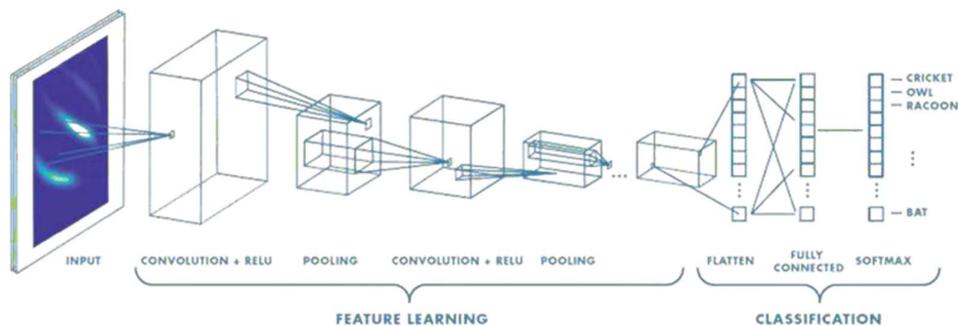


keluaran atau kesimpulan yang ditarik oleh jaringan didasarkan pada pengalamannya selama mengikuti proses pembelajaran/pelatihan (Puspitaningrum, 2006). JST mampu mengidentifikasi struktur model dan efektif dengan menghubungkan input serta output simulasi dan model peramalan. (Setiawan dan Rudiyanto, 2004).

2.8 Convolution Neural Network (CNN)

Convolutional Neural Network (CNN) merupakan pengembangan dari Multilayer Perceptron (MLP) yang didesain untuk mengolah data dua dimensi dalam bentuk citra. CNN ini termasuk kedalam jenis *Deep Neural Network* karena kedalaman jaringan yang tinggi dan banyak diaplikasikan pada data citra. Pada dasarnya klasifikasi citra dapat digunakan dengan MLP, akan tetapi dengan metode MLP kurang sesuai untuk digunakan karena tidak menyimpan informasi spasial dari data citra dan menganggap setiap piksel adalah fitur yang independen sehingga menghasilkan hasil yang kurang baik.

Penelitian awal yang mendasari penemuan CNN ini pertama kali dilakukan oleh Hubel dan Wiesel (Hubel & Wiesel, T, 1968) mengenai virtual cortex pada indera penglihatan kucing. Secara teknis, CNN adalah sebuah arsitektur yang dapat dilatih dan terdiri dari beberapa tahap. Masukan (input) dan keluaran (output) dari setiap tahap adalah terdiri dari beberapa array yang biasa disebut feature map. Setiap tahap terdiri dari tiga layer yaitu konvolusi, fungsi aktivasi layer dan pooling layer (Ramba, 2020). Berikut adalah jaringan arsitektur Convolutional Neural Network :



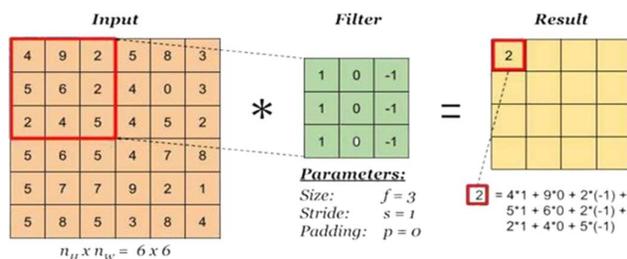
Gambar 2. 4 Arsitektur CNN

2.9 Feature Learning

Feature learning adalah proses encoding dari sebuah gambar menjadi feature yang berupa nilai-nilai yang merepresentasikan gambar tersebut. Proses ini terdiri dari beberapa layer yang saling bekerjasama untuk mengambil ciri dari 11 sebuah gambar (Ramba, 2020). Berikut ini penjelasan dari setiap layer pada *feature learning* :

2.9.1 Convolution Layer

Convolution Layer adalah suatu istilah matematis yang berarti mengaplikasikan sebuah fungsi pada output fungsi lain secara berulang. Convolution layer terdiri dari neuron yang membentuk sebuah filter dengan panjang dan tinggi (pixels) tertentu. Filter akan digeser pada keseluruhan bagian dari gambar. Setiap pergeseran akan dilakukan operasi 'dot' antara input dan filter, sehingga menghasilkan sebuah nilai output tertentu (Ramba, 2020).



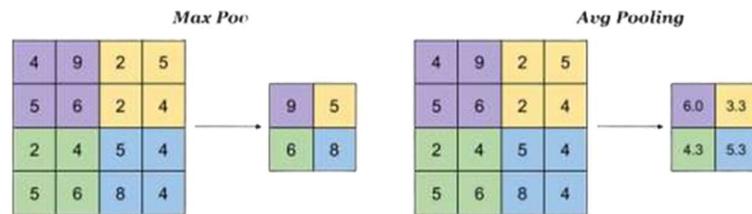
Gambar 2. 5 Operasi 'dot' antara input filter

2.9.2 Rectified Linear Unit (ReLU) activation

ReLU adalah fungsi aktivasi yang diperkenalkan oleh Richard HR Hahnloser, yang memiliki dasar biologis dan matematika yang kuat. Pada 2011, hal itu ditunjukkan untuk lebih meningkatkan pelatihan jaringan saraf yang dalam. Ia bekerja dengan menetapkan nilai pada 0, yaitu $f(x) = \max(0, x)$. Sederhananya, output 0 ketika $x < 0$, dan sebaliknya, output fungsi linear ketika $x \geq 0$. ReLU secara konvensional digunakan sebagai fungsi aktivasi untuk neural network, dengan softmax menjadi fungsi klasifikasinya di akhir node. Kemudian, jaringan tersebut menggunakan fungsi cross-entropy softmax untuk mempelajari parameter berat θ dari jaringan saraf (Agarap, 2018).

2.9.3 Pooling Layer

Pooling layer merupakan bagian dari feature learning pada CNN yang berperan untuk mengurangi dimensi dari matriks hasil konvolusi (*convolved feature*). Tujuan dari penggunaan *pooling layer* adalah untuk mempercepat proses komputasi. Hal ini dapat terjadi karena setelah melewati *pooling layer*, parameter yang harus di-*update* semakin sedikit, sehingga resiko terjadinya overfitting semakin minim. Sama seperti pada *convolution layer*, dan *pooling layer* juga memiliki filter dengan ukuran tertentu yang akan melakukan proses *sliding window* terhadap matriks input. Terdapat dua jenis pooling yang sering digunakan, yakni *max pooling* dan *average pooling* (Ramba, 2020).



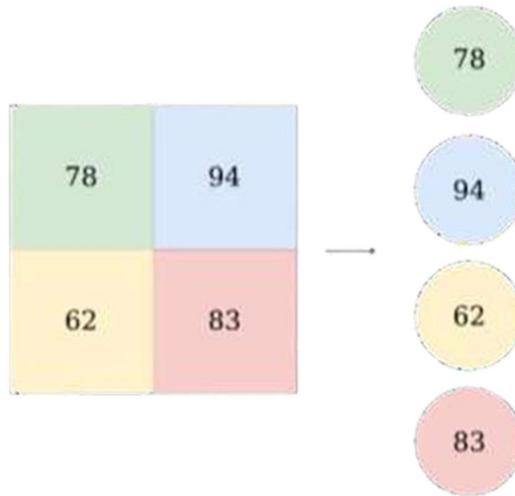
Gambar 2. 6 Perbedaan Max dan Arg Pool

2.10 Classification

Proses *classification* berfungsi untuk melakukan klasifikasi terhadap setiap neuron yang telah di ekstraksi pada proses feature learning. Bagian ini terdiri dari beberapa layer yang saling berkaitan satu sama lain. Berikut ini adalah penjelasan setiap fungsi pada bagian classification.

2.10.1 Flatten

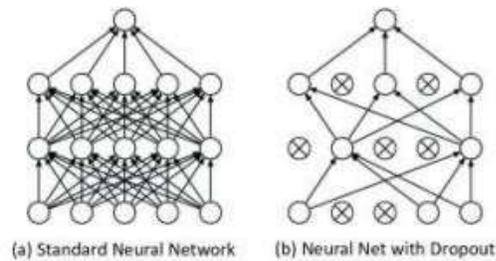
Flatten berfungsi untuk membentuk ulang (*reshape*) feature map dari *multidimensional array* menjadi vektor. Hal ini diperlukan agar nilai- nilai tersebut dapat digunakan sebagai input pada fully connected layer (Ramba, 2020).



Gambar 2. 7 Proses reshape feature map pada flatten

2.10.2 *Dropout*

Dropout adalah teknik regularisasi neural network dimana beberapa neuron akan dipilih secara acak dan tidak dipakai dalam proses training. Dengan menghilangkan suatu neuron.

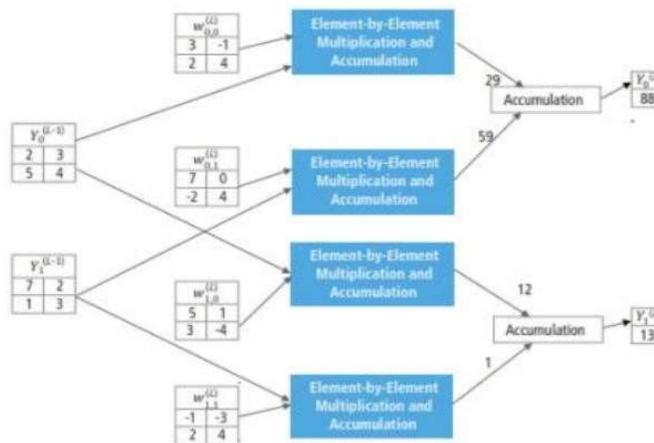


Gambar 2. 8 Regulasi Dropout

Pada bagian a merupakan jaringan syaraf biasa yang memiliki dua *hidden layer*. Sedangkan pada bagian b merupakan jaringan syaraf dengan menggunakan dropout. Dari gambar tersebut terlihat terdapat beberapa neuron aktivasi yang tidak dipakai lagi.

2.10.3 *Fully Connected Layer*

Fully connected layer merupakan *feed forward neural networks* yang terdiri dari *hidden layer*, *activation function*, *ouput layer*, dan *loss function*.



Gambar 2. 9 Proses Fully Conected Layer

Pada dasarnya lapisan ini biasanya digunakan pada MLP (Multi Layer Perceptron) yang mempunyai tujuan untuk melakukan transformasi pada dimensi data agar data dapat diklasifikasikan secara linier.

2.10.4 *Softmax*

Softmax adalah fungsi aktivasi yang mengubah angka alias log menjadi probabilitas yang berjumlah satu. Ini juga merupakan elemen inti yang digunakan dalam tugas klasifikasi di *Deep Learning*. *Softmax* adalah fungsi aktivasi yang merupakan bagian dari ReLU dan Sigmoid. Ini sering digunakan dalam klasifikasi. Output softmax besar jika skor (input disebut logit) besar. Outputnya kecil jika nilainya kecil. Proporsi tidak seragam. Softmax bersifat eksponensial, dapat memperbesar perbedaan sampai mendorong satu hasil lebih dekat ke 1 sementara yang lain lebih dekat ke 0. Ternyata skor alias log menjadi probabilitas. *Cross entropy* (fungsi biaya) sering dihitung untuk output *Softmax* label *softmax* dan label sebenarnya dengan (*One Hot Encoding*). (Uniqtech, 2018). *Softmax* digunakan untuk mendapatkan hasil klasifikasi. Nilai softmax berada pada interval $[0 - 1]$ dan memiliki jumlah 1 jika seluruh elemennya dijumlahkan. Fungsi ini biasanya digunakan diujung lapisan dari fully connected untuk klasifikasi lebih dari dua kelas yang digunakan pada CNN untuk mendapat nilai probabilitas suatu objek terhadap kelas yang ada (Wulandari et al., 2019).