

BAB II

LANDASAN TEORI

2.1 State of The Art

Berdasarkan topik permasalahan yang telah dirumuskan maka dibuatlah *literature review* dari jurnal penelitian sebelumnya yang berkaitan dengan pengenalan ASL menggunakan CNN *Hand Keypoint*. Hasil dari *literature review* yang telah dibuat dapat dilihat pada lampiran 1.

Penelitian mengenai pengenalan sistem bahasa isyarat telah dilakukan pada tahun 1999 oleh Vogler and Metaxas dengan judul *Parallel Hidden Markov Models for American Sign Language Recognition*. Penelitian yang dilakukan Pugeault & Bowden, R berjudul *Spelling It Out: Real-Time ASL Fingerspelling Recognition* berkontribusi membuat dataset ASL, selanjutnya dataset ASL tersebut di dipakai oleh (Tao, Leu, and Yin 2018) dalam penelitiannya dengan judul *American Sign Language alphabet recognition using Convolutional Neural Networks with multiview augmentation and inference fusion*. selain pengenalan bahasa isyarat Amerika telah dilakukan penelitian pengenalan bahasa lainnya seperti bahasa india oleh (Rao et al. 2018) *Deep Convolutional Neural Networks for Sign Language Recognition* dan bahasa argentina *Real-Time Sign Language Gesture (Word) Recognition from Video Sequences Using CNN and RNN* (Masood et al. 2018). Matriks penelitian terkait pengenalan bahasa isyarat dapat dilihat pada tabel 2.1.

Tabel 2.2 berisi tentang penelitian terkait tentang hand keypoint. *On Geometric Features for Skeleton-Based Action Recognition using Multilayer*

LSTM Networks (Zhang, Liu, and Xiao 2017) menerapkan model LSTM sebagai pengenalan bahasa isyarat. Selain LSTM dalam pengenala bahasa isyarat bisa juga menggunakan CNN seperti yang dilakukan oleh (Cao et al. 2017) *Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields* dan *A Deep learning Approach for Analyzing Video and Skeletal Features in Sign Language Recognition*. (Konstantinidis, Dimitropoulos, and Daras 2018a). *Sign Language Recognition Based on Hand and Body Skeletal Data* melanjutkan penelitian sebelumnya dengan menggabungkan model CNN dengan LSTM sehingga lebih akurat dalam pengenalan bahasa isyarat pada video yang berurutan(Konstantinidis, Dimitropoulos, and Daras 2018). (Suharjito, Thiracitta, and Gunawan 2021) yang berjudul *SIBI Sign Language Recognition Using Convolutional Neural Network Combined with Parameters Transfer Learning and non-trainable Parameter*. Penelitian ini menggunakan *convolutional neural network* dengan *transfer learning* dari microsoft kinect dengan hasil akurasi rata-rata 80%. Sistem Pengenalan Bahasa Isyarat Indonesia dengan Menggunakan Metode Fuzzy K-Nearest Neighbor(Gafar and Sari 2018) dengan akurasi rata-rata 88.8%. Pengenalan Bahasa Isyarat Indonesia (SIBI) Menggunakan *Leap Motion Controller* dan Algoritma Data Mining Naïve Bayes(Ridwang 2017). *Construction of SIBI Datasets for Sign Language Recognition Using a Webcam*(Kuswardhana et al. 2021). Matriks penelitian terkait pengenalan bahasa isyarat dapat dilihat pada tabel 2.1.

Pada tabel 2.3 yang berisi tentang penelitian terkait dengan penggunaan dataset Massey. Dataset Massey merupakan dataset yang dibuat oleh Barczak

pada penelitiannya yang berjudul *A New 2D Static Hand Gesture Colour Image Dataset for ASL Gestures*. Penelitian yang dilakukan oleh Bheda dan Radpour dengan judul *Using Deep Convolutional Neural Network for Gesture Recognition in American Sign Language* mengimplementasikan model deep convolutional neural network (deepCNN) sebagai klasifikasi pengenalan bahasa isyarat Amerika dan menggunakan dataset Massey memperoleh akurasi sebesar 82.5%.

Tabel 2. 1 Matrxs Penelitian Terkait Pengenalan Bahasa Isyarat

No.	Penulis/Tahun	Judul	Model/Algoritma	Kontribusi	Dataset
1.	Nikam dan Ambekar (2016)	Sign Language Recognition Using Image Based Hand Gesture Recognition Techniques	hull convexity	Mengimplementasikan algoritma hull convexity untuk tracking hand gesture	-
2.	Von Agris, et. Al (2008)	Rapid Signer Adaptation for Continuous Sign Language Recognition Using a Combined Approach of Eigenvoices, MLLR, and MAP	EV + MLLR dan MAP	Menggabungkan pendekatan EV + MLLR dan MAP yang menghasilkan tingkat akurasi yang besar	Video corpus
3	Kai-Yin Fox, et. Al (2015)	A Real-Time ASL Recognition System Using Leap Motion Sensors	multi sensor data fusion (MSDF) + HMM	Menggabungkan multi sensor data fusion (MSDF) dengan HMM	Dari sensor Leap Motion
4.	Ananda Rao, et. Al (2018)	Deep Convolutional Neural Networks for Sign Language Recognition	deepCNN	Penerapan model CNN pada Bahasa isyarat india	Indian sign language (ISL)
5	Nicolas Pugeault & Richard	Spelling It Out: Real-Time ASL Fingerspelling Recognition	Multi-class random forest	Membuat dataset ASL dan kalsifikasi pengenalan bahasa	Dataset ASL yang dibuat

No.	Penulis/Tahun	Judul	Model/Algoritma	Kontribusi	Dataset
	Bowden(2011)		classification	isyarat Amerika	peneliti
6.	Wenjin Tao, et. Al(2018)	American Sign Language alphabet recognition using Convolutional Neural Networks with multiview augmentation and inference fusion	CNN + 3D moddan Kinel-based algorithm	Memberikan gambaran tentang recognition menggunakan Multiview dan 3D model-based algorithms dan perbandingan nilai akurasi Nicolas Pugeault & Richard Bowden(2011)	Alfabet dataset ASL Nicolas Pugeault & Richard Bowden(2011) dan NTU digit dataset
7.	Rabet Fatmi, et. Al(2019)	Comparing ANN, SVM, and HMM based <i>Machine learning</i> Methods for American Sign Language Recognition using Wearable Motion Sensors	ANN + SVM, and HMM	Membandingkan algoritma ANN + SVM dengan HMM dalam pengenalan bahasa isyarat Amerika	Dataset ASL

No.	Penulis/Tahun	Judul	Model/Algoritma	Kontribusi	Dataset
8	Christian Vogler and Dimitris Metaxas(1999)	Parallel Hidden Markov Models for American Sign Language Recognition	HMM dan PaHMM	Membuat perbandingan dan modifikasi metode HMM	Dataset dibuat oleh peneliti
9.	Sarfaraz Masood dkk(2018)	Real-Time Sign Language Gesture (Word) Recognition from Video Sequences Using CNN and RNN	CNN + RNN	menyajikan sistem berbasis vision untuk menafsirkan gerakan tangan yang terisolasi dari Bahasa Isyarat Argentina.	Dataset bahasa isyarat Argentina LSA64
10	Suharjito dkk (2020)	SIBI Sign Language Recognition Using Convolutional Neural Network Combined with Parameters Transfer Learning and non-trainable Parameter	Convolutional neural network CNN inflated 3D dengan transfer learning dari microsoft kinect.	Menerapkan model convolutional neural network (CNN) inflated 3D dengan metode transfer learning dari ImageNet dan Kinect dataset untuk mengatasi masalah data yang sedikit.	Dataset dibuat oleh peneliti

No.	Penulis/Tahun	Judul	Model/Algoritma	Kontribusi	Dataset
11	Agum Agidrama Gafar dan Jayanti Yusmah Sari(2017)	Sistem Pengenalan Bahasa Isyarat Indonesia dengan Menggunakan Metode Fuzzy K-Nearest Neighbor	Fuzzy K-Nearest Neighbor	Membuat sistem pengenalan SIBI sekaligus menjadi media pembelajaran untuk menjembatani tunarungu dan orang normal dalam berkomunikasi.	Dataset dibuat oleh peneliti
12	Ridwan(2017)	Pengenalan Bahasa Isyarat Indonesia (SIBI) Menggunakan Leap Motion Controller dan Algoritma Data Mining Naïve Bayes	Naïve Bayes + leap motion controller	Menggunakan Leap Motion Controller untuk pengenalan SIBI	Dataset dibuat oleh peneliti
13	Dandhi Kuswardhana dkk(2020)	Construction of SIBI Datasets for Sign Language Recognition Using a Webcam	HSV (Hue Saturation Value)	membuat dataset SIBI melalui gerakan tangan menggunakan metode eksperimental.	Dataset dibuat oleh peneliti

No.	Penulis/Tahun	Judul	Model/Algoritma	Kontribusi	Dataset
14	Mochamad Bagus Setiyo Bakti dan Yuliana Melita Pranoto(2019)	Pengenalan Angka Sistem Isyarat Bahasa Indonesia Dengan Menggunakan Metode Convolutional Neural Network	LeNet CNN	Memperkenalkan arsitektur LeNet dari CNN sebagai pengenalan SIBI	Dataset digit dibuat oleh peneliti

Tabel 2. 2 Matriks Penelitian Terkait *Hand Keypoint*

No	Penulis/Tahun	Judul	Model/Algoritma	Kontribusi	Dataset
1	Songyang Zhang dkk(2017)	On Geometric Features for Skeleton-Based Action Recognition using Multilayer LSTM Networks	LSTM 3-lapis	merangkum evolusi pekerjaan sebelumnya pada pengenalan aksi 3D berbasis RNN menggunakan kerangka dan berhipotesis bahwa menjelajahi hubungan di	Dataset NTU-RGB+D dataset, SBU-Kinect dataset, UT-Kinect dataset, and Berkeley MHAD dataset.

				antara semua sendi dapat menghasilkan kinerja yang lebih baik.	
2	Zhe Cao dkk(2017)	Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields	CNN	mengakui upaya dari penulis MPII dan COCO dataset human pose. Kumpulan data ini memungkinkan estimasi pose pada manusia	Dataset MPII dan COCO human pose datasets
3	Tomas Simon dkk(2017)	<i>Hand Keypoint</i> Detection in Single Images using Multiview Bootstrapping	Convolutional Pose Machines (CPM) + Multiview Bootstrapping	menyajikan pendekatan untuk meningkatkan kinerja detektor keypoint yang diberikan menggunakan pengaturan multi-kamera. Pendekatan ini, yang disebut sebagai bootstrap multiview.	Dataset MPII
4	Dimitrios Konstantinidis	<i>A Deep learning</i> Approach for Analyzing Video and	VGG16 CNN	menyajikan metodologi pengenalan bahasa isyarat	Dataset LSA64 dan RWTH-PHOENIX-

	dkk(2018)	Skeletal Features in Sign Language Recognition		baru yang menjembatani kesenjangan antara pengukuran langsung dan pendekatan berbasis vision, sehingga memanfaatkan kedua metode dan mengatasi keterbatasannya	Database cuaca.
5	Dimitrios Konstantinidis dkk(2018)	Sign Language Recognition Based on Hand and Body Skeletal Data	VGG-19, deppCNN dan LSTM	mengusulkan metodologi berbasis deeplearning yang akurat dan kuat untuk pengenalan bahasa isyarat dari video sequences.	Dataset LSA64

Tabel 2. 3 Matriks Penelitian Terkait Pengenalan Isyarat menggunakan dataset Massey Barczak dkk (2011)

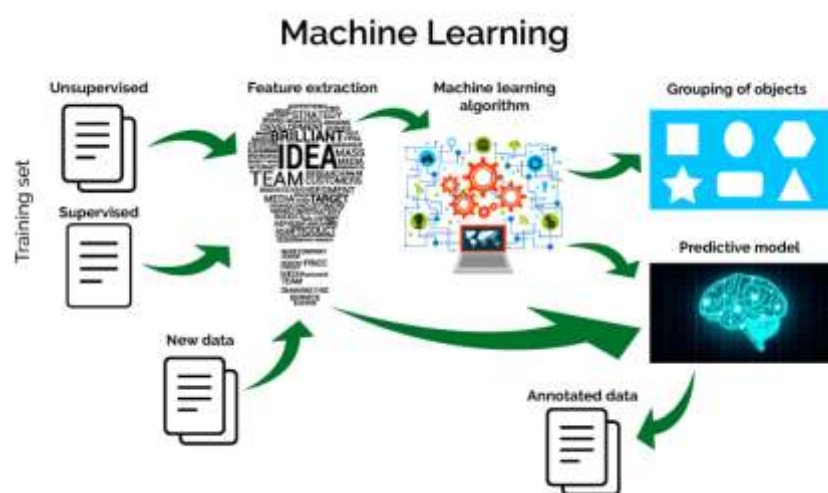
No	Penulis/Tahun	Judul	Model/Algoritma	Kontribusi	Dataset
1	A.L.C. Barczak dkk(2011)	A New 2D Static Hand Gesture Colour Image Dataset for ASL Gestures	Merekam gerakan tangan ASL standar dari sukarelawan	Membuat dataset bahasa isyarat Amerika standar	Dataset dibuat peneliti Massey
2	Yong Soon Tan(2020)	Convolutional neural network with spatial pyramid pooling for hand gesture recognition	CNN-SPP	Mengenalkan model convolutional neural network (CNN) terintegrasi dengan spasial piramida pooling (SPP), dijuluki CNN-SPP, untuk pengenalan gerakan tangan berbasis vision.	Massey Barczak dkk (2011), NUS hand gesture datase dan ASL finger spelling dataset
3	Murat Taskiran(2018)	A Real-Time System For Recognition Of American Sign Language By Using Deep Learning	CNN+ Skin Detection dan Convex Hull	Perbandingan nilai akurasi metode yang diusulkan dengan penelitis sebelumnya.	Massey Barczak dkk (2011)

4	Pulkit Rathi dkk(2019)	Sign Language Recognition Using ResNet50 Deep Neural Network Architecture	ResNet50 Deep Neural Network	merancang Arsitektur Jaringan Syaraf Dalam 2 tingkat berbasis ResNet50 baru untuk mengklasifikasikan kata- kata yang dieja jari	Massey Barczak dkk (2011)
5	Garcia dan Viesca (2016)	Real-time American Sign Language Recognition with Convolutional Neural Networks	Standford deepCNN + <i>pre- trained</i> GoogLeNet	Menggunakan pretrained GoogLeNet pada model deepCNN untuk pengenalan ASL	Massey Barczak dkk (2011),
6	Vivek Bheda and Dianna Radpour(2017)	Using Deep Convolutional Networks for Gesture Recognition in American Sign Language	deepCNN	Menormalisasi input dataset untuk model yang diajukan (deepCNN)	Massey Barczak dkk (2011),

2.2 Tinjauan Pustaka

2.1.1 Machine Learning

Machine learning merupakan serangkaian teknik yang dapat membantu dalam menangani dan memprediksi data yang sangat besar dengan cara mempresentasikan data-data tersebut dengan algoritma pembelajaran (Danukusumo, Pranowo, and Maslim 2017).



Gambar 2. 1 *Machine learning* (Pantech, 2018)

Istilah *machine learning* pertama kali didefinisikan oleh Arthur Samuel pada tahun 1959. Menurut Arthur Samuel, *machine learning* adalah suatu bidang ilmu komputer yang memberikan kemampuan pembelajaran kepada komputer untuk mengetahui sesuatu tanpa pemrograman yang jelas. *machine learning* dapat didefinisikan sebagai metode komputasi berdasarkan pengalaman untuk meningkatkan performa atau membuat prediksi yang akurat. Definisi pengalaman disini ialah informasi sebelumnya yang telah tersedia dan bisa dijadikan data pembelajar (Mehryar Mohri 2018)

Menurut Suyanto pada bukunya yang berjudul *Machine learning* tingkat dasar dan lanjut, *Machine learning* dapat dikelompokkan menjadi enam kelompok yaitu:

1. *Supervised Learning*

Algoritma ini membangkitkan suatu fungsi yang memetakan *input* ke *output* yang diinginkan. Kualitas hasil pembelajaran sangat bergantung pada kesesuaian *input* ke *output* yang diberikan. Algoritma jenis ini disebut pembelajaran terawasi.

2. *Unsupervised Learning*

Algoritma ini memodelkan sekumpulan *input* secara otomatis tanpa ada panduan (berupa *output* yang diinginkan). Artinya, data-data yang dipelajari hanya berupa *input* tanpa label kelas. Algoritma ini biasanya digunakan untuk masalah klusterisasi(*clustering*).

3. *Reinforcement Learning*

Algoritma ini mempelajari suatu kebijakan bagaimana melakukan aksi berdasarkan hasil pengamatan terhadap lingkungan yang ada. Setiap aksi menghasilkan akibat bagi lingkungan tersebut, dan lingkungan memberikan umpan balik untuk memandu algoritma tersebut.

4. *Transduction*

Algoritma ini mirip dengan *supervised learning*, tetapi tidak secara eksplisit membangun suatu fungsi. Algoritma ini justru berlatih

memproduksi *output* baru berdasarkan *training inputs*, *training output*, dan *testing output* yang tersedia selama proses pembelajaran.

5. *Learning to Learn*

Algoritma ini mempelajari bias induktifnya sendiri berdasarkan pengalaman-pengalaman sebelumnya.

2.1.2 Deep learning

Deep learning merupakan salah satu bidang dari *machine learning* yang memanfaatkan jaringan syaraf tiruan untuk implementasi permasalahan dengan dataset yang besar. Teknik *deep learning* memberikan arsitektur yang sangat kuat untuk supervised learning. Dengan menambahkan lebih banyak lapisan maka model pembelajaran tersebut bisa mewakili data citra berlabel dengan lebih baik. Pada *machine learning* terdapat teknik untuk menggunakan ekstraksi fitur dari data pelatihan dan algoritma pembelajaran khusus untuk mengklasifikasi citra maupun untuk mengenali suara. Namun, metode ini masih memiliki beberapa kekurangan baik dalam hal kecepatan dan akurasi.

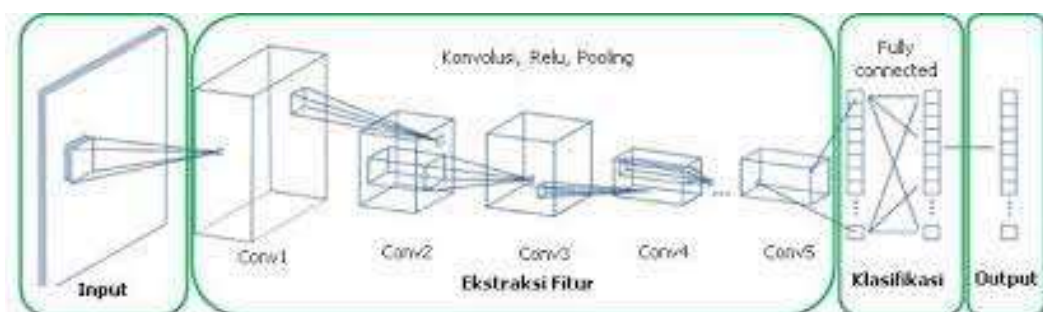
Aplikasi konsep jaringan syaraf tiruan yang dalam (banyak lapisan) dapat ditanggihkan pada algoritma *machine learning* yang sudah ada sehingga komputer sekarang bisa belajar dengan kecepatan, akurasi, dan skala yang besar. Prinsip ini terus berkembang hingga *deep learning* semakin sering digunakan pada komunitas riset dan industri untuk membantu memecahkan banyak masalah data besar seperti *Computer vision*, *Speech recognition*, dan *Natural Language Processing*. *Feature*

Engineering adalah salah satu fitur utama dari *deep learning* untuk mengekstrak pola yang berguna dari data yang akan memudahkan model untuk membedakan kelas. *Feature Engineering* juga merupakan teknik yang paling penting untuk mencapai hasil yang baik pada tugas prediksi. Namun, sulit untuk dipelajari dan dikuasai karena kumpulan data dan jenis data yang berbeda memerlukan pendekatan teknik yang berbeda juga. Dalam *deep learning*, metode CNN atau *Convolutional Neural Network* sangatlah bagus dalam menemukan fitur yang baik pada citra ke lapisan berikutnya untuk membentuk hipotesis nonlinier yang dapat meningkatkan kekompleksitasan sebuah model. Model yang kompleks tentunya akan membutuhkan waktu pelatihan yang lama sehingga di dunia *deep learning* penggunaan GPU sudah sangatlah umum (Danukusumo, Pranowo, and Maslim 2017).

2.1.3 Convolutional Neural Network

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 (Hubel and Wiesel 1962) mengenai

visual 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*. Berikut adalah jaringan arsitektur *Convolutional Neural Network*.

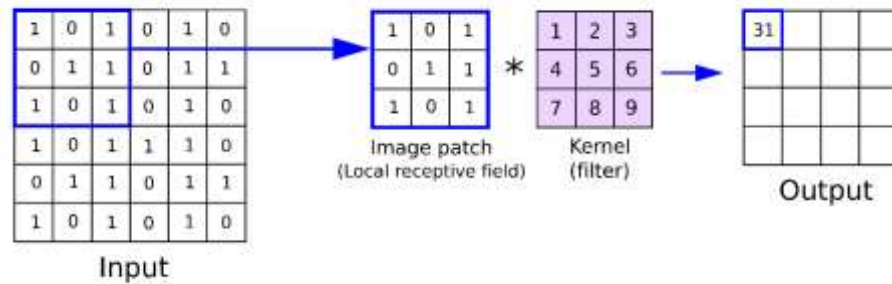


Gambar 2. 2 Arsitektur *machine learning* (krizhevsky dkk 2012)

Berdasarkan gambar 2.2, Tahap pertama pada arsitektur CNN adalah tahap konvolusi. Tahap ini dilakukan dengan menggunakan sebuah kernel dengan ukuran tertentu. Perhitungan jumlah kernel yang dipakai tergantung dari jumlah fitur yang dihasilkan. Kemudian dilanjutkan menuju fungsi aktivasi, biasanya menggunakan fungsi aktivasi ReLU (Rectifier Linear Unit), Selanjutnya setelah keluar dari proses fungsi aktivasi kemudian melalui proses pooling. Proses ini diulang beberapa kali sampai didapatkan peta fitur yang cukup untuk dilanjutkan ke *fully connected neural network*, dan dari *fully connected network* adalah *output class*.

A. Convolution layer

Convolution layer merupakan bagian dari tahap pada arsitektur CNN. Tahap ini melakukan operasi konvolusi pada *output* dari *layer* sebelumnya. *Layer* tersebut adalah proses utama yang mendasari jaringan arsitektur CNN. Konvolusi adalah istilah matematis dimana pengaplikasian sebuah fungsi pada *output* fungsi lain secara berulang. Operasi konvolusi merupakan operasi pada dua fungsi argumen bernilai nyata. Operasi ini menerapkan fungsi *output* sebagai *feature map* dari *input* citra. *Input* dan *output* ini dapat dilihat sebagai dua argumen bernilai riil. *Convolutional Layer* terdiri dari neuron yang tersusun sedemikian rupa sehingga membentuk sebuah filter dengan panjang dan tinggi (*pixels*). Sebagai contoh, *layer* pertama pada *feature extraction layer* biasanya adalah conv. *Layer* dengan ukuran 5x5x3. Panjang 5 *pixels*, tinggi 5 *pixels* dan tebal/jumlah 3 buah sesuai dengan channel dari *image* tersebut. Ketiga filter ini akan digeser keseluruh bagian dari gambar. Setiap pergeseran akan dilakukan operasi “dot” antara *input* dan nilai dari filter tersebut sehingga menghasilkan sebuah *output* atau biasa disebut sebagai activation map atau *feature map*. Perhatikan ilustrasi berikut:



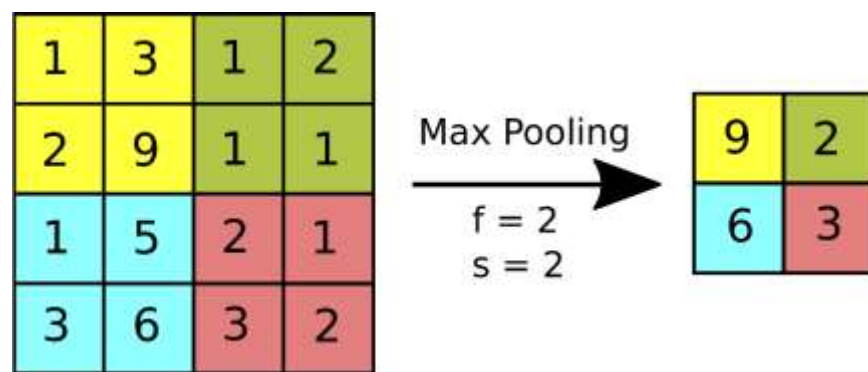
Gambar 2. 3 *convolution layer* (Reynolds. A.H., 2019)

B. Pooling

Pooling merupakan pengurangan ukuran matriks dengan menggunakan operasi *pooling*. *Pooling layer* biasanya berada setelah lapisan konvolusi. Pada dasarnya *pooling layer* terdiri dari sebuah filter dengan ukuran dan stride tertentu yang akan secara bergantian bergeser pada seluruh area *feature map*. Dalam *pooling layer* terdapat dua macam *pooling* yang biasa digunakan yaitu *average pooling* dan *max-pooling*. Nilai yang diambil pada *average pooling* adalah nilai rata-rata, sedangkan pada *max-pooling* adalah nilai maksimal. Lapisan *Pooling* yang dimasukkan diantara lapisan konvolusi secara berturut-turut dalam arsitektur model CNN dapat secara progresif mengurangi ukuran volume *output* pada *feature map*, sehingga mengurangi jumlah parameter dan perhitungan di jaringan, untuk mengendalikan *overfitting*. Lapisan *pooling* bekerja di setiap tumpukan *feature map* dan melakukan pengurangan pada ukurannya. Bentuk lapisan *pooling* umumnya dengan menggunakan

filter dengan ukuran 2x2 yang diaplikasikan dengan langkah sebanyak dua dan beroperasi pada setiap irisan dari *inputnya*.

Berikut ini adalah contoh gambar operasi max-pooling :



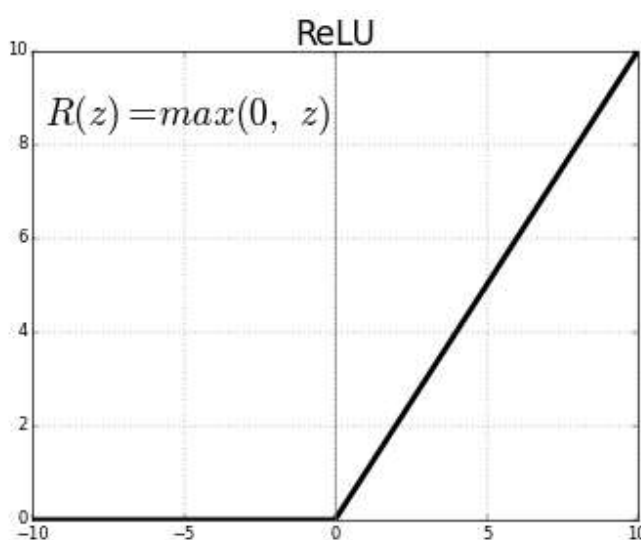
Gambar 2. 4 Max Pooling (Reynolds. A.H., 2019)

Gambar 2.4 menunjukkan proses dari max-pooling. *Output* dari proses pooling adalah sebuah matriks dengan dimensi yang lebih kecil dibandingkan dengan citra awal. Lapisan pooling diatas akan beroperasi pada setiap irisan kedalaman volume *input* secara bergantian. Jika dilihat dari gambar diatas operasi max-pooling dengan menggunakan ukuran filter 2x2. Masukan pada proses tersebut berukuran 4x4, dari masing-masing 4 angka pada *input* operasi tersebut diambil nilai maksimalnya kemudian dilanjutkan membuat ukuran *output* baru menjadi ukuran 2x2.

C. Relu Layer

Rectified Linear Units (ReLU) layer ini mengaplikasikan fungsi $f(x) = \max(0, x)$ untuk meningkatkan sifat nonlinearitas fungsi

keputusan dan jaringan secara keseluruhan tanpa mempengaruhi bidang-bidang reseptif pada *convolutional layer*. Berikut ini contoh operasi ReLU *layer*:

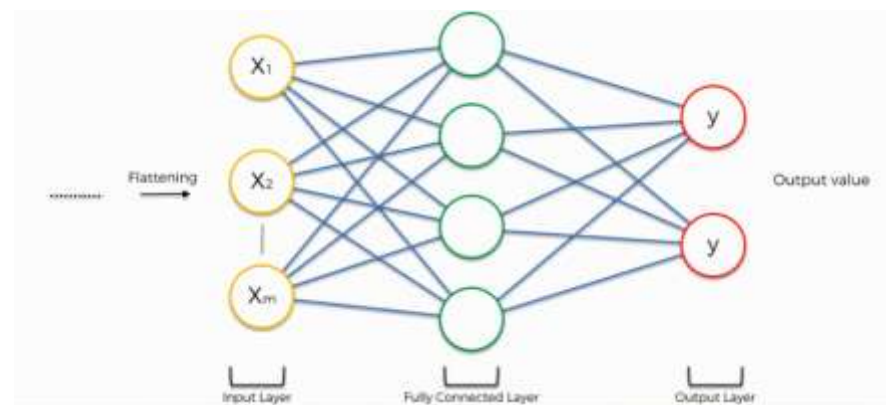


Gambar 2. 5 ReLU (Sena. S., 2017)

D. *Fully-Connected Layer*

Fully-Connected Layer adalah sebuah lapisan dimana semua neuron aktivasi dari lapisan sebelumnya terhubung semua dengan neuron di lapisan selanjutnya sama seperti halnya dengan neural network biasa. 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. Perbedaan antara lapisan *fully-connected* dan lapisan konvolusi biasa adalah neuron di lapisan konvolusi terhubung hanya ke daerah tertentu pada *input*, sementara

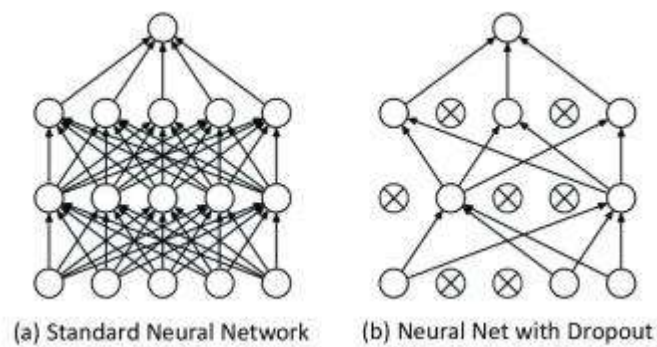
lapisan *fully-connected* memiliki neuron yang secara keseluruhan terhubung. Namun, kedua lapisan tersebut masih mengoperasikan produk dot, sehingga fungsinya tidak begitu berbeda. Berikut ini adalah proses *fully-connected*:



Gambar 2. 6 *Processing of Fully-Connected Layer*
(SuperDataScience Team 2018)

E. Dropout

Dropout merupakan sebuah teknik regulasi jaringan syaraf dengan tujuan memilih beberapa neuron secara acak dan tidak akan dipakai selama proses pelatihan, dengan kata lain neuron-neuron tersebut dibuang secara acak. Hal ini berarti bahwa kontribusi neuron yang dibuang akan diberhentikan sementara jaringan dan bobot baru juga tidak diterapkan pada neuron pada saat melakukan backpropagation (Srivastava et al. 2014). Berikut adalah gambar proses dropout:



Gambar 2. 7 *Dropout* (Srivastava et al 2014)

F. *Softmax Classifier*

Softmax Classifier adalah generalisasi dari fungsi logistik. *Output* dari softmax ini dapat digunakan untuk mewakili distribusi sebuah kategori. *Softmax function* digunakan dalam berbagai macam metode klasifikasi contohnya *multinomial logistic regression*, *multiclass linear discriminant analysis*, *naive Bayes classifier*, dan *neural network*. Secara spesifiknya fungsi ini biasa digunakan pada metode klasifikasi *multinomial logistic regression* dan *multiclass linear discriminant analysis*. Berikut adalah fungsi yang diberikan:

$$f_j(z) = \frac{e^{z_j}}{\sum_k e^{z_k}}$$

Notasi f_j menunjukkan hasil fungsi untuk setiap elemen ke- j pada vektor keluaran kelas. Argumen z adalah hipotesis yang diberikan oleh model pelatihan agar dapat diklasifikasi oleh fungsi softmax. Softmax juga memberikan hasil yang lebih intuitif dan juga memiliki

interpretasi probabilistik yang lebih baik dibanding algoritma klasifikasi lainnya. Softmax memungkinkan kita untuk menghitung probabilitas untuk semua label. Dari label yang ada akan diambil sebuah vektor nilai bernilai riil dan merubahnya menjadi vektor dengan nilai antara nol dan satuyang bila semua dijumlah akan bernilai satu.

2.1.4 Hand Keypoint

Deteksi *Hand Keypoint* adalah proses menemukan sendi pada jari serta ujung jari pada gambar yang diberikan. Ini mirip dengan menemukan titik kunci di Wajah (alias Deteksi *Landmark* Wajah) atau Tubuh (*Estimasi Pose* Tubuh Manusia), tetapi, berbeda dari Deteksi Tangan karena dalam kasus itu, kita memperlakukan seluruh tangan sebagai satu objek. Model menghasilkan 22 *keypoint*. Tangan memiliki 21 poin sedangkan poin ke-22 menandakan latar belakang. Poin-poinnya seperti gambar di bawah ini:



Gambar 2. 8 *hand keypoint* (Gupta. V, 2018)