



Klasifikasi klon teh berbasis deep CNN dengan residual dan densely connections

Tea clone classification using deep CNN with residual and densely connections

Ade Ramdan^{1*)}, Vicky Zilvan¹⁾, Endang Suryawati¹⁾, Hilman F. Pardede¹⁾, Vitria Puspitasari Rahadi²⁾

¹⁾Pusat Penelitian Informatika, Lembaga Ilmu Pengetahuan Indonesia
Jl. Sangkuriang Komplek LIPI, Bandung, Indonesia 40135

²⁾Pusat Penelitian Teh dan Kina
Kampung Gambung-Pasirjambu, Kotak Pos 1013 Bandung, Indonesia 40010

Cara sitasi: A. Ramdan, V. Zilvan, E. Suryawati, H. F. Pardede, and V. P. Rahadi, "Klasifikasi klon teh berbasis deep CNN dengan residual dan densely connections," *Jurnal Teknologi dan Sistem Komputer*, vol. 8, no. 4, pp. 289-296, 2020. doi: [10.14710/jtsiskom.2020.13768](https://doi.org/10.14710/jtsiskom.2020.13768), [Online].

Abstract – Tea clone of Gambung series is a superior variety of tea that has high productivity and quality. Smallholder farmers usually plant these clones in the same areas. However, each clone has different productivity or quality, so it is difficult to predict the production quality in the same area. To uniform the variety of clones in an area, smallholder farmers still need experts to identify each plant because one and other clones share the same visual characteristics. We propose a tea clone identification system using deep CNN with skip connection methods, i.e., residual connections and densely connections, to tackle this problem. Our study shows that the proposed method is affected by the hyperparameter setting and the combining feature maps method. For the combining method, the concatenation method on a densely connected network shows better performance than the summation method on a residual connected network.

Keywords – Gambung tea clone; deep CNN; skip connection; densely connected networks; residual connected networks

Abstrak – Klon-klon teh seri Gambung hasil pengembangan merupakan bahan tanaman unggul yang memiliki produksi dan kualitas tinggi untuk meningkatkan produk teh di Indonesia. Perkebunan rakyat biasanya menanam klon-klon tersebut dalam satu area kebun yang sama sehingga sulit memprediksi kualitas produk teh yang dihasilkan. Akan sangat membantu apabila area pada kebun tersebut diidentifikasi berdasarkan jenis klon yang ditanam. Namun, klon teh tersebut memiliki banyak kemiripan sehingga sulit bagi non pakar melakukan identifikasi untuk penyediaan dan penanaman bahan tanam pada perkebunan yang memiliki area sangat luas. Kajian ini mengusulkan sistem identifikasi klon teh berbasis deep CNN dengan menerapkan metode

skip connection, yaitu residual connections dan densely connections. Studi menunjukkan bahwa kinerja sistem dipengaruhi oleh pengaturan hyperparameter dan metode penggabungan peta fitur yang diperkenalkan kembali ke lapisan berikutnya melalui koneksi, dimana metode concatenation pada densely connected network dapat mencapai performa yang lebih baik dibandingkan metode penjumlahan pada residual connected networks.

Kata Kunci – klon teh Gambung; deep CNN; skip connection; densely connected networks; residual connected networks

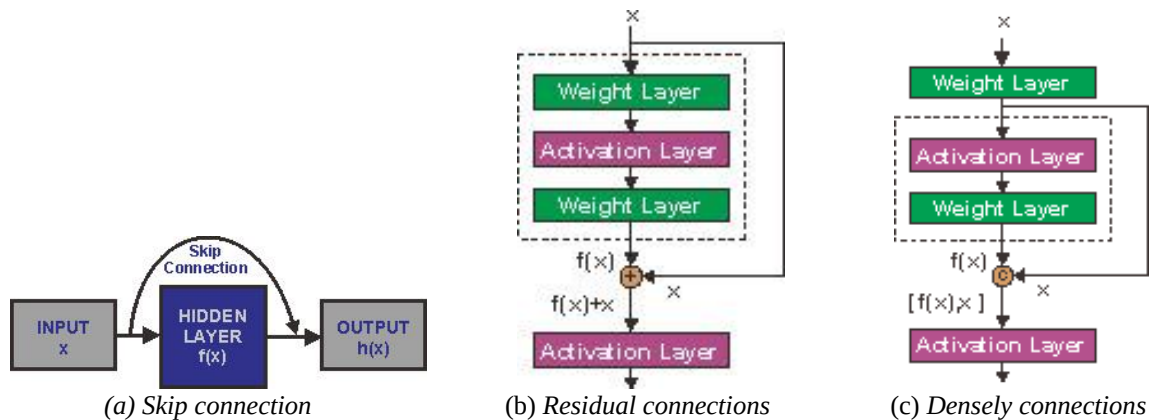
I. PENDAHULUAN

Permintaan dan persaingan di pasar dunia terhadap teh merupakan tantangan bagi Indonesia untuk meningkatkan produksi dan kualitas tanaman teh. Salah satu aspek yang dapat meningkatkan produksi dan kualitas tanaman teh adalah persiapan bahan tanaman unggul dalam budidaya teh. Pusat Penelitian Teh dan Kina (PPTK) telah mengembangkan dan mengeluarkan klon-klon unggulan dengan produksi tinggi dan kualitas baik sehingga produk teh Indonesia dapat ditingkatkan dan mampu bersaing di pasar global.

Klon unggulan yang dikembangkan PPTK adalah klon unggulan teh seri Gambung yang merupakan klon unggulan teh dari varietas assamika. Klon-klon unggulan varietas assamika tersebut terdiri dari GMB 1 sampai GMB 11 [1]. Setiap klon tersebut memiliki karakteristik masing-masing. Klon tanaman teh seri GMB 3, GMB 7, dan GMB 9 memiliki potensi produksi dan kadar katekin yang tertinggi [2], sedangkan GMB 1, GMB 4, dan GMB 11 juga memiliki kandungan katekin besar dan potensi produksi tinggi [3], [4].

Beberapa Seri Gambung ini memiliki karakteristik fisik yang mirip, yaitu antara lain klon seri GMB 1, GMB 3, GMB 4, GMB 7, GMB 9, dan GMB 11. Hal ini dapat menyulitkan dalam identifikasi klon seri GMB tersebut apabila ditanam dalam sebuah area perkebunan yang sama. Masalah tersebut sering ditemui di

^{*)}Penulis korespondensi (Ade Ramdan)
Email: ader001@lipi.go.id



Gambar 1. Skip connection secara umum dan diterapkan pada arsitektur CNN

lapangan. Lebih lanjut, prediksi kualitas teh yang dihasilkan dari area perkebunan tersebut sulit dilakukan. Identifikasi diperlukan ketika pihak perkebunan dan petani melakukan penyediaan dan penanaman bahan tanam dari keenam klon teh seri GMB tersebut.

Para ahli dibutuhkan untuk dapat mengidentifikasi klon seri GMB dengan kasat mata dalam memeriksa perbedaan dari ukuran fisik, tekstur, struktur tulang, dan warna daun [5]. Sulit bagi non-pakar atau bahkan petani untuk melakukan identifikasi dengan hasil yang akurat. Padahal, keterbatasan tenaga ahli yang dapat melakukan identifikasi secara manual dapat menyebabkan tidak terpenuhinya kebutuhan perkebunan yang memiliki area yang sangat luas. Metode yang dapat mengidentifikasi klon teh seri GMB tersebut perlu dikembangkan. Dengan adanya sistem identifikasi klon teh, petani dapat secara mandiri mengidentifikasi area perkebunan dengan jenis klon teh yang berbeda sehingga kualitas teh dapat lebih diprediksi.

Salah satu solusi untuk membuat sistem identifikasi otomatis klon teh seri GMB adalah dengan penggunaan pembelajaran mesin. Saat ini, teknologi terkemuka untuk identifikasi objek dalam *computer vision* adalah *deep learning* [6], [7]. *Deep learning* ini merupakan sistem pembelajaran mesin yang sangat mumpuni. *Convolutional Neural Network* (CNN) yang merupakan teknologi *deep learning* dimana penggunaannya telah berhasil diterapkan di bidang *computer vision* dan menunjukkan kinerja yang luar biasa dalam banyak aplikasi modern [8].

Namun, CNN yang sangat dalam memiliki masalah *overfitting* dan *vanishing gradient*, seperti yang terjadi pada studi sebelumnya dalam menggunakan model AlexNet dan VGGNet [9]. Metode mengidentifikasi klon teh berbasis CNN menggunakan kedalaman arsitektur CNN kedua model tersebut mengalami kesulitan mencapai konvergensi ketika digunakan untuk mengidentifikasi lebih dari 3 jenis klon teh seri GMB. Kedalaman arsitektur CNN terutama ketika diterapkan pada dataset kecil dapat menyebabkan *overfitting* [10]-[12] dan terjadinya *vanishing gradient* yang dapat memperburuk hasil pelatihan [9], [13], [14].

Metode *skip connection* yang diterapkan pada jaringan *deep learning* dapat mengatasi masalah tersebut. *Skip connection* dapat menangani masalah

vanishing gradien dengan efektif dalam [15], [16]. *Skip connection* ini merupakan koneksi tambahan antar-lapisan selain koneksi langsung di lapisan yang berdekatan [16] untuk meningkatkan aliran informasi di sepanjang lapisan [17]-[19].

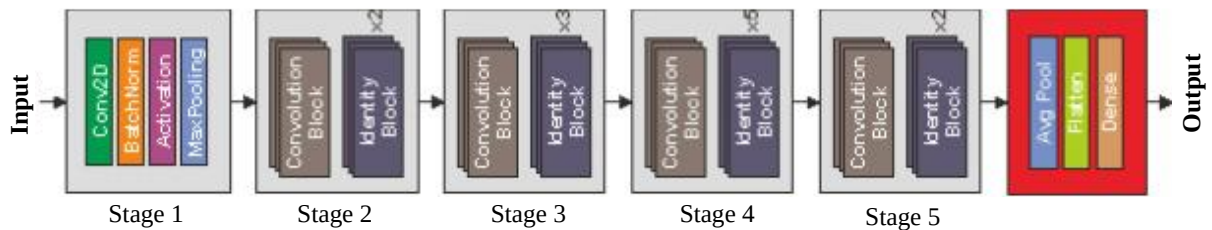
Makalah ini mengusulkan penerapan metode *skip connection* [16] pada *baseline deep CNN*, yaitu *residual connections* [15] dan *densely connections* [20] untuk mengatasi masalah yang terjadi pada model *deep CNN* konvensional dalam mengidentifikasi klon teh seri GMB. Pengaturan laju pembelajaran/*learning rate* (LR) dan *optimizer* yang berbeda pada pelatihan model *residual connected networks* dan model *densely connected networks* dapat menunjukkan bahwa kontribusi dari peta fitur yang diberikan kembali ke lapisan dalam melalui *skip connection* mampu mencapai kinerja generalisasi yang lebih baik.

II. METODE PENELITIAN

Arsitektur *deep CNN* dengan *skip connection* secara umum ditunjukkan pada Gambar 1. Dengan *skip connection*, sebuah jaringan digunakan untuk men-*skip* koneksi *hidden layers*. Hal ini dilakukan agar *loss* dari lapisan tersebut dapat dibawa ke jaringan berikutnya sehingga *loss* setelah proses *skip* adalah kombinasi *loss* dengan *loss* yang dibawa oleh *skip connection*. Efeknya adalah bertambahnya nilai gradien sehingga *vanishing gradient* dapat dihindarkan. Dalam penelitian ini, *skip connection* yang ditambahkan di arsitektur CNN adalah *residual connections* [15] dan *densely connections* [20] yang digunakan untuk identifikasi klon daun teh seri GMB.

A. Residual connections

Residual connections merupakan arsitektur *Deep CNN* (DCNN) yang sedang populer saat ini. He dkk. [15] mengusulkan *residual connections* ke dalam jaringan. *Residual connections* dikenal juga sebagai *skip connection* yang ditambahkan di *baseline* model CNN [21]. *Residual connections* memiliki keluaran yang dapat dihitung sebagai $f(x)+x$ dimana x adalah masukan yang ditambahkan ke output $f(x)$. Keluaran $f(x)$ adalah



Gambar 2. Arsitektur *residual connected networks*

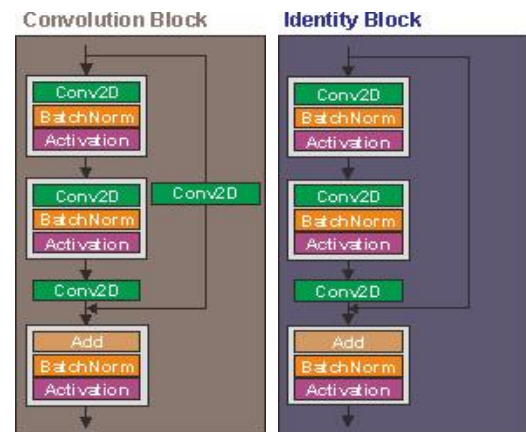
keluaran sejumlah lapisan aktivasi-bobot (*weight activation*), seperti yang ditunjukkan pada Gambar 1.

Residual connections dapat meneruskan informasi input secara langsung ke lapisan berikutnya. Informasi tersebut selalu mengalir tanpa hambatan dan gradien tidak akan pernah hilang tanpa terpengaruh dengan kedalaman jaringan, karena gradien pada setiap lapisan dapat dihitung dengan input asli [22].

Arsitektur *residual connections* yang digunakan dalam penelitian ini terdiri dari 50 lapisan yang terbagi menjadi 5 tahap seperti yang ditunjukkan pada Gambar 2. Tahap pertama menggunakan konvolusi 2D dengan 64 filter (7x7) dan *stride* 2x2, *batch normalization*, aktivasi menggunakan ReLU dan *Max Pooling* 3x3 dengan *stride* 2x2. Tahap kedua sampai kelima pada masing-masing stage terdiri dari *convolution block* dan *identity block*. Setiap *convolution block* dan *identity block* terdiri dari tiga lapisan konvolusional, seperti yang ditunjukkan pada Gambar 3. Setiap lapisan konvolusional diikuti oleh lapisan *batch normalization* dan fungsi aktivasi ReLU. *Identity block* x2, x3, dan x5 berarti terdapat dua, tiga, dan lima *identity block* yang menumpuk secara bersamaan.

Tahap kedua, lapisan konvolusional menggunakan konvolusi 2D pertama dengan 64 filter (1x1), konvolusi 2D kedua dengan 64 filter (3x3), dan konvolusi 2D ketiga dengan 256 filter (1x1). Tahap ketiga, lapisan konvolusional menggunakan konvolusi 2D pertama dengan 128 filter (1x1), konvolusi 2D kedua dengan 128 filter (3x3), dan konvolusi 2D ketiga dengan 512 filter (1x1). Tahap keempat, lapisan konvolusional menggunakan konvolusi 2D pertama dengan 256 filter (1x1), konvolusi 2D kedua dengan 256 filter (3x3), dan konvolusi 2D ketiga dengan 1024 filter (1x1). Tahap kelima, lapisan konvolusional menggunakan konvolusi 2D pertama dengan 512 filter (1x1), konvolusi 2D kedua dengan 512 filter (3x3), dan konvolusi 2D ketiga dengan 2048 filter (1x1).

Avg Pool merupakan *average pooling* (7x7) untuk meminimalkan *overfitting* dengan mengurangi jumlah parameter dalam model [23]. Lapisan *Avg Pool* digunakan untuk mengurangi dimensi spasial dari tensor dengan hanya mengambil nilai rata-rata fitur dari peta fitur pada lapisan terakhir. Lapisan *Flatten* digunakan untuk mengubah dimensi peta fitur menjadi nilai vector satu dimensi [24] untuk masukan lapisan *Dense* yang berfungsi untuk menambahkan lapisan yang *fully connected* di ujung jaringan.

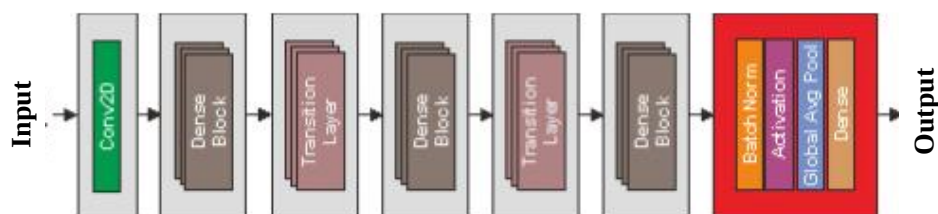


Gambar 3. Blok konvolusi dan identitas

B. Densely connections

Berdasarkan *residual connections*, Huang dkk. [20] mengusulkan konsep *densely connections*. Berbeda dari *residual connections* yang menggabungkan peta fitur melalui penjumlahan, *densely connections* tidak menjumlahkan, namun menggabungkan (*concatenate*) keluaran peta fitur dari lapisan dengan peta fitur yang masuk. *Densely connections*, sebagai arsitektur DCNN, menyajikan koneksi baru antara lapisan. Setiap lapisan terhubung ke semua lapisan ke depan. Akibatnya, setiap lapisan menerima fitur dari semua lapisan sebelumnya. Metode yang digunakan untuk menghubungkan fitur tersebut adalah metode konkatenasasi dari fitur, seperti yang ditunjukkan pada Gambar 1. Hasil keluaran fitur dari bobot lapisan tersebut yaitu $f(x)$, digabungkan dengan fitur yang masuk (x) menghasilkan keluaran $[f(x), x]$.

Struktur CNN *densely connected* yang digunakan dalam penelitian ini terdiri dari 40 lapisan yang terbagi menjadi 3 bagian, yaitu *dense blocks*, *transition layer* (lapisan transisi), dan *classification layer* (lapisan klasifikasi), seperti yang ditunjukkan pada Gambar 4. *Dense block* merupakan sekelompok lapisan *dense* yang terdiri dari *batch normalization*, aktivasi (ReLU), dan 3x3 lapisan konvolusi dengan 12 filter seperti yang ditunjukkan pada Gambar 5. Setiap lapisan *dense* terhubung langsung ke semua lapisan sehingga setiap lapisan menerima fitur dari semua lapisan sebelumnya. *Dense block* yang digunakan berjumlah 3 buah dengan menggunakan rasio pertumbuhan 12. Rasio pertumbuhan merupakan jumlah kanal di setiap peta fitur.



Gambar 4. Arsitektur *densely connected networks*

Lapisan transisi merupakan lapisan antara *dense block* yang melakukan konvolusi dan *pooling* yang berfungsi untuk menurunkan jumlah saluran dan melakukan *down-sampling* fitur. Lapisan transisi yang digunakan terdiri dari *batch normalization*, aktivasi (*ReLU*), 1×1 lapisan konvolusi, dan 2×2 *average pooling layer* dengan *stride* 2 seperti yang ditunjukkan pada Gambar 6. Lapisan klasifikasi merupakan bagian terakhir dari arsitektur *densely connections*. Lapisan klasifikasi ini terdiri dari *batch normalization*, aktivasi (*ReLU*), lapisan *Global Average Pooling* dan lapisan *Dense*.

C. Dataset

Dalam penelitian ini, ada enam klon seri GMB yang berjumlah 4066 citra dataset daun teh klon Gambung yang dikumpulkan. Citra diambil menggunakan berbagai DSLR dan kamera gawai cerdas di PPTK.

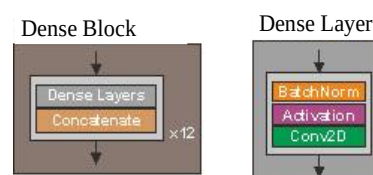
Percobaan dilakukan menggunakan enam jenis klon seri GMB, yaitu GMB 1, GMB 3, GMB 4, GMB 7, GMB 9 dan GMB 11 dari klon seri GMB. Dataset ini berisi 713 citra klon GMB 1, 598 citra klon GMB 3, 749 citra klon GMB 4, 638 citra klon GMB 7, 699 citra klon GMB 9, dan 669 citra klon GMB 11, seperti yang ditunjukkan pada Tabel 1. Dalam eksperimen, ukuran citra dari dataset tersebut diubah menjadi 224×224 piksel.

D. Persiapan percobaan

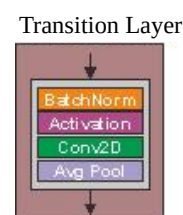
Dalam percobaan, dataset dibagi menjadi tiga bagian, yaitu 80 % dari total dataset keseluruhan sebagai data latih, 10 % dari total dataset keseluruhan sebagai data validasi, dan sisanya 10 % dari total dataset keseluruhan untuk pengujian dalam percobaan ini. Hal ini berarti data latih adalah sebanyak 3.281 dataset, data validasi sebanyak 398 dataset, dan data uji sebanyak 387 dataset.

Data pelatihan digunakan untuk melatih atau membangun model. Data validasi digunakan untuk mencari hasil *hyperparameter* yang paling baik saat melatih model. Data uji digunakan untuk menguji model setelah proses pelatihan selesai dengan menggunakan *hyperparameter* terbaik dari hasil data validasi.

Pada penelitian ini, model *residual connected networks*, selanjutnya disebut ResNet, dan *densely connected networks*, selanjutnya disebut DenseNet, dilatih dengan dua skenario. Skenario pertama menggunakan seluruh data latih dari total 80 % data sehingga data pelatihan sebanyak 4066 dataset. Skenario kedua menggunakan dataset hanya 75 % dari seluruh data latih atau 3050 jumlah data.



Gambar 5. Blok *Dense* dan lapisan *Dense*



Gambar 6. Lapisan transisi

Tabel 1. Data ringkasan klon seri GMB

Seri GMB	Sampel data citra	Jumlah data citra
1		713
3		598
4		749
7		638
9		699
11		669

Pada tiap skenario, kedua model menggunakan pengaturan *hyperparameter*, yaitu laju pembelajaran dan *optimizer* yang berbeda. *Optimizer* tersebut adalah Adam [25]-[27] dan *Stochastic Gradient Descent* (SGD) [28]-[30] yang merupakan algoritma optimasi paling banyak digunakan untuk pelatihan model DNN.

Pengaturan nilai laju pembelajaran yang diujikan pada eksperimen ini adalah 10^{-3} , 10^{-4} dan 10^{-5} dengan 500 *epochs* untuk setiap model. Hal tersebut dilakukan untuk melihat efek terhadap performa kedua model dalam mengidentifikasi klon teh dari gambar daun teh seri GMB.

Tabel 2. Hasil kinerja model arsitektur ResNet dari data uji skenario pertama

LR	SGD		Adam	
	Akurasi (%)	Galat	Akurasi (%)	Galat
1e-3	71,32	4,51	59,17	2,46
1e-4	82,69	5,56	64,86	2,11
1e-5	70,28	5,98	64,08	3,06

Tabel 3. Hasil kinerja model arsitektur DenseNet dari data uji skenario pertama

LR	SGD		Adam	
	Akurasi (%)	Galat	Akurasi (%)	Galat
1e-3	92,25	0,86	86,3	0,65
1e-4	93,02	1,22	93,28	0,44
1e-5	74,42	1,71	92,76	1,00

III. HASIL DAN PEMBAHASAN

Hasil eksperimen secara keseluruhan ditunjukkan pada [Tabel 2](#), [Tabel 3](#), [Tabel 4](#), dan [Tabel 5](#). Arsitektur ResNet dan DenseNet mencapai akurasi yang baik, yaitu di atas 80 % ketika menggunakan LR 10^{-4} . Hal tersebut mengindikasikan bahwa CNN dengan penambahan *skip connection* dapat digunakan dalam mengidentifikasi klon daun teh dengan baik. Hasil akurasi tertinggi untuk ResNet adalah 82,69 %, dicapai dengan menggunakan SGD seperti yang ditunjukkan pada [Tabel 2](#), sedangkan akurasi tertinggi yang dicapai DenseNet adalah 93,28 % dengan menggunakan Adam seperti yang ditunjukkan pada [Tabel 3](#).

Dari hasil eksperimen pada skenario pertama, diperoleh hasil validasi terbaik dalam mendapatkan *hyperparameter* yang paling baik dalam melakukan generalisasi. Nilai LR 10^{-4} adalah nilai terbaik untuk model DenseNet dan ResNet. Model DenseNet memiliki akurasi paling tinggi dengan galat paling kecil pada LR 10^{-4} menggunakan Adam dibandingkan ResNet yang memiliki performa maksimum dengan LR 10^{-4} dan SGD. Nilai *hyperparameter* terbaik tersebut digunakan untuk menguji model dengan menggunakan data uji. Hasil pengujian ditunjukkan pada [Tabel 2](#) dan [Tabel 3](#), sedangkan hasil kurva akurasi dan *loss* dari kedua model tersebut dinyatakan pada [Gambar 7](#) dan [Gambar 8](#).

Dalam skenario kedua, kedua model dilatih dengan data pelatihan yang hanya menggunakan 75 % dari seluruh data latih. Pengaturan nilai *hyperparameter* terbaik yang diperoleh dari hasil data validasi pada skenario kedua masih sama dengan nilai *hyperparameter* terbaik yang didapat pada skenario pertama. Hasil pengujian kinerja model dengan menggunakan *hyperparameter* terbaik dari hasil data validasi Hasil eksperimen ditunjukkan pada [Tabel 4](#) dan [Tabel 5](#).

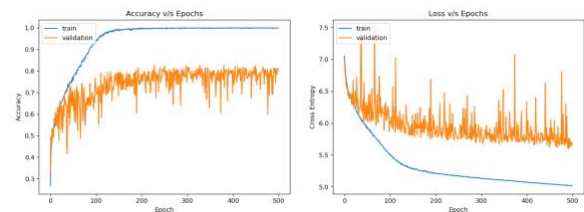
Hasil eksperimen tersebut menunjukkan pentingnya pemilihan laju pembelajaran LR. LR yang terlalu besar dapat mengakibatkan solusinya tidak mencapai konvergensi akibat galat yang fluktuatif, sedangkan jika terlalu kecil, model dapat terjebak dalam *local minimum* sehingga performa sulit membaik, walaupun jumlah epoch

Tabel 4. Hasil kinerja model arsitektur ResNet dari data uji skenario kedua

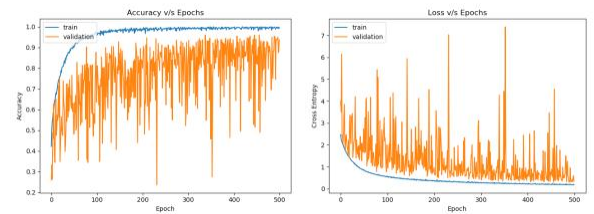
LR	SGD		Adam	
	Akurasi (%)	Galat	Akurasi (%)	Galat
1e-3	69,75	5,05	56,37	3,46
1e-4	77,39	5,84	52,23	3,94
1e-5	69,75	6,08	70,38	2,77

Tabel 5. Hasil kinerja model arsitektur DenseNet dari data uji skenario kedua

LR	SGD		Adam	
	Akurasi (%)	Galat	Akurasi (%)	Galat
1e-3	78,34	1,43	85,03	0,62
1e-4	64,65	2,28	91,72	0,45
1e-5	59,55	2,12	90,44	1,11



Gambar 7. Kurva akurasi dan *loss* model ResNet menggunakan LR 10^{-4} dan SGD pada skenario pertama

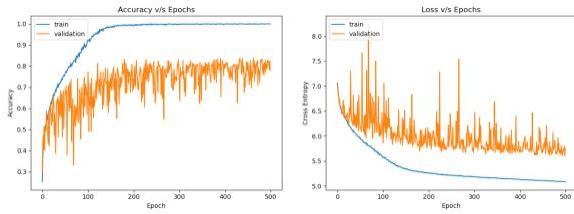


Gambar 8. Kurva akurasi dan *loss* model DenseNet menggunakan LR 10^{-4} dan Adam pada skenario pertama

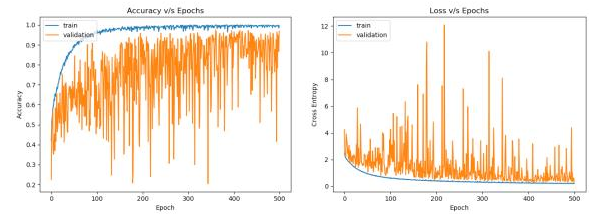
ditambah. Secara umum, Adam dapat mencapai akurasi yang lebih baik karena Adam menggunakan metode adaptif untuk mengubah nilai LR untuk epoch berikutnya.

Hasil pengamatan menunjukkan bahwa model DenseNet memiliki performa model yang relatif stabil dalam keadaan fit dibanding model ResNet seperti yang ditunjukkan pada [Gambar 9](#) dan [Gambar 10](#). Pada model DenseNet, setelah data pelatihan dikurangi kuantitasnya menghasilkan nilai akurasi 91,72 % atau turun 1,56 dari 93,28 % dengan nilai galat masih di bawah 0,5. Model ResNet menghasilkan nilai akurasi 77,39 % atau turun 5,3 dari 82,69 % dengan nilai galat yang masih di atas 5.

Dari kurva model ResNet pada [Gambar 7](#) dan [Gambar 9](#), nilai galat (*loss*) tidak ada tanda-tanda menaik seiring pertambahan epoch, bahkan cenderung sedikit menurun. Hal tersebut menandakan kinerja dari model ResNet masih bisa ditingkatkan dengan melakukan regularisasi [31], [32], seperti *dropout* [33]-



Gambar 9. Kurva akurasi dan loss model ResNet menggunakan LR 10^{-4} dan SGD pada skenario kedua



Gambar 10. Kurva akurasi dan loss model DenseNet menggunakan LR 10^{-4} dan Adam pada skenario kedua

Tabel 6. Matriks konfusi dari model ResNet menggunakan LR 10^{-4} dan SGD pada skenario pertama.

		Kelas Prediksi					
		GMB 1	GMB 3	GMB 4	GMB 7	GMB 9	GMB 11
Kelas Aktual	GMB 1	62	1	0	1	0	1
	GMB 3	4	47	2	1	0	1
	GMB 4	2	0	57	1	2	1
	GMB 7	1	0	3	55	3	3
	GMB 9	0	3	6	2	61	7
	GMB 11	0	2	2	4	14	38

Tabel 7. Matriks konfusi dari model ResNet menggunakan LR 10^{-4} dan SGD pada skenario kedua.

		Kelas Prediksi					
		GMB 1	GMB 3	GMB 4	GMB 7	GMB 9	GMB 11
Kelas Aktual	GMB 1	56	3	4	2	0	0
	GMB 3	5	38	0	1	1	0
	GMB 4	1	0	59	0	4	0
	GMB 7	3	1	3	16	7	3
	GMB 9	0	3	3	2	47	2
	GMB 11	0	1	4	5	13	27

[35], *label smoothing* [36]-[38], dan *mixup* [32], [39]. Regularisasi tersebut dapat mempengaruhi peningkatan kinerja dan ketahanan model yang akan diselidiki dalam studi selanjutnya.

Studi menunjukkan bahwa dengan mendapatkan *hyperparameter* yang paling baik dapat membantu model dalam melakukan generalisasi dengan lebih baik. Hal ini menunjukkan bahwa upaya mendapatkan *hyperparameter* yang baik dari sisi model masih diperlukan oleh jenis model apapun untuk membantu meningkatkan akurasi klasifikasi.

Hasil akurasi berdasarkan matriks konfusi dari model ResNet dengan data uji pada skenario pertama dan kedua disajikan pada Tabel 6 dan Tabel 7, sedangkan matriks konfusi dari model DenseNet pada skenario pertama dan kedua disajikan pada Tabel 8 dan Tabel 9. Hasil tersebut menunjukkan bahwa perbedaan performa antara ResNet dan DenseNet disebabkan performa ResNet pada kelas GMB 11 yang sering salah dikenali sebagai GMB 9. DenseNet, meskipun mampu meningkatkan akurasi untuk kelas GMB 11 dibandingkan ResNet, tetap memiliki mis-klasifikasi yang tidak sebaik kelas-kelas lain. Hal ini disebabkan karena secara visual GMB 11 dan GMB 9 sangatlah mirip.

Penelitian ini mengindikasikan bahwa penerapan CNN dengan *skip connection* berpotensi lebih baik

dibandingkan CNN tanpa *skip connection* yang sudah dilaporkan dalam [9], meskipun dengan jumlah data yang sama. Dengan jumlah parameter ResNet dan DenseNet yang lebih sedikit dibandingkan VGGNet, menyebabkan kebutuhan jumlah data untuk optimasi parameter menjadi lebih sedikit. Namun, berdasarkan observasi, DenseNet dan ResNet membutuhkan memori yang lebih besar dalam proses pelatihannya dan memerlukan komputasi yang lebih besar dibandingkan VGGNet.

IV. KESIMPULAN

Kedalaman suatu jaringan untuk mendapatkan akurasi yang lebih baik dapat tercapai jika CNN memiliki koneksi langsung di berbagai lapisan. Hasil kontribusi dari peta fitur yang diperkenalkan kembali ke lapisan yang lebih dalam melalui koneksi dapat meningkatkan kinerja model. Hasil eksperimen dengan menurunkan kuantitas dataset pelatihan dalam pengaturan yang berbeda menunjukkan bahwa DenseNet dan ResNet masih mampu mencapai performa yang baik dan stabil. Model DenseNet dalam identifikasi klon teh seri GMB mampu mencapai akurasi pelatihan yang sempurna dan mampu menunjukkan kinerja generalisasi yang lebih baik daripada ResNet.

Tabel 8. Matriks konfusi dari model DenseNet menggunakan LR 10^{-4} dan Adam pada skenario pertama

		Kelas Prediksi					
		GMB 1	GMB 3	GMB 4	GMB 7	GMB 9	GMB 11
Kelas Aktual	GMB 1	65	0	0	0	0	0
	GMB 3	0	55	0	0	0	0
	GMB 4	0	1	61	0	0	1
	GMB 7	3	1	1	50	3	7
	GMB 9	0	0	0	1	74	4
	GMB 11	0	0	0	1	3	56

Tabel 9. Matriks konfusi dari model DenseNet menggunakan LR 10^{-4} dan Adam pada skenario kedua

		Kelas Prediksi					
		GMB 1	GMB 3	GMB 4	GMB 7	GMB 9	GMB 11
Kelas Aktual	GMB 1	63	1	1	0	0	0
	GMB 3	0	44	0	1	0	0
	GMB 4	0	0	62	1	1	0
	GMB 7	0	0	0	30	2	1
	GMB 9	0	0	1	5	51	0
	GMB 11	1	0	3	3	5	38

UCAPAN TERIMA KASIH

Makalah ini difasilitasi oleh Pusat Penelitian Informatika dan hasil komputasi diperoleh dengan menggunakan fasilitas HPC, Lembaga Ilmu Pengetahuan Indonesia (LIPI). Terima kasih disampaikan kepada rekan peneliti di Pusat Penelitian Informatika LIPI, Pusat Pemanfaatan dan Inovasi IPTEK LIPI, dan Pusat Penelitian Teh dan Kina (PPTK) Gombang yang telah memberikan bantuan dalam penelitian ini.

DAFTAR PUSTAKA

- [1] M. S. Haq and K. Karyudi, "Upaya peningkatan produksi teh (*Camelia Sinensis (L.) O. Kuntze*) melalui penerapan kultur teknis," *Warta PPTK*, vol. 24, no. 1, pp. 71-84, 2013.
- [2] M. S. Haq and A. I. Mastur, "The growth of seedlings generated from cleft grafting of several superior tea clones," *Journal of Industrial and Beverage Crops*, vol. 5, no. 3, pp. 105-112, 2018. doi: [10.21082/jtidp.v5n3.2018.p105-112](https://doi.org/10.21082/jtidp.v5n3.2018.p105-112)
- [3] B. Sriyadi, "Penilaian hubungan genetika klon teh berdasarkan komponen senyawa kimia utama dan potensi hasil," *Jurnal Penelitian Teh dan Kina*, vol. 18, no. 1, pp. 1-10, 2015.
- [4] H. Mawarti and R. Ratnawati, "Pengaruh peningkatkan kadar kolesterol pada diet tinggi lemak oleh epigallocatechin gallate (EGCG) teh hijau klon Gmb4," *Prosiding Seminas Competitive Advantage*, vol. 1, no. 2, pp. 1-5, 2012.
- [5] Pusat Penelitian Teh dan Kina, "Klon GMB 1-11," [Online]. Available: <https://www.gamboeng.com/pages/detail/2015/59/146>. [Accessed: April. 8, 2020].
- [6] A. R. Pathak, M. Pandey, and S. Rautaray, "Application of deep learning for object detection," *Procedia Computer Science*, vol. 132, pp. 1706-1717, 2018. doi: [10.1016/j.procs.2018.05.144](https://doi.org/10.1016/j.procs.2018.05.144)
- [7] Y. Sun, Y. Liu, G. Wang, and H. Zhang, "Deep learning for plant identification in natural environment," *Computational Intelligence and Neuroscience*, vol. 2017, 7361042, 2017. doi: [10.1155/2017/7361042](https://doi.org/10.1155/2017/7361042)
- [8] C. Szegedy et al., "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition*, Boston, USA, Jun. 2015, pp.1-9. doi: [10.1109/CVPR.2015.7298594](https://doi.org/10.1109/CVPR.2015.7298594)
- [9] A. Ramdan et al., "Deep CNN detection for tea clone identification," *Jurnal Elektronika dan Telekomunikasi*, vol. 19, no. 2, pp. 45-50, 2019. doi: [10.14203/jet.v19.45-50](https://doi.org/10.14203/jet.v19.45-50)
- [10] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929-1958, 2014.
- [11] B. Wu, Z. Liu, Z. Yuan, G. Sun, and C. Wu, "Reducing overfitting in deep convolutional neural networks using redundancy regularizer," in *International Conference on Artificial Neural Networks*, Alghero, Italy, Sept. 2017, pp. 49-55. doi: [10.1007/978-3-319-68612-7_6](https://doi.org/10.1007/978-3-319-68612-7_6)
- [12] W Liu, Y. Zhang, X. Li, Z. Liu, B. Dai, T. Zhao, and L. Song, "Deep hyperspherical learning," in *31st International Conference on Neural Information Processing Systems*, California, USA, Dec. 2017, pp. 3953-3963.
- [13] B. Barz and J. C. Denzler, "Deep learning on small datasets without pre-training using cosine loss," in *IEEE Winter Conference on Applications of Computer Vision*, Snowmass Village, USA, Mar. 2020, pp. 1360-1369. doi: [10.1109/WACV45572.2020.9093286](https://doi.org/10.1109/WACV45572.2020.9093286)

- [14] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Thirteenth International Conference on Artificial Intelligence and Statistics*, Sardinia, Italy, May 2010, pp. 249-256.
- [15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, USA, Jun. 2016, pp. 770-778. doi: [10.1109/CVPR.2016.90](https://doi.org/10.1109/CVPR.2016.90)
- [16] A. E. Orhan and X. Pitkow, "Skip connections eliminate singularities," in *International Conference on Learning Representations*, Vancouver, Canada, May 2018, pp. 1-22.
- [17] H. Wu, J. Zhang, and C. Zong, "An empirical exploration of skip connections for sequential tagging," 2016, [arXiv:1610.03167](https://arxiv.org/abs/1610.03167).
- [18] T. Raiko, H. Valpola, and Y. Lecun, "Deep learning made easier by linear transformations in perceptrons," in *Fifteenth International Conference on Artificial Intelligence and Statistics*, La Palma, Spain, Apr. 2012, pp.924-932.
- [19] A. Graves, "Generating sequences with recurrent neural networks", 2013, [arXiv:1308.0850v5](https://arxiv.org/abs/1308.0850v5).
- [20] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, USA, Jul. 2017, pp. 2261-2269. doi: [10.1109/CVPR.2017.243](https://doi.org/10.1109/CVPR.2017.243)
- [21] W. Ma, Q. Yang, Y. Wu, W. Zhao, and X. Zhang, "Double-Branch multi-attention mechanism network for hyperspectral image classification," *Journal Remote Sensing*, vol. 11, no. 11, 1307, 2019. doi: [10.3390/rs11111307](https://doi.org/10.3390/rs11111307)
- [22] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *European Conference on Computer Vision*, Amsterdam, Netherland, Oct. 2016, pp. 630-645.
- [23] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: a comprehensive review," *Journal Neural Computation*, vol. 29, no. 9, pp. 2352-2449, 2017. doi: [10.1162/neco_a_00990](https://doi.org/10.1162/neco_a_00990)
- [24] W. Wartini, B. Minasny, M. Montazerolghaem, J. Padarian, R. Ferguson, S. Bailey, and A. B. Mcbratney, "Convolutional neural network for simultaneous prediction of several soil properties using visible/near-infrared, mid-infrared, and their combined spectra," *Geoderma*, vol. 352, pp. 251-267, 2019. doi: [10.1016/j.geoderma.2019.06.016](https://doi.org/10.1016/j.geoderma.2019.06.016)
- [25] D. P. Kingma and J. L. Ba, "Adam: a method for stochastic optimization," in *International Conference on Learning Representations*, Banff, Canada, Apr. 2014, pp. 1-15.
- [26] P. Isola, J. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, USA, Jul. 2017, pp. 5967-5976. doi: [10.1109/CVPR.2017.632](https://doi.org/10.1109/CVPR.2017.632)
- [27] K. Xu et al., "Show, attend and tell: neural image caption generation with visual attention," in *32nd International Conference on Machine Learning*, Lille, France, Jul. 2015, pp. 2048-2057.
- [28] N. S. Keskar and R. Socher, "Improving generalization performance by switching from adam to SGD," 2017, [arXiv:1712.07628](https://arxiv.org/abs/1712.07628).
- [29] S. Merity, N. S. Keskar, and R. Socher, "Regularizing and optimizing LSTM language models," 2017, [arXiv:1708.02182](https://arxiv.org/abs/1708.02182).
- [30] A. C. Wilson, R. Roelofs, M. Stern, N. Srebro, and B. Recht, "The marginal value of adaptive gradient methods in machine learning," 2017, [arXiv:1705.08292](https://arxiv.org/abs/1705.08292).
- [31] J. Lee, T. Won, T. K. Lee, H. Lee, G. Gu, and K. Hong, "Compounding the performance improvements of assembled techniques in a convolutional neural network," 2020, [arXiv:2001.06268](https://arxiv.org/abs/2001.06268).
- [32] Y. Yamada, M. Iwamura, T. Akiba, and K. Kise, "Shakedrop regularization for deep residual learning," *IEEE Access*, vol. 7, pp. 186126-186136, 2019. doi: [10.1109/ACCESS.2019.2960566](https://doi.org/10.1109/ACCESS.2019.2960566)
- [33] J. Guo and S. Gould, "Depth dropout: efficient training of residual convolutional neural networks," in *International Conference on Digital Image Computing: Techniques and Applications*, Gold Coast, Australia, Dec. 2016, pp. 1-7. doi: [10.1109/DICTA.2016.7797032](https://doi.org/10.1109/DICTA.2016.7797032)
- [34] H. Wang, G. Wang, G. Li, and L. Lin, "CamDrop: a new explanation of dropout and a guided regularization method for deep neural networks," in *28th ACM International Conference on Information and Knowledge Management*, New York, USA, Nov. 2019, pp. 1141-1149. doi: [10.1145/3357384.3357999](https://doi.org/10.1145/3357384.3357999)
- [35] G. Ghiasi, T. Lin, and Q. V. Le, "DropBlock: a regularization method for convolutional networks," 2018, [arXiv:1810.12890](https://arxiv.org/abs/1810.12890).
- [36] M. Mezzini, "Empirical study on label smoothing in neural networks," in *International Conferences in Central Europe on Computer Graphics, Visualization and Computer Vision*, Prague, Czech Republic, Jun. 2018, pp. 200-205. doi: [10.24132/CSRN.2018.2802.25](https://doi.org/10.24132/CSRN.2018.2802.25)
- [37] M. Goibert and E. Dohmatob, "Adversarial robustness via label-smoothing," 2019, [arXiv:1906.11567](https://arxiv.org/abs/1906.11567).
- [38] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. B. Wojna, "Rethinking the inception architecture for computer vision," in *IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, USA, Jun. 2016, pp. 2818-2826. doi: [10.1109/CVPR.2016.308](https://doi.org/10.1109/CVPR.2016.308)
- [39] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, "CutMix: regularization strategy to train strong classifiers with localizable features," in *IEEE/CVF International Conference on Computer Vision*, Seoul, South Korea, Nov. 2019, pp. 6022-6031. doi: [10.1109/ICCV.2019.00612](https://doi.org/10.1109/ICCV.2019.00612)