

KLASIFIKASI LEVEL NON-PROLIFERATIF RETINOPATI DIABETIK DENGAN ENSEMBLE CONVOLUTIONAL NEURAL NETWORK

Ruvita Faurina¹, Endina Putri Purwandari², Mario Tiara Pratama³, Indra Agustian⁴

^{1,2} Teknik Informatika, Universitas Bengkulu, Jl. W.R Supratman Kandang Limun, Bengkulu, Indonesia

^{3,4} Teknik Elektro, Universitas Bengkulu, Jl. W.R Supratman Kandang Limun, Bengkulu, Indonesia

¹ruvita.faurina@unib.ac.id

²endinaputri@unib.ac.id

³mario.tiarapratama@gmail.com

⁴indraagustian@unib.ac.id

Abstrak: Penelitian ini mengusulkan penggunaan algoritma *CNN ensemble classifier* untuk klasifikasi level non-proliferatif Retinopati diabetik. Penelitian ini membandingkan metode *transfer learning feature-extraction* dan *fine-tuning* untuk memilih yang terbaik untuk digunakan. Pada lapisan klasifikasi, dibandingkan penggunaan lapisan *GAP* dan *Flatten* dengan menggunakan metode *dropout*, mode terbaik digunakan sebagai mode final klasifikasi. Arsitektur yang digunakan adalah DenseNet201, InceptionV3 dan MobileNetV2, Masing-masing model diuji dengan optimasi SGD dan ADAM. Keputusan prediksi diambil berdasarkan metode *average voting*. Hasil pengujian masing-masing arsitektur menunjukkan hasil terbaik adalah *fine tuning*, GAP, dan optimasi ADAM. Model final *fine-tuning* DenseNet201, InceptionV3 dan MobileNetV2 dapat mengklasifikasi level retinopati diabetik dengan akurasi pada data uji masing-masing 93%, 94% dan 89%. Sedangkan performa klasifikasi model *ensemble* untuk keseluruhan kelas memiliki akurasi terendah 95,6% dan *F1-Score* terendah 91.3%.

Kata Kunci: retinopati diabetik, *deep learning*, *convolutional neural network*, *ensemble classifier*, DenseNet201, InceptionV3, MobileNetV2.

Abstract: This research proposes the use of deep learning algorithm *CNN ensemble classifier* for the classification of non-proliferative diabetic retinopathy. This study compares transfer learning feature-extraction and fine-tuning methods to select the best one to use. The classification layer is compared using the GAP and Flatten layers with the dropout method, the best layer will be used in the classification system. The architecture used is DenseNet201, InceptionV3, and MobileNetV2. Each model is tested with SGD and ADAM optimization. The final prediction is decided by the average voting method. The test results for each architecture show that the best results are fine-tuning, GAP, and ADAM optimization. The final fine-tuning models DenseNet201, InceptionV3, and MobileNetV2 were able to classify the rate of diabetic retinopathy on the test data with accuracy at 93%, 94%, and 89%, respectively. Meanwhile, the performance of the ensemble model classification for the entire class has the lowest accuracy of 95.6% and the lowest F1-Score of 91.3%.

Keywords: diabetic retinopathy, deep learning, convolutional neural network, ensemble classifier, DensNet, InceptionV3, MobileNetV2.

I. PENDAHULUAN

Diabetic Retinopathy atau retinopati diabetik(RD) adalah komplikasi dari penyakit diabetes yang memicu terjadinya penyumbatan pembuluh darah pada daerah retina mata, dan dapat menyebabkan kebutaan. RD disebut sebagai penyebab utama kebutaan pada usia produktif di negara Barat [1]. Riset Kesehatan Dasar (Riskesdes) 2013 menemukan 6,9% penduduk Indonesia yang berusia diatas 15 tahun menderita Diabetes Mellitus(DM), dan Rumah Sakit Cipto Mangunkusumo mencatat persentase komplikasi kedua terbesar setelah neuropati adalah retinopati [2].

Metode konvensional yang digunakan untuk mendiagnosa level retinopati diabetik adalah dengan melakukan pemeriksaan langsung ke dalam bola mata pasien oleh dokter ahli dengan

menggunakan oftalmoskop atau melalui pengamatan terhadap hasil rekaman citra digital dari kamera fundus [3]. Pemeriksaan manual oleh dokter dapat memakan waktu dan hasil pemeriksaan tergantung pada tingkat keahlian dokter yang melakukan pemeriksaan. Kesalahan diagnosa pada pasien RD beresiko memperparah komplikasi. Data penelitian menyebutkan bahwa 90 % dari meningkatnya keparahan RD dapat dicegah dengan pemeriksaan dan perawatan pada pasien yang tepat [4]. Oleh karena itu, diperlukan sebuah sistem yang dapat mendiagnosis level retinopati diabetik secara tepat dan cepat serta dengan biaya yang murah dan juga dapat digunakan untuk deteksi awal yang dapat menggantikan ketidakhadiran dokter atau pakar.

Banyak penelitian yang telah dilakukan untuk membuat model sistem deteksi otomatis level keparahan RD. Metode yang banyak dilakukan sebelumnya adalah dengan melakukan ekstraksi fitur atau informasi yang berhubungan dengan level RD pada citra fundus [5], fitur atau informasi tersebut dimasukan ke algoritma klasifikasi tertentu seperti *random forest*[6], *support vector machine*[7], dan *AdaBoost classifier*[8].

Berbeda dengan algoritma klasifikasi yang memerlukan ekstraksi fitur di awal, *Convolutional Neural Network* (CNN) merupakan sebuah teknik *deep learning* yang dapat melakukan ekstraksi fitur pada data citra, dengan model *deep learning*, umumnya CNN lebih baik dalam merepresentasikan data citra [9]–[11]. Proses pelatihan model CNN membutuhkan data pelatihan yang besar, umumnya antara puluhan ribu hingga jutaan data. Memaksakan melakukan pelatihan model CNN dengan layer yang dalam dengan data yang kecil akan menghasilkan sistem klasifikasi yang tidak maksimal. Sementara itu, data citra

fundus yang tersedia masih terbatas. Solusi dari masalah tersebut dengan metode *transfer learning* yang memungkinkan untuk melakukan pelatihan model *deep learning* dengan memanfaatkan bobot awal model *pre-train* yang sudah ada [12].

Penelitian ini mengusulkan pendekatan *ensemble classifier* dari tiga arsitektur CNN, yaitu arsitektur DensNet, InceptionV3 dan MobileNetV2 dengan pengambilan keputusan prediksi berdasarkan metode *average voting*. Pemilihan arsitektur berdasarkan penelitian[13] yang mereview performa berbagai jenis arsitektur CNN untuk klasifikasi retinopati diabetik.

II. PENELITIAN TERKAIT

Model komputasi untuk klasifikasi level RD sebenarnya telah sejak lama dikembangkan oleh para peneliti, bahkan sebelum berkembangnya algoritma *convolutional neural network*. Para peneliti terdahulu umumnya melakukan rekayasa tingkat keabuan piksel pada suatu gambar untuk mengekstraksi fitur atau informasi yang berhubungan dengan level RD pada gambar fundus. Fitur yang telah diekstraksi tersebut kemudian dimasukan ke algoritma klasifikasi tertentu seperti *random forest*, *support vector machine*, dan *AdaBoost classifier*. Metode ini dinilai masih kurang efektif, dan tingkat akurasi yang tidak konsisten, karena sifat dari ekstraksi ciri yang sangat dipengaruhi oleh kualitas warna dan kontras gambar.

Convolutional Neural Network secara implisit memungkinkan melakukan ekstraksi fitur dan klasifikasi, yang sekaligus menjawab kesulitan dalam pembuatan model klasifikasi level RD. Secara umum pekerjaan dalam bidang ini yang memanfaatkan jaringan *Convolutional neural*

network terbagi menjadi kelompok klasifikasi biner dan klasifikasi multi kelas.

Pada kelompok klasifikasi biner, penelitian [14] merancang model *convolutional neural network* untuk mendeteksi citra fundus dengan RD. Mereka menggunakan dataset pribadi yang terdiri dari 800 data pelatihan dan 200 data validasi dalam penelitian tersebut. Hasil evaluasi model mereka menunjukkan 94.5% dari data validasi dapat diprediksi dengan benar yang juga menunjukkan bahwa model mereka lebih unggul dari model komputasi tradisional. Penelitian [15] mengusulkan model jaringan CNN untuk mendeteksi RD. Model tersebut dilatih dengan *open dataset* kaggle dan divalidasi dengan data DiaretdB1. Berdasarkan klasifikasi biner yang mereka lakukan, model tersebut menghasilkan sensitivitas 93.6%.

Sementara, pada klasifikasi multi kelas, penelitian [16] mengusulkan *rancangan* model *convolutional neural network* untuk mendeteksi dan mengklasifikasi level keparahan RD. Penelitian ini menggunakan 78.000 gambar sebagai data pelatihan dan 5000 gambar sebagai data validasi yang diambil dari *open dataset Kaggle*. Hasil evaluasi model ini menghasilkan akurasi 75% pada data validasi. Penelitian [17] mengajukan model CNN dengan memodifikasi arsitektur Xception. Penelitian ini menggunakan dataset dari kompetisi Kaggle APTOS 2019. Hasil pengujian menunjukkan model usulan mereka menghasilkan akurasi 83.09 % dan lebih unggul dibandingkan dengan model dengan arsitektur ResNet59, MobileNet, InceptionV3 dan Xception original. [17] mengajukan pendekatan *transfer learning* sebagai solusi dalam mengatasi keterbatasan dan ketidak seimbangan data kaggle yang mereka gunakan pada saat pelatihan jaringan CNN. Hasil pengujian mereka menunjukkan jaringan dengan

arsitektur Inception ResNet V2 menghasilkan skor kappa paling tinggi 0.76.

III. METODOLOGI

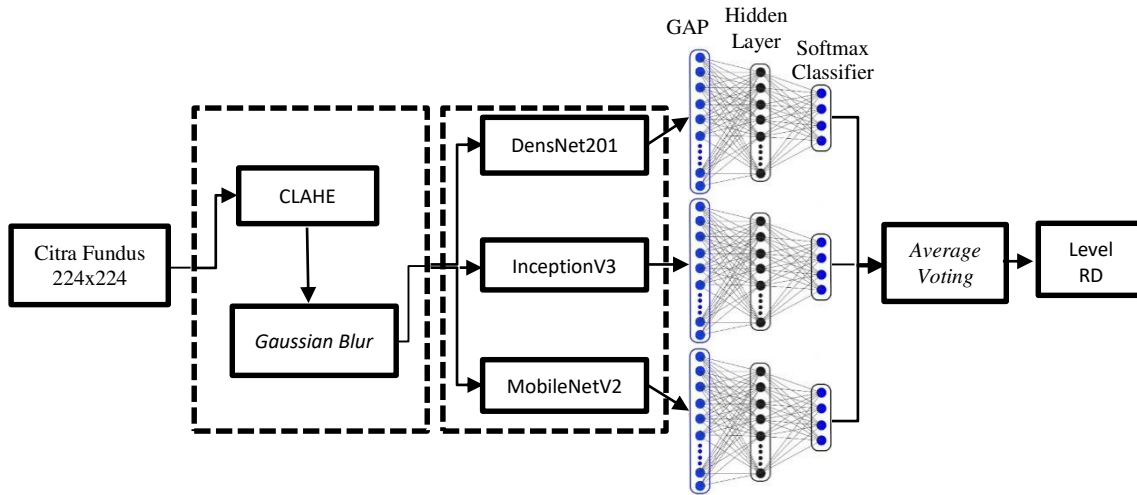
Dataset awal dari penelitian ini terdiri dari 1460 citra fundus yang dipilih secara acak dari data APTOS *competition* 2019[18]. Masing-masing kelas memiliki 365 data. Label data telah disediakan oleh APTOS dalam format csv. Dari data ini jumlah data pada masing-masing level RD tidak seimbang, sehingga dilakukan augmentasi data untuk menyeimbangkannya. Augmentasi yang dilakukan juga memperbesar dataset dan memvariasikan citra baru berdasarkan citra dataset awal. Augmentasi pra-pelatihan terdiri dari *cropping* (memotong citra menjadi beberapa bagian), *shearing*(menggeser sisi citra sehingga citra membentuk jajaran genjang). Sedangkan augmentasi pada saat proses pelatihan terdiri dari rotasi, flip horizontal dan flip vertikal.

Sebelum masuk ke tahapan *ensemble*, citra fundus terlebih dahulu dilakukan pra-proses normalisasi kontras dengan metode *Contrast Limited Adaptive Histogram Equalization* (CLAHE)[19] dan dilanjutkan dengan konversi ke *grayscale* dan *Gaussian Blur*[20]. Citra fundus dari dataset, dikumpulkan dari kamera fundus yang berbeda, sehingga memiliki tingkat kontras yang berbeda, CLAHE berfungsi untuk menyeragamkan tingkat kontras citra fundus tersebut. *Gaussian Blur* berfungsi untuk mereduksi *noise* pada citra fundus. Keluaran *Gaussian Blur* merupakan input ensemble CNN.

Model *ensemble* yang diusulkan terdiri dari tiga model yaitu DensNet201, InceptionV3 dan MobileNetV2. Output masing-masing model berupa probabilitas level RD, kemudian diputuskan prediksi akhir dengan menggunakan metode

average voting. Blok diagram sistem ditunjukkan pada gambar 1.

Vektor yang dihasilkan oleh GAP akan masuk ke *hidden layer* jaringan *neural network*. Pada



Gambar 1. Skema jaringan *Ensemble Deep Learning* yang diusulkan.

Penggunaan tiga arsitektur pada model *ensemble convolutional neural network* bertujuan agar model dapat mengekstraksi fitur RD yang lebih beragam pada gambar fundus. *Ensemble convolutional neural network* ini terdiri dari 2 jaringan utama yaitu jaringan konvolusi dan jaringan *neural network*.

Jaringan konvolusi atau *convolutional layer* merupakan kumpulan sejumlah filter yang pada dasarnya merupakan matriks dengan ukuran dan pola tertentu yang disebut juga dengan bobot. Setiap filter dikalikan dengan matriks gambar untuk menghasilkan fitur yang berguna pada proses klasifikasi oleh jaringan *neural network*.

Sebelum masuk ke jaringan *neural network*, *feature map* yang dihasilkan oleh jaringan konvolusi yang masih dalam dimensi matriks ($h \times w \times d$) akan diubah menjadi dimensi vektor ($1 \times 1 \times d$) oleh lapisan *Global Average Pooling* (GAP). Selain berfungsi untuk mengubah format data *feature map*, penggunaan GAP dimaksudkan untuk mereduksi jumlah parameter yang akan masuk ke jaringan *neural network* yang berguna untuk meminimalkan resiko *overfitting*.

Algoritma yang diusulkan

Input: Fundus Images (X, Y); where $Y = \{y/y \in \{\text{Normal, Mild, Moderate, Severe, PDR}\}\}$

Output : The trained model that classifies the fundus image $x \in X$

Pra-proses:

- Normalisasi dengan skala diameter yang sama menjadi 600x600
- Normalisasi Contrast dengan Contrast Limited Adaptive Histogram Equalization (CLAHE)
- Grayscale dan Gaussian Blur
- Augmentasi : Cropping dan Shearing gambar secara acak.
- Resize gambar ke 224 x 224
- Kalkulasi α awal

Impor pre-train model $H = \{\text{Densnet201, InceptionV3, MobileNetV2}\}$.

Tukar layer FCN terakhir tiap pre-train model dengan jaringan NN dengan 1 GAP, 1 Hidden Layer dan 1 Softmax classifier dengan dimensi (4×1).

untuk Setiap $\forall h \in H$ lakukan:

$\alpha = 0.0001$

untuk $= 1$ to 50 do

untuk setiap mini batch (X_i, Y_i) $\in (X_{\text{train}}, Y_{\text{train}})$ lakukan:

update parameter pada model $h(\cdot)$ menggunakan optimizer.

jika error validasi tidak mengecil setelah 3 epoch **maka:**

$\alpha = \alpha \times 0.5$

jika error validasi tidak mengecil setelah 5 epoch **maka:** pelatihan berhenti.

untuk setiap $x \in X_{\text{test}}$ lakukan:

Ensemble: Hitung rata-rata probabilitas model terhadap masing-masing kelas.

Diagnosis level RD= Probabilitas tertinggi dari *average voting*.

selesai

lapisan *neural network*, sebelum dan setelah *hidden layer* disisipkan fungsi *dropout* yang bertujuan

untuk mendeaktivasi neuron pada jaringan *neural network* secara acak berdasarkan bobot pada masing-masing neuron selama proses pelatihan. Pada lapisan terakhir jaringan *neural network*, dipasang fungsi *softmax* yang bertugas untuk menghitung probabilitas masing-masing neuron terhadap masing-masing kelas (level RD).

Penggunaan layer *average voting* mewakili proses *ensemble* pada penelitian ini. Probabilitas dari 3 fungsi softmax yang dimiliki oleh masing-masing model *convolutional neural network* akan diambil nilai probabilitas rata-rata. Nilai tersebutlah yang akan menjadi probabilitas kelas klasifikasi level RD.

Metode Pelatihan

Proses pelatihan pada penelitian ini menggunakan metode *transfer learning*[22]. *Transfer learning* pada penelitian ini menggunakan pendekatan feature extraction dan *Fine-tuning*. Pada tahapan pelatihan *feature extraction*, pembaharuan bobot hanya dilakukan pada jaringan *neural network* sementara jaringan konvolusi menggunakan bobot dari ImageNet. Selanjutnya, pada pendekatan *Fine-tuning* semua jaringan akan dilatih. Bobot awal model adalah bobot dari pelatihan *feature extraction*. Pengujian dilakukan terhadap beberapa variasi jumlah neuron dan *dropout*, hingga menemukan variasi jumlah neuron dan rasio terbaik pada masing-masing model.

Learning rate merupakan *hyperparameter* yang berpengaruh untuk terciptanya model optimal pada proses pelatihan jaringan *convolutional neural network* yang berbasis *backpropagation*. Penelitian ini melakukan tuning *learning rate* model pada *optimizer* SGD[23] dan Adam[24]. Variasi nilai *learning* dan jenis *optimizer* dengan akurasi tertinggi dan *loss* paling minimum akan diambil

sebagai model optimal dan akan digunakan pada *ensemble* model.

Metode Pengujian

Model pada penelitian ini dievaluasi dengan menghitung akurasi model terhadap keseluruhan data uji dengan menggunakan *confusion matrix*[21]. Selanjutnya, evaluasi lanjutan dilakukan dengan menghitung tingkat *Accuracy*, *precision*, *recall* dan *f1-score* untuk tiap kelas (level RD), masing-masing dihitung dengan menggunakan persamaan (1)(2)(3) dan(4).

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall (R) = \frac{TP}{TP+FN} \quad (3)$$

$$F1 = \frac{2RP}{R+P} \quad (4)$$

IV. HASIL DAN PEMBAHASAN

Total dataset setelah augmentasi pra-pelatihan yang digunakan pada penelitian ini adalah 8000 citra fundus ukuran 224x224 piksel², dibagi menjadi 6000 citra untuk pelatihan, 1500 untuk validasi dan 500 untuk pengujian. Distribusi dataset dan level RD ditunjukkan pada tabel 1. Citra fundus hasil setelah dinormalisasi ukuran, dilanjutkan dengan normalisasi kontras dan *Gaussian Blur*, seperti ditunjukkan pada gambar 2.

Tabel 1. Distribusi dataset

| Level RD | Pelatihan | Validasi | Testing |
|---------------|-------------|-------------|------------|
| 0-Normal | 1500 | 375 | 125 |
| 1-Mild | 1500 | 375 | 125 |
| 2-Moderate | 1500 | 375 | 125 |
| 3-Severe | 1500 | 375 | 125 |
| Jumlah | 6000 | 1500 | 500 |

Penelitian ini menggunakan dua model *transfer learning* yang diujikan, yang pertama adalah model ekstraksi fitur dan yang kedua adalah model *fine-tuning*. Pada model pertama, model jaringan pre-train hanya berfungsi sebagai ekstraktor fitur, tidak

ada modifikasi atau update bobot pada model pre-train, output pre-trained model dimasukkan ke dalam jaringan klasifikasi dengan mode *dropout regulation*. Pada model kedua, dilakukan perbaikan bobot pada jaringan konvolusi model pre-train dan mengganti lapisan terakhir dengan jaringan klasifikasi dengan *dropout regulation*.



Gambar 2. Pra-proses olah dataset citra fundus

Pada tahap *transfer learning* mode ekstraksi fitur, pada lapisan klasifikasi layer pertama dilakukan pengujian dua jenis layer yaitu, *Global Average Pooling*(GAP) dan *Flatten*, dengan jumlah neuron 2048; 1024; 512; 256 dan 128 serta rasio *dropout* 0.5; 0.25 dan 0.15. Pada arsitektur Densnet201, lapisan GAP memiliki kinerja yang lebih baik dibanding *flatten*. Akurasi terbaik model DensNet201 adalah 73% pada pelatihan dan 69% pada validasi dengan struktur dua *hidden layer* dengan 512 neuron dan rasio *dropout* 0.25. GAP dapat meminimalkan *overfitting* pada model, hal ini terlihat pada kecilnya selisih antara akurasi pelatihan dan akurasi validasi. Pada arsitektur InceptionV3, lapisan GAP juga memiliki kinerja yang lebih baik dengan akurasi terbaik pada pelatihan 56% dan validasi 58% dengan struktur 2 *hidden layer* dengan 256 neuron dan rasio *dropout* 0.15. pada arsitektur MobileNetV2, GAP juga menunjukkan kinerja yang lebih baik dengan akurasi terbaik pada pelatihan 64% dan validasi 61% dengan struktur 2 *hidden layer* dengan 1024 neuron dan rasio *dropout* 0.25.

Berdasarkan hasil uji pada tahap pertama, dipilih lapisan GAP sebagai lapisan pertama

jaringan klasifikasi untuk tahap pengujian *transfer learning* mode *fine-tuning*. Pada tahap *fine-tuning* Jumlah neuron yang digunakan adalah 2048; 1024; 512; 256 dan 128 serta rasio *dropout* 0.5; 0.25 dan 0.15. Hasil pengujian pada arsitektur Densnet201 menunjukkan kinerja terbaik dengan *hidden layer* yang memiliki 512 neuron dengan rasio *dropout* 0.15 dalam mengklasifikasi level RD, akurasi validasi yang mencapai 80% dan loss yang rendah 0.48. Dengan fungsi *early stopping*, karena akurasi model sudah tidak mengalami peningkatan signifikan dari epoch ke 19, proses pelatihan berhenti pada epoch ke 24 dari target 100 epoch. *Fine-tuning* pada arsitektur InceptionV3 memberikan hasil terbaik dengan satu *hidden layer* yang memiliki 1024 neuron dengan rasio *dropout* 0.25, hal ini terlihat pada akurasi validasi yang mencapai 81% dan *loss* 0.44. arsitektur InceptionV3 terhenti oleh *early stopping* pada epoch 25. Kurva konvergensi pelatihan InceptionV3 menunjukkan model sudah cukup konvergen yang ditandai dengan eror minimal yang sudah stabil. Sedangkan pada arsitektur MobileNetV2, hasil terbaik dengan satu *hidden layer* yang memiliki 512 neuron dengan rasio *dropout* 0.5. Akurasi terbaik model ini adalah 0.74 % dengan *loss* 0.7. MobileNetV2 terhenti oleh *early stopping* pada epoch 18. Akurasi validasi model masih di bawah 80%.

Berdasarkan hasil pengujian tahap kedua, tahapan selanjutnya adalah optimasi *learning rate*. Dari tahap kedua diperoleh jumlah neuron dan rasio *dropout* optimal untuk masing-masing model. Proses optimasi *learning rate* dilakukan untuk memperbaiki akurasi dan konvergensi model. Metode optimasi yang diuji pada penelitian ini adalah, *Stochastic Gradient Descent*(SGD) dan *Adaptive Moment Estimation*(Adam). Optimasi

learning rate, dilakukan dengan dua pendekatan, pertama dengan memberikan nilai *learning rate* tetap selama proses pelatihan, dengan variasi *learning rate* 0.1; 0.01; 0.001; 0.0001; 0.00001. Pendekatan kedua dengan metode *automatic reduce learning rate*, dengan cara memberikan nilai *learning rate* awal sebesar 0.0001 kemudian sistem akan memperkecil nilai *learning rates* sebesar *learning rate* dikalikan 0.5 ($\alpha_{\text{baru}} = \alpha - 0.5$). α jika akurasi validasi model tidak membaik setelah 3 *epoch*.

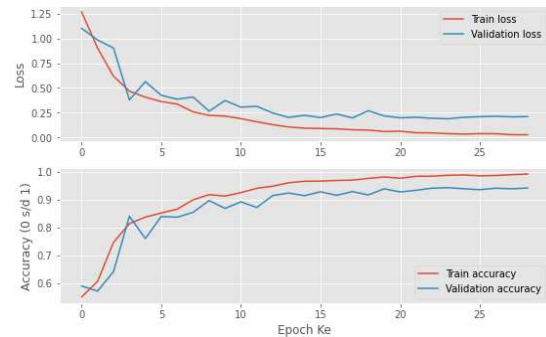
Tabel 2. Optimasi *Learning-rate*

| Optimizer | (α) | Akurasi (0 s/d 1) | | |
|-----------|--------------|-------------------|-------------|-------------|
| | | D201 | MV2 | InV3 |
| SGD | 0.1 | 0.72 | 0.66 | 0.81 |
| | 0.01 | 0.83 | 0.79 | 0.89 |
| | 0.001 | 0.87 | 0.75 | 0.87 |
| | 0.0001 | 0.8 | 0.77 | 0.79 |
| | 0.00001 | 0.79 | 0.74 | 0.5 |
| | Auto * | 0.8 | 0.75 | 0.76 |
| Adam | 0.1 | 0.25 | 0.24 | 0.42 |
| | 0.01 | 0.46 | 0.59 | 0.47 |
| | 0.001 | 0.68 | 0.74 | 0.81 |
| | 0.0001 | 0.85 | 0.85 | 0.92 |
| | 0.00001 | 0.92 | 0.81 | 0.89 |
| | Auto * | 0.94 | 0.90 | 0.93 |

D201: DensNet201, MV2: MobileNetV2, InV3: InceptionV3
SGD *: DensNet201 = 2.5e-5; MobileNetV2=2.5e-5; InceptionV3= 5e-5.
Adam *: DensNet201 = 1.25e-5; MobileNetV2=2.5e-5; InceptionV3= 1.25e-5

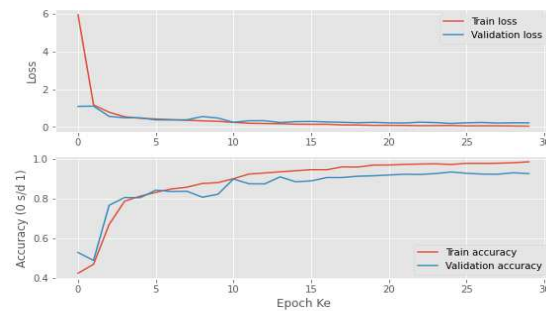
Tabel 2 menunjukkan hasil pengujian optimasi, akurasi terbaik model dengan arsitektur DensNet201 adalah 94 % dengan *learning rate* 1.125e-5 menggunakan Adam optimizer, akurasi terbaik arsitektur InceptionV3 93% dengan *learning rate* 1.25e-5 dengan Adam optimizer sementara akurasi terbaik MobileNetV2 saat *learning rate* 2.5e-5 dengan Adam optimizer adalah 90%. Kualitas model juga diperlihatkan dengan kurva konvergensi hasil pelatihan masing-masing arsitektur pada gambar 3, 4 dan 5. Kurva konvergensi mengindikasikan bahwa masing-masing model sudah berhasil konvergen ke titik

error minimal. Akurasi validasi model juga telah konvergen dan stabil. Sehingga model tersebut dijadikan sebagai model optimal yang kemudian

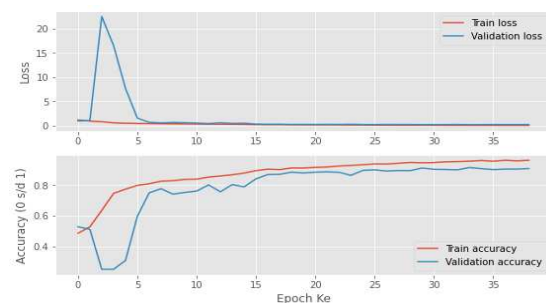


Gambar 3. Kurva konvergensi pelatihan DensNet201

akan dijadikan basis model pada algoritma *ensemble* yang diusulkan.



Gambar 4. Kurva konvergensi pelatihan InceptionV3

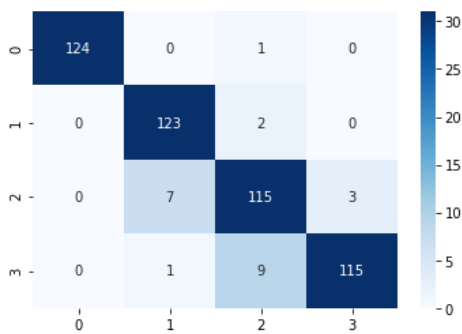


Gambar 5. Kurva konvergensi pelatihan MobileNetV2

Total 500 data citra fundus diujikan, diperoleh *Confusion matrix* dari model *ensemble* pada gambar 6. Pada klasifikasi multi-kelas, keberhasilan klasifikasi sesuai aktual adalah *True Positive* dan *True Negative* sebagai keberhasilan klasifikasi untuk data bukan kelas aktual, baik prediksi pada kelas yang lain benar atau salah, diperoleh rekapitulasi parameter-parameter *confusion matrix*

untuk mengukur performa model pada masing-masing kelas ditunjukkan pada tabel 3. Dengan menggunakan indikator pada persamaan (1)-(4) diperoleh data kinerja untuk kelas 0(Normal),1(*Mild*), 2(*Moderate*), dan 3(*Severe*) pada tabel 4.

TP adalah *True Positive*, yaitu jumlah data positif yang diklasifikasi dengan benar oleh sistem. TN adalah *True Negative*, yaitu jumlah data negatif yang diklasifikasi dengan benar oleh sistem. FN adalah *False Negative*, yaitu jumlah data negatif namun diklasifikasi salah oleh sistem. FP adalah *False Positive*, yaitu jumlah data positif namun diklasifikasi salah oleh sistem.



Gambar 6. *Confusion Matrix* final model ensemble

Tabel 3. Total parameter *Confusion Matrix*

| Kelas | TP | TN | FP | FN |
|-------|-----|-----|----|----|
| 0 | 124 | 375 | 0 | 1 |
| 1 | 123 | 367 | 8 | 2 |
| 2 | 115 | 363 | 12 | 10 |
| 3 | 115 | 372 | 3 | 10 |

Tabel 4. Parameter performa model ensemble

| Kelas | Accuracy | Precision | Recall | F1 |
|-------|----------|-----------|--------|-------|
| 0 | 0,998 | 1,000 | 0,992 | 0,996 |
| 1 | 0,980 | 0,939 | 0,984 | 0,961 |
| 2 | 0,956 | 0,906 | 0,920 | 0,913 |
| 3 | 0,974 | 0,975 | 0,920 | 0,947 |

Berdasarkan tabel 4, performa model memprediksi kelas untuk aktual kelas sesungguhnya memiliki akurasi di atas 95%, rata-rata 97,7%, dengan catatan bahwa kesalahan prediksi ke kelas lain adalah *True Negative*. Performa *Precision* diatas 90%, rata-rata 95,5% mengindikasikan rasio prediksi data positif

terhadap jumlah data positif ditambah data negatif yang terdeteksi positif. Performa *Recall* di atas 92%, rata-rata 95,54%, mengindikasikan rasio prediksi data positif terhadap data positif ditambah data positif terdeteksi negatif. *F1-Score* yang merupakan rata-rata harmonik dari *precision* dan *recall*, di atas 91%, dengan rata-rata 95,4%. *F1-Score* mengindikasikan bahwa model memiliki *precision* dan *recall* yang baik.

V. KESIMPULAN

Pada penelitian ini diusulkan algoritma *ensemble CNN* untuk klasifikasi Retinopati Diabetik Non-proliferative dengan pra-proses CLAHE dan *Gaussian Blur*, *transfer learning fine-tuning* arsitektur DensNet, InceptionV3 dan MobileNetV32, lapisan klasifikasi terdiri dari GAP, *dropout hidden layer* dan *softmax classifier*, dan dioptimasi menggunakan algoritma ADAM. Model final fine-tuning DensNet, InceptionV3 dan MobileNetV32 dapat mengklasifikasi level retinopati diabetik dengan akurasi pada data uji masing-masing 93%, 94% dan 89%. Sedangkan performa klasifikasi model *ensemble* untuk masing-masing kelas memiliki akurasi terendah 95,6% dan *F1-Score* terendah 91.3%. Untuk penelitian selanjutnya model *ensemble* yang diusulkan perlu diuji untuk dataset baru.

UCAPAN TERIMA KASIH

Tim Peneliti mengucapkan terima kasih kepada Universitas Bengkulu yang telah memberikan hibah melalui skema Penelitian Unggulan Fakultas di Fakultas Teknik Universitas Bengkulu Tahun Anggaran 2020 dengan Nomor Kontrak: 2098/UN30.13/HK/2020.

REFERENSI

- [1] Erlvira and E. E. Suryawijaya, "Retinopati Diabetes," *Cermin Dunia Kedokt.*, vol. 46, no. 3, pp. 220–224, 2019.
- [2] Kementerian Kesehatan RI, "WASPADA DIABETES, Eat well Live well," *Pusat Data dan Informasi*, 2014.
- [3] M. S. Sari, Ratna, Rasmala Dewi, "Pola Retinopati Diabetik Pada Pasien Diabetes Mellitus Rawat Jalan Di RSUD Raden Mattaher Jambi," *J. Healthc. Technol. Med.*, vol. 5, no. 2, pp. 287–296, 2019.
- [4] P. Z. Z. Tapp, Robyn J., Jonathan E. Shaw, C. Alex Harper, Maximilian P. De Courten, Beverley Balkau, Daniel J. McCarty, Hugh R. Taylor, Timothy A. Welborn, "The prevalence of and factors associated with diabetic retinopathy in the Australian population," *Diabetes Care*, vol. 26, no. 6, pp. 1731–1737, 2003.
- [5] Z. Gao, J. Li, J. Guo, Y. Chen, Z. Yi, and J. Zhong, "Diagnosis of Diabetic Retinopathy Using Deep Neural Networks," *IEEE Access*, vol. 7, pp. 3360–3370, 2019.
- [6] R. Casanova, S. Saldana, E. Y. Chew, R. P. Danis, C. M. Greven, and W. T. Ambrosius, "Application of random forests methods to diabetic retinopathy classification analyses," *PLoS One*, vol. 9, no. 6, pp. 1–8, 2014.
- [7] M. Faisal, D. Wahono, I. K. E. Purnama, M. Hariadi, and M. H. Purnomo, "Classification of diabetic retinopathy patients using support vector machines (SVM) based on digital retinal image," *J. Theor. Appl. Inf. Technol.*, vol. 59, no. 1, pp. 197–204, 2014.
- [8] A. Dhakal and S. Shakya, "Detection and Classification of Diabetic Retinopathy using Adaptive Boosting and Artificial Neural Network," no. August, 2019.
- [9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Adv. Neural Inf. Process. Syst.*, vol. 2, pp. 1097–1105, 2012.
- [10] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [11] Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, 2013.
- [12] H. C. Shin *et al.*, "Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning," *IEEE Trans. Med. Imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.
- [13] I. Kandel and M. Castelli, "Transfer learning with convolutional neural networks for diabetic retinopathy image classification. A review," *Appl. Sci.*, vol. 10, no. 6, 2020.
- [14] Xu K, Feng D, Mi H. Deep convolutional neural network-based early automated detection of diabetic retinopathy using fundus image. *Molecules*. 2017 Dec;22(12):2054.
- [15] W. M. Gondal, J. M. Köhler, R. Grzeszick, G. A. Fink, and M. Hirsch, "Weakly-supervised localization of diabetic retinopathy lesions in retinal fundus images," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 2069–2073.
- [16] H. Pratt, F. Coenen, D. M. Broadbent, S. P. Harding, and Y. Zheng, "Convolutional neural networks for diabetic retinopathy," *Procedia Comput. Sci.*, vol. 90, pp. 200–205, Jan. 2016.
- [17] Krishnan AS, Bhat V, Ramteke PB, Koolagudi SG. "A Transfer Learning Approach for Diabetic Retinopathy Classification Using Deep Convolutional Neural Networks". In 2018 15th IEEE India Council International Conference (INDICON) 2018 Dec 16 (pp. 1-6). IEEE.
- [18] "APTOS", 2019, [online] Available: <https://www.kaggle.com/c/aptos2019-blindness-detection/overview/aptos-2019>
- [19] Pizer SM, Amburn EP, Austin JD, Cromartie R, Geselowitz A, Greer T, ter Haar Romeny B, Zimmerman JB, Zuiderveld K. "Adaptive histogram equalization and its variations". *Computer vision, graphics, and image processing*. 1987 Sep 1;39(3):355-68.
- [20] Haddad RA, Akansu AN. "A class of fast Gaussian binomial filters for speech and image processing". *IEEE Transactions on Signal Processing*. 1991 Mar 1;39(3):723-7.
- [21] Townsend JT. "Theoretical analysis of an alphabetic confusion matrix". *Perception & Psychophysics*. 1971 Jan 1;9(1):40-50.
- [22] Shin HC, Roth HR, Gao M, Lu L, Xu Z, Nogues I, Yao J, Mollura D, Summers RM. "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer

- learning*". IEEE transactions on medical imaging. 2016 Feb 11;35(5):1285-98.
- [23] Bottou L. "*Stochastic gradient descent tricks*". In Neural networks: Tricks of the trade 2012 (pp. 421-436). Springer, Berlin, Heidelberg.
- [24] Zhang Z. "*Improved adam optimizer for deep neural networks*". In 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS) 2018 Jun 4 (pp. 1-2). IEEE