

BAB II DASAR TEORI

2.1 KAJIAN PUSTAKA

Penelitian [8] ini mempelajari mengenai pengaruh pengoptimasian dalam kinerja model, pengoptimalan yang digunakan adalah *Root Mean Square Propagation* (RMSProp), Adam, dan SGD. Menggunakan RMSProp 5 lapis didapatkan hasil akurasi 91%, Adam 5 lapis didapatkan hasil akurasi 89%, dan *Stochastic Gradient Descent* (SGD) 5 lapis didapatkan hasil 88%. Dapat disimpulkan bahwa menggunakan pengoptimasian RMSProp mendapatkan akurasi terbaik. Penelitian [9] Ini melakukan evaluasi efisiensi CapsNet untuk klasifikasi Tuberkolosis dalam gambar CRX dan dibandingkan dengan AlexNet serta VGG-16. Dilakukan klasifikasi Tuberkolosis menggunakan 2 kelas yaitu Tuberculosis dan Normal, citra yang digunakan yaitu chest x-ray. Pada penggunaan arsitektur VGG-16 berhasil mendapatkan hasil akurasi paling baik sebesar 94.56%.

Penelitian [6] ini melakukan identifikasi penyakit pneumonia pada gambar x-ray dengan menggunakan salah satu metode yaitu CNN berdasarkan ekstraksi fitur sobel. Pada penelitian ini diklasifikasikan menjadi 2 kelas yaitu pneumonia dan non pneumonia, pada pengujian 102 x-ray pneumonia didapatkan hasil 95 citra x-ray benar dideteksi oleh sistem dan pada 101 data uji x-ray non pneumonia didapatkan hasil 91 citra x-ray benar di deteksi. Didapatkan nilai keakurasian sebesar 91,54% dan didapatkan dari nilai *epoch* 50, nilai dari *learning rate* sebesar 0,0001 serta *batch* yang bernilai 20. Pada Penelitian [7] ini melakukan perbandingan arsitektur ResNet-50 dengan ResNet-152 untuk melakukan identifikasi penyakit pneumonia. Dalam melakukan perbandingan ini menggunakan 2 kelas citra x-ray yaitu normal dan pneumonia, hasil yang didapatkan akurasi masing-masing arsitektur ResNet 50 sebesar 88% dan ResNet 152 sebesar 89%. Ukuran citra x-ray dan jumlah layer dapat mempengaruhi hasil akurasi pengujian. .

Pada Penelitian [10] dari melakukan implementasi dari salah satu metode yaitu ResNet-50 serta metode 101 *backbone* dalam melakukan deteksi penyakit pneumonia menggunakan citra x-ray. Pada penelitian ini juga menggunakan empat macam skema pengujian yang telah dilakukan yaitu menggunakan gambar x-ray asli yang tidak teraugmentasi, serta gambar x-ray yang telah teraugmentasi, gambar x-ray yang dilakukan perbaikan CLAHE tidak teraugmentasi, dan citra diperbaiki CLAHE

teraugmentasi. Setelah dilakukan analisis hasil terbaik yang didapatkan adalah menggunakan skema gambar yanggg dilakukan perbaikan CLAHE teraugmentasi yang mendapatkan nilai hasil akurasi sebesar 76%, nilai presisi 62%, nilai sensitivitas 74%, dan dengan nilai spesifisitas 77%.

2.2 DASAR TEORI

2.2.1 Pneumonia

Pneumonia adalah penyakit pada paru-paru yang menyebabkan sistem pernapasan mengalami peradangan dan menimbulkan cairan. Pneumonia dapat disebabkan karena virus, bakteri, dan jamur [11]. Pada alveolus paru-paru terjadi pembengkakan yang menyebabkan cairan atau lendir dapat menumpuk dan mengganggu saluran pernafasan. Penyakit pneumonia akan mengakibatkan sulit untuk bernafas dan akan menyebabkan suhu tubuh serta menimbulkan batuk pada penderita [6]. Penyakit Pneumonia merupakan salah satu jenis penyebab kematian yang besar pada anak-anak di dunia karena infeksi tunggal [3]. Penyakit pneumonia juga dapat disebabkan oleh mikroorganisme yang bersifat patogen yang menyebabkan sistem kekebalan pada tubuh melemah sehingga akan menyebabkan paru-paru meradang.



Gambar 2.1 X-ray paru-paru terinfeksi.[12]



Gambar 2.2 X-ray paru-paru normal[12]

Pada citra diatas dapat dilihat pada gambar 2.1 merupakan paru-paru yang terinfeksi oleh mikroorganisme penyebab penyakit pneumonia dan pada gambar 2.2 merupakan x-ray paru-paru yang normal. Penyakit pneumonia ini kerap menjangkit bayi, anak-anak, maupun lansia diatas usia 65 tahun dan dengan orang yang memiliki tingkat kekebalan tubuh yang lemah [13]. Berdasarkan kuman penyebab pneumonia, pneumonia di klasifikasikan menjadi empat jenis sebagai berikut :

2.2.1.1 Pneumonia Bacterial

Pneumonia *bacterial* merupakan salah satu jenis klasifikasi penyakit pneumonia yang disebabkan oleh *myctoplasma*. Organisme yang biasa menyerang yaitu bakteri *Streptococcus pneumonia* dan bakteri *Haemofilus influenza*.

2.2.1.2 Pneumonia Atipikal

Pneumonia atipikal merupakan salah satu klasifikasi penyakit pneumonia yang disebabkan oleh bakteri *Chlamidia trachomatis*, dan bakteri *C. pneumonia* serta bakteri *Pneumocytis*.

2.2.1.3 Pneumonia Virus

Pneumonia virus merupakan jenis klasifikasi pneumonia yang biasanya disebabkan oleh jenis virus *parainfluenza*, virus *influenza* serta *Cytomegalovirus*

2.2.1.4 Pneumonia Jamur

Pneumonia jamur merupakan salah satu klasifikasi penyakit pneumonia yang menyerang karena imun tubuh sedang lemah sehingga mudah terserang.

2.2.2 Citra

Citra atau *image* adalah sebuah gambar bidang yang berbentuk bidang dua dimensi dan disusun dari banyak pixel. Gambar terbentuk dari sebuah kotak yang teratur hingga jarak secara *horizontal* maupun *vertikal* datar pada pixel akan sama di keseluruhan bagian gambar [14]. Gambar digital tersusun dari beberapa elemen, dan pada setiap komponen akan memiliki nilai dan posisi. Elemen tersebut disebut juga dengan *picture elements*, atau *image elements*, dan *pels*[15]. Pada *grid pixel* memiliki nilai angka yang menggambarkan *channel* warna, dan angka tersebut disimpan secara urut oleh komputer.

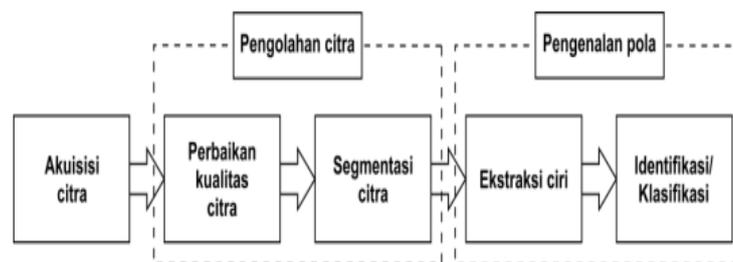


Gambar 2.3 Citra[14]

Gambar 2.3 merupakan salah satu contoh citra. Citra atau gambar dalam sistem komputer berbentuk matriks 2 dimensi. Setiap piksel mempunyai angka yang dapat mempresentasikan channel warna, dan angka tersebut di simpan secara berurutan oleh komputer dan sering di kurangi untuk kompresi maupun pengolahan tertentu.

2.2.3 Pengolahan Citra Digital

Pengolahan citra mampu mengidentifikasi jenis atau banyaknya objek pada suatu citra. Pengolahan citra digital dapat berfungsi untuk memperbaiki kualitas citra agar menjadi lebih baik sehingga mudah untuk diinterpretasi oleh sistem komputer. Pada penelitian ini pengolahan citra digital digunakan dalam proses mendeteksi penyakit pneumonia menggunakan citra x-ray. Pada gambar 2.4 merupakan langkah pengolahan citra digital.



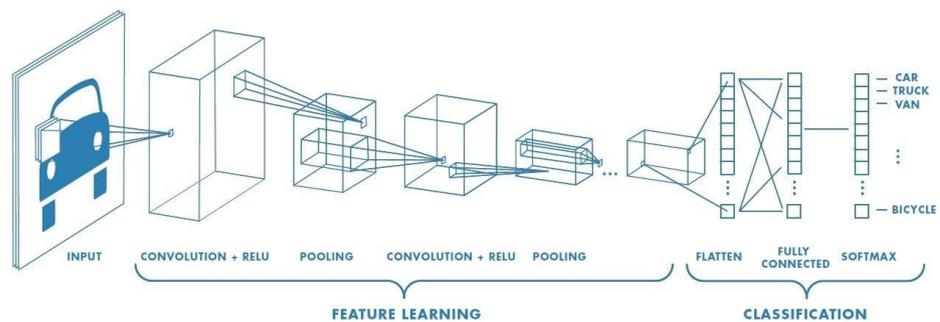
Gambar 2.4 Langkah pengolahan pada citra digital

Pengolahan citra atau gambar digital merupakan salah satu jenis keilmuan yang biasanya mempelajari tentang teknik dalam pengolahan gambar. Pengolahan citra atau gambar yang digunakan dalam melakukan perbaikan kesalahan data pada sinyal gambar atau citra yang diakibatkan transmisi dan digunakan dalam meningkatkan tingkat kualitas gambar agar mudah

diinterpretasikan dengan penglihatan mata manusia. Proses dari pengolahan gambar digital terbagi menjadi tiga garis besar yaitu pengakuisisian citra, melakukan pengolahan gambar atau citra , dan juga pengenalan pada pola.

2.2.4 Convolution Neural Network

Convolutional Neural Network berfungsi untuk melakukan klasifikasi data yang telah diberi label dengan menggunakan metode *supervised learning*. Cara kerjanya dengan ada data yang akan dilatih dan ada variabel yang dijadikan target yang memiliki tujuan untuk mengelompokkan data kedalam data yang telah ada.



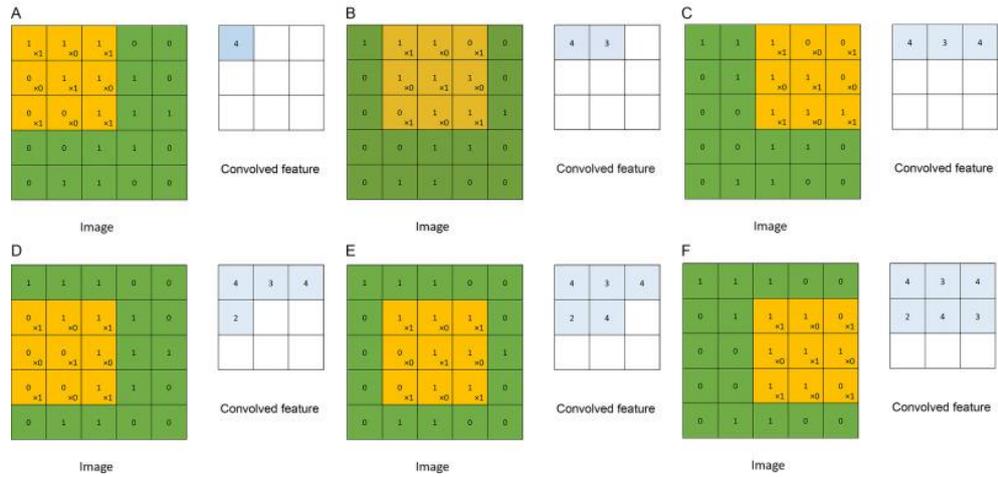
Gambar 2.5 Convolution neural network(CNN)[16]

Pada gambar 2.5 merupakan tahapan proses CNN. *Convolution Neural Network* atau (CNN) merupakan jenis dari *Deep Neural Network* karena banyak dipakai dalam pengolahan citra atau gambar. CNN juga pengembangan dari *Multilayer Perceptron* (MLP) diciptakan untuk melakukan pengolahan data yang berbentuk 2D. CNN juga digunakan dalam pengklasifikasian data yang menggunakan salah satu jenis metode yaitu metode *supervised learning*, *supervised learning* memiliki cara kerja yang dilakukan dengan data dilatih yang memiliki nilai variabel yang memiliki target, maka metode ini memiliki salah satu tujuan yang melakukan pengelompokan suatu informasi ke dalam data yang telah ada [16].

2.2.5 Convolution Layer

Convolution Layer merupakan lapisan bagian paling luar dan lapisan yang sangat penting untuk digunakan. *Convolution layer* memiliki tujuan operasi konvolusi di data gambar yang digunakan untuk melakukan ekstraksi fitur dari gambar yang dimasukan. Definisi dari *convolution* sendiri merupakan pengaplikasian sebuah kernel yang di analogikan berupa kotak kuning pada citra,

sedangkan kotak yang memiliki warna hijau adalah gambar yang akan dilakukan proses dari konvolusi[17].

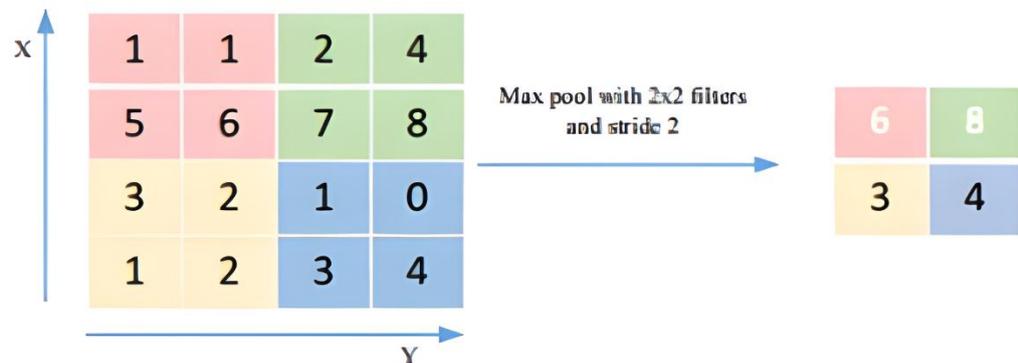


Gambar 2.6 Convolution layer[17]

Pada gambar 2.6 merupakan contoh dari *convolution layer*, Kernel yang memiliki pergerakan dari sudut kiri bagian atas ke sudut bagian kanan bawah, hingga nilai hasil dari konvolusi gambar dapat dilihat di gambar yang terletak di sebelah kanannya.

2.2.6 Pooling Layer

Pooling Layer merupakan *layer* yang masuknya berupa suatu fungsi bernama *feature map*. Fungsi ini kemudian diolah berdasarkan nilai piksel terdekat dengan berbagai macam operasi statistik [18]. *Pooling* layer dalam model *CNN* umumnya dimasukan di antara lapisan konvolusi secara berurutan. Operasi *down-sampling* yang digunakan untuk mengurangi jumlah parameter merupakan salah fungsi dari lapisan ini yaitu untuk mereduksi input secara spasial.



Gambar 2.7 Pooling layer[18]

Gambar 2.7 merupakan contoh dari *pooling layer*. *Pooling layer* biasanya dipakai untuk mengambil nilai maksimal atau *max pooling* dan nilai rata-rata atau *average pooling* yang didapatkan dari bagian piksel citra. Cara kerja yang digunakan dalam proses *max pooling* adalah dengan membagi keluaran dari *convolution layer* menjadi *grid* kecil lalu mengambil nilai maksimal pada setiap *grid* untuk digunakan dalam penyusunan matriks citra.

2.2.7 Fully Connected Layer

Fully connected layer merupakan lapisan dengan semua *neuron* di lapisan selanjutnya sudah dihubungkan dengan semua *neuron aktivasi* dari lapisan sebelumnya. Lapisan ini yang beroperasi pada dimensi data dengan melakukan perubahan dengan tujuan proses klasifikasi data secara linier. *Multi Layer Perceptron* (MLP) merupakan salah satu contoh penerapan *layer* ini [10].

2.2.8 Softmax Layer

Softmask merupakan fungsi untuk klasifikasi yang menghasilkan keluaran distribusi probabilitas. Untuk mendapatkan output, *softmax layer* biasanya digunakan pada *layer* terakhir *Neural Networks*. *Softmax layer* menentukan probabilitas terbesar untuk hasil kelasnya[19]. Pada persamaan 2.1 berikut merupakan persamaan fungsi softmask

$$p(x)_i = \frac{e^{a_i}}{\sum_{k=1}^N e^{a_k}} \quad (2.1)$$

P_i merupakan fungsi softmax yang diterapkan

e merupakan bilangan konstan euler (2,71828)

a merupakan vektor input yang akan diubah

menjadi probabilitas

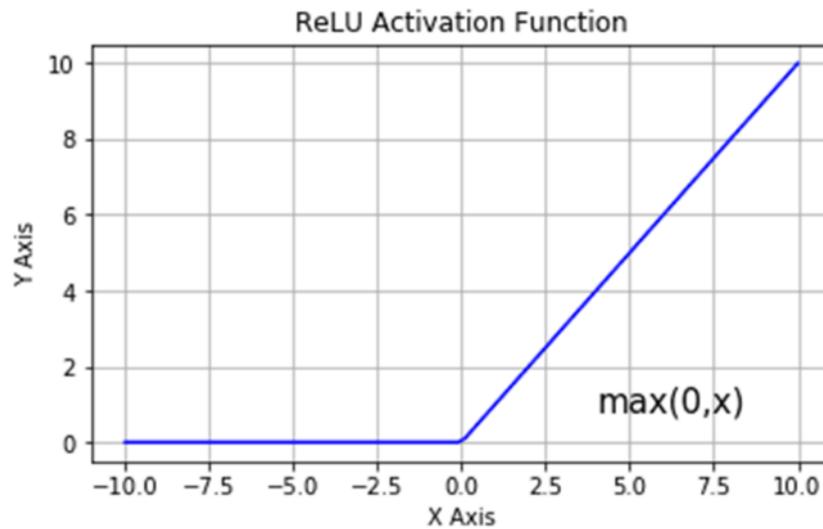
i merupakan indeks dari elemen vektor x

N merupakan jumlah elemen vektor x

2.2.9 Rectified Linear Units (ReLU) Activation

Rectified Linear unit (ReLU) memiliki fungsi aktivasi yang mengubah nilai input *negative* menjadi bernilai nol. ReLU digunakan untuk meningkatkan tahap pelatihan menjadi berkali-kali lebih cepat. Menggunakan aktivasi ReLU pada lapisan fungsi dapat meningkatkan kinerja jaringan[18]. ReLU bekerja dengan ketentuan nilai pada 0, yaitu $f(x) =$

maksimal ($0 \cdot x$). Jadi akan didapatkan nilai output 0 jika $x < 0$, dan begitupun sebaliknya jika nilai $x \geq 0$ maka output fungsinya linear.



Gambar 2.8 Rel-u activation

Dapat dilihat pada gambar 2.8. Secara konvensional ReLU digunakan sebagai fungsi aktivasi *neural network*, dengan menggunakan *softmax* sebagai fungsi aktivasi di akhir *node*. Selanjutnya jaringan menggunakan fungsi *cross-entropy softmax* untuk mempelajari parameter berat Θ dari jaringan saraf[18]. Pada persamaan 2.2 dibawah ini digunakan dalam ReLU.

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \quad (2.2)$$

2.2.10 Cross Entropy Loss Function

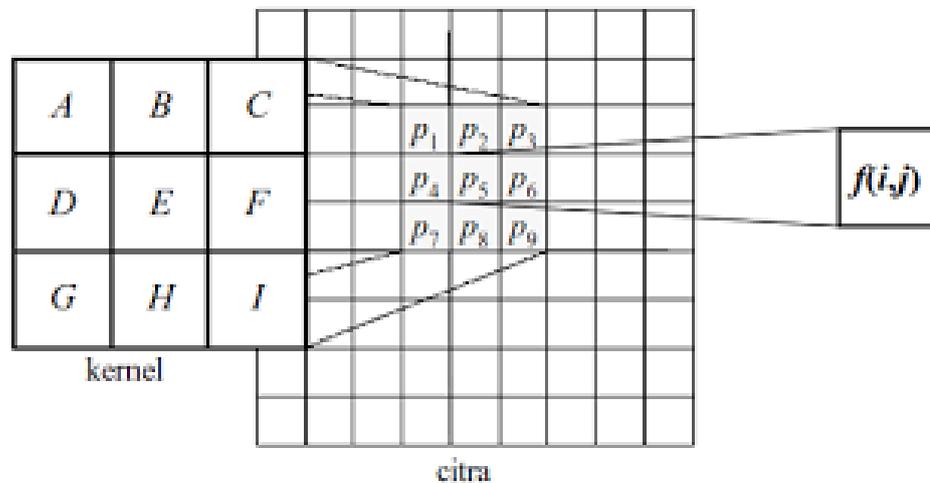
Loss Function berfungsi untuk meminimalisir nilai *error*. Model dengan banyak kelas memerlukan langkah untuk mencari perbandingan antara probabilitas hipotesis hasil hipotesis dengan probabilitas dengan kebenaran yang asli[20]. Algoritma akan banyak muncul selama dilakukan pelatihan akan muncul yang menyesuaikan dengan parameter dengan tujuan meminimalisir perbedaan.

2.2.11 Machine Learning

Salah satu cabang ilmu tentang cara komputer memiliki kecerdasan adalah *Machine learning*. Untuk memiliki kecerdasan, komputer harus dapat melakukan pembelajaran. Artinya, *machine learning* adalah salah satu bidang keilmuan dengan fokus pembelajaran mesin atau komputer untuk cerdas [21].

2.2.12 Konvolusi

Operator dengan masukan berupa kombinasi linier piksel yang menghasilkan piksel lain merupakan salah satu penerapan konvolusi dalam *image processing*. Konvolusi merupakan teknik untuk memperjelas atau menghaluskan citra dengan menggantikan nilai piksel dengan beberapa nilai piksel yang berdekatan dengan piksel yang asli [17]. Konvolusi dapat melakukan proses tersebut dengan tidak mengubah ukuran dari gambar. Tujuannya yaitu untuk mendapatkan fitur dari gambar gambar sebagai masukan. Perubahan yang linier akan dihasilkan oleh konvolusi dari data masukan yang sesuai dengan informasi yang ada. Optimalisasi keluaran pada lapisan konvolusi mengakibatkan kompleksitas yang signifikan. Gambar 2.9 merupakan contoh proses konvolusi matriks kernel dengan matriks gambar digital [22]:



Gambar 2.9 Matriks konvolusi[23]

$$f(i, j) = Ap_1 + Bp_2 + Cp_3 + Dp_4 + Ep_5 + Fp_6 + Gp_7 + Hp_8 + Ip_9 \quad (2.3)$$

f(i,j) = hasil konvolusi citra

A - I = matriks kernel

P₁P₂ = fungsi kernel

Pada persamaan 2.3, proses konvolusi dilakukan pada matriks citra $m \times n$ dengan matriks kernel berukuran 3×3 . Sub matriks berukuran 3×3 diambil dari matriks citra yang selanjutnya diubah menjadi vektor berukuran $3 \times 2 = 9$ serta matriks kernel yang diubah menjadi vektor $3 \times 2 = 9$ [23].

2.2.12.1 Grayscale

Citra yang memiliki hanya satu skala warna abu-abu dinamakan citra *grayscale*. Perubahan warna citra ke *grayscale* memiliki tujuan untuk mengurangi informasi yang dibutuhkan dalam suatu proses elemen citra. Hal ini karena warna abu-abu memiliki satu warna intensitas yang sama dalam komponen RGB (*red, green, blue*).

2.2.12.2 Kernel

Kernel merupakan matriks angka kecil yang biasanya digunakan dalam proses konvolusi pada sebuah gambar. Kernel dengan pola angka yang berbeda menghasilkan hasil yang berbeda juga pada konvolusi. Kernel umumnya memiliki ukuran $m \times n$, namun kernel 3×3 merupakan bentuk yang sering digunakan berukuran [19].

2.2.12.3 Stride

Stride digunakan sebagai parameter untuk menentukan pergeseran filter pada sebuah sistem. Apabila *stride* memiliki nilai 1, artinya *convolutional filter* akan mengalami pergeseran sebesar 1 piksel *horizontal* kemudian *vertikal*. Nilai *stride* yang semakin kecil menggambarkan detail informasi yang semakin kecil juga dari sebuah masukan. Nilai *stride* yang semakin besar tidak membutuhkan komputasi yang lebih besar jika dibandingkan dengan *stride* kecil. Akan tetapi, penggunaan *stride* tidak selalu mendapatkan hasil yang baik.

2.2.12.4 Padding

Penambahan ukuran pada sebuah piksel dengan suatu nilai tertentu di sekitar data masukan disebut *padding*. Penambahan ini memiliki tujuan agar bidang reseptif menghasilkan nilai yang tidak terlalu kecil agar tidak menghilangkan informasi yang penting. *Zero padding* merupakan salah satu contoh jenis *padding* yang bernilai nol. Data yang bernilai tunggal didapatkan dari hasil bidang reseptif. Hasil dari proses konvolusi ini akan digunakan sebagai masukan untuk proses konvolusi setelahnya. Jika hasilnya negatif, maka akan dikenakan proses *zero padding* atau diganti nilainya menjadi nol. Sebaliknya jika konvolusi mengeluarkan nilai piksel lebih besar daripada nilai *grayscale* maksimum yaitu 255, maka nilai itu akan dijadikan nilai *grayscale* maksimum yang baru.

2.2.12.5 Feature Map

Hasil keluaran dari sebuah proses konvolusi disebut *feature map*. Konvolusi merupakan proses yang mengalikan nilai masukan dengan nilai filter. Analisis klasifikasi dapat dilakukan melalui *Feature Map* berdasarkan blok dari berbagai aspek untuk memahami fitur-fitur dari masukan yang dipertahankan atau terdeteksi oleh model sistem.

2.2.13 Confusion matrix

Confusion matrix merupakan parameter untuk mengukur tingkat akurasi dari sebuah model yang dibangun. Matriks ini berisikan perbandingan antara nilai sebenarnya dengan nilai prediksi model. Jika nilai prediksi sama dengan nilai sebenarnya, maka akan bernilai *true negative* atau *true positive*. Jika nilainya berbanding terbalik, maka akan bernilai *false negative* atau *false positive*.

		Actual Condition		
		FALSE	TRUE	
Predicted Condition	FALSE	TN	FN	Predicted Negative
	TRUE	FP	TP	Predicted Positive
		Actual Negative	Actual Positive	

Gambar 2.10 Confusion matrix

Gambar 2.10 merupakan contoh tabel *confusion matrix*, secara kuantitatif perhitungan *confusion matrix* adalah sebagai berikut :

2.2.13.1 Akurasi

Akurasi merupakan suatu indikator pengujian seberapa akurat nilai rasio prediksi benar (*negative dan positif*) untuk dapat melakukan tahapan klasifikasi dengan benar. Persamaan dibawah ini digunakan untuk menghitung nilai akurasi dengan melakukan penjumlahan jumlah prediksi *true* dibagi semua prediksi.

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \% \quad (2.4)$$

TP : Jumlah x-ray pneumonia terdeteksi pneumonia

TN : Jumlah x-ray normal terdeteksi normal

FP : Jumlah x-ray normal terdeteksi pneumonia

FN : Jumlah x-ray pneumonia terdeteksi normal

2.2.13.2 Presisi

Presisi merupakan indikator pengujian yang menggambarkan seberapa tepatnya suatu model untuk melakukan prediksi kejadian positif dalam sebuah rangkaian kegiatan prediksi. Persamaan dibawah ini digunakan untuk melakukan perhitungan nilai presisi dengan melakukan pembagian nilai true positif dengan jumlah data positif.

$$\text{Presisi} = \frac{TP}{TP + FP} \times 100 \% \quad (2.5)$$

TP : Jumlah x-ray pneumonia terdeteksi pneumonia

FP : Jumlah x-ray normal terdeteksi pneumonia

2.2.13.3 Loss

Loss merupakan indikator pengujian yang menggambarkan hasil perhitungan yang menghasilkan nilai ketidakakurasian sistem dalam mengidentifikasi sebuah objek. Persamaan dibawah ini digunakan untuk mengetahui nilai loss. Nilai p merupakan probabilitas untuk setiap kategori sebuah keluaran, q merupakan nilai keluaran yang diinginkan, Dimana nilai $i \in [1, N]$ dan N merupakan jumlah neuron yang terdapat pada lapisan keluaran.

$$\text{Loss} = H(p, q) = - \sum_i q_i \log p_i \times 100 \quad (2.6)$$

q_i merupakan label actual untuk kelas ke- i (0 atau 1)

p_i merupakan probabilitas prediksi model untuk kelas ke- i

2.2.13.4 Recall

Recall merupakan suatu indikator pengujian yang menampilkan keberhasilan jenis gambar yang teridentifikasi benar setelah proses identifikasi. Persamaan dibawah ini digunakan untuk mendapatkan nilai *recall*.

$$\text{Recall} = \frac{TP}{TP + FN} \times 100 \% \quad (2.7)$$

TP : Jumlah x-ray pneumonia terdeteksi pneumonia

FN : Jumlah x-ray peneumonia terdeteksi normal

2.2.13.5 F1-Score

F1-Score merupakan perhitungan untuk melakukan pencarian suatu variable nilai baru dengan cara melakukan pengabungan nilai *recall* dan *presisi*. Karena keadaan tersebut nilai dari *precision* dan *recall* akan memiliki bobot yang berbeda. Persamaan dibawah ini digunakan untuk mengetahui nilai F1-Score

$$F1 - Score = 2 \times \frac{Recall \times presisi}{Recall + presisi} \times 100 \% \quad (2.8)$$

2.2.13.6 Optimizer

Optimizer adalah metode optimasi yang berfungsi untuk meningkatkan akurasi dan mengurangi kesalahan dengan mendapatkan nilai bobot yang optimal. Saat proses *training* dilakukan percobaan mengubah nilai parameter untuk mengurangi nilai *loss* dan memberikan hasil yang akurat [3]. Optimizer yang akan digunakan dalam percobaan yaitu *Adaptive Momentum* (Adam), *Stochastic Gradients Descent* (SGD), dan *Root Mean Square propagation* (RMSprop).

Optimizer Adam merupakan algoritma yang dikenalkan oleh Kingma & Ba. *Optimizer* Adam memiliki kelebihan yaitu implementasi yang mudah, membutuhkan memori yang kecil, cocok digunakan untuk data yang besar dan mampu menghasilkan akurasi yang lebih baik [22]. *Optimizer* SGD adalah algoritma yang bekerja dengan memperbaiki nilai *error* yang dihasilkan menggunakan nilai *learning rate* yang tidak berubah [21]. *Optimizer* RMSprop merupakan algoritma yang bekerja dengan mempertahankan nilai parameter *learning rate* yang didapatkan dari rata-rata *gradien* [21]. *Optimizer* RMSprop adalah *optimizer* yang diusulkan pertama kali oleh Geoff Hinton yang.

Formula mametis untuk *optimizer* Adam, SGD, dan RMSprop dapat dilihat dari persamaan berikut:

$$RMSprop = \theta_{t+1,i} = \theta_{t,i} - \frac{\eta}{\sqrt{\epsilon + E[g^2]_t}} \nabla J(\theta_{t,i}) \quad (2.9)$$

$$Adam = \theta_{n+1} = \theta_n - \frac{a}{\sqrt{\hat{v}_n + \epsilon}} \hat{m}_n \quad (2.10)$$

$$SGD = \theta = \theta - \eta \cdot \nabla_{\theta} J(\theta; x; y) \quad (2.11)$$

Dimana : θ merupakan parameter (Theta)
 η merupakan Learning Rate (Eta)
 ∇ merupakan gradient (Nabla)
 $J(\theta;x;y)$ merupakan input parameter θ
dengan contoh training dan label (kelas)

2.2.14 Learning Rate

Learning rate merupakan salah satu bagian parameter *training* yang memiliki fungsi untuk menghitung nilai koreksi bobot Ketika proses *training*. Nilai *learning rate* sendiri berada pada *range* nol (0) sampai satu (1). Semakin tinggi nilai *learning rate* maka proses *training* akan semakin cepat tetapi semakin tinggi nilainya maka ketelitian jaringan akan berkurang [24].

2.2.15 Epoch

Epoch adalah perwakilan jumlah iterasi yang dilakukan pada set data. *Epoch* mengindikasikan sebuah CNN belajar dari *training* dataset secara keseluruhan dan *epoch* merupakan tanda bahwa satu siklus CNN dari keseluruhan training dataset [25]. Konvergensi nilai *loss* berfungsi untuk menentukan apakah sistem sudah mencapai nilai *epoch* terbaik.