

Klasifikasi Non-Destruktif Kemanisan Semangka Manohara Menggunakan *Transfer Learning VGG-16*

Dicky Ryanto Fernandes^{1*}, Nur Rachmat²

^{1,2}Fakultas Ilmu Komputer dan Rekayasa, Universitas Multi Data Palembang

Jl. Rajawali No.14, 9 Ilir, Kec. Ilir Tim. II, Kota Palembang, Sumatera Selatan 30113, (0711) 376400

e-mail: dicky.ryanto2303@gmail.com, nur.rachmat@mdp.ac.id

ABSTRAK

Semangka (*Citrullus lanatus*) merupakan buah tropis populer di Indonesia karena rasanya yang manis dan kandungan airnya yang tinggi. Penentuan tingkat kemanisan masih banyak dilakukan secara destruktif dengan refraktometer, sehingga kurang efisien. Penelitian ini bertujuan mengklasifikasikan tingkat kemanisan semangka Manohara secara non-destruktif berdasarkan ciri fisik luar menggunakan *Convolutional Neural Network* (CNN) dengan arsitektur VGG-16 dan pendekatan *transfer learning*. Data dikumpulkan secara mandiri dan dibagi menjadi 80% data latih, 10% validasi, dan 10% uji. Model menggunakan *Adam Optimizer* dan *Softmax* sebagai *classifier*. Hasil terbaik diperoleh pada skenario ke-4 dengan akurasi 67,42%. Namun, model menunjukkan gejala *underfitting* dan kecenderungan mengklasifikasi ke satu kelas. Penelitian ini menunjukkan potensi awal penggunaan visi komputer dalam seleksi kualitas semangka secara otomatis dan non-destruktif, meskipun masih diperlukan peningkatan akurasi agar dapat diimplementasikan secara praktis di lapangan.

Kata Kunci: Adam Optimizer, CNN, Klasifikasi Kemanisan Semangka, VGG-16

ABSTRACT

Watermelon (*Citrullus lanatus*) is a popular tropical fruit in Indonesia due to its sweet taste and high water content. Determining sweetness level is still commonly done destructively using a refractometer, making the process less efficient. This study aims to classify the sweetness level of Manohara watermelon non-destructively based on external physical characteristics using a *Convolutional Neural Network* (CNN) with the VGG-16 architecture and a *transfer learning* approach. The dataset was self-collected and split into 80% training data, 10% validation, and 10% testing. The model employed *Adam Optimizer* and *Softmax* as the classifier. The best result was obtained in the fourth scenario with an accuracy of 67.42%. However, the model showed *underfitting* and a tendency to classify into only one class. This research demonstrates the initial potential of computer vision in automatic and non-destructive watermelon quality assessment, although improvements in accuracy are still needed for practical field implementation.

Keywords: Adam Optimizer, CNN, VGG-16, Watermelon Sweetness Classification

Pendahuluan

Semangka atau dalam bahasa latin *Citrullus lanatus* merupakan tumbuhan dalam famili *cucurbitaceae L*, dengan ciri khas kulitnya yang bergaris-garis hijau tua atau muda, mempunyai daya tarik dari dagingnya yang berwarna merah atau kuning, memiliki kandungan air yang tinggi, dan kenyal [1]. Semangka merupakan salah satu jenis buah yang populer di kalangan masyarakat Indonesia [2].

Berdasarkan data dari Badan Pusat Statistik (BPS), produksi semangka pada tahun 2018 total sebanyak 481.744 ton, tahun 2019 sebanyak 523.333 ton, dan tahun 2020 sebanyak 560.317 ton [3]. Dari data tersebut dapat dilihat bahwa produksi semangka di Indonesia meningkat setiap tahunnya, sehingga penting untuk melakukan pemilahan semangka sebelum dijual ke masyarakat umum.

Buah semangka yang dipanen perlu dilakukan pemilahan untuk mengklasifikasikan kualitas semangka yang ada di pasaran. Kualitas dan harga buah semangka di pasaran ditentukan oleh jenis dan variasi buah, berat buah, kondisi buah, dan manisnya buah semangka. Mengukur kemanisan buah penting untuk menilai kualitas buah, dan penjual dapat meningkatkan harga jual berdasarkan manisnya buah tersebut [1].

Berdasarkan data dari Kementerian Pertanian, konsumsi semangka antara tahun 2020 hingga 2022 menunjukkan tren peningkatan yang konsisten. Pada tahun 2020 total konsumsi semangka tercatat sebesar 1,896 kg, tahun 2021 sebesar 1,914 kg dan tahun 2022 sebesar 3,171 kg [4].

Berdasarkan preferensi konsumen, semangka yang digemari konsumen Indonesia adalah semangka yang rasanya manis [5]. Saat membeli semangka utuh tentu menginginkan semangka yang berwarna merah dan manis, sementara membeli buah-buahan manis dan segar bukanlah tugas yang mudah karena tidak mengetahui apakah

semangka sudah matang dan siap dipanen, untuk itu pembeli perlu memahami cara memilih buah ini dengan benar agar diperoleh buah dengan rasa yang manis dan segar [6].

Penelitian tentang klasifikasi buah yang membandingkan algoritma CNN dan SVM dengan total *dataset* 11.219 dan 17 label nama buah, dengan pembagian 70% data latih dan 30% data uji, serta ukuran gambar 100 x 100 piksel dan *epoch* sebanyak 25, mendapatkan akurasi sebesar 96,87% untuk CNN dan sebesar 93,09% untuk SVM [7]. Penelitian tentang klasifikasi buah pir yang membandingkan algoritma CNN dan NN dengan total *dataset* 3500, dengan pembagian 75% data latih dan 25% data uji, kemudian *dataset* di *resize* menjadi ukuran 100 x 100 piksel dan *epoch* sebanyak 10, mendapatkan akurasi, presisi, *recall* dan *F1-Score* sebesar 97% untuk CNN, dan pada NN akurasi sebesar 86%, presisi 89%, *recall* 86% dan *F1-Score* 86% [8]. Dari beberapa penelitian yang telah diuraikan, terbukti bahwa CNN memiliki akurasi yang lebih tinggi ketimbang metode lain seperti SVM dan NN dalam mengklasifikasi objek khususnya buah-buahan.

Kelemahan CNN adalah memerlukan data pelatihan yang banyak, proses pelatihan membutuhkan waktu yang lama, dan sering terjadi *overfitting* [9]. *Transfer learning* merupakan teknik atau metode yang menggunakan model yang telah dilatih sebelumnya (sudah dilatih pada kumpulan data) untuk melatih model lain dengan menggunakannya sebagai titik awal dan mengubah serta memperbarui parameter agar sesuai dengan kumpulan data kasus masalah baru [10]. Teknik *transfer learning* merupakan solusi yang dapat digunakan ketika *dataset* berukuran kecil dan tidak cukup ideal untuk melakukan proses pelatihan dari awal [10].

Telah dilakukan beberapa penelitian terdahulu yang menggunakan metode *transfer learning CNN* (*Convolutional Neural Network*) dengan arsitektur VGG-16 (*Visual Geometry Group 16-layer network*) dalam mengklasifikasi buah. Misalnya pada penelitian yang mengklasifikasikan kualitas buah salak mendapatkan akurasi sebesar 95,38% [11]. Pada penelitian lain yang mengklasifikasikan kualitas buah pisang mendapatkan akurasi sebesar 83,53% [12]. Pada penelitian sebelumnya [6], untuk mendeteksi tingkat kemanisan buah semangka menggunakan CNN dengan arsitektur *EfficientNetV2S* masih dilakukan secara manual menggunakan alat refraktometer dan bersifat destruktif yang perlu memotong buah sehingga kurang efisien dan praktis.

Penelitian ini akan mengklasifikasi semangka manohara dengan cara non destruktif (tanpa membelah/memotong buah) menggunakan metode CNN dengan arsitektur VGG-16 serta *dataset* yang diambil secara mandiri dan telah divalidasi oleh pakar. Pemilihan arsitektur VGG-16 dalam penelitian ini didasarkan pada struktur yang sederhana, stabil, dan terbukti efektif untuk klasifikasi gambar, khususnya dengan jumlah data terbatas. Dibandingkan *EfficientNet*, *ResNet*, atau *MobileNet* yang lebih ringan atau canggih, VGG-16 lebih mudah diimplementasikan dan sangat kompatibel dengan teknik *transfer learning*.

Berdasarkan penelitian yang telah diuraikan sebelumnya, menunjukkan bahwa efektivitas metode *transfer learning CNN* dengan arsitektur VGG-16 dalam mengklasifikasikan objek berupa gambar, khususnya dalam pemilihan buah. Oleh karena itu, studi ini mengusulkan metode CNN dengan arsitektur VGG-16 dalam mengklasifikasi tingkat kemanisan buah semangka manohara non destruktif berdasarkan ciri fisik luar buah.

Metode Penelitian

Tahapan Penelitian

Penelitian ini diawali dengan kajian pustaka yang relevan, setelah itu dilanjutkan dengan mengumpulkan *dataset* semangka manohara, kemudian perancangan sistem, implementasi dan yang terakhir pengujian sistem. Gambar 1 merupakan tahapan penelitian yang dilakukan.



Gambar 1. Tahapan Penelitian

Pengumpulan Dataset

Pada tahap ini, dilakukan proses pengumpulan *dataset* dengan cara mengangkat satu persatu buah semangka kemudian mendokumentasikan bagian *groundspot* semangka manohara. Pengambilan gambar dilakukan menggunakan kamera *smartphone* dengan spesifikasi 50 MP dengan pengambilan foto diluar dan didalam ruangan pada jarak 30 cm. Data yang didapat sebanyak 108 citra berformat .jpg yang dibagi menjadi 2 kelas yaitu semangka premium yang manis ditandai dengan *groundspot* besar berwarna *orange* kekuningan dan semangka non premium yang tidak manis ditandai dengan *groundspot* kecil berwarna kuning samar dan sedikit putih pucat. *Dataset* telah mendapatkan validasi dari Dinas Pertanian Tanaman Pangan dan Hortikultura Provinsi Sumatera Selatan. Tabel 1 menunjukkan contoh gambaran *dataset* yang digunakan.

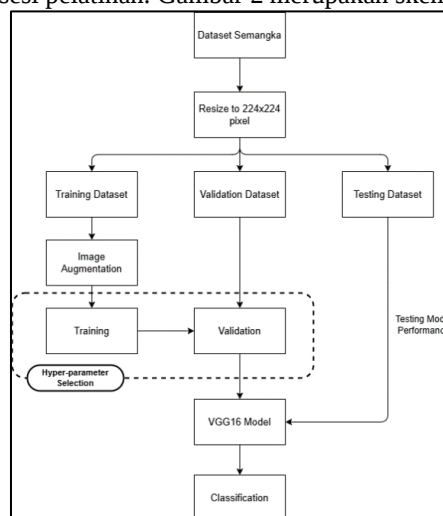
Tabel 1. Contoh Dataset

Kelas	Foto Groundspot	Pengujian °Brix
Premium (Manis)	 Groundspot orange kekuningan	 11 °Brix
Non-Premium (Tidak Manis)	 Groundspot putih pucat dan samar	 7 °Brix

Perancangan Sistem

Pada tahapan ini dilakukan perancangan sistem yang dibutuhkan. Diawali dengan *input dataset* semangka, lalu dilakukan *resize* ke ukuran 224x224 piksel untuk menyamakan ukuran yang dibutuhkan arsitektur VGG-16. Kemudian *dataset* dipecah dengan bobot 80% data *training*, 10% data *validation*, dan 10% data *testing*. Pada data *training* dilakukan *augmentasi* untuk memperbanyak variasi data pelatihan dengan menerapkan *rescale* (1.0/255), *rotation_range* (30), *width_shift_range* (0.2), *height_shift_range* (0.2), *zoom_range* (0.2), *horizontal_flip* (True), *brightness_range* [0.7, 1.3], dan *fill_mode* 'nearest'. Proses selanjutnya adalah *feature learning* yang terdiri dari lapisan lapisan konvolusi, lapisan aktivasi, dan lapisan *pooling*. Dari proses *feature learning* menghasilkan keluaran (*output*) berupa *neuron* yang juga biasa disebut sebagai *feature map*.

Lalu tahap selanjutnya tahap klasifikasi, di mana terdapat proses *flattening* yang bertujuan untuk mengonversi matriks menjadi vektor satu dimensi yang akan digunakan sebagai input dalam lapisan *fully-connected*. Setelah itu, fungsi aktivasi *softmax* diterapkan untuk menghitung kemungkinan dari setiap kelas yang tersedia. Kemudian model dioptimalkan dengan memakai *optimizer* dengan tujuan mengurangi tingkat kesalahan dan meningkatkan akurasi selama sesi pelatihan. Gambar 2 merupakan skema perancangan VGG-16.



Gambar 2. Skema Perancangan VGG-16

Implementasi

Pada tahapan ini dilakukan implementasi dari sistem yang telah dirancang sebelumnya menggunakan *Google Colaboratory* dengan *library TensorFlow* dan *Keras* agar sistem dapat mengenali dan melakukan klasifikasi

terhadap data *training* yang telah diolah sebelumnya.

Pengujian

Pada tahapan ini, sistem yang telah dibuat sebelumnya akan melakukan uji coba terhadap data uji dengan 4 skenario pengujian untuk mencari model terbaik dari kombinasi *epoch*, *batch size*, dan *learning rate*. Tabel 2 menunjukkan skenario kombinasi pengujian model.

Tabel 2. Skenario Kombinasi Pengujian Model

Skenario	Epoch	Batch Size	Learning Rate
1	30	8	0,001
2	30	8	0,0001
3	50	8	0,001
4	50	8	0,0001

Pengacuan Pustaka

Semangka

Semangka dengan nama ilmiah *Citrullus lanatus* adalah tanaman yang memiliki nilai ekonomi tinggi dan telah ditanam secara komersial di berbagai negara, termasuk Indonesia. Buah ini tidak hanya dikenal oleh masyarakat Indonesia karena kesegarannya, tetapi juga memiliki peranan ekonomi yang signifikan bagi sektor pertanian di Indonesia [13].

Tanaman semangka berasal dari wilayah Afrika dan merupakan salah satu jenis tanaman hortikultura yang sangat disukai oleh masyarakat Indonesia karena rasa manisnya, banyaknya kandungan air, serta teksturnya yang renyah. Buah semangka memiliki kandungan air dan vitamin yang tinggi, di mana air berfungsi untuk mencegah dehidrasi dan vitamin berperan dalam menjaga kesehatan tubuh. Dengan kandungan tersebut, semangka menawarkan beberapa manfaat, termasuk pencegahan dehidrasi, perlindungan terhadap fungsi saraf, pengurangan frekuensi serangan asma, pemeliharaan kesehatan kulit dan rambut, serta dukungan dalam proses penurunan berat badan. Berbagai keuntungan dapat diperoleh dari buah semangka, termasuk bagian buah, kulit, dan bijinya [14].

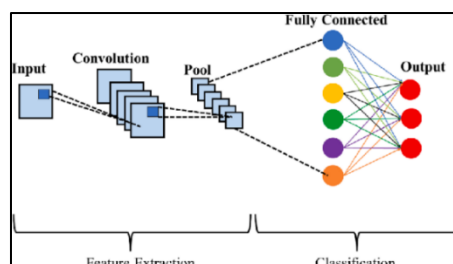
°Brix

Rasa manis yang ada pada buah serta sayuran dipengaruhi oleh jumlah zat padat yang terlarut. Zat padat terlarut ini diwakili oleh nilai derajat Brix (°Brix). Pentingnya nilai °Brix terletak pada fakta bahwa ia memberikan ukuran kualitas secara objektif, termasuk tingkat kemanisan buah. Pengukuran nilai °Brix dapat dilakukan dengan mudah dan terpercaya di lapangan menggunakan alat yang disebut refractometer [6]. Berdasarkan buku Indonesia *Customs and Excise Laboratory Bulletin* [15], rasa manis pada semangka yang dianalisis dalam penelitian ini terbagi menjadi 2 kelas, yaitu manis (≥ 8 °Brix) dan tidak manis (< 8 °Brix).

Convolutional Neural Network (CNN)

Convolutional neural network (CNN) merupakan jenis jaringan saraf yang dikembangkan khusus untuk menyelesaikan masalah dalam pengolahan gambar. *CNN* adalah teknik yang sangat efisien untuk mengkategorikan, mendeteksi, dan mengenali pola dalam gambar. Dengan arsitektur yang menyerupai cara otak manusia memproses visual, *CNN* memiliki kemampuan yang lebih baik dalam menangkap detail citra. Tipe data yang diterapkan pada *CNN* adalah data dua dimensi, seperti gambar atau audio, yang mengandalkan operasi konvolusi di dalam matriks serta bobot yang berbentuk empat dimensi, yaitu kumpulan kernel konvolusi. Karena karakteristik dari proses konvolusi ini, *CNN* hanya dapat diterapkan pada data yang memiliki struktur dua dimensi [16].

Secara umum *CNN* memiliki komponen yang mencakup masukan, pengambilan fitur, pengelompokan, dan hasil. Tahapan pengambilan fitur dalam *CNN* melibatkan beberapa lapisan, termasuk lapisan konvolusi, lapisan aktivasi (dengan *ReLU*), serta lapisan *pooling*. Pada tahap pengelompokan, lapisan yang digunakan meliputi *fully-connected layer* dan fungsi aktivasi (dengan *Softmax*) [17]. Gambar 3 menunjukkan lapisan-lapisan yang ada pada arsitektur *CNN*.



Gambar 3. Lapisan Pada Arsitektur CNN

Lapisan Masukan (Input)

Lapisan ini berfungsi untuk menyimpan nilai piksel dari gambar yang dimasukkan. Gambar yang dimasukkan memiliki 3 saluran warna RGB (Merah, Hijau, Biru) [18].

Lapisan Konvolusi

Lapisan Konvolusi terdiri dari sel-sel yang diatur sedemikian rupa untuk membentuk sebuah filter dengan dimensi panjang dan lebar (piksel). Proses konvolusi memanfaatkan kernel dan *stride*, yang merupakan kombinasi antara dua matriks yang berbeda untuk menghasilkan matriks baru [18].

Lapisan Aktivasi

Pada lapisan ini terjadi modifikasi nilai-nilai pada *feature map* dengan jarak tertentu yang tergantung pada fungsi aktivasi yang digunakan, yaitu fungsi aktivasi *ReLU*. Secara fundamental, fungsi *ReLU* (*Rectified Linear Unit*) melakukan “*threshold*” dari 0 sampai tak terhingga. Fungsi ini telah menjadi salah satu fungsi yang sangat terkenal saat ini [18]. Fungsi *ReLU* adalah fungsi *non-linier* di mana neuron tidak diaktifkan secara simultan, melainkan hanya saat hasil dari transformasi linier menunjukkan nilai nol. Fungsi *ReLU* dapat dilihat pada persamaan (1) [19].

$$f(x) = \max(0, x) \quad (1)$$

x = Data input yang digunakan,

$f(x)$ = Output dari fungsi *ReLU* yang berupa nilai 0 dan 1.

Lapisan Pooling

Lapisan *pooling* berfungsi di setiap lapisan peta fitur dan mengurangi dimensinya. Biasanya, bentuk lapisan *pooling* menggunakan filter berukuran 2x2 yang diterapkan dengan langkah dua dan beroperasi pada setiap potongan dari inputnya [18].

Lapisan Fully Connected

Lapisan yang disebut *Fully Connected*, atau yang lebih umum dikenal sebagai *dense layer* adalah jenis lapisan di mana setiap neuron dari lapisan sebelumnya terhubung dengan semua neuron di lapisan berikutnya, mirip dengan cara kerja jaringan syaraf buatan [18].

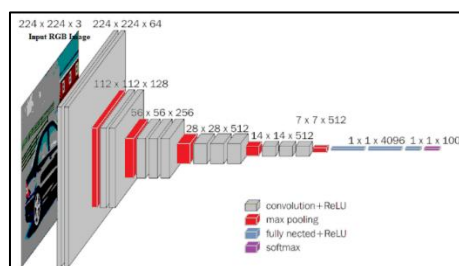
Lapisan Keluaran (Output)

Lapisan yang terakhir adalah melakukan klasifikasi, di mana dengan bantuan fungsi aktivasi *softmax*, input akan dikelompokkan sesuai dengan kategori tujuannya [18].

VGG-16

VGG-16 adalah arsitektur neurologis dalam yang terdiri dari 16 lapisan dan merupakan salah satu arsitektur pertama yang digunakan dalam *transfer learning*. VGG-16 dikembangkan oleh Karen Simonyan dan Andrew Zisserman dan diterbitkan pada tahun 2014. Model ini menang dalam kompetisi ILSVRC *challenge* 2014 dengan perolehan *top 5* tes akurasi hingga 92,7% pada klasifikasi 14 juta citra dalam 1000 kelas [20].

Arsitektur VGGNet adalah pengembangan dari *alexnet* yang berfokus pada proses pengambilan fitur di lapisan konvolusi sehingga dapat menghasilkan banyak representasi citra untuk kebutuhan klasifikasi. VGG-16 merupakan komponen dari *model deep learning* yang terdiri dari 16 lapisan. Dalam arsitektur VGG-16, terdapat 13 lapisan konvolusi, ditambah 2 lapisan yang berfungsi sebagai *fully_connected*, serta 1 lapisan untuk klasifikasi [20]. Gambar 4 merupakan arsitektur pada VGG-16.



Gambar 4. Arsitektur VGG-16

Data Preprocessing

Data *preprocessing* merupakan tahap-tahap yang dilakukan untuk mengatur data agar sesuai dengan format yang dibutuhkan, sehingga bisa digunakan sebagai data uji ataupun data pelatihan [22].

Resize

Resize adalah proses di mana gambar mengalami pemotongan atau modifikasi ukuran dari gambar aslinya dengan ukuran yang telah ditentukan, misalnya 224x224 piksel [23].

Augmentasi

Augmentasi merupakan langkah yang diambil untuk meningkatkan jumlah data dengan menghasilkan data baru berdasarkan data yang telah ada [24]. *Augmentasi* data melibatkan sejumlah metode seperti pemutaran, pengubahan ukuran, pergeseran, dan cerminan citra yang dirancang untuk memperluas variasi data tanpa mengubah label aslinya [25].

Python

Python merupakan sebuah bahasa pemrograman yang terkenal dan gampang untuk dipelajari. Banyak pengembang memilih *Python* dalam membuat perangkat lunak, kecerdasan buatan, pembuatan situs web, pembelajaran mesin, serta analisis data. Bahasa ini menawarkan beragam pustaka, seperti *NumPy* yang digunakan untuk perhitungan numerik dan *Pandas* yang difungsikan dalam analisis data, sehingga pengguna dapat menyelesaikan tugas tertentu dengan cepat dan efisien [26].

TensorFlow

TensorFlow merupakan pustaka sumber terbuka yang dikembangkan oleh tim Google Brain untuk keperluan komputasi numerik serta pembelajaran mesin dalam skala besar. *TensorFlow* menyatukan berbagai model, algoritma pembelajaran mesin, dan algoritma pembelajaran mendalam (jaringan saraf). *TensorFlow* memanfaatkan *Python* untuk menawarkan *API front-end* yang digunakan dalam pembuatan aplikasi dengan kerangka kerja, serta menjalankan aplikasi tersebut dengan kinerja yang tinggi. *TensorFlow* mampu melatih dan menerapkan jaringan saraf buatan untuk berbagai tujuan, termasuk pengenalan tulisan tangan, identifikasi gambar, penyematan kata, jaringan saraf berulang, model *sequence-to-sequence* untuk terjemahan, pemrosesan bahasa alami, dan simulasi yang berbasis pada *PDE* (Persamaan Diferensial Parsial) [27].

Keras

Keras adalah sebuah pustaka pembelajaran mesin yang bersifat *open source* dan berbasis *Python*. Pustaka ini dirancang untuk memudahkan implementasi model *deep learning* dengan cepat dan dengan proses yang sederhana mungkin, terutama untuk penelitian dan pengembangan yang dikeluarkan dengan lisensi MIT. Pustaka jaringan syaraf yang bersifat terbuka ini dibuat untuk mendukung eksperimen yang cepat pada jaringan syaraf dalam, dan dapat berfungsi di atas *CNTK*, *Tensorflow*, serta *Theano*. *Keras* mempunyai fokus pada modularitas, kemudahan penggunaan, dan kemampuan untuk diperluas. Kerangka kerja ini tidak menangani perhitungan pada tingkat rendah tetapi malah menyerahkan pekerjaan tersebut kepada pustaka lain yang dikenal sebagai *backend* [28].

Optimizer

Optimizer merupakan elemen krusial dalam proses pelatihan model *CNN* yang dapat memengaruhi laju serta ketepatan konvergensi dari model tersebut [29]. *Optimizer* merupakan sebuah algoritma yang berfungsi untuk mengatur pembaruan bobot dan bias model dengan sasaran untuk mengecilkan fungsi kerugian selama tahap pelatihan [30].

Adam Optimizer

Adam Optimizer merupakan teknik optimasi yang sangat terkenal dalam bidang pembelajaran mesin, khususnya dalam melatih jaringan saraf dalam berbagai aplikasi pembelajaran mendalam. *Adam*, yang merupakan akronim dari "*Adaptive Moment Estimation*," dengan cerdas mengintegrasikan keunggulan dari dua metode optimasi sebelumnya, yaitu *RMSProp* dan *Momentum*, untuk mencapai proses konvergensi yang cepat dan efektif. *Adam Optimizer* menghitung estimasi pertama (rata-rata) dan kedua (varians tidak terpusat) dari gradien, dan memanfaatkan kedua estimasi tersebut untuk menyesuaikan tingkat pembelajaran untuk setiap parameter [31]. Rumus untuk menentukan *optimizer Adam* dijabarkan dalam persamaan (2) hingga persamaan (6) [30]:

Perkiraan momen awal (*mean*) :

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (2)$$

Perkiraan momen kedua (varians yang tidak terpusat) :

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t \quad (3)$$

Perkiraan momen pertama yang sudah diperbaiki dari bias :

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \quad (4)$$

Perkiraan momen kedua yang telah diperbaiki dari bias:

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \quad (5)$$

Pembaruan pada parameter:

$$\hat{\theta}_t = \hat{\theta}_{t-1} - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} \quad (6)$$

Dengan

g_t = Gradien pada saat waktu (t),

β_1 dan β_2 = koefisien untuk penurunan eksponensial dalam perhitungan momen,

θ = parameter yang akan dioptimalkan,

α = berfungsi sebagai laju pembelajaran,

ϵ = sebuah konstanta yang sangat kecil untuk menghindari pembagian dengan nol.

Confusion Matrix

Akurasi dapat dipahami sebagai seberapa dekat nilai prediksi dengan nilai sebenarnya. Akurasi menunjukkan seberapa akurat atau benar klasifikasi tersebut. Pada saat yang sama, fungsi *recall* dimanfaatkan untuk menghitung

proporsi kasus positif sebenarnya yang dikenali dengan benar. Matriks kebingungan biasanya digunakan untuk mengukur akurasi, presisi, dan perolehan. Matriks *konfusi* merupakan alat pengukuran berbentuk matriks yang memungkinkan untuk menentukan keakuratan klasifikasi suatu kelas berdasarkan algoritma yang digunakan [32]. Pada Tabel 3 menunjukkan bentuk matriks *konfusi*.

Tabel 3. Bentuk Matriks Konfusi

<i>Confusion Matrix</i>		Nilai Sebenarnya	
		True	False
Nilai Prediksi	True	TP (True Positif) Hasil yang benar	FP (False Positif) Hasil yang tidak terduga
	False	FN (False Negatif) Hasil yang hilang	TN (True Negatif) Tidak adanya hasil yang benar

Tabel 3 menjelaskan bahwa nilai TP (true positif) dan TN (true negatif) mewakili tingkat akurasi klasifikasi. Umumnya semakin besar nilai TP dan TN, maka semakin tinggi akurasi, presisi, dan recall pengklasifikasiannya. Jika label prediksi keluaran benar dan nilai sebenarnya salah maka disebut positif palsu (FP). Sebaliknya jika label keluaran yang diprediksi bernilai salah dan nilai sebenarnya bernilai benar maka disebut false negative (FN) [32]. Berikut rumus perhitungan *accuracy*, *precision*, *recall* dan *f1-score* dalam membangun model klasifikasi ditunjukkan pada persamaan (10), (11), dan (12).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (7)$$

$$Precision = \frac{TP}{TP + FP} \quad (8)$$

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

$$F1\ score = 2 \times \frac{presisi \times recall}{presisi + recall} \quad (10)$$

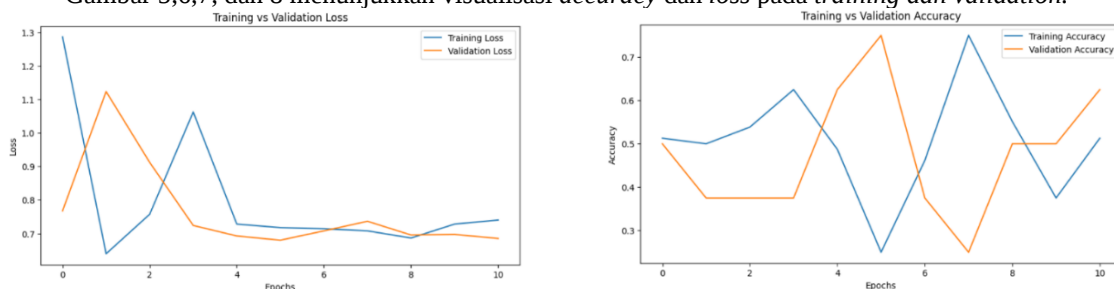
Hasil Dan Pembahasan

Penelitian ini membekukan 5 *layer* awal pada VGG16, menambahkan 2 *layer dense* (512 dan 256) dengan fungsi aktivasi *ReLU*, menggunakan *Early Stopping* (*patience* = 5) dan *Adam Optimizer*. Di bawah ini adalah hasil dari setiap skenario dengan memanfaatkan berbagai kombinasi pengujian. Tabel 4 merupakan hasil pengujian skenario.

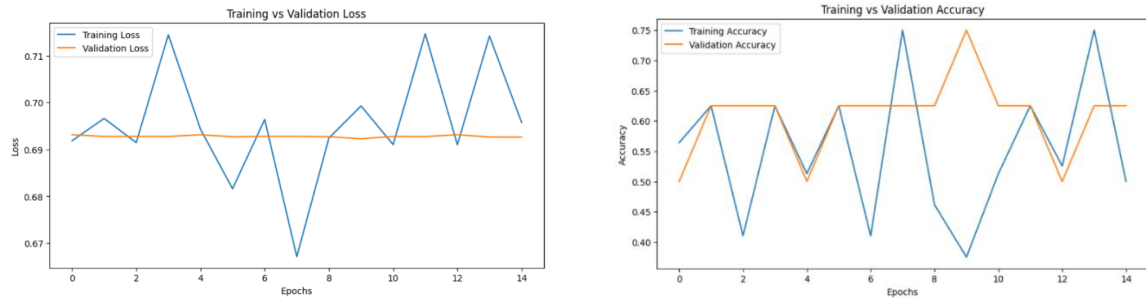
Tabel 4. Hasil Pengujian Skenario

Skenario	Accuracy	Training Accuracy	Validation Accuracy	Loss	Precision	Recall	F1-Score
1	0,6654	0,51	0,62	0,6379	0,6654	1,00	0,7991
2	0,6685	0,50	0,62	0,6924	0,6685	1,00	0,8013
3	0,6563	0,46	0,62	0,6558	0,6563	1,00	0,7925
4	0,6742	0,62	0,62	0,6906	0,6742	1,00	0,817

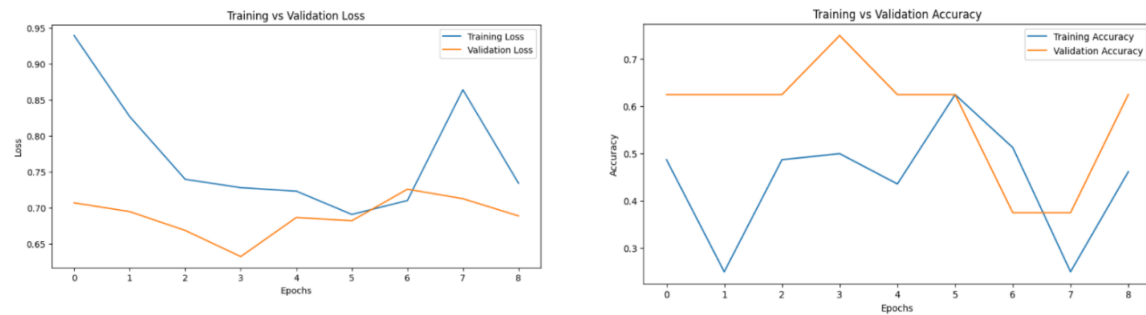
Gambar 5,6,7, dan 8 menunjukkan visualisasi *accuracy* dan *loss* pada *training* dan *validation*.



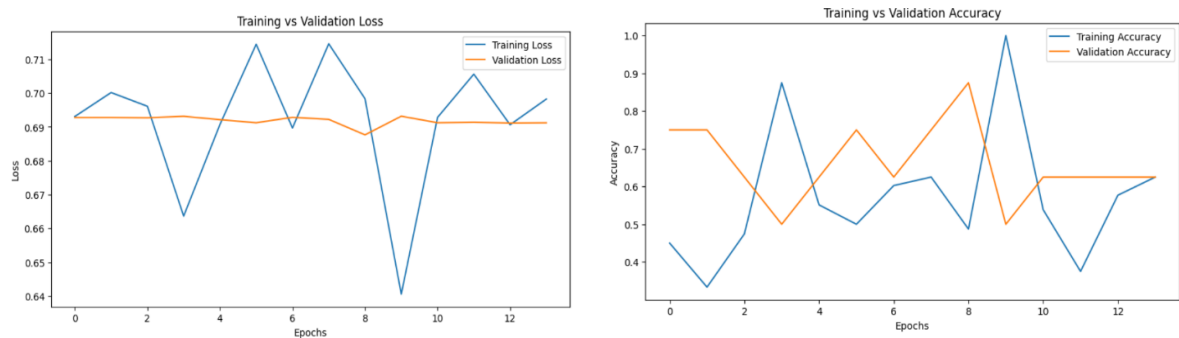
Gambar 5. Visualisasi Accuracy dan Loss Pada Skenario 1



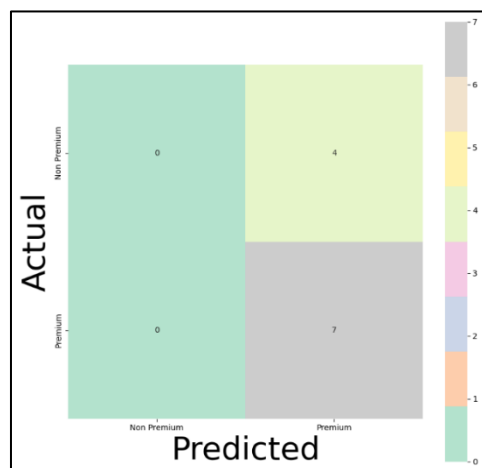
Gambar 6. Visualisasi Accuracy dan Loss Pada Skenario 2



Gambar 7. Visualisasi Accuracy dan Loss Pada Skenario 3

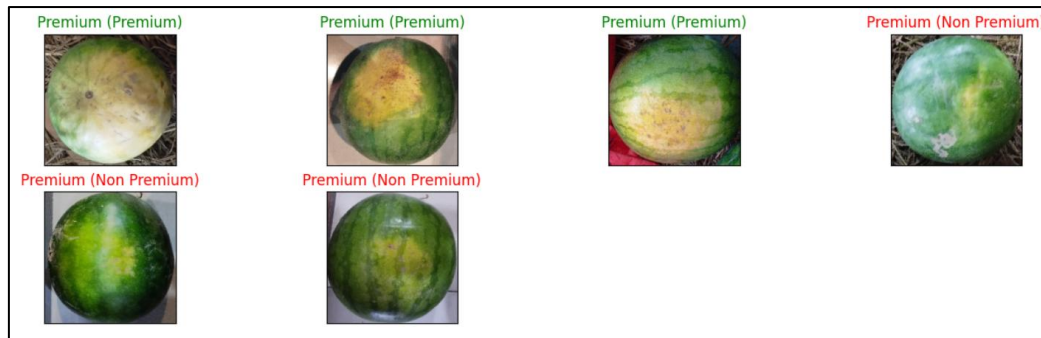


Gambar 8. Visualisasi Accuracy Dan Loss Pada Skenario 4



Gambar 9. Hasil matriks konfusi terbaik

Gambar 9 menunjukkan hasil terbaik dari matriks konfusi. Dari semua skenario pengujian model menunjukkan indikasi model belum belajar dengan baik (*underfitting*) dan selalu mengklasifikasi tingkat kemanisan semangka manohara ke kelas premium (manis), hal ini dikarenakan keterbatasan dalam hal kualitas dan banyaknya jumlah *dataset*. Gambar 10 merupakan tes visualisasi klasifikasi.



Gambar 10. Tes Visualisasi Klasifikasi

Simpulan

Hasil dari penelitian ini adalah model VGG-16 berhasil diimplementasikan, namun akurasi model masih terbatas (maks 67,42%) pada skenario 4 dan menunjukkan gejala *underfitting* serta kecenderungan mengklasifikasi ke satu kelas. Permasalahan *underfitting* disebabkan oleh keterbatasan jumlah dan kualitas data. Penelitian ini menggunakan pendekatan tahap awal dan butuh skalabilitas dengan lebih banyak data dan eksperimen lebih lanjut. Penelitian mendatang disarankan memperbanyak dan meningkatkan kualitas *dataset*, menggunakan teknik augmentasi lanjutan, dan membandingkan arsitektur alternatif seperti *MobileNet*, *ResNet*, *EfficientNet* dan lainnya.

Daftar Pustaka

- [1] Q. A'yun and F. Utamingrum, "Rancang Bangun Deteksi Kemanisan Buah Semangka menggunakan Metode Gray Level Co-Occurrence Matrix dan Backpropagation Neural Network berbasis Raspberry Pi," 2022. [Online]. Available: <http://j->
- [2] A. U. S. Ghouri and A. N. Sugiharto, "Kajian Penampilan Semangka (*Citrullus lanatus*) Dari Beberapa Kombinasi Persilangan Diploid-tetraploid Study on the Performance of Watermelon (*Citrullus lanatus*) from Several Cross Combination of Diploid-Tetraploid," *Jurnal Produksi Tanaman*, vol. 12, no. 9, pp. 403–412, 2024, doi: 10.21776/ub.protan.2024.012.09.05.
- [3] Dinas Pertanian, "Produksi Tanaman Buah-buahan," Badan Pusat Statistik. Accessed: Sep. 18, 2024. [Online]. Available: <https://www.bps.go.id/indicator/55/62/1/produksi-tanaman-buah-buahan.html>
- [4] Kementerian Pertanian, *Statistik Konsumsi Pangan Tahun 2022*. 2022.
- [5] M. Otang, Y. Da Rato, and S. Noni, "Preferensi Konsumen Terhadap Pembelian Buah Semangka (*Citrullus Vugaris*) di Kebun Praktek Fakultas Pertanian Universitas Nusa Nipa Indonesia," *Jurnal Ilmiah Wahana Pendidikan*, vol. 7, no. 2, May 2021, doi: 10.5281/zenodo.4736475.
- [6] M. A. Ikhsanudin, E. Setyati, and H. Junaedi, "Deteksi Tingkat Kemanisan Buah Semangka (*Citrullus Lanatus*) Berdasarkan Ciri Kulit Buah Dengan Menggunakan Metode CNN (Convolutional Neural Network)," *JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 8, no. 4, pp. 1501–1513, Dec. 2023, doi: 10.29100/jupi.v8i4.4118.
- [7] B. W. Kurniadi, H. Prasetyo, G. L. Ahmad, B. Aditya Wibisono, and D. Sandya Prasvita, "Analisis Perbandingan Algoritma SVM dan CNN untuk Klasifikasi Buah," *Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA) Jakarta-Indonesia*, p. 1, Sep. 2021.
- [8] H. F. Setyawan, B. Nugroho, and Y. V. Via, "Analisis Perbandingan Performansi Algoritma CNN dan NN Pada Buah Pir," *Jurnal Mahasiswa Teknik Informatika*, vol. 7, no. 6, pp. 3114–3120, Dec. 2023.
- [9] B. S. E. Dwi and D. R. I. M. Setiadi, "Deteksi Tumor Otak Dengan Metode Convolutional Neural Network," *Jurnal Eksplora Informatika*, vol. 13, no. 2, pp. 188–197, Mar. 2024, doi: 10.30864/eksplora.v13i2.971.
- [10] A. E. Wijaya, W. Swastika, and O. H. Kelana, "Implementasi Transfer Learning Pada Convolutional Neural Network Untuk Diagnosis Covid-19 dan Pneumonia Pada Citra X-Ray," *SAINSBERTEK Jurnal Ilmiah Sains & Teknologi*, vol. 2, Sep. 2021.
- [11] Rismiyati and A. Luthfiarta, "VGG16 Transfer Learning Architecture for Salak Fruit Quality Classification," *Telematika*, vol. 18, p. 37, Mar. 2021, doi: 10.31315/telematika.v18i1.4025.
- [12] J. Halim and A. N. Fajar, "Klasifikasi Pisang Berbasis Algoritma VGG16 Melalui Metode CNN Deep Learning," *INFORMASI (Jurnal Informatika dan Sistem Informasi)*, vol. 15, no. 1, May 2023.
- [13] R. R. Cameron, S. R. Afriani, Z. Multazam, S. R. Yusticia, A. Febriani, and R. Saputra, "Evaluasi Frekuensi Penanaman Tanaman Semangka (*Citrullus lanatus*) Terhadap Keparahan Penyakit dan

- Produktivitas Tanaman Semangka,” *J-Plantasimbiosa*, vol. 5, no. 2, pp. 1–8, Nov. 2023, doi: 10.25181/jplantasimbiosa.v5i2.3299.
- [14] Y. Safitri and D. S. Juwita, “Pengabdian Kepada Masyarakat Tentang Diversifikasi Buah Semangka Di Desa Ridan Permai Kecamatan Bangkinang Kota Kabupaten Kampar Tahun 2021,” *Community Development Journal : Jurnal Pengabdian Masyarakat*, vol. 3, no. 3, pp. 2193–2195, Nov. 2022, doi: 10.31004/cdj.v3i3.13266.
- [15] B. P. dan I. B. T. A. DJBC, *Nilai Brix Untuk Menentukan Kualitas Pada Buah-Buahan. Indonesia Customs And Excise Laboratory Bulletin*. Jakarta, 2016.
- [16] Anhar and R. A. Putra, “Perancangan dan Implementasi Self-Checkout System pada Toko Ritel menggunakan Convolutional Neural Network (CNN),” *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, vol. 11, no. 2, p. 466, Apr. 2023, doi: 10.26760/elkomika.v11i2.466.
- [17] M. D. Darajat, Y. A. Sari, and R. C. Wihandika, “Convolutional Neural Network untuk Klasifikasi Citra Makanan Khas Indonesia,” vol. 5, no. 11, pp. 4764–4769, 2021, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [18] K. Azmi, S. Defit, and S. Sumijan, “Implementasi Convolutional Neural Network (CNN) Untuk Klasifikasi Batik Tanah Liat Sumatera Barat,” *JURNAL UNITEK*, vol. 16, no. 1, pp. 28–40, Jun. 2023, doi: 10.52072/unitek.v16i1.504.
- [19] I. Firmansyah and B. H. Hayadi, “Komparasi Fungsi Aktivasi Relu Dan Tanh Pada Multilayer Perceptron,” *JIKO (Jurnal Informatika dan Komputer)*, vol. 6, no. 2, p. 200, Sep. 2022, doi: 10.26798/jiko.v6i2.600.
- [20] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” in *ICLR 2015*, Sep. 2014, pp. 1–14. [Online]. Available: <http://arxiv.org/abs/1409.1556>
- [21] A. Saputro, S. Mu'min, Moch. Lutfi, and H. Putri, “Deep Transfer Learning Dengan Model Arsitektur VGG16 Untuk Klasifikasi Jenis Varietas Tanaman Lengkeng Berdasarkan Citra Daun,” *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 6, no. 2, pp. 609–614, Sep. 2022, doi: 10.36040/jati.v6i2.5456.
- [22] N. K. Pratama and F. T. Anggraeny, “Deteksi Lampu Lalu Lintas Dan Zebra Cross Menggunakan Mobilenetv2 Single Shot Detector,” *Jurnal Informatika Kaputama (JIK)*, vol. 7, no. 2, pp. 225–232, Jul. 2023.
- [23] lip Supiyani and N. Arifin, “Identifikasi Nomor Rumah Pada Citra Digital Menggunakan Neural Network,” *METHODIKA: Jurnal Teknik Informatika dan Sistem Informasi*, vol. 8, no. 1, pp. 18–21, Mar. 2022, doi: 10.46880/mtk.v8i1.921.
- [24] W. M. Pradnya D and A. P. Kusumaningtyas, “Analisis Pengaruh Data Augmentasi Pada Klasifikasi Bumbu Dapur Menggunakan Convolutional Neural Network,” *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 6, no. 4, pp. 2022–2031, Oct. 2022, doi: 10.30865/mib.v6i4.4201.
- [25] D. Husen, “Evaluasi Teknik Augmentasi Data Untuk Klasifikasi Tumor Otak Menggunakan Cnn Pada Citra Mri,” *TEKNIMEDIA: Teknologi Informasi dan Multimedia*, vol. 5, no. 2, pp. 219–228, Dec. 2024, doi: 10.46764/teknimedia.v5i2.220.
- [26] N. M. Surbakti et al., “Penggunaan Bahasa Pemrograman Python dalam Pembelajaran Kalkulus Fungsi Dua Variabel,” *Algoritma : Jurnal Matematika, Ilmu pengetahuan Alam, Kebumihan dan Angkasa*, vol. 2, no. 3, pp. 98–107, May 2024, doi: 10.62383/algoritma.v2i3.67.
- [27] R. F. Muharram and A. Suryadi, “Implementasi Artificial Intelligence Untuk Deteksi Masker Secara Realtime Dengan Tensorflow Dan Ssdmobilenet Berbasis Python,” *Jurnal Widya*, vol. 3, no. 2, pp. 281–290, Oct. 2022, [Online]. Available: <https://jurnal.amikwidyaloka.ac.id/index.php/awl>
- [28] M. Chandra and E. Pramana, “Image Recognition Menggunakan Metode Cosine Distance untuk Aplikasi Penanganan Food Waste,” *Journal of Intelligent System and Computation*, vol. 4, no. 2, pp. 77–84, Oct. 2022, doi: 10.52985/insyst.v4i2.250.
- [29] P. A. Setyawan, A. Danuarta, F. N. Alam, and T. Agustin, “Analisis Perbandingan Optimizer Pada Model Convolutional Neural Network Untuk Deteksi Kanker Paru-paru,” *SEMINAR NASIONAL AMIKOM SURAKARTA (SEMNAS)*, vol. 2, pp. 320–330, Dec. 2024.
- [30] D. Kurniadi, M. R. Shidiq, and A. Mulyani, “Perbandingan Penggunaan Optimizer dalam Klasifikasi Sel Darah Putih Menggunakan Convolutional Neural Network,” *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*, vol. 14, no. 1, pp. 77–86, Mar. 2025, doi: 10.22146/jnteti.v14i1.17162.
- [31] N. S. Mahajaya, P. D. W. Ayu, and R. R. Huizen, “Pengaruh Optimizer Adam, AdamW, SGD, dan LAMB terhadap Model Vision Transformer pada Klasifikasi Penyakit Paru-paru,” *Prosiding Seminar Hasil Penelitian Informatika dan Komputer (SPINTER)*, vol. 1, no. 2, pp. 818–823, Apr. 2024, [Online]. Available: <https://www.kaggle.com/datasets/tawsifurrahman/covid19-radiography-database>,
- [32] L. Qadrini, A. Sepperwali, and A. Aina, “Decision Tree Dan Adaboost Pada Klasifikasi Penerima Program Bantuan Sosial,” *Jurnal Inovasi Penelitian*, vol. 2, no. 7, pp. 1959–1966, 2021.