

# PENGENALAN KUE TRADISIONAL INDONESIA MENGUNAKAN ALGORITMA CONVOLUTIONAL NEURAL NETWORK

Devid Sumarlie<sup>1</sup> Chairisni Lubis<sup>2</sup> Teny Handhayani<sup>3</sup>

<sup>1,2,3</sup> Teknik Informatika, Fakultas Teknologi Informasi, Universitas Tarumanagara

Jl. Letjen S Parman no 1, Jakarta 11440 Indonesia

Corresponding Author: [chairisnil@fti.untar.ac.id](mailto:chairisnil@fti.untar.ac.id)

## Abstrak

*Di Indonesia banyak sekali kue yang termasuk ke dalam kategori jajanan tradisional. Jajanan tradisional merupakan budaya khas nusantara yang wajib dilestarikan oleh orang Indonesia. Kue tradisional adalah makanan ringan yang disukai masyarakat karena bentuknya padat dan mengenyangkan. Kue tradisional memiliki beragam Tekstur, bentuk serta warna nya yang sangat beragam dan ada yang mirip/identik satu sama sama lain, sehingga agak menyulitkan seseorang untuk mengidentifikasi kue. Masalah yang dihadapi oleh pembeli kue seringkali adalah tidak mengetahui nama dari suatu kue karena dari banyaknya jenis kue yang dijual di pasar. Berkat kemajuan teknologi banyak masyarakat sekitar memanfaatkan sosial media untuk mengambil foto makanan tetapi untuk mengenali kue tersebut masih banyak orang yang belum benar-benar memahami betul kue tradisional ketimbang kue modern. Permasalahan di atas dapat diselesaikan jika dibuat suatu sistem untuk mengenali citra/foto kue tersebut dan komputer dapat diprogram dan dilatih untuk mengklasifikan kue ke dalam kategori kue tertentu dengan memanfaatkan citra kue tadi dengan menggunakan algoritma Convolutional Neural Network(CNN). Hasil pengujian terbaik yang didapat adalah pengujian dengan mengikutsertakan data augmentation pada saat training, yang dimana DenseNet121 memiliki akurasi yang lebih tinggi daripada VGG-16 yakni sebesar 80% dan pengujian Convolutional Neural Network, DenseNet121, yang menggunakan k-fold cross validation dengan hasil akurasi fold 1 yang lebih tinggi yakni sebesar 77% dan peningkatan drastis sampai hingga fold 5. Jika tanpa menggunakan data augmentasi maka DenseNet121 mencapai nilai akurasi terbaik sebesar 83% dan tanpai menggunakan transfer learning dengan nilai learning rate sebesar 0,00001 dan batch size sebesar 16*

**Kata kunci:** Algoritma, Convolutional Neural Network, DenseNet121, Kue tradisional Indonesia, VGG-16

## Abstract

*In Indonesia, a lot of cakes are included in the category of traditional snacks. Traditional snacks are a unique culture of the archipelago that must be preserved by Indonesians. Traditional cakes are snacks that people like because they are dense and filling. Traditional cakes have a variety of textures, shapes and colors are very diverse and some are similar to each other, so it is rather difficult to identify the cake. The problem faced by buyers is that they often do not know the name of a cake because of the many types of cakes sold in the market. Technological advances have also caused many local people to use social media to take photos of food, but to recognize these cakes, there are still many people who do not really understand traditional cakes compared to modern cakes. The above problem can be solved if a system is made to recognize the image/photo of the cake and the computer can be programmed and to classify the cake into a certain category of cake by utilizing the image of the cake using the Convolutional Neural Network (CNN) algorithm. The best test results are tests that include data*

*augmentation during training, where VGG-16 has a higher accuracy than DenseNet121 which is 80% and DenseNet121 testing which uses k-fold cross validation with an accuracy of fold 1 which is 77% and a drastic increase up to fold 5. If without using data augmentation, the best result obtained is an accuracy of 83% achieved by DenseNet121 without transfer learning, learning rate 0,00001 and batch size 16.*

**Keywords**— *Algoritma, Convolutional Neural Network, DenseNet121, Indonesian traditional snacks, VGG-16.*

## 1. PENDAHULUAN

Kebutuhan pangan merupakan salah satu kebutuhan sehari-hari yang penting dalam hidup manusia. Makanan ringan merupakan salah satu kudapan favorit masyarakat dan juga merupakan salah satu cara untuk memenuhi kebutuhan pangan. Kue tradisional adalah makanan ringan yang disukai masyarakat karena bentuknya padat dan mengenyangkan. Saat ini kue tradisional yang dijual di pasar sangatlah beragam jenisnya. Kue tradisional memiliki beragam Tekstur, bentuk serta warna nya yang sangat beragam dan ada yang mirip/identik satu sama sama lain, sehingga agak menyulitkan seseorang untuk mengidentifikasi kue. Masalah yang dihadapi oleh pembeli kue seringkali adalah tidak mengetahui nama dari suatu kue karena dari banyaknya jenis kue yang dijual di pasar. Disamping itu, berkat kemajuan teknologi yang pesat, banyak masyarakat bisa memanfaatkan sosial media untuk mengambil foto makanan tetapi untuk mengenali kue tersebut masih banyak orang yang belum benar-benar memahami betul kue tradisional. Kebutuhan pangan juga mengikuti kemajuan jaman sehingga kue tradisional kian terlupakan dan dapat hilang dari pasaran karena tergerus dengan menjamurnya makanan kekinian. Oleh karena itu makanan kue tradisional harus dilestarikan keberadaannya.

Permasalahan di atas dapat diselesaikan jika dibuat suatu sistem untuk mengenali citra/foto kue tersebut dan komputer dapat diprogram dan dilatih untuk mengklasifikasikan kue ke dalam kategori kue tertentu dengan memanfaatkan citra kue dengan menggunakan algoritma *Convolutional Neural Network (CNN)*, yang merupakan salah satu algoritma Machine Learning dari pengembangan *Multi Layer Perceptron (MLP)* yang dapat digunakan untuk melakukan pengenalan citra. CNN dipilih karena mampu melakukan ekstraksi fitur secara otomatis dan memiliki akurasi tinggi untuk pengenalan citra. Pada perancangan ini, akan digunakan arsitektur CNN berupa *VGG16* dan *DenseNet121* untuk pengenalan citra. *VGG16* dipilih karena memiliki tingkat akurasi yang baik dan kedalaman layer yang lebih dalam dan parameter yang lebih banyak, tetapi pada *VGG16* terdapat masalah yang memerlukan waktu pelatihan yang lebih lama dan model yang lebih berat karena ukurannya besar dibandingkan dengan *DenseNet121*. Terdapat 2 opsi arsitektur *VGG* yang umum digunakan, yaitu *VGG16* dan *VGG19*, *VGG16* dipakai karena performa *VGG16* tidak berbeda jauh dengan *VGG19*, hanya saja *VGG19* memiliki 3 convolutional layer tambahan dan jumlah parameter yang lebih banyak. *Densenet121* dipilih karena asitektur *DenseNet* pola konektivitasnya sederhana antar lapisan. Jaringan ini menghubungkan semua lapisan sedemikian rupa sehingga setiap lapisan memperoleh masukan tambahan dari semua lapisan sebelumnya dan meneruskan peta fiturnya sendiri ke semua lapisan berikutnya. Kelebihan ini yang tidak dimiliki oleh *VGG16*.

## 2. METODE PENELITIAN

Pada penelitian pengenalan kue tradisional Indonesia ini, *Convolutional Neural Network (CNN)* digunakan untuk mengenali kue. Ada 2 arsitektur CNN yang digunakan dalam proses pengenalanya, yaitu: *VGG16* dan *DenseNet121*. Kedua arsitektur ini digunakan untuk mencari

arsitektur mana yang terbaik nilai akurasinya. Kue yang akan dikenali berjumlah 13 klas yaitu: kue cucur, dadar gulung, kue cucur, kue gemblong, kue klepon, kue lumpur, kue lupis, kue nagasari, kue ongol-ongol, kue putu, kue serabi, kue cenil, kue lemet dan kue timphan. Kelas kue yang tersebut dipilih karena beberapa kue tersebut cukup sulit untuk dibedakan

Dataset dikumpulkan melalui Google Image Search dengan 13 kelas nama kue tradisional Indonesia yang masing-masing kelas berjumlah antara 250-300 gambar. Pembagian data dilakukan dengan porsi 80:20 untuk porsi training dan testing. Jika menggunakan metode K-Fold Cross Validation maka data akan dibagi ke dalam 5 fold dan keseluruhan data training dan testing akan memiliki slot tersendiri dan berganti ganti posisi antara training dan testing agar merata dan adil dalam proses pelatihan model.

## 2.1 VGG-16

VGG16 adalah CNN yang dikembangkan bersama oleh *Visual Geometry Group* di Universitas Oxford dan *Google DeepMind*. Hal terpenting dalam *Convolutional Neural Network* adalah tingkat kedalaman layernya. *VGG* dibuat untuk memperbaiki parameter arsitektur lainnya, dan terus meningkatkan kedalaman jaringan dengan menambahkan lebih banyak lapisan konvolusi, yang layak karena penggunaan filter konvolusi kecil ( $3 \times 3$ ) di semua lapisan [3]

Ukuran default citra yang dapat diterima oleh *VGG-16* adalah  $224 \times 224$  pikseel. Pada penelitian ini, arsitektur yang digunakan adalah *VGG-16*. *VGG16* adalah jaringan saraf dalam arsitektur yang terdiri dari 16 lapisan. Arsitektur *VGG16* ditunjukkan pada Gambar 1. *VGG16* memiliki 13 lapisan lapisan konvolusi, 2 lapisan yang sepenuhnya terhubung, dan 1 lapisan pengklasifikasi (*classification layer*).

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input ( $224 \times 224$ RGB image)					
conv3-64	conv3-64 LRN	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 conv1-256	conv3-256 conv3-256 conv3-256	conv3-256 conv3-256 conv3-256 conv3-256
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

**Gambar 1** Arsitektur VGG-16

Sumber: Karen Simonyan and Andrew Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, <https://arxiv.org/abs/1409.1556>, 2 Maret 2022.

Konfigurasi stride konvolusi tetap ke 1 pixel, setelah konvolusi, yaitu padding adalah 1 piksel untuk lapisan konvolusional  $3 \times 3$ . Pengelompokan spasial dilakukan dengan lima lapisan *max-pooling*, yang mengikuti beberapa lapisan konvolusional (tidak semua lapisan konvolusional diikuti dengan *max-pooling*). *Max-pooling* dilakukan melalui jendela  $2 \times 2$  piksel, dengan stride 2. Tumpukan lapisan konvolusi (yang memiliki kedalaman berbeda dalam arsitektur berbeda) diikuti oleh tiga lapisan Terhubung Sepenuhnya (*Fully Connected Layer*) yakni dua yang pertama memiliki 4096 saluran masing-masing, yang selanjutnya melakukan 1000 kelas klasifikasi *ILSVRC*, jumlah diatur 1000 karena berisi 1000 saluran (satu untuk setiap kelas).

## 2.2 Densenet121

*Dense Convolutional Network (DenseNet)* adalah arsitektur CNN yang menghubungkan setiap lapisan/blok ke setiap lapisan/blok lainnya dengan cara *feedforward*. Sedangkan jaringan convolution tradisional dengan Lapisan memiliki koneksi  $L - 1$  antara setiap lapisan dan lapisan berikutnya jaringan memiliki koneksi langsung  $L (L + 1) / 2$ . *DenseNet* memiliki beberapa kelebihan diantaranya masalah gradien yang dapat teratasi, memperkuat lapisan, lapisan yang berulang-ulang atau redundan, dan mengurangi jumlah parameter [4]

Untuk memastikan kemaksimalan aliran informasi antar lapisan dalam jaringan, maka semua layer akan dihubungkan (dengan ukuran peta fitur yang cocok) secara langsung dengan satu sama lain. Untuk mempertahankan sifat *feed-forward*, setiap lapisan memperoleh input tambahan dari semua lapisan sebelumnya dan meneruskan peta fiturnya sendiri ke semua lapisan-lapisan berikutnya.

Pada gambar 2 di bawah, DenseNet yang menggunakan 4 blok. Lapisan yang berada pada *dense block* berisikan batch normalization, ReLU activation dan convolution dengan filter  $3 \times 3$ . Lapisan antara 2 blok yang berdekatan disebut sebagai transition layer dan perubahan ukuran fitur melalui convolution dan pooling (average pooling).

Layers	Output Size	DenseNet-121	DenseNet-169	DenseNet-201	DenseNet-264
Convolution	$112 \times 112$	$7 \times 7$ conv, stride 2			
Pooling	$56 \times 56$	$3 \times 3$ max pool, stride 2			
Dense Block (1)	$56 \times 56$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$
Transition Layer (1)	$56 \times 56$	$1 \times 1$ conv			
	$28 \times 28$	$2 \times 2$ average pool, stride 2			
Dense Block (2)	$28 \times 28$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$
Transition Layer (2)	$28 \times 28$	$1 \times 1$ conv			
	$14 \times 14$	$2 \times 2$ average pool, stride 2			
Dense Block (3)	$14 \times 14$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 64$
Transition Layer (3)	$14 \times 14$	$1 \times 1$ conv			
	$7 \times 7$	$2 \times 2$ average pool, stride 2			
Dense Block (4)	$7 \times 7$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 16$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$
Classification Layer	$1 \times 1$	$7 \times 7$ global average pool			
		1000D fully-connected, softmax			

**Gambar 2** DenseNet dengan jumlah 4 blok layer

Sumber : Gao Huang et al, *Densely Connected Convolutional Networks*, <https://arxiv.org/abs/1608.06993>, 2 Maret 2022

Fungsi aktivasi ReLU (ReLU *activation*) digunakan untuk mengubah nilai  $x$  menjadi 0 jika nilai  $x$  tersebut bernilai negatif, sebaliknya untuk nilai  $x$  tidak berubah apabila nilai tidak kurang dari 0. Hal tersebut ditunjukkan untuk meningkatkan pelatihan *Deep Neural Network*. Persamaan fungsi aktivasi ReLU dijelaskan dalam persamaan di bawah.

$$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases} \quad (1)$$

Keterangan:

$f(x)$  = Fungsi ReLU

$x$  = Nilai piksel

### 2.3 K-Fold Cross Validation

*Cross Validation* adalah metode statistik yang mengevaluasi dan membandingkan algoritma pembelajaran dengan membagi data menjadi dua yaitu data training dan data testing. Baik set data pelatihan dan data testing/validasi harus di *cross-over* dalam beberapa iterasi secara berkelanjutan sedemikian rupa sehingga setiap blok data memiliki peluang untuk divalidasi. Salah satu bentuk dari *cross validation* adalah *k-fold cross validation*.

Prinsip dari *k-fold cross validation* adalah membagi-bagi tiap kelompok data menjadi  $k$  bagian kelompok data yang selanjutnya data tersebut secara bergantian akan digunakan untuk *training* dan *testing* sejumlah  $k$  pengujian [5].

### 2.4 Transfer Learning

*Transfer learning* dalam *machine learning* adalah peningkatan proses pembelajaran baru untuk sistem yang dilakukan dengan transfer pengetahuan dari tugas terkait yang sebelumnya sudah dipelajari. Tujuan dari *transfer learning* adalah untuk meningkatkan pembelajaran dalam pelatihan dengan memanfaatkan pengetahuan dari beberapa rangkaian pelatihan yang sudah ada sebelumnya. Beberapa keunggulan *transfer learning* adalah performa awal pada saat training menggunakan pengetahuan dari *transfer learning* dapat dicapai lebih baik dibanding tanpa *transfer learning*, lalu yang kedua adalah jumlah waktu yang diperlukan untuk proses pembelajaran dapat lebih cepat dan efisien dan yang ketiga adalah kinerja akhir yang dapat dicapai lebih baik menggunakan *transfer learning* [6].

## 3. HASIL DAN PEMBAHASAN

Pengujian pertama yang dilakukan adalah pengujian pada algoritma CNN yaitu *VGG-16* dan *Densenet121*. Pengujian dilakukan di *Google Colaboratory* dan kedua algoritma CNN tersebut diuji untuk memastikan apakah keduanya dapat bekerja dengan baik untuk tugas *multiclass classification* dan apakah program ini mampu untuk melakukan klasifikasi citra kue tradisional Indonesia dengan benar.

Pengujian algoritma dilakukan di lingkungan sistem *Google Colaboratory*. Pengujian dilakukan dengan menggunakan dataset yang dikumpulkan melalui *Google Image Search*. Dalam dataset yang ada terdapat 13 kelas nama kue tradisional Indonesia yang masing-masing kelas berjumlah antara 250-300 gambar. Pembagian data dilakukan dengan porsi 80:20 untuk porsi *training* dan *testing*. Jika menggunakan metode *K-Fold Cross Validation* maka data akan dibagi ke dalam 5 *fold* dan keseluruhan data training dan testing akan memiliki slot tersendiri dan berganti ganti posisi antara training dan testing agar merata dan adil dalam proses pelatihan

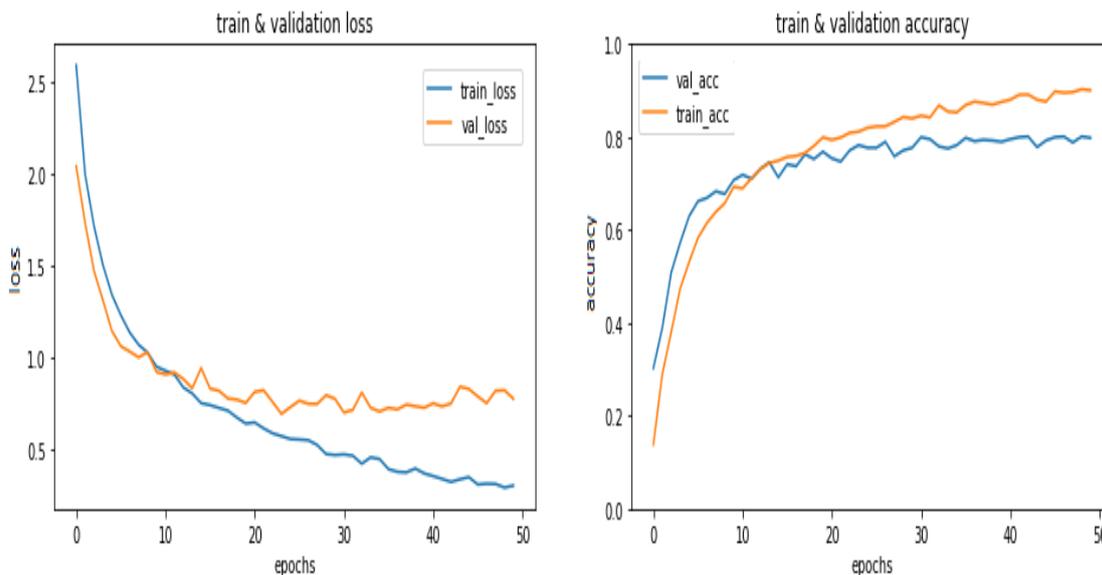
model (*VGG-16* dan *DenseNet-121*). Model dengan performa terbaik akan dipilih untuk sistem pengenalan kue tradisional Indonesia.

Ada 4 skenario pengujian yang dilakukan pada penelitian ini, yaitu:

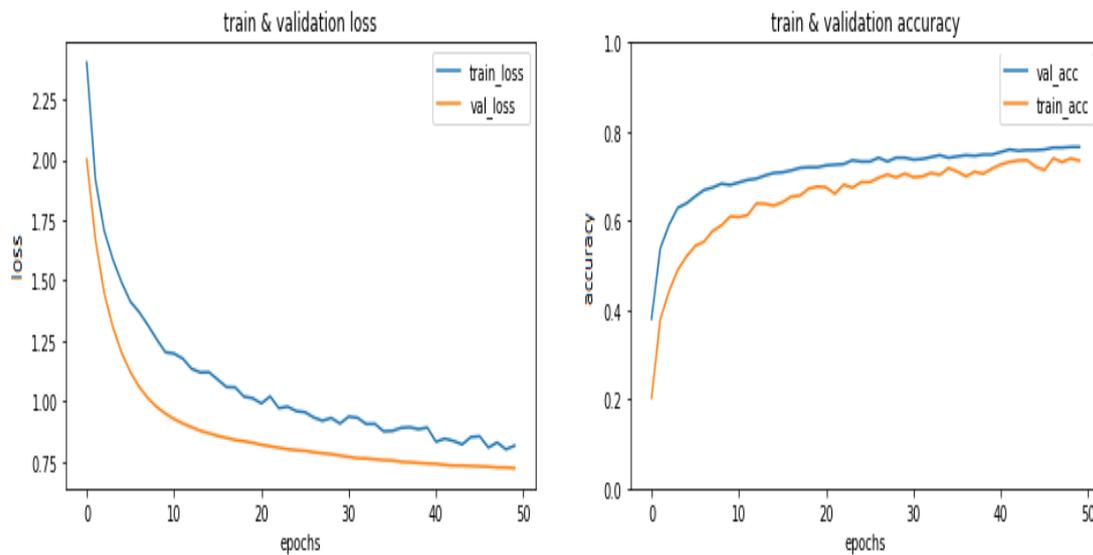
1. Pengujian dengan menggunakan **VGG-16 tanpa K-Fold Cross Validation**
2. Pengujian dengan menggunakan **DenseNet121 tanpa K-Fold Cross Validation**
3. Pengujian dengan menggunakan **VGG-16 dengan K-Fold Cross Validation**
4. Pengujian dengan menggunakan **DenseNet121 dengan K-Fold Cross Validation**

Dari hasil pengujian, hasil terbaik didapat dari pengujian dengan mengikutsertakan *data augmentation* pada saat *training*. *Data augmentation* digunakan untuk memperkaya jumlah dataset yang jumlah data dari keseluruhan kelasnya cukup sedikit. *VGG-16* memiliki akurasi yang lebih tinggi daripada *DenseNet121* yakni sebesar 80% dan memiliki kemampuan yang baik untuk kekonvergenan terhadap tingkat loss, serta pengujian *DenseNet121* yang menggunakan *k-fold cross validation* dengan hasil akurasi *fold 1* yang lebih tinggi yakni sebesar 77% dan peningkatan akurasi secara drastic dari *fold 2* sampai hingga *fold 5* dengan akurasi berkisar diantara 97-100%. Jika tanpa menggunakan *data augmentation*, hasil terbaik yang didapat nilai akurasi sebesar 83% yang diraih oleh *DenseNet121* dengan *transfer learning*, *learning rate*  $1e-5$  dan *batch size* 16.

Grafik akurasi dan *loss* dari pengujian yang menggunakan *data augmentation* serta hasil training algoritma CNN *VGG-16* maupun *DenseNet121* ditampilkan pada Gambar 2 dan 3 di bawah ini. Dan hasil Pengujian kedua metode ditampilkan pada Tabel 1:



**Gambar 2** Grafik training dari VGG-16 menggunakan data augmentation



Gambar 3 Grafik training dari DenseNet121 menggunakan data augmentation

Tabel 1 Hasil Pengujian dengan menggunakan VGG-16 dan DenseNet121

Model	Tingkat Akurasi
VGG-16 tanpa transfer learning (semua layer diaktifkan, learning rate 1e-5, batch size 32, 2 dense layer sebesar 512 unit, 1 layer batchnormalization dan 1 layer dropout)	71%
VGG-16 dengan transfer learning, learning rate 1e-5, batch size 32, 1 unit Dense Layer sebesar 512 neuron, penambahan layer batch normalization serta ditambahkan 1 dropout layer	57% (buruk)
VGG16 dengan transfer learning, learning rate 1e-4, batch size 32	57% (buruk)
VGG-16 tanpa transfer learning, batch size 16, dan learning rate 1e-5	73%
<b>VGG-16 (dengan data augmentation)</b>	<b>80%</b>
DenseNet121 dengan transfer learning, learning rate 1e-4, batch size 32, 2 buah Dense layer dengan 256 unit neuron dan batch normalization layer	80%
DenseNet121 tanpa transfer learning, learning rate 1e-4, batch size 32, batch normalization layer 2 buah, dropout 0.8	44% (buruk)
DenseNet121 dengan transfer learning, learning rate 1e-5, batch size 32	63%
DenseNet121 tanpa transfer learning, learning rate 1e-5, batch size 16	83%
<b>DenseNet121 (dengan data augmentation)</b>	<b>77%</b>

#### 4. KESIMPULAN

Kesimpulan yang didapat setelah melakukan penelitian: Pengenalan kue tradisional Indonesia dengan menggunakan Convolutional Neural Network ini adalah kedua model algoritma Convolutional Neural Network (CNN) yang digunakan sangat optimal bila diaplikasikan dengan menggunakan *data augmentation*. Hasil yang unggul diraih oleh VGG-16 dengan tingkat akurasi 80% dengan selisih 3% daripada DenseNet121 yang memiliki tingkat akurasi 77%

### DAFTAR PUSTAKA

- [1] Agustina, Astrid dan Sutrisna, Sutarki. “Ruang Komunal Kue Tradisional Di Senen”. Jurnal Stupa Universitas Tarumanagara. Vol. II, Nomor 2. Oktober 2020.
- [2] Khan, Salman, Rahmani, Hossein; Shah, Syed Afaq Ali; Bennamoun, Mohammed; Medioni, Gerard; and Dickinson, Sven. A Guide to Convolutional Neural Networks for Computer Vision. San Rafael: Morgan & Claypool Publisher, 2018.
- [3] Simonyan, Karen dan Zisserman, Andrew. Very Deep Convolutional Networks for Large-Scale Image Recognition. <https://arxiv.org/abs/1409.1556>, 2 Maret 2022.
- [4] Huang, Gao; Liu, Zhuang; and Maaten, Laurens van der. Densely Connected Convolutional Networks. <https://arxiv.org/abs/1608.06993>, 2 Maret 2022.
- [5] Refaeilzadeh Payam; Tang, Lei; and Liu, Huan, Cross Validation. [https://doi.org/10.1007/978-0-387-39940-9\\_565](https://doi.org/10.1007/978-0-387-39940-9_565), 27 Maret 2022.
- [6] Jose Rafael Magdalena; Lopez, Antonio Jose Serrano. Handbook Of Research On Machine Learning Applications and Trends: Algorithms, Methods and Techniques. Hershey: IGI Publishing, 2009.