

Pengembangan Aplikasi berbasis Android untuk Mengenali Jenis Lesi Kulit Menggunakan *Convolutional Neural Network*

Development of an Android-based Application to Recognize Types of Skin Lesions Using Convolutional Neural Network

Blessynta Christesa Sengkey¹, Debby Paseru^{*2}, dan Steven Pandelaki³

^{1,2,3}Program Studi Teknik Informatika; Fakultas Teknik, Universitas Katolik De La Salle

^{1,2,3}Kairagi 1 Kombos, Manado, Indonesia

dpaseru@unikadelasalle.ac.id

Diterima : 22 Maret 2024 || Direvisi : 11 April 2024 || Disetujui: 30 April 2024

Abstrak – Lesi kulit merupakan kelainan atau gangguan kulit berupa perubahan, kerusakan, pertumbuhan yang abnormal terhadap kulit, seperti perubahan tekstur, warna, munculnya benjolan dan bintik pada kulit. Penyakit ini tentu mengganggu aktivitas dan perilaku orang setiap hari karena reaksi yang ditimbulkan, seperti sensasi gatal, nyeri, perih dan panas yang berlebihan. Akan tetapi pengetahuan akan jenis-jenis lesi kulit oleh masyarakat umum masih kurang dan diperlukan sebuah sistem yang mampu memberikan informasi terkait lesi kulit primer. Pengolahan citra sebagai bagian dari *machine learning* dapat mengenali jenis-jenis lesi kulit primer melalui aplikasi yang menggunakan *Convolutional Neural Network* (CNN). Metode ini mampu melakukan ekstraksi fitur dan klasifikasi yang baik sehingga sangat cocok dimanfaatkan untuk pendeteksian gambar. Penelitian dilakukan terhadap 4 kelas lesi, yakni makula, urtikaria, papula dan vesikular. Berdasarkan hasil pengujian dengan model CNN didapati nilai rata-rata akurasi, yaitu sebesar 95% dengan perhitungan nilai pada kelas makula dengan *precision* 91%, *recall* 100%, *f-1 score* 95%, kelas urtikaria dengan *precision* 100%, *recall* 91%, *f-1 score* 95%, kelas papula dengan *precision* 98%, *recall* 93%, *f-1 score* 96% dan kelas vesikular dengan *precision* 93%, *recall* 99%, *f-1 score* 96%.

Kata Kunci: lesi kulit, *convolutional neural network*, aplikasi, Android, klasifikasi

Abstract – Skin lesions are skin abnormalities or disorders in the form of changes, damage, abnormal growth of the skin, such as changes in texture, color, appearance of lumps and spots on the skin. This disease certainly disrupts people's activities and behavior every day because of the reactions it causes, such as sensations of itching, pain, stinging and excessive heat. However, knowledge of the types of skin lesions by the lay public is still lacking and a system is needed that can provide information regarding primary skin lesions. Image processing as part of machine learning can recognize types of primary skin lesions through applications that use *Convolutional Neural Network* (CNN). This method can perform good feature extraction and classification, so it is very suitable for image detection. Research was carried out on 4 classes of lesions, namely macular, urticarial, papular and vesicular. Based on the test results with the CNN model, it was found that the average accuracy value was 95% with the calculation of values in the macular class with *precision* 91%, *recall* 100%, *f-1 score* 95%, urticaria class with *precision* 100%, *recall* 91%, *f-1 score* 95%, papule class with *precision* 98%, *recall* 93%, *f-1 score* 96% and vesicular class with *precision* 93%, *recall* 99%, *f-1 score* 96%.

Keywords: skin lesions, *convolutional neural network*, application, Android, Classification

PENDAHULUAN

Lesi kulit merupakan kelainan atau gangguan kulit berupa perubahan, kerusakan, pertumbuhan yang abnormal terhadap kulit, seperti perubahan tekstur, warna, munculnya benjolan dan bintik pada kulit. Adapun lesi kulit ini secara medis dibedakan menjadi lesi primer dan sekunder. Lesi primer merupakan kondisi perubahan pada kulit yang didapati sejak lahir atau didapati sepanjang hidup, sedangkan lesi sekunder

adalah perkembangan gangguan lesi primer yang terjadi karena iritasi, infeksi, luka, dan pemicu lainnya. Kelainan ini memicu perubahan terhadap tampilan, warna, tekstur dan ketebalan pada kulit akibat pertumbuhan jaringan abnormal di permukaan kulit. Perubahan terhadap permukaan kulit muncul dalam bentuk bercak, luka, benjolan dan warna (Anum, et al., 2021).

Beberapa faktor penyebab kelainan kulit ini, seperti lingkungan, obat-obatan, makanan, ataupun

gejala suatu penyakit. Penyakit ini tentu mengganggu aktivitas dan perilaku orang setiap hari karena reaksi yang ditimbulkan, seperti sensasi gatal, nyeri, perih dan panas yang berlebihan. Akan tetapi pengetahuan akan jenis-jenis lesi kulit oleh masyarakat awam masih kurang dan diperlukan diagnosis dini atau pengenalan untuk melakukan pencegahan lewat konsultasi dokter dan mengonsumsi obat-obatan yang tepat (Anum, et al., 2021).

Penelitian serupa telah dilakukan terhadap klasifikasi lesi kulit menggunakan metode *transfer learning* dan augmentasi dengan salah satu arsitektur *Convolutional Neural Network*, yakni *AlexNet*. Penelitian tersebut membahas lesi kulit yang disebabkan penyakit kanker kulit, yakni melanoma dan nevus (Hosny, Kassem, & Foad, 2019); (Cullell-Dalmau, Noe, Otero-Vinas, Meic, & Manzo, 2021; Warsito). Hasil klasifikasi yang dilakukan memperoleh hasil akurasi yang tinggi sehingga sistem yang dibangun dapat dikatakan mampu melakukan klasifikasi dari kasus yang diangkat. Berbeda dengan penelitian yang akan dilakukan ini yang lebih menjelaskan dasar dari kelainan lesi kulit, dalam hal ini jenis lesi kulit primer menggunakan algoritma CNN.

Convolutional Neural Network (CNN) termasuk salah satu algoritma dalam bidang *Deep Learning* yang merupakan bagian dari *Machine Learning* sangat efektif dalam tugas pengolahan citra karena terdiri atas serangkaian lapisan atau layer yang secara otomatis mempelajari fitur-fitur kompleks dari sebuah citra atau gambar dimulai dari bagian-bagian terperinci, seperti sudut dan tepi gambar dan fitur yang abstrak, seperti objek dan bentuk (Hosny, Kassem, & Foad, 2019) (Maier, Syben, Lasser, & Riess, 2019) (Periyanto, Yudhana, & Umar, 2019) (Setiawan W. , 2021). Penelitian (Li, Liu, Yang , Peng, & Zhou, 2021) (Rozaqi, Sunyoto, & Arief, 2021) menyatakan dengan jelas bahwa CNN dapat melakukan ekstraksi fitur secara otomatis dan klasifikasi secara baik dan cocok dimanfaatkan untuk pendeteksian gambar. Kedua penelitian mendeteksi gambar berdasarkan hasil unggahan file. Demikian juga penelitian (Setiawan A. W., 2019) (Kassem, 2021), hanya mampu mendeteksi gambar berdasarkan hasil unggahan file namun, tidak bisa mendeteksi gambar langsung melalui kamera. Olehnya penelitian ini akan mendeteksi gambar langsung dari kamera. Dari sisi klasifikasi medis, jenis-jenis lesi kulit primer, yakni makula, papula, nodul, plak, urtikaria, vesikuler, bula, pustula, kista dan

purpura (Polevaya, Ravodin, & Filchenkov, 2019). Namun, pada penelitian ini, jenis lesi yang akan dideteksi terbatas pada 4 kelas, yakni makula, urtikaria, papula, dan Vesikuler. Jenis-jenis lesi primer tersebut akan dideteksi dengan model CNN yang akan dibangun. CNN akan bekerja dengan memproses citra dalam hal ini gambar yang dimasukkan via file atau kamera, kemudian mengklasifikasi gambar tersebut berdasarkan kategori atau kelas lesi kulit primer. Aplikasi akan dibangun berbasis Android menggunakan CNN agar proses pendeteksian langsung menjangkau kamera *handphone* untuk pengambilan gambar atau mengunggah gambar dari galeri *handphone*. Adanya penerapan aplikasi ini dapat membantu masyarakat untuk mengenali secara awal jenis lesi kulit primer yang dideritanya sehingga dapat melakukan pemeriksaan lanjut pada dokter spesialis kulit.

METODOLOGI PENELITIAN

Dalam penelitian ini, digunakan metode Scrum untuk pengembangan perangkat lunak berbasis Android. Scrum adalah metodologi pengembangan perangkat lunak yang fleksibel dan adaptif dalam menghadapi perubahan. Metode ini merupakan bagian dari Agile Software Development yang digunakan oleh pengembang untuk mengatasi kelemahan, kekurangan, dan perubahan dalam pengembangan perangkat lunak. Scrum memiliki struktur terstruktur dan dapat melibatkan perulangan untuk meningkatkan efisiensi dalam pengembangan perangkat lunak (Ariza, Mozo, & Quintero, 2018). Pada Scrum terdapat tahapan pengembangan (Rizky & Sugiarti, 2022) sebagai berikut:

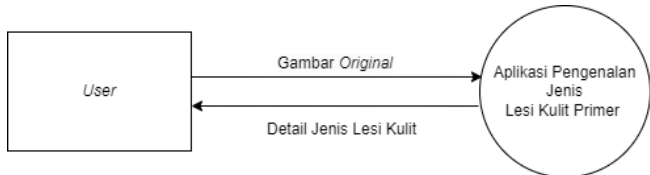
Initiate

Merupakan tahap awal untuk menganalisis sistem lama dan mengidentifikasi kebutuhan sistem baru. Dari penelitian sebelumnya, aplikasi Android untuk mendeteksi jenis lesi kulit menggunakan metode CNN melalui kamera belum pernah dilakukan sehingga pada penelitian ini, akan dibuat aplikasi Android untuk mendeteksi penyakit kulit melalui kamera atau unggahan file. Selain itu pula, penelitian sebelumnya hanya mendeteksi penyakit kulit berdasarkan file unggahan sebagai masukan namun, pada penelitian ini, penyakit kulit dapat dideteksi langsung melalui kamera, yaitu menangkap gambar langsung dari kulit yang dicurigai sebagai penyakit kulit. Aplikasi ini

menjadi media awal untuk mendeteksi penyakit kulit yang diderita seseorang.

Plan and Estimate

Merupakan tahap kedua dari metode *Scrum* yang mencakup perancangan sistem dan antarmuka. Perancangan sistem digambarkan dalam bentuk DFD. Sedangkan perancangan antarmuka digambarkan dalam bentuk *storyboard*.



Gambar 1 DFD Level 0

Pada gambar 1, pengguna dapat mengirimkan gambar ke aplikasi via unggahan file atau kamera dan hasilnya akan diketahui jenis lesi kulit yang diderita. Pengguna bisa berupa masyarakat yang ingin mengetahui jenis penyakit lesi kulit yang dideritanya. Dari gambar 1, akan diturunkan ke DFD level 1 untuk detail dari proses utama yang ada.



Gambar 2 DFD Level 1

Gambar 2 menjelaskan proses lebih detail dari aplikasi yang dibuat termasuk metode CNN yang digunakan. Terdapat 4 proses, yaitu memasukkan gambar, mengubah RGB gambar, menerapkan metode

CNN dan menampilkan hasil. Untuk proses 3.0, akan dijelaskan detail pada gambar 3. Gambar 3 menjelaskan detail langkah dari metode CNN, yang dimulai dari menghitung *convolutional layer*, menetapkan *pooling layer*, mengubah matriks menjadi *vector*, melakukan klasifikasi dan memperoleh hasil dari deteksi yang dilakukan.



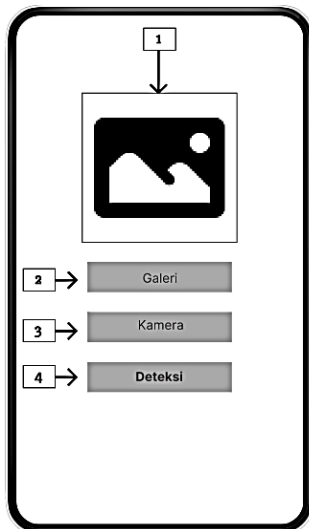
Gambar 3 DFD Level 2 Proses 3.0

Pada tahap ini pula, desain antarmuka dari aplikasi telah dirancang sesuai tahapan sebelumnya sebagai berikut:



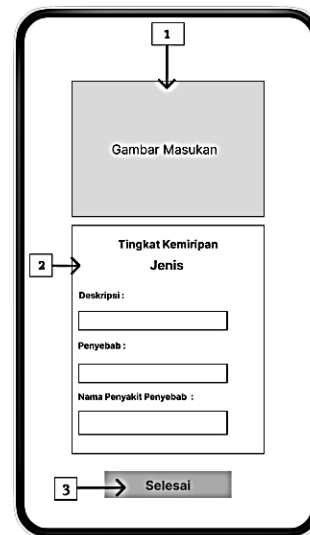
Gambar 4 Storyboard Halaman Muka Aplikasi

Pada gambar 4, terdapat angka 1 untuk menunjukkan gambar pengenalan lesi kulit primer, angka 2 sebagai teks “Selamat Datang di Aplikasi *Skin Lessons*” dan angka 3 merupakan tombol untuk memulai aplikasi.



Gambar 5 Storyboard Halaman Pengenalan Gambar

Gambar 5 merupakan halaman utama untuk mendeteksi penyakit kulit yang diderita pengguna. Angka 2 merupakan tombol untuk mengambil gambar dari galeri, angka 3 adalah tombol untuk mengambil gambar dengan kamera dan angka 4 merupakan tombol deteksi untuk gambar yang dimasukkan.



Gambar 6 Storyboard Halaman Hasil

Gambar 6 merupakan halaman hasil deteksi yang menunjukkan tingkat kemiripan, jenis lesi, deskripsi, penyebab dan nama penyakit penyebab.

Implementation

Tahap ketiga dari metode *Scrum* adalah tahapan implementasi desain antarmuka, penulisan kode program, dan simulasi untuk aplikasi pengenalan jenis lesi kulit berbasis Android menggunakan metode *Convolutional Neural Network*. Aplikasi akan dibangun menggunakan *Kotlin* dan *Python*, juga digunakan *Visual Code Studio* dan *Google Colab* sebagai teks editor. *Library* yang digunakan ada *Tensorflow*, *Keras* dan *Flask* sebagai pendukung.

Review and Retrospect

Setelah implementasi selesai dilakukan, maka langkah berikutnya adalah meninjau sistem melalui pengujian untuk mengidentifikasi celah, kekurangan, atau perubahan yang diperlukan. Untuk aplikasi ini, akan diterapkan sistem pengujian *black box*, yaitu teknik pengujian yang berfokus pada sisi fungsionalitas, khususnya apakah *CNN* berhasil mendeteksi penyakit lesi kulit.

Tabel 1 Pengujian

Pengujian yang Dilakukan	Hasil Pengujian
Melakukan pelatihan model CNN terhadap data 4 kelas lesi kulit primer.	CNN berhasil melakukan pendeteksian terhadap data 4 kelas lesi kulit primer.

Pengujian yang Dilakukan	Hasil Pengujian			
1/1 [.....] - 35 35/step	True: Makula Pred: Makula (99.9%) 	True: Makula Pred: Makula (100.0%) 	True: Makula Pred: Makula (100.0%) 	True: Makula Pred: Makula (53.1%)
	True: Makula Pred: Makula (100.0%) 	True: Makula Pred: Makula (100.0%) 	True: Makula Pred: Makula (100.0%) 	True: Makula Pred: Makula (100.0%)
	True: Makula Pred: Makula (96.9%) 	True: Makula Pred: Makula (82.9%) 	True: Makula Pred: Makula (99.9%) 	True: Makula Pred: Makula (100.0%)

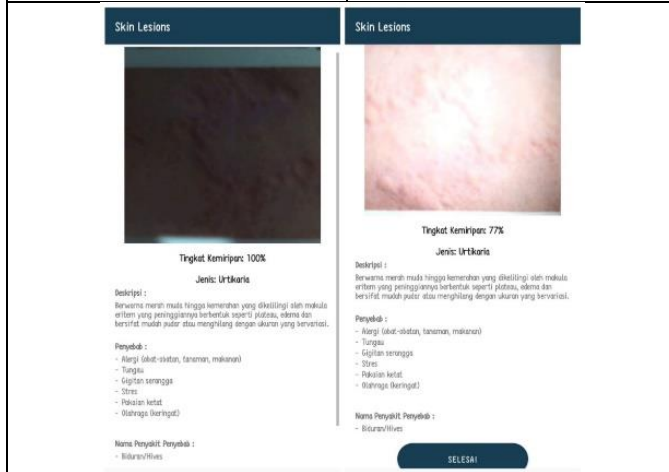
Melakukan pengujian data latihan dengan 100 epoch. Didapati nilai rata-rata precision, recall, f-1 score dan accuracy.

	precision	recall	f1-score
Makula	0.91	1.00	0.95
Urtikaria	1.00	0.91	0.95
Papula	0.98	0.93	0.96
Vesikular	0.93	0.99	0.96
accuracy	0.95		

Pengenalan atau deteksi lesi makula. Aplikasi dapat mengenali lesi makula.

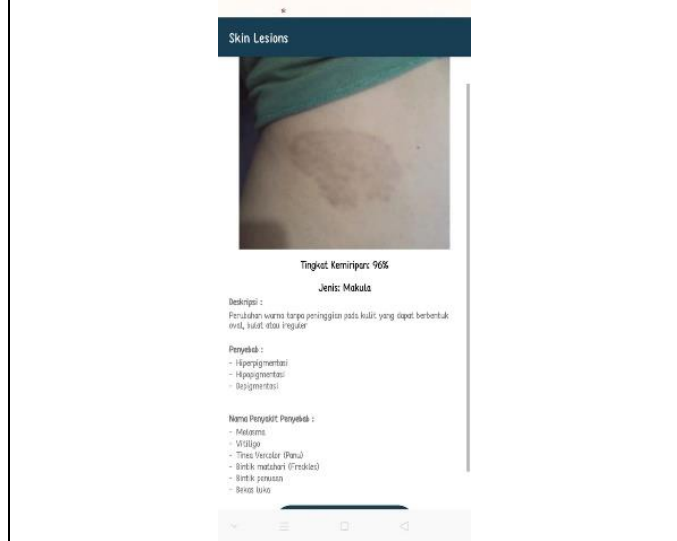


Pengenalan atau deteksi lesi urtikaria. Aplikasi dapat mengenali lesi urtikaria



Mengambil gambar menggunakan kamera. Aplikasi dapat mendeteksi gambar yang diambil.

Pengujian yang Dilakukan	Hasil Pengujian
dengan jarak kurang dari 50cm.	



Mengambil gambar menggunakan kamera dengan jarak 50cm. Aplikasi dapat mendeteksi gambar yang diambil.

HASIL DAN PEMBAHASAN

Setelah dilakukan pengembangan aplikasi Android menggunakan metode CNN, selanjutnya perlu dijelaskan mengenai data yang digunakan dalam aplikasi tersebut.

A. Data Preparation

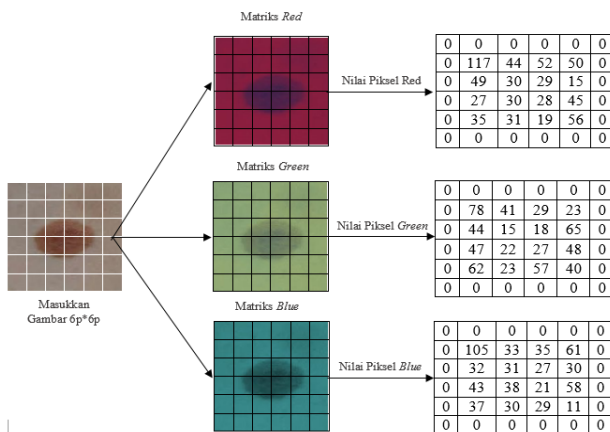
Pengumpulan data dilakukan terhadap dataset berupa kumpulan gambar lesi kulit primer. Terdapat 4 kelas pengenalan, yaitu makula, papula, urtikaria dan Vesikuler. Sumber data dari penelitian ini diambil dari DermNet dan Google Image. Selanjutnya data diperbanyak data dan dilatih dengan teknik augmentasi, seperti rescale, rotation, flip, zoom, brightness dan shear untuk memastikan tiap data yang akan dilatih dapat dideteksi dengan baik. Data Training untuk makula sebanyak 297, Urtikaria sebanyak 390, Papula sebanyak 343 dan Vesikuler sebanyak 315 data sehingga total data menjadi 1345.



Gambar 7 Data Training

Data yang ada kemudian divalidasi dengan membagi data latih sebesar 80% dan data uji sebesar 20%, yang menghasilkan data terbanyak ada pada kelas urtikaria, yakni sebesar 29% dan yang kurang pada kelas makula, yakni sebesar 22%.

Selanjutnya dilakukan perhitungan menggunakan CNN. Asumsi perhitungan terhadap sebuah citra atau gambar berukuran 6 piksel*6 piksel untuk melewati proses perhitungan pada tahap awal, yaitu *convolutional layer*. Proses *convolution* akan dilakukan pada gambar dengan piksel citra 3 *channel*, yaitu RGB (*Red, Green, Blue*) dengan menetapkan *padding* = 0 (same) pada nilai-nilai piksel yang dihasilkan oleh 3 *channel* RGB. Setelah mengetahui nilai piksel dari ketiga *channel*, langkah selanjutnya akan dilakukan perhitungan matematis sebanyak berapa kali tahap konvolusi dilakukan. Jika matriksnya 6x6, sebanyak 16 kali perhitungan akan dilakukan pada masing-masing *channel* RGB. Perhitungan matematis tersebut dapat dilihat pada gambar berikut.



Gambar 8 Nilai piksel dari Matriks RGB

Langkah selanjutnya adalah mengatur ukuran kernel konvolusi agar model CNN dapat menghasilkan *output* yang lebih ringkas atau tereduksi dari citra tersebut. Kernel *size* yang ditetapkan berukuran 3x3.

1	2	-1
0	1	2
2	-1	0

Gambar 9 Kernel Konvolusi 3x3

Selanjutnya adalah memulai perhitungan matematis terhadap tiap matriks pada *convolutional layer*.

1. Convolutional Layer

Matriks Red :

0*1	0*2	0*(-1)	0	0	0
0*0	117*1	44*2	52	50	0
0*2	49*(-1)	30*0	29	15	0
0	27	30	28	45	0
0	35	31	19	56	0
0	0	0	0	0	0

Matriks Red * Kernel 3x3

=

156			

Fitur Konvolusi 4x4

Gambar 10 Perhitungan 1 Konvolusi Matriks Red

Gambar 10 merupakan tahap awal konvolusi yang dilakukan terhadap matriks *red* dengan perkalian antara matriks *red* dan kernel konvolusi pertama. Berikut ini merupakan contoh perhitungan pertama terhadap matriks *red* :

Perhitungan 1 :

$$(0*1)+(0*2)+(0*(-1))+(0*0)+(117*1)+(44*2)+(0*2)+(49*(-1))+(30*0) = 156$$

Perhitungan yang sama dilakukan sesuai dengan berapa banyak *stride* yang dihasilkan pada matriks 6*6, yakni sebanyak 16 kali konvolusi.

Setelah total 16 kali perhitungan dilakukan terhadap matriks *red*, maka diperoleh hasil berupa fitur konvolusi 4x4.

156	216	183	93
272	265	189	178
120	205	234	86
121	174	172	174

Gambar 11 Fitur Konvolusi Matriks Red

Perhitungan yang sama diterapkan pada matriks *Green* dan *Blue* dan menghasilkan *feature maps* yang akan digunakan pada tahapan atau *layer* selanjutnya.

156	216	183	93
272	265	189	178
120	205	234	86
121	174	172	174

Hasil Channel Red

116	172	87	-6
142	254	241	146
102	233	98	270
180	201	116	163

Hasil Channel Green

139	72	192	85
228	269	124	171
115	191	223	196
100	186	73	148

Hasil Channel Blue

+

=

411	460	462	172
642	778	554	495
337	629	555	552
401	561	361	485

Feature Maps

Gambar 12 Feature Maps Hasil Konvolusi

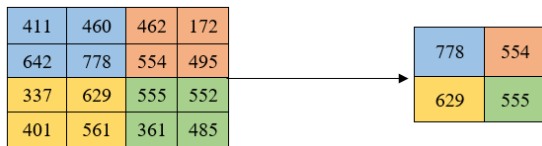
Gambar 12 memperlihatkan hasil perhitungan matematis yang dilakukan terhadap ketiga *channel* RGB. Nilainya akan saling dijumlahkan untuk memperoleh *feature maps* untuk kemudian dilanjutkan menjadi *input* pada proses atau tahapan selanjutnya.

2. Activation Function ReLu

Fungsi ini digunakan untuk mengubah setiap nilai negatif yang terdapat pada fitur menjadi bernial 0 dengan rumus fungsi $f(x) = \max(0, x)$. Akan tetapi *feature maps* yang dihasilkan pada tahap konvolusi sebelumnya tidak memuat nilai negatif, maka bersifat tetap dan tidak berubah.

3. *Pooling Layer*

Pada tahapan ini *feature maps* yang dihasilkan pada tahap *convolution* akan melalui proses *pooling*, dimana *feature maps* yang berukuran 4x4 tersebut akan diambil nilai *max pooling* atau nilai tertinggi dengan menetapkan *kernel size* 2x2 dan *stride* 2.



Gambar 13 Operasi Max Pooling

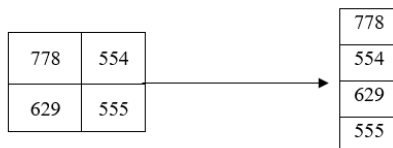
Gambar 13 menunjukkan bahwa *kernel size* 2x2 dan *stride* 2 menghasilkan *pool size* atau fitur baru berukuran 2x2. Pada perhitungan *max pooling* pertama nilai yang didapatkan, yaitu 778, perhitungan kedua dengan nilai 554, perhitungan ketiga dengan nilai 629, dan perhitungan keempat dengan nilai 555.

4. *Dropout Layer*

Tahapan ini bertujuan untuk mengurangi kompleksitas model yang sudah dibangun dengan membuang atau *dropout* setiap *neural network* yang tidak terpakai sehingga mengurangi beban komputasi dan kinerja model menjadi lebih cepat dengan ukuran 2x2x32 yang tidak diubah.

5. *Flatten Layer*

Flatten diterapkan untuk mengubah matriks atau tensor menjadi bentuk vektor atau matriks satu dimensi. Dalam konteks ini, dari hasil *output dropout layer* berukuran 2x2x32, akan diubah menjadi bentuk vektor baru. Ilustrasinya dapat dilihat pada gambar berikut ini.

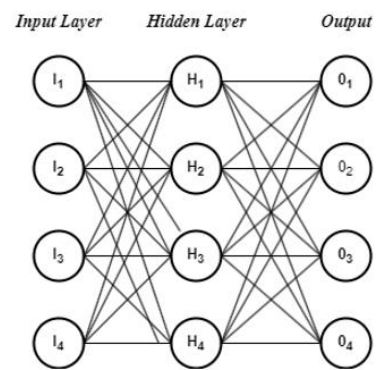


Gambar 14 Operasi Flatten Terhadap Matriks 2x2

Gambar 14 memperlihatkan proses *flatten* dengan matriks 2x2 yang ditransformasi menjadi matriks 4x1 atau berbentuk sebuah vektor. Vektor yang dihasilkan tersebut akan menjadi masukan untuk *fully connected layer* (*dense*).

6. *Fully Connected Layer*

Fully Connected merupakan proses yang menghubungkan setiap neuron dalam jaringan saraf pada setiap *layer* yang ada. *Layer* pada *fully connected layer* terdiri dari *input layer* yang berasal dari nilai vektor 1 dimensi yang telah dibuat pada tahap sebelumnya, *hidden layer* atau *layer* tersembunyi yang jumlahnya tidak dapat diperkirakan dan memuat nilai-nilai piksel dari *image processing* yang dilakukan dengan CNN dan *output* yang akan menunjukkan hasil akhir klasifikasi berdasarkan jumlah kelas yang ditetapkan, dalam hal ini sebanyak 4 kelas (gambar 15).



Gambar 15 Fully Connected Layer

Setelah melalui rangkaian tahapan dalam CNN, maka sampailah pada tahap terakhir dari model yang telah dibuat, yakni *dense* menggunakan fungsi aktivasi *softmax*. Contoh perhitungan pada *fully connected layer*.

a. Perhitungan *input layer*

Nilai yang akan dimasukkan pada *input layer* diambil dari output yang dihasilkan pada proses *flatten*. Setelah itu lakukan perhitungan dengan rumus:

$$\sum_{i=1}^N I_i * w_{ij} = H_i \dots\dots\dots(1)$$

Keterangan :

I = Nilai dari *Flatten Layer* sebelumnya

w = Bobot (*weight*)

H = Nilai yang akan dimasukkan pada *hidden layer*

Selanjutnya ambil nilai *output* dari proses *flatten*, yakni $I_1 = 778, I_2 = 554, I_3 = 629, I_4 = 555$ kemudian tentukan bobot atau *weight* yang acak dari *range* (-1,1), yaitu nilai $H_1 = 251,6; H_2 = 1006,4; H_3 = 1208;$ dan nilai $H_4 = 754,8$.

b. Perhitungan *hidden layer*

Pada perhitungan *hidden layer* yang diilustrasikan dengan H_1, H_2, H_3, H_4 , setiap neuron yang ada pada *input layer*, yakni I_1, I_2, I_3 dan I_4 dilakukan

perkalian dengan menentukan nilai bobot atau sehingga diperoleh nilai $H_1 = 251,6$, $H_2 = 1.006,4$, $H_3 = 1.208$ dan $H_4 = 754,8$. Berikut rumus perhitungan *hidden layer*.

$$\sum_{i=1}^N H_1 * w_{ij} = O_i \quad \dots\dots\dots (2)$$

Keterangan :

H = Nilai *hidden layer*

w = Bobot (*weight*) yang baru

O = Nilai *output*

Nilai yang telah diperoleh pada H_1, H_2, H_3 dan H_4 kemudian akan dikalikan lagi dengan bobot yang berbeda untuk memperoleh nilai *output* O_1, O_2, O_3 dan O_4 , yaitu $O_1 = 1.791,52$, $O_2 = 644,15$, $O_3 = 322,08$ dan $O_4 = 966,24$ yang mewakili nilai *output* dari 4 kelas yang akan dilatih dengan model CNN.

7. Fungsi Aktivasi *Softmax*

Selanjutnya masuk pada perhitungan untuk *softmax* menggunakan rumus eksponensial O_i dengan melakukan pembagian antara O_1, O_2, O_3 dan O_4 . Rumus perhitungan *softmax* dapat dilihat di bawah ini (Warsito, Deep Learning : Penentuan Bobot Neural Network, 2021):

$$s(O_i) = \frac{e^{O_i}}{\sum_{j=1}^n e^{O_j}} \quad \dots\dots\dots (3)$$

dengan hasil

$$s(O_1) + s(O_2) + s(O_3) + s(O_4) = 1$$

$$0,60 + 0,15 + 0,05 + 0,20 = 1$$

Berdasarkan hasil perhitungan menggunakan algoritma CNN terhadap data yang ada, diperoleh bobot nilai probabilitas yang lebih tinggi pada O_1 , yaitu 0,60 yang merujuk pada kelas 1 ‘makula’. Cara perhitungan data selanjutnya sama untuk setiap jenis lesi.

Data yang ada digunakan untuk melakukan perhitungan akurasi, presisi dan *recall* sebagai berikut:

- Total Dataset = 442
- True Positive = 422
- False Positive = 20
- False Negative = 20

1. *Precision*

Diketahui :

- FP (Makula) = 10
- FP (Urtikaria) = 0
- FP (Papula) = 2
- FP (Vesikular) = 8

Makula : = 0,90

- Urtikaria : = 1
- Papula : = 0,98
- Vesikular : = 0,92

All Precision = 0,95

2. *Recall*

Diketahui :

- FN (Makula) = 0
- FN (Urtikaria) = 11
- FN (Papula) = 8
- FN (Vesikular)= 1

- Makula : = 1
- Urtikaria : = 0,91
- Papula : = 0,93
- Vesikular : = 0,99

All Recall = 0,96

3. *Accuracy* = 0,95

Dari hasil *precision* 95%, *recall* 96%, dan *accuracy* sebesar 95% dapat dinyatakan bahwa aplikasi berhasil mengimplementasikan CNN dan aplikasi mampu mendeteksi jenis lesi kulit dengan akurasi yang tinggi melalui kamera dan unggah file.

KESIMPULAN

Berdasarkan hasil dari penelitian yang dilakukan, maka kesimpulan yang dapat ditarik, yaitu :

1. Pengembangan aplikasi Android berhasil dibangun dan mampu mengimplementasikan metode *convolutional neural network* untuk melakukan klasifikasi terhadap 4 kelas lesi kulit primer, yakni makula, urtikaria, papula dan vesikuler.
2. Pengujian terhadap model CNN menunjukkan performa dan akurasi yang baik dengan nilai rata-rata akurasi dari pelatihan model dengan 100 *epoch* sebesar 95%.
3. Pelatihan model CNN dengan *confusion matrix* memperoleh hasil, yakni pada kelas makula dengan *presicion* 91%, *recall* 100%, *f-1 score* 95%, kelas urtikaria dengan *precision* 100%, *recall* 91%, *f-1 score* 95%, kelas papula dengan *precision* 98%, *recall* 93%, *f-1 score* 96% dan kelas vesikular dengan *precision* 93%, *recall* 99%, *f-1 score* 96%.

UCAPAN TERIMA KASIH

Terima kasih kepada keluarga, dosen, dan semua teman-teman atas dukungannya dalam penelitian ini.

DAFTAR PUSTAKA

- Anum, Q., Gustia, R., Yenny, S. W., Akhayar, G., Asri, E., Ariani, T., Asri, R. C. (2021). Identifikasi Penyakit Kulit dan Pemeriksaan Lesi Kulit kepada Masyarakat di Daerah Binaan Universitas Andalas. *Jurnal Ilmiah Pengembangan dan Penerapan Ipteks*, 28(3), 335-340.
- Ariza, H. M., Mozo, V. R., & Quintero, H. M. (2018). Methodology for the Agile Development of Software Based on a Guide for the Body of Knowledge of SCRUM (SBOK TM Guide). *International Journal of Applied Engineering Research*, 13(14), 11479-11483.
- Cullell-Dalmau, M., Noe, S., Otero-Vinas, M., Meic, I., & Manzo, C. (2021). Convolutional Neural Network for Skin Lesion Classification: Understanding the Fundamentals Through Hands-On Learning. *Frontiers in Medicine*, 1-8.
- Hosny, K. M., Kassem, M. A., & Foad, M. M. (2019). Classification of skin lesions using transfer learning and augmentation with Alex-net. *Plos One*, 1-17.
- Kassem, M. A. (2021). Machine Learning and Deep Learning Methods for Skin Lesion Classification and Diagnosis: A Systematic Review. *Diagnostics*, 11, 1-29.
- Li, Z., Liu, F., Yang, W., Peng, S., & Zhou, J. (2021). A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects. *IEEE Transactions on Neural Networks and Learning Systems*, 33(12), 6999 - 7019.
- Maier, A., Syben, C., Lasser, T., & Riess, C. (2019). A gentle introduction to deep learning in medical image processing. *Z Med Phys*, 29(2), 86-101.
- Periyanto, A., Yudhana, A., & Umar, R. (2019). Rancang Bangun Klasifikasi Citra Dengan Teknologi Deep Learning Berbasis Metode Convolutional Neural Network. *Jurnal Format*, 8(2), 138-147.
- Polevaya, T., Ravodin, R., & Filchenkov, A. (2019). Skin Lesion Primary Morphology Classification With End-To-End Deep Learning Network. *International Conference on Artificial Intelligence in Information and Communication (ICAIIIC)*. Okinawa, Japan.
- Rizky, M., & Sugiarti, Y. (2022). Penggunaan Metode Scrum Dalam Pengembangan Perangkat Lunak: Literature Review. *Journal of Computer Science and Engineering (JCSE)*, 3(1), 41-46.
- Rozaqi, A. J., Sunyoto, A., & Arief, M. R. (2021). Deteksi Penyakit Pada Daun Kentang Menggunakan Pengolahan Citra dengan Metode Convolutional Neural Network. *CITEC Journal*, 22-31.
- Setiawan, A. W. (2019). Perbandingan Prescreening Lesi Kulit Berbasis Convolutional Neural Network: Citra Asli Dan Tersegmentasi. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, 8(4), 793-799.

- Setiawan, W. (2021). *Deep Learning menggunakan Convolutional Neural Network: Teori dan Aplikasi*. Malang: Media Nusa Creative (MNC Publishing).
- Warsito, B. (2021, Juli). *Deep Learning : Penentuan Bobot Neural Network*. Retrieved Juli 30, 2023, from https://eprints.undip.ac.id/44090/1/Warsito_Makalah_KNM16_25juli.pdf.

Halaman ini sengaja dikosongkan