

Klasifikasi Pisang Berbasis Algoritma VGG16 Melalui Metode CNN Deep Learning

Jayadi Halim, Ahmad Nurul Fajar

Binus Graduate Program – *Master of Information System Management*
Bina Nusantara University, Jl. Kebon Jeruk Raya no 27 Kebon Jeruk, Jakarta 11530

Email : jayadi.halim@binus.ac.id

ABSTRAK

Pisang *cavendish* banyak dikonsumsi di Indonesia dan berpotensi menjadi komoditas utama di Indonesia. Namun, proses pemilihan kualitas pisang di Indonesia masih banyak yang dilakukan secara tradisional. Hal ini menjadi penghambat pisang dalam menjadi komoditas utama. Klasifikasi mutu modern dapat dilakukan untuk memperbaiki proses seleksi kualitas pisang dalam meningkatkan penjualan di sektor pertanian. Peningkatan penjualan pisang di sector pertanian akan menjadikan pisang sebagai komoditas utama dan meningkatkan ekonomi Indonesia. Metode *deep learning* yaitu CNN dengan model VGG16 dapat diimplementasikan sebagai solusi dari permasalahan tersebut. Peneliti akan mencoba menggunakan berbagai jumlah epoch untuk mendapatkan hasil evaluasi yang terbaik. Variabel dibagi 5 dengan total kumpulan data gambar adalah 550. Kumpulan data juga dibagi dengan latihan dan tes dengan persentase 70%: 30%. Hasil eksperimen menunjukkan hasil performa terbaik pada epoch 50 dengan akurasi train 98.96% dan akurasi test 83.53%. Model akan disimpan dan akan digunakan oleh para pelaku dalam industri pertanian di Indonesia.

Kata kunci: *Convolutional Neural Network (CNN), VGG16, Klasifikasi Gambar, Pisang, Deep Learning*

ABSTRACT

Cavendish bananas are widely consumed in Indonesia and have the potential to become a major commodity in Indonesia. However, the process of selecting the quality of bananas in Indonesia is still mostly done traditionally. This is an obstacle for bananas to become the main commodity. Modern quality classification can be done to improve the process of quality selection of bananas in increasing sales in the agricultural sector. Increased sales of bananas in the agricultural sector will make bananas a major commodity and boost the Indonesian economy. The deep learning method, namely CNN with the VGG16 model, can be implemented as a solution to this problem. Researchers will try to use various epoch numbers to get the best evaluation results. The variable divided by 5 with the total image data set is 550. The data set is also divided by exercises and tests with the percentage of 70%: 30%. Experimental results show the best performance results on epoch 50 with 98.96% train accuracy and 83.53% test accuracy. The model will be stored and will be used by actors in the agricultural industry in Indonesia.

Keywords: *Convolutional Neural Network (CNN), VGG16, Image Classification, Banana, Deep Learning*

1. PENDAHULUAN

Indonesia merupakan negara yang dikenal dengan keanekaragaman buah-buahan tropis dan memiliki potensi besar untuk mengembangkan industri pertanian. Indonesia disebut negara agraris karena sektor pertanian memegang peranan penting dalam perekonomian dan penduduknya banyak bekerja di sektor pertanian. Sesuai dengan namanya, Indonesia memiliki banyak sumber daya alam yang melimpah karena letak geografis Indonesia yang sangat baik dan mendukung pertumbuhan tanaman. Indonesia berada di daerah tropis yang memiliki curah hujan yang tinggi, sehingga membuat kondisi tanah dan lahan di Indonesia menjadi subur dan banyak berbagai jenis tumbuhan yang dapat tumbuh dengan kualitas yang baik. Salah satu buah yang berpotensi menjadi komoditas di Indonesia adalah pisang. Pisang merupakan buah yang mudah didapatkan, memiliki nilai ekonomi, budaya, dan gizi yang tinggi (Nuramanah et al., 2013).

Pisang merupakan buah yang tumbuh dimana saja, di desa, kota, tidak musiman dan sangat mudah didapatkan di seluruh Indonesia. Hal ini menyebabkan pisang sangat bermanfaat karena pisang berperan sebagai buah yang bergizi, sebagai sumber vitamin, mineral dan karbohidrat yang melimpah. Pisang memiliki kandungan utama berupa tepung atau pati dan berbagai jenis gula. Pisang mengandung gula yang terdiri dari senyawa-senyawa yang sangat mudah dicerna oleh tubuh manusia, seperti sukrosa 2%, fruktosa 3,6%, dan fruktosa 4,6% (Utomo et al., 2019). Daging buah pisang mengandung berbagai vitamin seperti vitamin A, vitamin B, dan vitamin C. Pisang juga mengandung mineral seperti kalsium, fosfor, dan zat besi.

Pisang merupakan buah yang memiliki tingkat produksi yang sangat tinggi di Indonesia. Badan Pusat Statistik (BPS) mencatat produksi pisang di Indonesia mencapai 8,18 juta ton pada 2020. Angka itu meningkat 12,39% dari 7,28 juta ton pada 2019. Data lain dari Badan Pusat Statistik juga mencatat pisang merupakan buah yang paling banyak diproduksi di Indonesia. Pada tahun 2020, produksi pisang akan mencapai 8,18 juta ton dan jika dibandingkan dengan buah-buahan lain yang diproduksi di Indonesia, pisang memiliki keunggulan yang sangat jauh. Melalui data tersebut dapat diasumsikan bahwa pisang memiliki potensi yang sangat tinggi di Indonesia untuk menjadi komoditas utama di sektor pertanian.

Permasalahan yang sering terjadi pada industri perkebunan pisang adalah cara pemilahan pisang yang didistribusikan ke pasar masih manual. Misalnya, berbagai jenis perkebunan pisang yang merupakan daerah usaha di Lampung dengan jumlah produksi yang sangat banyak juga masih menggunakan cara manual dalam memilah kualitasnya. Petani akan memberikan semua hasil panennya dan akan diangkut dengan transportasi. Kemudian, mulai dari pengepul daerah hingga industri provinsi dan daerah yang akan memilah kualitas pisang yang dikirim ke pengecer secara manual. Kegiatan ini tentunya melibatkan lebih banyak sumber daya manusia dan tidak lepas dari human error. Hal ini tentunya akan merugikan retailer dan akan mempengaruhi kepuasan pelanggan.

Pemilihan kualitas pisang di Indonesia sangat penting untuk meningkatkan penjualan dan jumlah konsumsi. Selain itu, kualitas pisang yang baik juga memungkinkan distribusi ekspor untuk membantu meningkatkan perekonomian sektor pertanian. Proses pemilihan dan pengelompokan buah pisang dapat dilihat melalui warna buah pisang.

Dalam beberapa tahun terakhir, telah banyak penelitian tentang klasifikasi citra menggunakan *CNN* dengan algoritma *VGG16*. Hasil tersebut juga menyatakan bahwa *VGG16* merupakan *model* yang baik karena memiliki tingkat akurasi yang tinggi. S. Zhenzhen, L. Fu, J. Wu, Z. Liu, R. Li dan Y. Cui (Song et al., 2019) menemukan bahwa *model Faster R-CNN* yang diimplementasikan oleh *VGG16* dikembangkan dan dievaluasi pada gambar buah kiwi yang diambil di kebun pada waktu yang berbeda-beda, pagi, siang dan malam, masing-masing dengan atau tanpa flash. Hasil penelitian menunjukkan bahwa *model VGG16* berdasarkan *Faster R-CNN* memiliki kinerja pendeteksian yang baik dan ketahanan terhadap situasi oklusi yang berbeda, bidang yang tumpang tindih, dan isyarat yang berdekatan. *Model VGG16* berbasis *R-CNN* yang lebih cepat secara efektif mengklasifikasikan dan mendeteksi kiwi di wilayah gambar kandidat. Selain itu, *model VGG16* mengenali gambar kiwi yang dikumpulkan dalam kondisi pencahayaan berbeda (pagi, siang, dan malam) dengan baik, dan ada juga kiwi yang tidak terdeteksi sebagai negatif palsu. Ketika kumpulan data kompleks dan berisi banyak gambar, *VGG16* mencapai pengenalan yang lebih baik daripada *ZFNet* (72,50%) yang dilatih dan diuji pada kumpulan data gambar yang sama dengan *VGG16*. Ini menunjukkan bahwa fitur yang diekstraksi dari jaringan *VGG16* yang digunakan dalam pekerjaan ini lebih efisien daripada *ZFNet* pada kumpulan data gambar yang

sama. *Model Faster R-CNN* berbasis *VGG16* dapat lebih akurat mengenali buah kiwi di lapangan. Sistem tersebut dapat secara efektif mendeteksi berbagai jenis buah di lapangan dan memberikan dukungan yang kuat untuk robot pemanen yang dapat bekerja sepanjang hari selama musim sibuk.

Li J, Ablan C, Wu R, Guan S, Yao J. (Li et al., 2021) menemukan bahwa peningkatan kinerja jaringan *F-RCNN VGG16* dengan menggunakan metode peningkatan warna terbukti paling berpengaruh dalam meningkatkan presisi rata-rata detektor, peningkatan warna yang kecil tidak memengaruhi perubahan kinerja detektor yang dihasilkan, tetapi peningkatan yang terlalu banyak akan berdampak negatif pada kinerja, faktor lainnya adalah variasi *dataset* yang menyebabkan fitur yang bertentangan dan memengaruhi kinerja seperti pada uji peningkatan warna. Ucat RC, Cruz JC (Ucat dan Cruz, 2019) menggunakan teknik *python OpenCV* dan *Tensorflow* untuk mengklasifikasikan grade pisang cavendish. *Dataset* berisi citra 1116 buah citra pisang dalam 4 kelas (*Class A Big Hand, Class A Small Hand, Class B Big Hand, Class B Small Hand*). Selain *dataset*, 279 gambar diklasifikasikan ke dalam setiap kelas. Hasil akhir *model* klasifikasi *CNN* pada *train, test* dan validasi ke-4 kelas menunjukkan akurasi di atas 90%.

Studi lain juga dilakukan (Maulana dan Rochmawati, 2020) menemukan bahwa *model CNN* yang menggunakan kombinasi tiga convolutional layer dan dua hidden layer dapat mengklasifikasikan citra buah dengan akurasi yang baik. Akurasi yang diperoleh dari proses pengujian menggunakan 345 citra uji adalah 97,97%. *Model CNN* yang dibuat pada penelitian ini juga dapat mengklasifikasikan citra buah yang ditangkap oleh kamera smartphone. *Model CNN* mengklasifikasikan citra buah yang tidak dikenal ke dalam kelas buah yang dianggap paling mirip dari kelas buah yang diteliti.

Segmentasi penelitian lain (H. A. Moh et al., 2021) menggunakan algoritma K-means clustering, proses ekstraksi fitur menggunakan teknik pembelajaran *transfer VGG16*, dan klasifikasi menggunakan *CNN*. *Dataset* berasal dari Kaggle dan berisi gambar 4000 tanaman merambat daun dalam 4 kelas (bintik hitam, bercak daun, daun sehat, busuk). Selain kumpulan data, 100 gambar *Google* juga digunakan sebagai data uji. Akurasi pelatihan *model CNN* adalah 99,50%. Klasifikasi menghasilkan akurasi 97,25% menggunakan data uji, sedangkan akurasi 95% dicapai dengan menggunakan

data gambar uji *off-dataset*. Metode pengolahan citra yang dirancang dapat diterapkan untuk mengidentifikasi dan mengklasifikasikan pola penyakit daun anggur.

Melalui penjelasan di atas, solusinya adalah pembelajaran mesin yang membutuhkan kumpulan data untuk dipelajari atau diawasi. Pengembangan *machine learning* akan dibantu dengan menggunakan *convolutional neural network* dengan arsitektur *VGG16*. Oleh karena itu, penelitian ini juga akan menggunakan arsitektur *VGG16* yang dioptimalkan sesuai dengan kebutuhan. Hasil penelitian ini diharapkan memiliki tingkat akurasi yang tinggi dengan minimal 80%, sehingga dapat diimplementasikan dengan baik dalam proses pemilihan kualitas citra pisang sehingga proses bisnis pada perkebunan pisang menjadi lebih efektif dan efisien, yang berdampak pada tingkat kepuasan konsumen terhadap buah pisang yang beredar di pasaran. pasar yang menyebabkan perekonomian sektor perkebunan pisang meningkat.

2. METODE PENELITIAN

Metodologi penelitian dalam penelitian ini menggunakan *model* pembelajaran mendalam *CNN*, dengan algoritma *pre-trained model* VGG 16 yang telah dioptimalkan mengikuti *model build* yang telah diuji sebelumnya oleh A. Mohit, S. K. Gupta dan K. K. Biswas (Agarwal et al., 2020).

Pengumpulan Data






Populasi yang akan digunakan dalam penelitian ini adalah citra buah pisang yang banyak dijumpai di Indonesia dan banyak beredar di pasaran. Sedangkan sampelnya adalah jenis pisang *cavendish* dengan jumlah sampel yang digunakan dalam penelitian ini sebanyak 550 buah data citra pisang yang dibagi menurut masing-masing variabel kategori yaitu 110 buah data citra pisang mentah, cukup matang 110 buah. data citra pisang, 110 data citra pisang matang, 110 data citra pisang terlalu matang, dan 110 data citra pisang busuk.

Sumber data dalam penelitian ini menggunakan data primer yang artinya data diambil langsung oleh peneliti dengan menggunakan kamera *smart phone Apple iPhone 11*. Selain itu, penempatan foto setiap gambar pisang akan dilakukan di tempat yang sama dengan kualitas cahaya yang baik. Hal ini berguna agar kualitas citra pada foto memiliki pencahayaan yang sama pada setiap citra dan meningkatkan akurasi pendataan citra pisang (Pardede et al., 2021).

Metode analisis data dalam penelitian ini adalah convolutional neural *network* yang merupakan bagian dari *deep learning*. Model pra-pelatihan arsitektur yang digunakan adalah *VGG16*. *Anaconda Navigator*, *Jupyter Notebook* dan *Google Studio* merupakan *software* dengan bantuan *hard package* yang akan digunakan untuk membantu proses analisis data dalam penelitian ini. Metode convolutional neural *network* dengan arsitektur *VGG16* akan digunakan untuk membantu mengklasifikasikan citra dari 5 variabel pisang.

Variabel

Tabel 1. *Research Variables*

Gambar Citra	Variabel	Definisi
	<i>raw banana</i>	Gambar pisang mentah adalah pisang dengan warna dominan hijau.
	<i>slightly ripe banana</i>	Citra pisang yang agak matang adalah pisang yang memiliki kualitas bagus dan memiliki warna dominan kuning kehijauan.
	<i>ripe bananas</i>	Citra pisang matang adalah pisang yang memiliki kualitas bagus dan warna dominan kuning, disertai sedikit bercak hitam.
	<i>bananas too ripe</i>	Citra pisang yang terlalu matang memiliki banyak bintik hitam.
	<i>rotten banana</i>	Citra pisang busuk adalah pisang yang memiliki tekstur gelap, dan warna yang dominan adalah hitam.

Convolutional Neural Network - VGG16 Model

Perancangan model *CNN-VGG16* pada penelitian ini akan menggunakan *filter* 64, 128, 256 dan 512. Kemudian, *kennel* yang digunakan adalah 3x3 dan *max pooling*

adalah 2x2. Perancangan arsitektur *CNN-VGG16* pada penelitian ini dibagi menjadi 5 bagian.

Tabel 2. *CNN VGG16 Algorithm Model*

<i>Layer</i>	<i>Output</i>
Input	224 x 224 x 3
64 Conv <i>Filter</i> dan Kernel (3x3)	224 x 224 x 64
64 Conv <i>Filter</i> dan Kernel (3x3)	224 x 224 x 64
Max Pooling (2x2)	112 x 112 x 64
128 Conv <i>Filter</i> dan Kernel (3x3)	112 x 112 x 128
128 Conv <i>Filter</i> dan Kernel (3x3)	112 x 112 x 128
Max Pooling (2x2)	56 x 56 x 128
256 Conv <i>Filter</i> dan Kernel (3x3)	56 x 56 x 256
256 Conv <i>Filter</i> dan Kernel (3x3)	56 x 56 x 256
256 Conv <i>Filter</i> dan Kernel (3x3)	56 x 56 x 256
Max Pooling (2x2)	28 x 28 x 256
512 Conv <i>Filter</i> dan Kernel (3x3)	28 x 28 x 512
512 Conv <i>Filter</i> dan Kernel (3x3)	28 x 28 x 512
512 Conv <i>Filter</i> dan Kernel (3x3)	28 x 28 x 512
Max Pooling (2x2)	14 x 14 x 512
512 Conv <i>Filter</i> dan Kernel (3x3)	14 x 14 x 512
512 Conv <i>Filter</i> dan Kernel (3x3)	14 x 14 x 512
512 Conv <i>Filter</i> dan Kernel (3x3)	14 x 14 x 512
Max Pooling (2x2)	7 x 7 x 512
Flatten	25088
Dense	5

1. *64 convolution filter*

Konvolusi pertama akan menggunakan *kennel* 3x3 dengan *filter* 64. Langkah selanjutnya adalah *max pooling* dengan ukuran 2x2.

2. *128 convolution filter*

Konvolusi kedua dan ketiga akan menggunakan *kennel* 3x3 dengan *filter* 64. Langkah selanjutnya adalah *max pooling* dengan ukuran 2x2.

3. *256 convolution filter*

Konvolusi keempat, kelima dan keenam akan menggunakan *kennel* 3x3 dengan *filter* 64. Langkah selanjutnya adalah *max pooling* dengan ukuran 2x2.

4. *512 convolution filter*

Konvolusi ketujuh hingga kedua belas akan menggunakan *filter* 512 yang terbagi menjadi 2 bagian. Bagian pertama pada konvolusi ketujuh sampai kesembilan

menggunakan *kernel* 3x3, serta *max pooling* dengan ukuran 2x2. Kemudian, bagian kedua adalah tahap konvolusi kesepuluh hingga kedua belas yang menerapkan proses yang sama dengan *filter* 512 bagian pertama. Pada bagian kedua, proses berfungsi untuk mengekstrak data citra yang digunakan. Setiap proses konvolusi, *filter* akan bertambah karena untuk mengekstrak data citra pada proses sebelumnya yang lebih kecil, diperlukan *filter* yang lebih besar pada konvolusi berikutnya.

5. *Fully Connected Layer*

Langkah terakhir dalam *model* arsitektur *CNN-VGG16* adalah proses pembentukan ulang peta fitur. Proses konvolusi sebelumnya akan menghasilkan *array* multidimensi. Proses reshape peta fitur berguna untuk mengubah format *array* multidimensi yang dihasilkan oleh proses konvolusi menjadi format vektor. Format vektor adalah format input untuk lapisan yang terhubung sepenuhnya. Langkah selanjutnya adalah secara bertahap menggunakan 25088 *neuron* pada lapisan tersembunyi. Setelah melalui proses tersebut, akan ditampilkan hasil klasifikasi data citra berdasarkan kategori masing-masing.

Data Pre-Processing

Pada tahap ini, peneliti perlu mencari pisang *cavendish* mentah. Kemudian peneliti harus menyediakan tempat yang sama, dengan tingkat cahaya dan pencahayaan yang sama sebagai tempat pengambilan gambar pisang. Kertas putih juga disediakan sebagai alas pisang dalam berfoto. Peneliti memotret pisang setiap hari, mulai dari pisang mentah hingga busuk dalam jumlah sebanyak mungkin.

Dalam penelitian ini, peneliti membutuhkan waktu 7 hari untuk mengambil gambar dari mentah hingga busuk. Dengan ini, peneliti memiliki 7 variabel citra pisang dengan jumlah yang berbeda. Peneliti harus memilih 5 variabel pisang yang paling cocok dengan citra pisang yang telah ditentukan di awal dan memilih 110 citra terbaik untuk masing-masing variabel. Terakhir, peneliti harus menyediakan 1 *folder* untuk menampung gambar pisang dan *folder* tersebut berisi 5 *folder* sesuai jumlah variabel dan berisi 110 foto gambar pisang di dalamnya.

Ketika semua gambar pisang sudah terkumpul, peneliti harus mengubah ekstensi *file* menjadi ekstensi yang sama (.png) dan memotong gambar hingga fokus pada objek gambar pisang (menghilangkan warna putih dari latar belakang kertas).

Pixel citra juga perlu diubah menjadi ukuran 600 x 600 karena *pixel* awal setelah di-*crop* masih terlalu tinggi, yaitu sekitar 1200 x 1200. Kemudian, citra awal yang dimasukkan saat proses pengujian algoritma adalah 224 x 224. Dengan ukuran tersebut, *pixel* yang disarankan adalah 2-3 kali lipat dari 224 x 224 dengan tujuan agar selama proses uji coba citra tidak terlalu banyak kehilangan *pixel* yang menyebabkan citra menjadi kabur dan tingkat akurasi menurun (Liu et al., 2020).

3. HASIL DAN PEMBAHASAN

Hasil yang diperoleh dari akurasi dan kehilangan data ditampilkan setelah mengimpor perpustakaan, mengatur direktori dari drive, deteksi objek, memuat data, eksplorasi data, *model* VGG khusus yang dibuat dan mengevaluasi kinerja. Semua ini adalah proses yang dilakukan di perangkat lunak *Google Collab* yang terhubung ke drive dan perpustakaan online. Pada penelitian ini semua *model* diuji dengan menggunakan perbandingan data latih: pengujian yaitu 70%:30%. Selanjutnya, peneliti mencoba membandingkan hasil dari total 10, 20, 30, 40 dan 50 *epoch* untuk melihat hasil *epoch* mana yang terbaik. Hasil terbaik akan digunakan sebagai *model* masa depan untuk meningkatkan tingkat evaluasi kinerja dan hasil prediksi klasifikasi yang lebih baik. Hasil performance *model* pada *epoch* 10, 20, 30, 40 dan 50 dapat dilihat pada sub bab dibawah ini.

Hasil Performance Model

Hasil performance *model* pada *epoch* 10, 20, 30, 40 dan 50 dapat dilihat pada tabel dibawah ini.

Tabel 3. Summary *Train* dan *Test Accuracy* Result

Total Epochs	Train Accuracy	Test Accuracy	Total Epochs	Train Accuracy
10	93.54%	77.57%	10	93.54%
20	97.92%	81.21%	20	97.92%
30	98.22%	82.42%	30	98.22%
40	98.44%	80.01%	40	98.44%
50	98.96%	83.53%	50	98.96%

Epoch pada proses *model* jaringan saraf yang berulang untuk melihat keseluruhan *dataset*. Dengan kata lain, berapa kali *model* neural network mengulang pengujian pada

dataset untuk mendapatkan nilai akurasi (Hasan et al., 2018). *Loss* merepresentasikan *model* neural *network* yang mencoba untuk meminimalkan: perbedaan jarak antara label sebenarnya dan label yang diprediksi. Untuk meminimalkan jarak ini, neural *network* belajar dengan menyesuaikan bobot dan bias sedemikian rupa sehingga mengurangi jumlah tingkat kerugian (Tang et al., 2018).

Melalui tabel 3 di atas, total *epoch* 10 memiliki tingkat akurasi *train* dan *test* yang paling rendah. Ini juga memiliki kesenjangan yang cukup besar antara yang lain. Sedangkan total *epoch* 20, 30, 40 dan 50 memiliki tingkat akurasi yang tidak signifikan yaitu dengan *train accuracy* \pm 98% dan *test accuracy* \pm 81%. Namun melalui data diatas, tingkat akurasi tertinggi berada pada *epoch* 50 dengan *train accuracy* 98,96% dan *test accuracy* 83,53%.

Data pada tabel 3 di atas merupakan akurasi dan angka *loss* yang muncul saat pertama kali *model* neural *network* dijalankan. Angka akurasi dan *loss* tiap *model* saat dijalankan akan selalu berbeda karena data *train* dan *test* yang digunakan selalu dikocok secara acak. Misalnya, pada data latih pertama, gambarnya mungkin nomor 1, 3, 7, 10, 15, 2, 7, dst. Dan tesnya mungkin nomor 20,23,19,13, dst. upaya kedua, gambar data latih mungkin nomor 8, 3, 2, 1, 9, 14, dst. dan tesnya mungkin nomor 21,13,29,23, dst. Ini membuat data latih acak dan tidak beraturan. Akurasi dan angka *loss* bisa lebih tinggi dan bahkan lebih rendah (Khamparia et al., 2020).

Perbandingan *epoch* pada *Accuracy* dan *Loss Result*

Hasil perbandingan evaluasi tingkat akurasi dan *loss CNN VGG16* pada penelitian ini pada masing-masing *epoch* akan dijelaskan dibawah ini.

Tabel 4. *Epoch 10 Accuracy dan Loss Result*

<i>Test No</i>	<i>Train Accuracy</i>	<i>Test Accuracy</i>	<i>Train Loss</i>	<i>Test Loss</i>
1	93.15%	72.44%	27.38%	53.71%
2	92.72%	74.54%	29.88%	56.84%
3	78.70%	69.68%	47.02%	67.82%
4	90.38%	76.97%	28.38%	53.73%
5	91.68%	75.15%	32.08%	53.18%

Untuk total 10 *epoch*, rata-rata *train accuracy* 89,33%, rata-rata *test accuracy* 73,76%, rata-rata *train loss* 32,95%, dan rata-rata *test loss* 57,05%. Melalui angka-

angka yang diperoleh di atas, percobaan ke-3 memiliki tingkat ketelitian paling rendah dan tingkat *loss* paling tinggi jika dibandingkan dengan percobaan lainnya. Bahkan perbedaannya juga cukup jauh. Artinya pada percobaan ke 3, data citra latih dan uji yang diuji tidak acak sehingga parameter neuron yang ditangkap tidak dapat memberikan hasil yang maksimal.

Tabel 5. *Epoch 20 Accuracy dan Loss Result*

<i>Test No</i>	<i>Train Accuracy</i>	<i>Test Accuracy</i>	<i>Train Loss</i>	<i>Test Loss</i>
1	90.99%	80.41%	21.53%	52.36%
2	96.36%	77.57%	14.35%	47.53%
3	91.84%	81.21%	22.96%	50.13%
4	97.66%	82.43%	13.42%	45.86%
5	90.85%	79.54%	21.66%	49.97%

Untuk total 20 *epoch*, rata-rata *train accuracy* 93,54%, rata-rata *test accuracy* 80,23%, rata-rata *train loss* 18,78%, dan rata-rata *test loss* 49,17%. Pada setiap percobaan juga tidak ada perbedaan akurasi dan *loss* yang cukup jauh. Namun, ini merupakan perkembangan yang baik antara hasil *epoch* 10 dan *epoch* 20. Pada *epoch* 20, akurasinya lebih tinggi dan tingkat kerugiannya lebih rendah.

Tabel 6. *Epoch 30 Accuracy dan Loss Result*

<i>Test No</i>	<i>Train Accuracy</i>	<i>Test Accuracy</i>	<i>Train Loss</i>	<i>Test Loss</i>
1	97.48%	81.45%	12.85%	45.29%
2	98.44%	82.80%	9.15%	40.61%
3	93.24%	80.58%	11.64%	48.46%
4	96.62%	82.83%	11.49%	45.62%
5	97.77%	83.04%	9.86%	40.13%

Untuk total 30 *epoch*, rata-rata *train accuracy* 96,71%, rata-rata *test accuracy* 82,10%, rata-rata *train loss* 11,00%, dan rata-rata *test loss* 44,02%. Data diatas juga menunjukkan bahwa tidak ada perbedaan yang signifikan antara *training* dan *loss* rate. Namun, terlihat juga adanya peningkatan akurasi dan penurunan tingkat *loss* antara *epoch* 20 dan 30.

Tabel 7. *Epoch 40 Accuracy dan Loss Result*

<i>Test No</i>	<i>Train Accuracy</i>	<i>Test Accuracy</i>	<i>Train Loss</i>	<i>Test Loss</i>
1	98.46%	83.03%	7.16%	42.82%
2	97.92%	84.84%	7.82%	39.83%
3	96.22%	82.52%	11.80%	42.45%
4	96.88%	82.81%	9.23%	45.11%
5	98.18%	82.23%	5.45%	49.04%

Untuk total 30 *epoch*, rata-rata *train accuracy* 97,53%, rata-rata *test accuracy* 83,09%, rata-rata *train loss* 8,29%, dan rata-rata *test loss* 43,85%. Pada *epoch* 40 juga terlihat adanya peningkatan akurasi dan penurunan *loss* yang cukup baik jika dibandingkan dengan *epoch* 30. Perbandingan pada *test* no 2 pada *epoch* 30 dan *test* no 3 pada *epoch* 40 juga menunjukkan tidak selalu *epoch* yang lebih tinggi akan meningkatkan tingkat akurasi dan mengurangi tingkat kerugian. Dalam hal ini akurasi pada *epoch* 30 lebih tinggi dan *loss* pada *epoch* 30 juga lebih rendah.

Tabel 8. *Epoch 50 Accuracy dan Loss Result*

<i>Test No</i>	<i>Train Accuracy</i>	<i>Test Accuracy</i>	<i>Train Loss</i>	<i>Test Loss</i>
1	90.99%	80.41%	21.53%	52.36%
2	96.36%	77.57%	14.35%	47.53%
3	91.84%	81.21%	22.96%	50.13%
4	97.66%	82.43%	13.42%	45.86%
5	90.85%	79.54%	21.66%	49.97%

Pada total 50 *epoch*, rata-rata *train accuracy* 98,50%, rata-rata *test accuracy* 83,64%, rata-rata *train loss* 5,37%, dan rata-rata *test loss* 39,36%. Perbandingan pada *test* no 1 di *epoch* 40 dan *test* no 4 di *epoch* 50 juga menunjukkan bahwa tidak selamanya *epoch* yang lebih tinggi akan memiliki tingkat akurasi yang lebih tinggi dan menurunkan tingkat *loss*. Dalam hal ini akurasi pada *epoch* 40 lebih tinggi dan *loss* pada *epoch* 40 juga lebih rendah.

Tingkat akurasinya paling tinggi dan tingkat kerugiannya paling rendah diantara yang lainnya. Cukup meyakinkan untuk mengatakan bahwa *model CNN VGG16* cukup baik karena semakin banyak *model* saraf melihat keseluruhan *dataset*, semakin banyak data yang dipelajari *model* saraf dan lebih sedikit kehilangan data untuk meminimalkan

pelatihan parameter dan label uji pada gambar pisang *Cavendish* klasifikasi (Chiang et al., 2019)

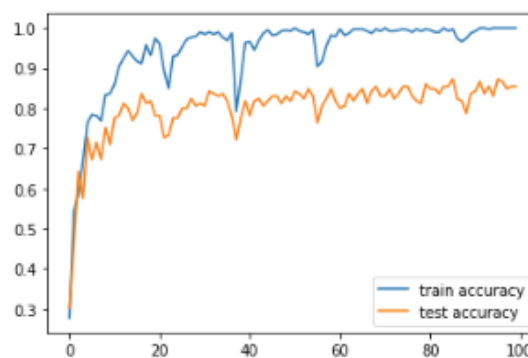
Uji Coba Epoch 100 Pada CNN VGG16 dan Analisis Grafik

Untuk membuktikan pernyataan bahwa *CNN VGG16* yang dibangun pada penelitian ini adalah *model* yang baik, peneliti mencoba melakukan pengujian dengan menjalankan *model VGG 16* dengan total 100 *epochs*. Hasilnya tercantum dalam tabel di bawah ini.

Tabel 9. *Epoch 100 Accuracy dan Loss Result*

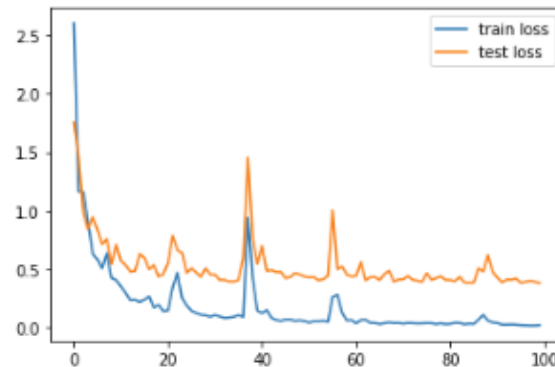
<i>Train Accuracy</i>	<i>Test Accuracy</i>	<i>Train Loss</i>	<i>Test Loss</i>
99.97%	86.73%	2.09%	34.81%

Melalui data ini, data untuk total *epoch 100* memiliki tingkat akurasi yang lebih tinggi pada *train* dan *test*, serta tingkat *loss* yang lebih rendah pada *train* dan *test* jika dibandingkan dengan *model CNN VGG 16* yang dijalankan pada *epoch 50*. Dengan ini, semakin meyakinkan kebenarannya bahwa *model CNN VGG 16* yang dibangun pada penelitian ini adalah *model* yang bagus. Hasil pergerakan akurasi *train* dan *test*, beserta *train* dan *test loss* dari *epoch 1-100* juga ditampilkan pada grafik di bawah ini untuk dianalisis



Grafik 1: *Epoch 100 Train dan Test Accuracy*

Grafik akurasi di atas menunjukkan pergerakan dari *epoch 0-100* mengalami peningkatan akurasi yang konstan, dimulai dari *epoch 1* dengan akurasi kurang dari 30% kemudian terus meningkat seiring dengan penambahan jumlah *epoch* yang terjadi pada *model training*.



Grafik 2: Epoch 100 Train dan Test Loss

Grafik *loss* di atas menunjukkan bahwa pergerakan dari *epoch* 0-100 mengalami penurunan tingkat *loss* secara konstan, dimulai dari *epoch* 1 dengan *loss* lebih dari 2 kemudian terus meningkat seiring dengan penambahan jumlah *epoch* yang terjadi pada pelatihan *model*.

Melalui 2 angka di atas, *epoch* ± 20 , ± 40 , ± 55 dan ± 90 mengalami penurunan tingkat akurasi, serta peningkatan tingkat *loss* pada *train* dan *test*. Ini terjadi karena *overfitting*. *Overfitting* sering terjadi saat menjalankan *model* neural network dan saat *model* neural network dijalankan, *model* tidak mempelajari trend dari kumpulan data tetapi menghafal setiap kumpulan data yang ada. Dengan kata lain, *model* tidak dapat melakukan klasifikasi generalisasi data karena model menjadi sangat cocok dengan kumpulan data pelatihan (Zhang et al., 2021). Saat terjadi *overfitting*, *loss rate* akan tiba-tiba meningkat dan tingkat akurasi akan menurun karena *dataset* yang dipelajari sebelumnya tidak cukup akurat untuk menyediakan data baru (d'Ascoli et al., 2020).

Hasil Diskusi

Melalui hasil *performance model* yang diperoleh, *model CNN VGG16* yang diuji pada penelitian ini merupakan *model* algoritma well pretrained. Hal ini dibuktikan dengan peningkatan tingkat akurasi dan penurunan tingkat *loss* untuk setiap jumlah *epoch* yang dilakukan. Ketika lebih banyak zaman dilakukan dalam pengujian *model*, itu berarti lebih banyak kumpulan data yang ditangkap dan dilatih sepenuhnya oleh *model* untuk dipelajari. Ketika *model* mempelajari lebih banyak kumpulan data, lebih banyak tren kumpulan data harus dipelajari oleh *model* jaringan saraf ini, menyebabkan peningkatan akurasi dan penurunan kerugian. Namun, semakin banyak *dataset* yang

digunakan untuk *model* pra-pelatihan uji akan menyebabkan keakuratannya semakin dapat diandalkan (Makantasis et al., 2015).

Melalui pengujian *model* menggunakan *epoch* 10, 20, 30, 40 dan 50, *model CNN VGG16* terbaik pada penelitian ini adalah pada *epoch* terbanyak yaitu 50. Namun perlu diingat bahwa *epoch* yang lebih tinggi tidak selalu memiliki tingkat akurasi yang lebih tinggi. terlalu tinggi dan tingkat kerugian yang lebih rendah. Contoh kejadian yang terjadi dapat dilihat pada hasil antara *epoch* 30 (*test* no 2) dan *epoch* 40 (*test* no 3), serta *epoch* 40 (*test* no 1) dan *epoch* 50 (*test* no 4). Kasus tersebut sangat mungkin terjadi dengan adanya *overfitting* pada *model* yang telah dilatih sebelumnya dan perlu dimodifikasi dengan menambahkan normalisasi (Srivastava et al., 2014). Meskipun insiden seperti itu terkadang terjadi, hasil penelitian membuktikan bahwa dengan *epoch* yang lebih tinggi, tingkat akurasi rata-rata lebih tinggi dan tingkat kerugian rata-rata lebih rendah.

4. SIMPULAN

Melalui hasil penelitian tersebut, peneliti menyarankan untuk menggunakan *epochs* 50 saat menguji *model* neural network. Peneliti tidak merekomendasikan *epochs* 100 karena runtime yang dibutuhkan sangat lama dan dapat menyebabkan *hardware* yang digunakan *overheat*. Namun, jika memiliki banyak waktu dan perangkat keras yang bagus, *epochs* 100 juga disarankan. Namun, perlu diingat bahwa untuk mendapatkan nilai *accuracy* tertinggi dan nilai *loss* terendah, *model* perlu dijalankan berulang kali. Iterasi ini merupakan proses penting dalam pengujian *model* jaringan syaraf tiruan untuk mendapatkan hasil evaluasi *model* yang terbaik.

Saran peneliti untuk penelitian selanjutnya adalah menggunakan *dataset* yang lebih banyak lagi, pada penelitian ini peneliti hanya menggunakan total 550 foto (110 foto pada setiap variabel). Ke depan sebaiknya disiapkan 1500 foto (300 foto tiap variabel) agar hasil yang didapat juga lebih reliabel dan relevan karena banyak *dataset* yang diproses selama *model* neural network berjalan. Memodifikasi *model CNN VGG 16* untuk mencegah *overfitting* juga diperlukan. Modifikasi tersebut dapat dilakukan dengan menambahkan aktivasi ReLU, normalisasi batch, regularisasi dropout, dan menambahkan batasan pada learning rate. Modifikasi ini sangat dianjurkan karena langkah pada *model* neural network ini memang berfungsi untuk mencegah *overfitting*

namun terkadang juga membuat tingkat *loss* sangat tinggi dan membuat akurasi mandek. Oleh karena itu, modifikasi *model neural network* harus diperhatikan dan diuji dengan baik agar hasil evaluasi lebih baik dari *model CNN VGG16* pada penelitian ini.

5. DAFTAR PUSTAKA

- Agarwal, M., Gupta, S. K., dan Biswas, K. K. (2020). Development of Efficient CNN model for Tomato crop disease identification. *Sustainable Computing: Informatics and Systems*, 28. <https://doi.org/10.1016/j.suscom.2020.100407>
- Chiang, W. L., Li, Y., Liu, X., Bengio, S., Si, S., dan Hsieh, C. J. (2019). Cluster-GCN: An efficient algorithm for training deep and large graph convolutional networks. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. <https://doi.org/10.1145/3292500.3330925>
- d'Ascoli, S., Sagun, L., dan Biroli, G. (2020). Triple descent and the two kinds of overfitting: Where dan why do they appear? *Advances in Neural Information Processing Systems, 2020-December*. <https://doi.org/10.1088/1742-5468/ac3909>
- H. A. Moh, R. Yan, dan R. Dwiza. (2021). Klasifikasi penyakit citra daun anggur menggunakan model CNN-VGG16. *Jurnal Teknologi Dan Sistem Komputer*, 9(4).
- Hasan, M. M., Chopin, J. P., Laga, H., dan Miklavcic, S. J. (2018). Detection and analysis of wheat spikes using Convolutional Neural Networks. *Plant Methods*, 14(1). <https://doi.org/10.1186/s13007-018-0366-8>
- Khamparia, A., Gupta, D., de Albuquerque, V. H. C., Sangaiah, A. K., dan Jhaveri, R. H. (2020). Internet of health things-driven deep learning system for detection and classification of cervical cells using transfer learning. *Journal of Supercomputing*, 76(11). <https://doi.org/10.1007/s11227-020-03159-4>
- Li, J., Ablan, C., Wu, R., Guan, S., dan Yao, J. (2021). Preprocessing method comparisons for VGG16 fast-RCNN pistol detection. *EPiC Series in Computing*, 76. <https://doi.org/10.29007/ml35>
- Liu, Z., Wu, J., Fu, L., Majeed, Y., Feng, Y., Li, R., dan Cui, Y. (2020). Improved Kiwifruit Detection Using Pre-Trained VGG16 with RGB and NIR Information Fusion. *IEEE Access*, 8. <https://doi.org/10.1109/ACCESS.2019.2962513>
- Makantasis, K., Karantzas, K., Doulamis, A., dan Doulamis, N. (2015). Deep supervised learning for hyperspectral data classification through convolutional neural networks. *International Geoscience and Remote Sensing Symposium (IGARSS), 2015-November*. <https://doi.org/10.1109/IGARSS.2015.7326945>
- Maulana, F. F., dan Rochmawati, N. (2020). Klasifikasi Citra Buah Menggunakan Convolutional Neural Network. *Journal of Informatics and Computer Science (JINACS)*, 1(02). <https://doi.org/10.26740/jinacs.v1n02.p104-108>
- Nuramanah, E., Hayat Sholihin, dan Siswaningsih, W. (2013). Kajian Aktivitas Antioksidan Kulit Pisang Raja Bulu (*musa paradisiaca* L. Var *sapientum*) Dan Produk Olahannya Eva. *Jurnal Sains Dan Teknologi Kimia*, 4(1).

- Pardede, J., Sitohang, B., Akbar, S., dan Khodra, M. L. (2021). Implementation of Transfer Learning Using VGG16 on Fruit Ripeness Detection. *International Journal of Intelligent Systems and Applications*, 13(2), 52–61. <https://doi.org/10.5815/ijisa.2021.02.04>
- Song, Z., Fu, L., Wu, J., Liu, Z., Li, R., dan Cui, Y. (2019). Kiwifruit detection in field images using Faster R-CNN with VGG16. *IFAC-PapersOnLine*, 52(30). <https://doi.org/10.1016/j.ifacol.2019.12.500>
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., dan Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15.
- Tang, M., Djelouah, A., Perazzi, F., Boykov, Y., dan Schroers, C. (2018). Normalized Cut Loss for Weakly-Supervised CNN Segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. <https://doi.org/10.1109/CVPR.2018.00195>
- Ucat, R. C., dan Cruz, J. C. D. (2019). Postharvest Grading Classification of Cavendish Banana Using Deep Learning and Tensorflow. *2019 International Symposium on Multimedia and Communication Technology, ISMAC 2019*. <https://doi.org/10.1109/ISMAC.2019.8836129>
- Utomo, B., Marsiti, C. I. R., dan Damiami, D. (2019). Uji Kualitas Tepung Pisang MAS (Musa Acuminata). *Jurnal BOSAPARIS: Pendidikan Kesejahteraan Keluarga*, 9(3). <https://doi.org/10.23887/jjpkk.v9i3.22146>
- Zhang, Y., Luo, T., Ma, Z., dan Xu, Z. Q. J. (2021). A Linear Frequency Principle Model to Understand the Absence of Overfitting in Neural Networks. *Chinese Physics Letters*, 38(3). <https://doi.org/10.1088/0256-307X/38/3/038701>