasi relevan menggunakan mesin pencari merupakan salah satu manfaat teknologi. Ketika informasi yang di tampilkan sangat berguna sesuai dengan preferensi dari pengguna, disitulah mesin pencari dikatakan sangat bermanfaat. Berdasarkan data dari Kristen Purcell, Joanna Brenner, dan Lee Rainie, 91% pengguna mesin pencari mengungkapkan bahwa mereka hampir selalu menemukan apa yang mereka cari, serta sebanyak 73% mengungkapka bahwa informasi

yang diperoleh dari mesin pencari dapat dipercaya [1].

Mesin pencari gambar adalah jenis mesin pencari yang menampilkan informasi tentang suatu objek melalui gambar yang diambil dari objek tersebut. Salah satu keunggulan mesin pencari gambar adalah bahwa pengguna tidak perlu mengetahui detail objek yang ingin mereka cari, karena mereka dapat menemukan informasi terkait langsung dengan mengambil gambar objek tersebut. Mesin pencari gambar utamanya menyajikan informasi terkait dengan foto yang digunakan [2].

Penelitian ini akan mengembangkan mesin pencari berbasis gambar terkait sistem

pencarian visual yang efisien untuk produk furnitur semakin meningkat. Sistem ini menggunakan neuron gambar untuk melakukan perbandingan antara gambar objek yang sedang dicari dan seluruh gambar yang ada dalam basis data. Mesin pencari gambar ini menggunakan jaringan saraf konvolusional yang telah dilatih sebelumnya bernama VGG-16. Disebut sebagai VGG-16 karena memiliki 16 lapisan bobot [3]. Untuk memperoleh lapisan konvolusional, diperlukan Kernel Filter, yaitu sebuah filter yang memiliki fungsi untuk memproses informasi lokal di area tertentu. Filter tersebut memiliki bobot yang diatur dalam bentuk matriks, dan hasilnya akan membentuk lapisan konvolusional dengan dimensi yang lebih kecil, tapi tetap mewakili informasi dari masukkan atau input [4].

Program ini akan dibuat menggunakan bahasa Python, Python cukup mudah dipahami karena merupakan bahasa pemrograman tingkat tinggi yang berfokus pada keterbacaan kode. [5]. Selain itu, Python memiliki berbagai paket yang mendukung pembuatan mesin pencari gambar, seperti Matplotlib, NumPy, OpenCV, Scikit-learn, dan SciPy. Selain itu, Python memiliki fitur PIP, manajer paket, yang memudahkan pengembang menginstal paket yang diperlukan.

[6]. Python juga dapat digunakan di berbagai sistem operasi, seperti Mac, Linux, dan Windows[7]. Selain itu, penelitian ini melakukan analisis performa dengan mengevaluasi bagaimana perbedaan dalam kualitas, sudut pengambilan, warna, dan latar belakang gambar yang dimasukkan berdampak pada hasil pencarian. Gambar yang dimasukkan digunakan sebagai gambar referensi untuk pencarian.

II. TINJAUAN PUSTAKA

Karena pencapaiannya yang luar biasa dalam bidang visi komputer, Deep Learning baru-baru ini mendapat perhatian yang lebih besar daripada pengembangan pembelajaran mesin [8]. Deep learning terinspirasi oleh struktur korteks manusia; itu menggunakan jaringan saraf model yang terdiri dari banyak lapisan tersembunyi atau tersembunyi yang saling bertumpuk. Lapisan-

lapisan ini berfungsi sebagai metode atau algoritma yang memproses input dan menghasilkan output.

[9]. Convolutional Neural Network (CNN) adalah teknik deep learning yang sedang berkembang [10]. Jaringan ini menerima gambar sebagai input, yang kemudian diproses melalui beberapa lapisan konvolusi menggunakan filter yang telah diterapkan. Pola dari bagian-bagian gambar ini membantu proses klasifikasi [11]. CNN adalah metode Deep Learning yang dibuat untuk mengatasi kelemahan dari metode sebelumnya. Meskipun memiliki kekurangan, CNN mampu mengurangi jumlah parameter bebas dan menangani deformasi input seperti rotasi, translasi, dan perubahan skala [12]. Berbagai macam lembaga dan antarmuka aplikasi (API) mulai muncul dengan fokus pada jaringan saraf tiruan karena penelitian Deep Learning semakin meningkat.

Penelitian telah dilakukan untuk menemukan cara yang paling akurat. Dalam penelitian yang dilakukan oleh Austin Darian Pratama dan Junita pada tahun 2023, yang disebut Implementasi Pembelajaran Mesin untuk Membuat Search Engine Gambar Berdasarkan Analisis Similaritas, VGG-16 memiliki tingkat akurasi sebesar 95,33%, dengan sudut pengambilan gambar 96,25% dan kualitas gambar 68,25% sebagai penurunan akurasi terbesar[13]. Hal ini menunjukkan bahwa arsitektur VGG-16 unggul dalam klasifikasi gambar yang jelas dan seragam, tetapi dia masih perlu memperbaiki keterbatasannya dalam menangani berbagai perspektif dan variasi kualitas gambar.

Kemudian penelitian lain yang dilakukan oleh R. Q. Hassan, Z. N. Sultani, dan B. N. Dhannoon pada tahun 2023, dalam penelitian mereka yang berjudul *Content-Based Image Retrieval Based On Corel Dataset* ini menggunakan Deep Learning, dan penelitian tersebut mengusulkan teknik CBIR menggunakan CNN untuk ekstraksi fitur yang menggunakan dua lapisan penggabungan berbeda yang digunakan untuk mengekstrak fitur yaitu maks dan rata-rata. Pengukuran kesamaan Euclidean dan Manhattan digunakan untuk menghitung jarak antara fitur gambar kueri dan database. Hasil percobaan pada

dataset Corel 1K dengan Euclidean menunjukkan peningkatan yang signifikan pada rata-rata presisi sebesar 0,88 atau 88% ketika menggunakan rata-rata pooling dengan ukuran fitur 256 untuk mengambil 10 gambar pertama jika dibandingkan dengan metode lain yang telah diusulkan sebelumnya, seperti CNN dan SVM, SIFT, CH lokal dan global untuk fitur warna, DWT dan EDH untuk fitur tekstur, serta BoVW yang menggunakan dua ekstraksi fitur seperti HOG dan SURF, metode yang diusulkan lebih akurat dibandingkan pendekatan mutakhir yang sudah ada, serta cukup baik dan menjanjikan [14]. Meskipun pada penelitian tersebut pengukuran jarak Euclidean masih dapat dianggap kurang optimal dalam menangani masalah tekstur kompleks atau fitur geometris yang memerlukan pengolahan lebih mendalam.

Penggunaan VGG-16 menunjukkan potensi besar dalam penerapan untuk tugas-tugas klasifikasi gambar, terutama dalam image search engine. Keunggulan utama dari VGG-16 terletak pada arsitekturnya yang sederhana namun efektif, di mana lapisan-lapisan konvolusi yang berulang dengan filter kecil 3x3 yang memungkinkan model untuk mengekstraksi fitur secara mendetail dari gambar. Meskipun VGG-16 mampu memberikan akurasi tinggi, model ini memiliki beberapa keterbatasan, terutama dalam menghadapi variasi sudut pengambilan gambar dan kualitas gambar yang rendah.

Dalam penelitian ini, penggunaan VGG-16 sebagai basis arsitektur diusulkan karena kemampuannya dalam menangani tugas klasifikasi gambar yang kompleks. Namun, untuk mengatasi keterbatasan yang ada, penelitian ini akan mengeksplorasi penggunaan data augmentation dan fine-tuning model untuk meningkatkan ketahanan terhadap variasi sudut dan kualitas gambar. Dengan melakukan optimasi ini, diharapkan metode yang diusulkan dapat lebih baik dalam berbagai kondisi gambar, sehingga lebih cocok untuk aplikasi visual search pada produk furnitur yang sering kali dihadapkan pada berbagai variasi visual.

III. METODE PENELITIAN

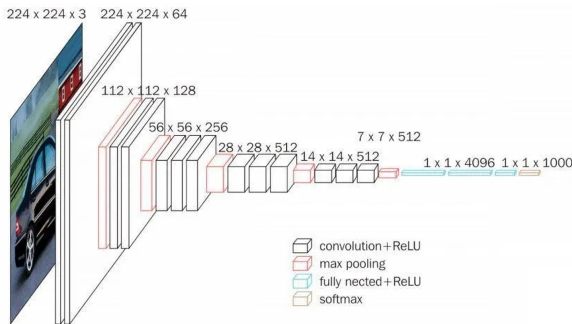
A. Convolutional Neural Network

Metode CNN yang digunakan untuk mengenali furnitur dalam penelitian ini terdiri dari beberapa tahap. Ini dilakukan untuk menutup kesenjangan semantik antara fitur mesin tingkat rendah dan persepsi manusia tingkat tinggi secara efektif, serta untuk menemukan gambar yang paling relevan dan meningkatkan kinerja pengambilan gambar. Penelitian ini dimulai dengan mengumpulkan data pelatihan untuk melatih jaringan saraf tiruan; arsitektur CNN yang digunakan adalah VGG-16, yang diambil dari Keras Library. Penelitian ini juga menggunakan jaringan saraf tiruan yang telah dilatih sebelumnya untuk tugas ekstraksi fitur. Jaringan saraf tersebut dilatih dengan data gambar furnitur yang dikumpulkan. Pelatihan adalah proses yang terdiri dari beberapa langkah. Langkah pertama adalah menemukan furnitur pada gambar. Setelah furnitur ditemukan, ciri-cirinya diekstraksi, yang menghasilkan nilai-nilai ciri untuk setiap furnitur. Nilai-nilai ini disimpan dan digunakan sebagai pengenalan atau classifier, dan pada tahap pengujian, klasifikasi furnitur dilakukan dengan classifier ini.

B. VGG-16

Dalam kompetisi ImageNet Large Scale Visual Recognition Challenge (ILSVRC) pada tahun 2014, Simonyan dan Zisserman memperkenalkan arsitektur Convolutional Neural Network VGG-16. VGG-16 memiliki 16 lapisan dengan parameter terlatih, termasuk 13 lapisan konvolusi dan 3 lapisan terhubung sepenuhnya. Setiap lapisan konvolusi menggunakan filter berukuran 3 kali 3 untuk mengekstraksi fitur-fitur penting dari gambar input, seperti tepi, tekstur, dan pola bentuk. Setelah setiap lapisan konvolusi, terdapat lapisan pooling yang mempertahankan informasi penting sambil mengurangi ukuran dimensi fitur. Karena arsitekturnya yang sangat sederhana, VGG-16 sangat baik untuk tugas klasifikasi gambar. Studi ini menggunakan VGG-16 sebagai model dasar, dengan modifikasi pada lapisan penuh terhubung untuk menyesuaikan dengan klasifikasi produk furnitur. Dengan menggunakan

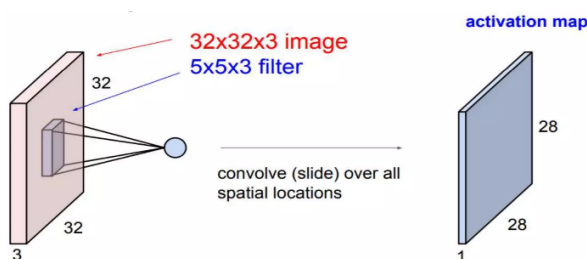
model pra-trained VGG-16, pelatihan dipercepat dan hasil yang baik dengan dataset yang ada. Gambaran arsitektur VGG-16 dapat dilihat pada Gambar 1.



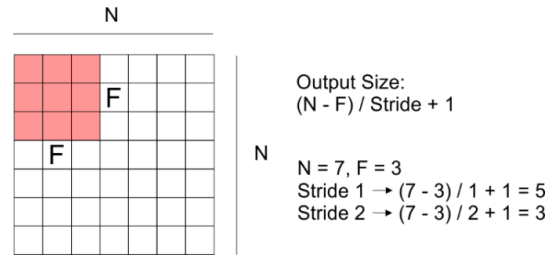
Gambar 1. Arsitektur VGG-16

B.I Convolutional Layer

Dalam arsitektur CNN, lapisan konvolusional adalah lapisan pertama yang menerima input gambar dan melakukan operasi konvolusi, menerapkan filter linier pada area lokal gambar. Setiap data yang masuk ke lapisan ini akan mengalami proses konvolusi. Lapisan filter ini memiliki dimensi panjang (piksel), lebar (piksel), dan kedalaman yang sesuai dengan channel gambar yang dimasukkan. Filter ini akan bergerak di seluruh bagian gambar, dan pada setiap perubahan, operasi "dot" dilakukan antara gambar masukan dan nilai-nilai filter, menghasilkan keluaran yang disebut sebagai peta fitur aktivasi. Tiga parameter, kedalaman (kedalaman), pergeseran (pergeseran), dan pengaturan zero padding, akan digunakan untuk mengoptimalkan kompleksitas model lapisan konvolusional [15]. Gambar 1 memperlihatkan proses konvolusi yang terjadi pada lapisan konvolusional, lalu Gambar 2 menunjukkan bagaimana cara menghitung nilai konvolusi tersebut.



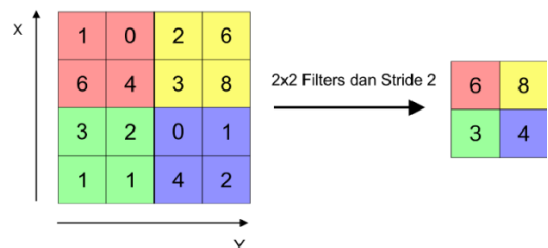
Gambar 2. Proses Dalam Konvolusi



Gambar 3. Rumus Perhitungan Konvolusi

B.II Pooling Layer

Setelah lapisan konvolusi, lapisan penyimpanan bertanggung jawab untuk mengurangi ukuran data citra dan menerima keluaran dari lapisan konvolusi. Filter dengan ukuran dan langkah tertentu digunakan pada lapisan penyimpanan. Seberapa jauh filter bergerak di seluruh area feature atau activation map akan ditentukan oleh jumlah langkah yang diambilnya setiap kali filter bergerak. Dengan Max Pooling 2x2 dan stride 2, nilai tertinggi dalam area 2x2 akan dipilih dalam setiap pergeseran filter. Pada saat yang sama, Average Pooling akan mengambil nilai rata-rata dari wilayah tersebut. Mengurangi ukuran data dan jumlah parameter yang diperlukan untuk tahap berikutnya, hasil akhir adalah gambar dengan ukuran yang lebih kecil dari yang pertama [16]. Gambar 3 memperlihatkan bagaimana proses di dalam Pooling Layer.

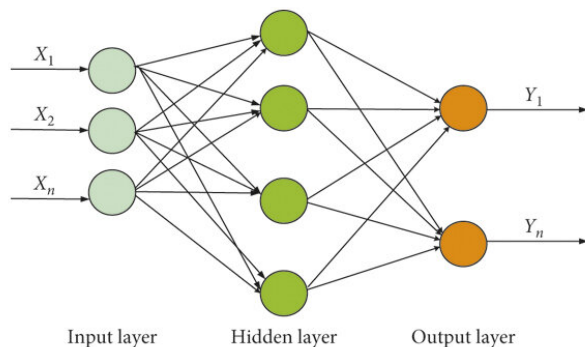


Gambar 4. Proses Pada Pooling Layer

B.III Fully Connected Layer

Karena feature map yang dihasilkan masih berupa array multidimensional, Lapisan Fully Connected menerima masukan dari Lapisan Pooling dalam bentuk feature map. Pada tahap ini, proses flatten, atau mengubah bentuk feature map menjadi vektor dengan n dimensi, di mana n adalah jumlah kelas output yang akan dipilih oleh program. Misalnya, jika layer tersebut memiliki 500 neuron, fungsi softmax akan digunakan

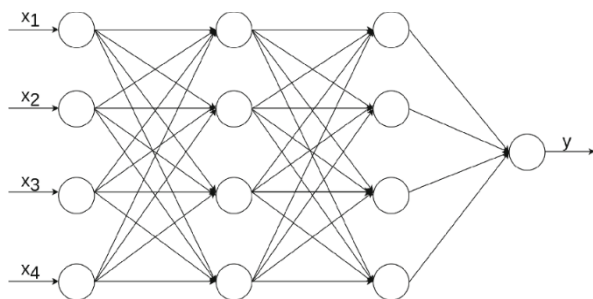
untuk mengembalikan daftar probabilitas tertinggi untuk 10 kelas label sebagai hasil akhir dari klasifikasi jaringan [17]. Gambar 4 menunjukkan bagaimana proses pada Fully Connected Layer.



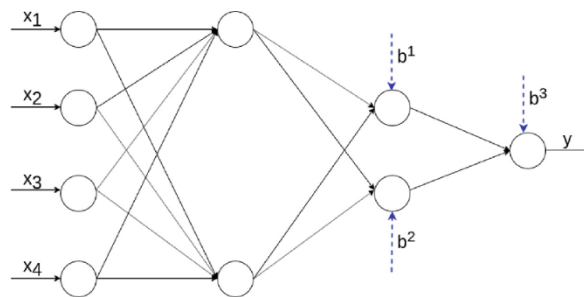
Gambar 5. Proses Dalam Lapisan Fully Connected

B.IV Dropout Layer

Dropout digunakan untuk mencegah overfitting serta mempercepat proses pembelajaran. Overfitting terjadi ketika model sangat baik dalam mengenali data yang digunakan selama pelatihan, tetapi tidak memberikan hasil yang baik pada data baru. Dalam praktiknya, Dropout bekerja dengan menonaktifkan sementara neuron di dalam jaringan, baik itu neuron di Hidden Layer maupun neuron di Visible Layer, sehingga membantu mengurangi kompleksitas model dan mencegah overfitting.



Gambar 6. Sebelum Terdapat Dropout Layer



Gambar 7. Setelah Terdapat Dropout Layer

B.V Softmax Layer

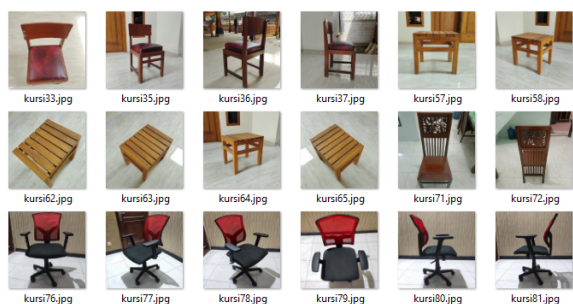
Lapisan softmax mengubah output jaringan saraf menjadi distribusi probabilitas di mana setiap nilai menunjukkan kemungkinan bahwa input termasuk dalam salah satu kelas yang ada. Lapisan ini digunakan pada lapisan terakhir model untuk mengklasifikasikan gambar produk furnitur ke dalam 14 kategori berbeda dalam tugas klasifikasi multi-kelas. Satu kelas produk diwakili oleh setiap output neuron softmax, dan nilai kemungkinan neuron tersebut menunjukkan tingkat keyakinan model terhadap prediksi kelas tersebut. Fungsi softmax memastikan bahwa jumlah total kemungkinan kelas sama dengan 1, yang memungkinkan untuk mengidentifikasi kelas dengan kemungkinan tertinggi sebagai hasil dari klasifikasi.

C. Dataset

Dataset penelitian ini dikumpulkan secara mandiri dengan mengambil foto dari berbagai jenis meja, kursi, dan sofa. Dataset foto akan digunakan sebagai input untuk proses di langkah-langkah berikutnya. Untuk memastikan bahwa semua data yang diperlukan sudah siap, persiapan dataset akan dilakukan. Dataset yang digunakan dalam penelitian ini terdiri dari empat belas produk, dengan masing-masing produk memiliki dua puluh gambar. Setiap subjek produk dibagi menjadi enam belas gambar, yang digunakan sebagai data pelatihan, dan empat gambar lagi digunakan sebagai data pengujian. Training, validation, dan testing adalah tiga komponen utama dari pengolahan dataset. Dataset tersebut mencakup beberapa kategori furnitur, masing-masing disusun dalam direktori berdasarkan labelnya. Gambar-gambar dari dataset pelatihan harus melalui tahapan wrapping

dan cropping sebelum masuk ke dalam proses pelatihan. Dalam proses pelatihan, ImageDataGenerator menerapkan peningkatan gambar yang mencakup fitur seperti rotasi, pergeseran, zoom, dan flip horizontal. Ini meningkatkan variasi data dan kemampuan model untuk mengenali pola.

Sebagian dataset yang digunakan pada penelitian ini dapat dilihat pada Gambar 7.



Gambar 8. Sampel Dataset Penelitian

C.I Dataset Preprocessing

Gambar-gambar dalam dataset awalnya memiliki resolusi 1080x1080 piksel, namun ukuran ini terlalu besar dan akan membebani sistem saat proses pelatihan. Oleh karena itu, dilakukan preprocessing data dengan mengubah ukuran gambar menjadi ukuran 224x224 piksel sebelum memasuki tahap pelatihan. Ukuran gambar tersebut sesuai dengan arsitektur VGG-16 yang digunakan dalam penelitian. Data training digunakan untuk melatih model, sementara data validation digunakan untuk mengukur performa model selama pelatihan dan mencegah overfitting.

C.II Pelatihan Dataset

Tahap pelatihan dataset adalah langkah awal yang memiliki tujuan untuk mengolah gambar-gambar yang sudah disiapkan. Pada proses ini, gambar inputan akan dilatih menggunakan metode Convolutional Neural Network (CNN) untuk membentuk model yang nanti akan diuji performanya. Model dilatih menggunakan arsitektur VGG-16, yang merupakan jaringan pre-trained yang berfungsi sebagai feature extractor. Augmentasi gambar digunakan selama pelatihan untuk meningkatkan kemampuan model dalam mengenali berbagai variasi gambar.

Pelatihan ini bertujuan untuk mendapatkan akurasi yang tinggi dalam klasifikasi..

C.III Pengujian Dataset

Setelah proses pelatihan, data yang sudah dilatih akan melalui proses klasifikasi. Evaluasi dilakukan menggunakan data testing yang belum pernah dilihat oleh model untuk menguji kemampuan generalisasi. Di mana hasil akhir dari tahap ini adalah tingkat akurasi yang mengukur kesesuaian antara data inputan dengan data yang ada di database. Hasil prediksi model diukur berdasarkan akurasi dan loss pada data validation dan testing untuk memastikan performa model terhadap data baru.

IV. HASIL DAN PEMBAHASAN

A. Ekstraksi Fitur

Proses ekstraksi fitur bertujuan untuk memperoleh informasi penting yang dapat digunakan untuk membedakan antara gambar produk yang beda dari citra produk yang sudah melalui proses penyalarsan. Setelah dilakukan preprocessing, termasuk deteksi, pemotongan (cropping), serta perubahan ukuran (resize) gambar produk, fitur-fitur diekstraksi menggunakan metode Convolutional Neural Network (CNN). Metode CNN yang digunakan dalam penelitian ini memanfaatkan Pre-Trained Neural Network untuk merepresentasikan produk furnitur. Proses ekstraksi ciri produk dengan metode Convolutional Neural Network memiliki beberapa tahapan layer deep neural network. Arsitektur Convolutional neural network selengkapnya dapat dilihat pada Tabel 1.

Tabel 1. Convolutional Neural Network Architecture Summary

Layer	Output Shape	Param #
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0

block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 0)	128
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 0)	256
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 0)	512
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
Block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
flatten (Flatten)	(None, 25088)	0
dense (Dense)	(None, 512)	12845568
dropout (Dropout)	(None, 512)	0
dense_1 (Dense)	(None, 14)	7182

Berdasarkan Tabel 1 digambarkan sebuah arsitektur model CNN pada penelitian ini mempunyai arsitektur yang terdiri dari beberapa lapisan konvolusi serta pooling, yang diikuti oleh layer fully-connected untuk melakukan klasifikasi. Pada input layer, model menerima gambar berukuran 224x224 piksel dengan 3 kanal

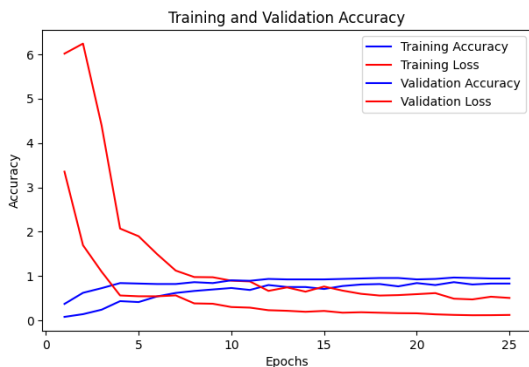
warna (RGB). Layer pertama terdiri dari dua layer konvolusi dengan 64 filter berukuran 3x3, diikuti oleh layer pooling dengan ukuran filter 2x2, yang mereduksi ukuran output menjadi 112x112x64. Proses ini dilanjutkan dengan dua layer konvolusi lagi, masing-masing dengan 128 filter, diikuti oleh layer pooling yang mereduksi ukuran output menjadi 56x56x128. Selanjutnya, terdapat tiga layer konvolusi masing-masing dengan 256 filter, diikuti oleh layer pooling yang mengurangi ukuran output menjadi 28x28x256. Kemudian, terdapat tiga layer konvolusi lagi dengan 512 filter, yang diikuti oleh layer pooling dengan ukuran output 14x14x512. Bagian akhir dari model konvolusi terdiri dari tiga layer konvolusi tambahan dengan 512 filter dan layer pooling yang menghasilkan output akhir berukuran 7x7x512.

Setelah semua layer konvolusi dan pooling, output diratakan menjadi vektor satu dimensi dengan ukuran 25088, yang kemudian dilewatkan ke layer fully-connected dengan 512 neuron. Untuk mencegah overfitting, digunakan layer dropout dengan tingkat dropout sebesar 50%. Terakhir, layer fully-connected dengan 14 neuron digunakan untuk menghasilkan output akhir yang sesuai dengan jumlah kelas yang akan diklasifikasikan. Model ini memiliki total 27.567.438 parameter, dengan 12.825.750 parameter yang dapat dilatih dan 14.741.688 parameter yang tidak dapat dilatih. Arsitektur ini dirancang untuk menangkap fitur-fitur kompleks dari gambar input dan melakukan klasifikasi dengan akurasi tinggi.

B. Hasil Pengujian dan Analisis

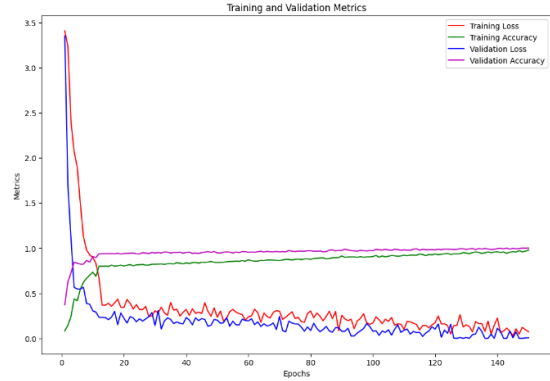
Dalam penelitian ini, pelatihan menggunakan hyperparameter meliputi learning rate, batch size, jumlah epoch, dropout rate, dan juga menggunakan optimizer adam yang dapat melakukan pembaruan bobot secara adaptif untuk menjaga keseimbangan antara kecepatan konvergensi dan stabilitas. Menggunakan batch size sebesar 32 untuk menentukan jumlah gambar yang diproses setiap kali model melakukan pembaruan bobot. Kemudian diterapkan dropout rate sebesar 0.5 pada lapisan fully connected, yang akan menonaktifkan 50% neuron secara

acak selama pelatihan untuk membantu meningkatkan kemampuan generalisasi model terhadap data baru dan juga untuk mencegah overfitting. Kombinasi hyperparameter ini dirancang untuk memastikan bahwa model dapat mempelajari fitur-fitur penting dari citra produk furnitur dengan baik, sekaligus meminimalkan kesalahan prediksi selama proses klasifikasi. Pengujian juga dilakukan untuk mengevaluasi pengaruh dari jumlah epoch atau langkah pelatihan pada performa sistem. Pada pengujian, dilakukan 25 epoch dan 150 epoch untuk memberikan waktu yang cukup bagi model untuk belajar dari dataset supaya mencapai performa yang optimal. Hasil pengujian dapat dilihat pada Gambar 7 dan Gambar 8.



Gambar 9. Hasil Accuracy dan Loss pada epoch 25

Pada pelatihan dengan epoch 25, model menunjukkan kinerja yang baik dengan training loss sebesar 56,10% dan accuracy sebesar 78,57%. Ini menandakan bahwa model mulai mempelajari pola dari data latih dengan cukup baik, karena akurasi sudah mencapai lebih dari 75%. Dengan validation loss sebesar 8,66%, menunjukkan bahwa model sudah dapat menangani data baru dengan baik, dan juga validation accuracy sebesar 96,88% menunjukkan bahwa model cukup mampu mengenali pola dari data validasi.



Gambar 10. Hasil Accuracy dan Loss pada epoch 150

Pada pelatihan dengan epoch 150, model mengalami peningkatan yang signifikan. Training loss turun menjadi 7,41% dan accuracy meningkat menjadi 97,78%, hal ini menunjukkan bahwa model semakin baik dalam mengenali pola dari data latih. Selain itu, validation loss turun drastis menjadi 0,68% dan validation accuracy mencapai 100%, yang menandakan bahwa model telah mempelajari pola dari data validasi dengan sangat bagus dan dapat menggeneralisasi pola dari data baru dengan sempurna. Hasil ini mengindikasikan bahwa model berhasil dioptimalkan dengan baik, karena baik akurasi maupun loss di data latih dan validasi konsisten berada di tingkat yang sangat baik.

Berdasarkan dari hasil pengujian, dengan dataset yang berjumlah 280 gambar berukuran 224x224 piksel dan epoch 25 selama proses pelatihan, maka diperoleh tingkat akurasi sebesar 78,57%. Sedangkan, pada pengujian dengan menggunakan epoch 150, menghasilkan tingkat akurasi yang meningkat menjadi 97,78%. Hal ini menunjukkan bahwasanya jumlah epoch sangat mempengaruhi tingkat akurasi, semakin banyak epoch, semakin baik pula akurasi.

V. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, metode Convolutional Neural Network berhasil diimplementasikan menggunakan library Keras dengan arsitektur VGG-16, menghasilkan tingkat akurasi klasifikasi produk furnitur yang mencapai 97,78%. Proses pelatihan menunjukkan bahwa semakin besar jumlah epoch yang digunakan, semakin baik akurasi yang diperoleh, dengan optimalisasi menggunakan optimizer Adam.

Dibandingkan dengan metode lain seperti Support Vector Machine dan k-Nearest Neighbors yang bergantung pada fitur manual, VGG-16 secara otomatis mengekstrak fitur lebih kompleks, yang terbukti lebih unggul dalam menangani variasi visual. Penggunaan VGG-16 memberikan performa akurasi yang baik dan terbukti efektif untuk klasifikasi produk furnitur dalam visual search.

DAFTAR PUSTAKA

- [1] Purcell K, Brenner J, and Rainie L, "Search Engine Use 2012," *Pew Research Center*. 2012, [Online]. Available: <https://www.pewresearch.org/internet/2012/03/09/search-engine-use-2012/>.
- [2] M. A. Sini and B. A. Ata, "Web Image Search Engine Evaluation," *International Arab Journal of e-Technology*, vol. 3, no. 2. 2013.
- [3] Kuruva Satya Ganesh, "What's The Role Of Weights And Bias In a Neural Network?," *Towards Data Science*. 2020, [Online]. Available: <https://towardsdatascience.com/whats-the-role-of-weights-and-bias-in-a-neural-network-4cf7e9888a0f>.
- [4] S. Hijazi, R. Kumar, and C. Rowen, "Using Convolutional Neural Networks for Image Recognition," *Cadence Design Systems Inc.: San Jose, CA, USA.*, vol. 8925. pp. 572–578, 2015.
- [5] K. R. Srinath, "Python—the fastest growing programming language." *International Research Journal of Engineering and Technology*, 4(12), 354-357, 2017.
- [6] A. Bogdanchikov, M. Zhaparov, and R. Suliyev, "Python to learn programming," *Journal of Physics: Conference Series*, vol. 423, no. 1. 2013, doi: 10.1088/1742-6596/423/1/012027.
- [7] V. K. Sharma, V. Kumar, S. Sharma, and S. Pathak, "Introduction to Python Programming," *Python Programming*. pp. 1–8, 2021, doi: 10.1201/9781003185505-1.
- [8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 2. pp. 1097–1105, 2012.
- [9] M. H. Ahmad, M. Hana, T. Ghazi Pratama, and H. Aulida, "Klasifikasi Empat Jenis Daun Herbal Menggunakan Metode Convolutional Neural Network," *Jurnal Ilmu Komputer dan Matematika*, vol. 4, no. 2. pp. 69–76, 2023.
- [10] K. U. Haqiqi, F. M. Hana, and H. Aulida, "Klasifikasi Jenis Golongan Kendaraan Di Gerbang Tol Menggunakan Arsitektur Cnn Vgg16," *Jurnal Ilmu Komputer dan Matematika*, vol. 5, no. 1. pp. 10–15, 2024.
- [11] Alwafi Ridho Subarkah, "IMPLEMENTASI DEEP LEARNING MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI ALAT TULIS (Studi)," *Nhk 技研*, vol. 151, no. 2. pp. 10–17, 2018.
- [12] S. Haykin and B. Kosko, "GradientBased Learning Applied to Document Recognition," *Intelligent Signal Processing*. 2010, doi: 10.1109/9780470544976.ch9.
- [13] F.-J. Sains and T. Vol, "Implementation of Machine Learning To Create an Image Search," vol. 7, no. 2, pp. 139–147, 2023.
- [14] R. Q. Hassan, Z. N. Sultani, and B. N. Dhannoon, "Content-based image retrieval based on corel dataset using deep learning," *IAES Int. J. Artif. Intell.*, vol. 12, no. 4, pp. 1854–1863, 2023, doi: 10.11591/ijai.v12.i4.pp1854-1863.
- [15] A. Saxena, "An Introduction to Convolutional Neural Networks," *International Journal for Research in Applied Science and Engineering Technology*, vol. 10, no. 12. pp. 943–947, 2022, doi: 10.22214/ijraset.2022.47789.
- [16] N. Shukla and F. Kenneth, "Machine learning with Tensorflow," *Manning*, vol. 07, no. 06. p. 274, 2018.
- [17] B. P. M, L. K. Likhitha, and D. S. Rajesh, "Handwritten Digit Recognition Using Deep Learning," *International Journal of Scientific Research in Science and Technology*. pp. 153–158, 2021, doi: 10.32628/cseit217439.