

## BAB II

### LANDASAN TEORI

Berikut ini merupakan beberapa teori dan tinjauan pustaka yang digunakan dalam penelitian.

#### 2.1. Tinjauan Pustaka

(J. Deng et al., 2016) melakukan penelitian yang berjudul *M3 CSR: Multi-view, multi-scale and multi-component cascade shape regression*. Penelitian ini melakukan modifikasi pada *Cascade Shape Regression* dimana menambahkan *M3 (Multi-view, Multi-scale, Multi-component)* untuk mendapatkan hasil *face alignment* yang lebih akurat. Hasil penelitian ini pada 300-W dataset yang diuji mendapatkan 4.39% *Normalize Mean Error* pada kondisi *indoor-outdoor* dengan *facial landmark 51* dan 5.45% *Normalize Mean Error* pada kondisi *indoor-outdoor* dengan *facial landmark 68*.

(Xiong & De La Torre, 2013) melakukan penelitian yang berjudul *Supervised Descent Method and its Applications to Face Alignment*. Penelitian ini melakukan minimalisasi fungsi *Non-linear Least Squares (NLS)* untuk mengatasi masalah tidak dapat membedakan secara analitis dan pendekatan numerik. Hasil penelitian ini mendapatkan *Normalize Error 3.47%* pada *facial landmark 17*.

(J. Yang et al., 2017) melakukan penelitian yang berjudul *Stacked Hourglass Network for Robust Facial Landmark Localisation*. Penelitian ini melakukan modifikasi pada *Hourglass Network* dengan menambahkan 2 langkah yaitu *supervised face transformation* untuk mengurangi variasi secara keseluruhan dan *deep convolution network* untuk membantu menentukan lokalisasi wajah akhir. Hasil dari penelitian ini menunjukkan *Failure Rate 1.8%* pada dataset COFW, 0.7% pada dataset IBUG dan 1% pada dataset 300W.

(Ma & Wang, 2019) melakukan penelitian yang berjudul *Multi-view Face Detection and Landmark Localization Based on MTCNN*. Penelitian ini melakukan deteksi wajah dari beberapa sudut pandang atau disebut *multi-view face detection*

and landmark localization. Metode yang dipakai yaitu *Multi-task Cascaded Convolutional Neural Network* (MTCNN).

(Yuen & Trivedi, 2017) melakukan penelitian yang berjudul *An Occluded Stacked Hourglass Approach to Facial Landmark Localization and Occlusion Estimation*. Penelitian ini melakukan pengamatan pada kegiatan pengemudi dengan wajah yang menjadi objeknya. Objek wajah diekstrak sehingga informasi seperti pose kepala, tingkat kedip, menguap berbicara dengan penumpang, yang kemudian dapat membantu memperoleh informasi tingkat yang lebih tinggi seperti gangguan, kantuk, dan lainnya. Hasil dari penelitian ini menunjukkan *Detection Rate* sebesar 93,2% dan *Success Rate* 93,4%.

(Uřičář et al., 2016) melakukan penelitian yang berjudul *Multi-View Facial Landmark Detector Learned by the Structured Output SVM*. Penelitian ini mengusulkan *multi-view landmark detector* secara *real-time* yang berbasis *Deformable Part Model*. *Detector* terdiri dari campuran DPM yang setiap komponennya mempunyai konfigurasi atau koordinatnya masing-masing dengan rentang sudut pandang tertentu. Evaluasi yang dilakukan pada dataset gambar “*in-the-wild*” menunjukkan usulan *detector* ini dapat bekerja dengan baik dengan metode *state-of-the-art* dalam hal akurasi dan kecepatan.

(Guo et al., 2019) melakukan penelitian yang berjudul *Multi-Scale Stacked Hourglass Network For Human Pose Estimation*. Penelitian ini melakukan perkiraan postur tubuh manusia dari informasi *keypoint* dan informasi lokal dari *local keypoint*. Hasil dari penelitian ini menunjukkan pada dataset MPII mendapatkan tingkat keberhasilan total yaitu 91.2% dan pada dataset LSP mendapatkan tingkat keberhasilan total 91.4%.

(Hong et al., 2019) melakukan penelitian yang berjudul *Facial Landmark Localization Based On Auto-Stacked Hourglass Network And Expectation Consensus*. Penelitian ini melakukan peningkatan model penyelarasan dengan *weighted heatmaps* untuk menghilangkan kesalahan kuantisasi dan dengan menerapkan ML otomatis untuk mencari struktur jaringan yang lebih baik. Hasil penelitian ini mendapatkan hasil AUC sebesar 84.01%

(Q. Liu et al., 2017), melakukan penelitian yang berjudul *Robust facial landmark tracking via cascade regression*. Penelitian ini dilakukan untuk mendeteksi wajah dengan metode *cascade regression*. Hasil dari penelitian ini mengatakan lokasi wajah dapat diketahui secara efektif dan akurat setelah melalui tahap *pose-based cascade shape regression model* dan *pose estimation model*.

(W. Deng et al., 2018) melakukan penelitian yang berjudul *Facial Landmark Localization by Enhanced Convolutional Neural Network*. Penelitian ini menggunakan *Enhanced Convolutional Neural Network* untuk mencari lokasi *Facial Landmark*. Hasilnya metode ECNN dapat memecahkan permasalahan penyelarasan wajah dengan 2 tahap yaitu dengan merekonstruksi peta respons pada setiap titik gambar dan pemetaan bentuk. Hasil model ini hanya mendapatkan tingkat kesalahan 4.38% pada subset umum dari dataset 300W.

(Lv et al., 2019) melakukan penelitian yang berjudul *3D facial expression modeling based on facial landmarks in single image*. Penelitian ini menggunakan *landmark* wajah, mengusulkan metode pemodelan ekspresi wajah 3D berdasarkan teori Kendall. Hasil pemodelan lebih akurat. Hasil pemodelan ekspresi wajah 3D nyaman diperoleh dari gambar wajah 2D dengan pose kepala yang berbeda.

(Tang et al., 2018a) melakukan penelitian yang berjudul *Facial landmark detection by semi-supervised deep learning*. Penelitian ini mengusulkan *semi-supervised facial landmark detection algorithm* (SEMI) yang berdasarkan *Convolutional Neural Network* (CNN), yang dapat mendeteksi komponen wajah dan landmark secara bersamaan. Hasil eksperimental pada Helen, LFPW dan 300-W dataset menunjukkan bahwa metode semi-supervised method kami melakukan lebih baik daripada banyak algoritma *state-of-the-art*. Hasil perbandingan yang lebih banyak antara pelatihan *semi-supervised* dan *supervised training schemes* menunjukkan keuntungan dari algoritma *semi-supervised* kami.

(Martinez & Valstar, 2016) melakukan penelitian yang berjudul *L2,1-based regression and prediction accumulation across views for robust facial landmark detection*. Penelitian ini mengusulkan metodologi baru untuk *facial landmark detection*. Dengan menggunakan representasi fitur yang dihasilkan dari

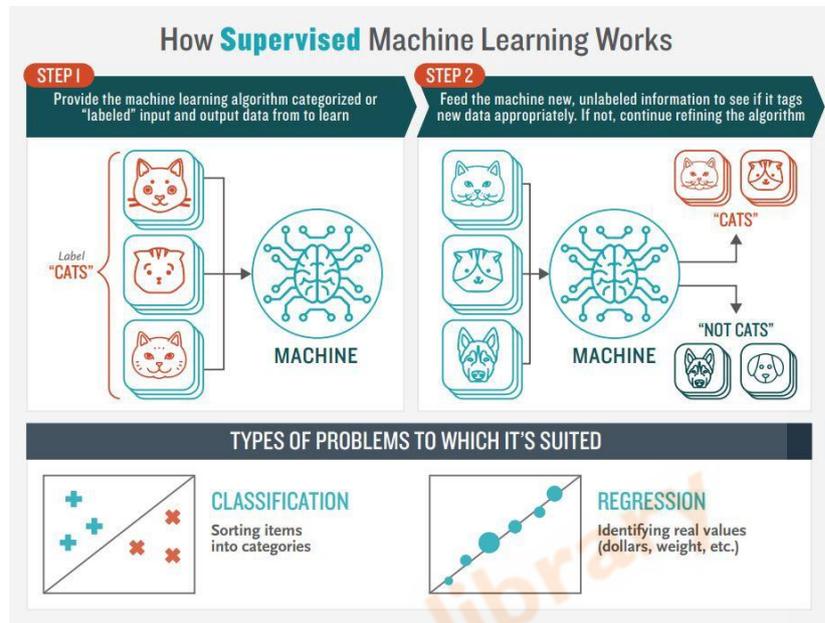
penggabungan 66 deskriptor HOG, satu per *landmark*. Namun, kami mengusulkan metode regresi baru yang menggantikan regresi *Least Squares* yang umum digunakan. Metode baru ini menggunakan norma L2,1, dan dirancang untuk meningkatkan kekokohan regresor terhadap inisialisasi yang buruk. Hasil percobaan mengkonfirmasi bahwa algoritma yang dihasilkan memang sangat kuat dan menghasilkan kinerja yang sangat baik.

Dalam meningkatkan akurasi *face detection* terdapat beberapa metode seperti yang dilakukan (J. Deng et al., 2016) yang memakai Metode *M3:CSR* pada penelitiannya didapatkan hasil *Normalize Mean Error* 5.45% dari data uji 596 citra. pada penelitian yang dilakukan (J. Deng et al., 2018) yang memakai metode *Cascade Multi-View Hourglass Model* mendapatkan *Normalize Mean Error* 5.52% dengan 2000 data uji. Pada kedua penelitian tersebut didapatkan hasil yang bagus dalam deteksi wajah dengan berbagai posisi citra wajah. Pada penelitian ini dilakukan pula peningkatan akurasi deteksi wajah dengan *proposed method* yang ditampilkan pada gambar 28. Dalam meningkatkan akurasi digunakan metode *Stacked Hourglass dan Multi-View*. Digunakan metode *Stacked Hourglass Network* ini karena pada metode dapat menangkap informasi pada setiap skala hingga ke ukuran terkecil dan menyatukan semua informasi tersebut sehingga didapat hasil keluaran yang dapat memprediksi kemungkinan *key-point* berada (B & B, 2016). Sedangkan metode *Multi-View* bekerja berdasarkan regresi sub ruang, dimana data latih dibagi menjadi tiga yaitu kanan, depan dan kiri. Kemudian dilatih berdasarkan *Cascade Shape Regression* model pada setiap *view*. Estimasi *view* ini berdasarkan *keypoint* yang menggunakan deteksi wajah dengan multi bentuk komponen sehingga dapat mengatasi variasi wajah. (J. Deng et al., 2016)

## 2.2. Supervised Learning

*Supervised learning* adalah sebuah pembelajaran pada *machine learning* dimana supervisornya sudah memiliki label pada tiap data nya. Label ini merupakan tag dari data yang telah ditambahkan dalam model *machine learning* atau data *machine learning*. Contoh algoritma *Machine learning* yaitu *classification* dan

*regression*. *Classification* dalam binatang seperti “kucing”, “burung”, dll. Sedangkan *regression* seperti berat badan, tinggi badan dsb.



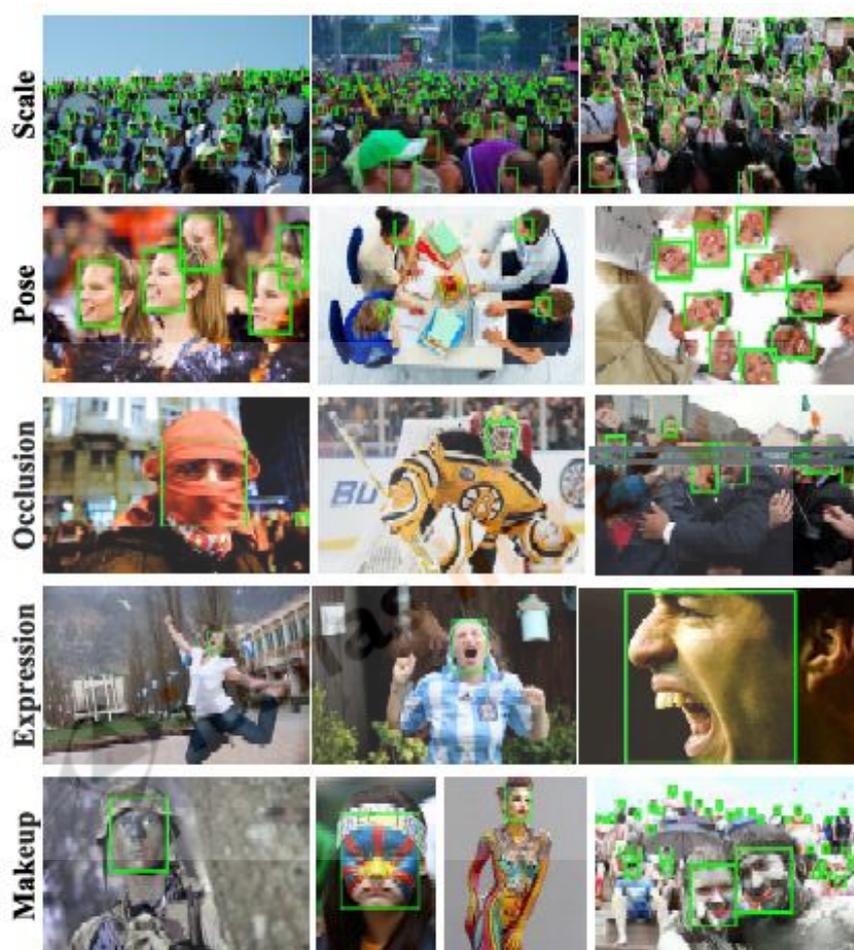
Gambar 2.1 *Supervised Learning*  
(sumber: medium.com)

### 2.3. *Face Detection*

*Face detection* atau Deteksi wajah adalah menentukan keberadaan wajah dalam gambar dan, jika ada, mencari lokasi gambar dan bagian masing-masing wajah. Deteksi wajah merupakan hal yang mudah bagi manusia, namun sangat sulit untuk komputer (S. Yang et al., 2016). Tujuan utama dari deteksi wajah adalah mengetahui semua posisi wajah pada gambar dan memberi tanda dengan kotak persegi. (Ma & Wang, 2019).

Tantangan yang terkait dengan deteksi wajah dapat dikaitkan dengan variasi dalam pose, skala, ekspresi wajah, oklusi atau halangan, dan kondisi pencahayaan, seperti yang ditunjukkan pada Gambar 2.2. Deteksi wajah telah membuat kemajuan yang signifikan setelah kontribusi dari Viola dan Jones. Detektor wajah modern dapat dengan mudah mendeteksi wajah *frontal* dan banyak digunakan dalam aplikasi dunia nyata, seperti kamera digital dan album foto elektronik. Penelitian

terbaru di bidang ini berfokus pada skenario yang sangat banyak, di mana sejumlah faktor rumit seperti pose ekstrem, wajah yang sangat ekspresif, dan sebagian besar oklusi atau halangan dapat menyebabkan variasi visual yang besar dalam penampilan wajah.



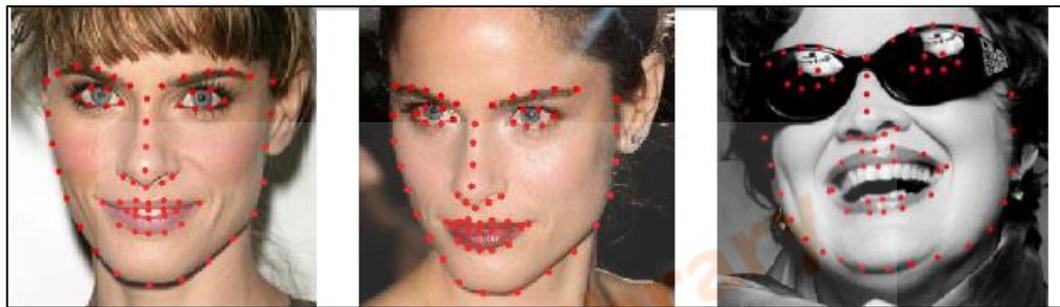
Gambar 2.2 Tantangan *Face Detection*

(Sumber : Yang et al., 2016)

#### 2.4. *Facial Landmark Detection*

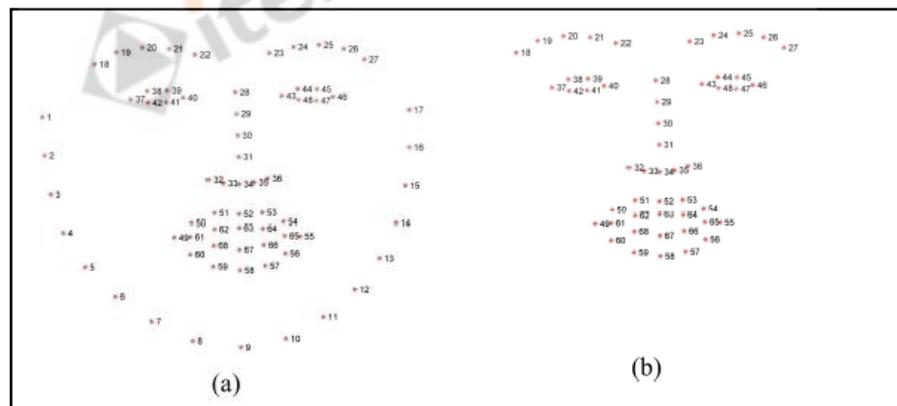
*Facial landmark* adalah titik wajah yang menonjol seperti pusat mata, ujung kebingangan, atau sudut mulut seperti yang ditunjukkan pada Gambar 2.3. Lokasi *facial landmarks* adalah masalah mendasar dalam banyak aplikasi *computer vision* tentang wajah, termasuk pengenalan wajah, pelacakan wajah, estimasi pose kepala, dan kartun. Metode ini telah banyak dipelajari secara luas selama dua dekade

terakhir. Namun banyak masalah yang muncul contohnya sangat sulit ketika wajah atau wajah *non-frontal* dengan perubahan ekspresi ekstrim diperhitungkan. Bahkan, dengan variasi dalam pose atau ekspresi, bentuk wajah mengalami deformasi nonlinier yang sulit dihadapi. *Facial landmark detection* mempunyai beberapa anotasi salah satunya yaitu dengan 106 point, 98 point, 68 point dan 51 point anotasi. Point anotasi ini berguna untuk menjadi ciri bagian wajah yang telah dideteksi. Ilustrasi point anotasi ini dapat dilihat pada gambar 2.4.



Gambar 2.3 Contoh *Facial Landmarks Detection*

(Sumber : Y. Chen, Yang, and Qian 2017)



Gambar 2.4 Contoh anotasi 68 dan 51 point *Facial Landmarks Detection*

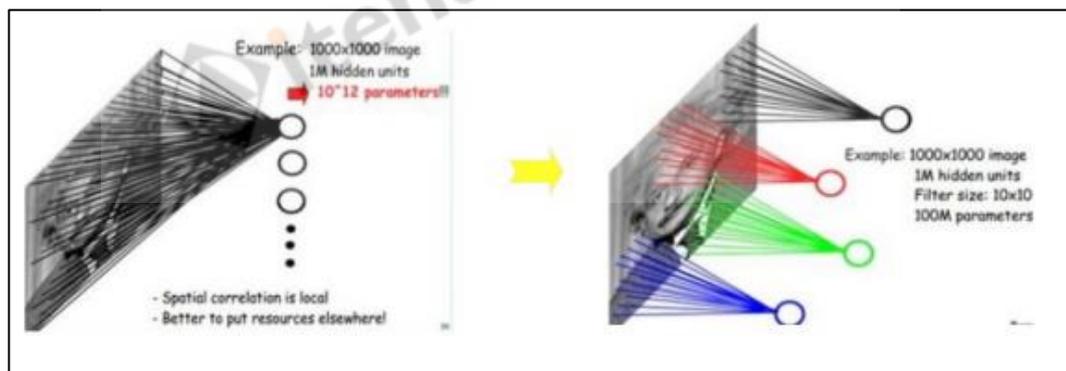
(Sumber :Sagonas et al. 2013)

## 2.5. *Convolutional Neural Network*

Algoritma *Convolutional Neural Network* atau lebih sering disebut CNN mempunyai struktur 2 lapisan yaitu *Feature Extraction* dan *Extracts Local Feature*. Lapisan *Feature extraction* merupakan input dari beberapa *neuron* yang saling

dihubungkan dari bidang reseptif local dari lapisan sebelumnya. Setelah *extracts local feature* atau fitur local diekstraksi, hubungan posisi antara fitur dan yang lainnya akan ditentukan. Lapisan peta fitur ini adalah komputasi dari setiap jaringan yang terdiri dari kemiripan antar peta fitur. Setiap peta fitur seperti pesawat mempunyai bobot yang serupa. Struktur dari lapisan peta fitur ini menggunakan fungsi sigmoid sebagai fungsi aktivasi dari *convolutional neural*, yang membuat peta fitur mempunyai pergeseran. Disamping itu, ketika *neuron* yang mempunyai peta fitur dengan bobot sama saling berbagi, angka parameter yang digunakan oleh jaringan akan dikurangi. Setiap lapisan konvolusi didalam *convolutional neural network* diikuti dengan lapisan perhitungan yang digunakan untuk menghitung rata-rata lokal dan ekstrak selanjutnya, 2 struktur ekstraksi fitur ini bersifat unik untuk mengurangi resolusi. (T. Liu et al., 2015)

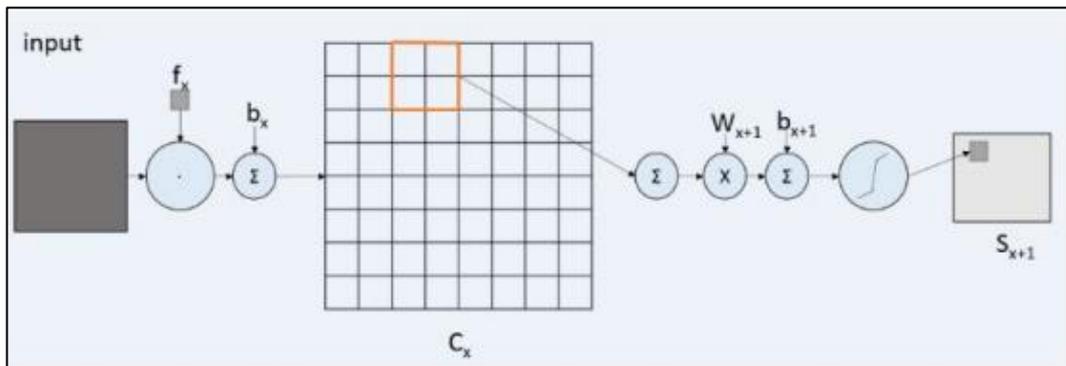
Secara metodologi algoritma CNN adalah *multilayer perceptron* yang digunakan untuk mengidentifikasi informasi dari gambar 2 dimensi. Mempunyai banyak lapisan: lapisan masukan, lapisan konvolusi, lapisan sample dan lapisan keluaran.



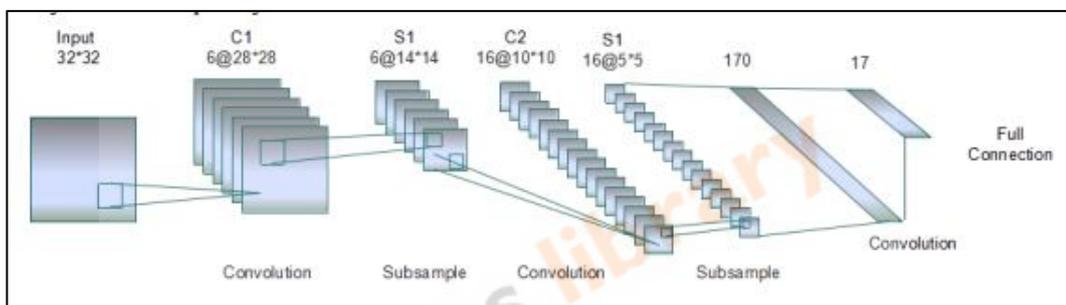
Gambar 2.5 Fully Connected dan Partial Connected

(Sumber: medium.com)

CNN mempunyai 2 proses utama yaitu konvolusi dan *sampling*. Proses konvolusi yaitu menggunakan filter  $f_x$  yang terlatih, *deconvolution* dari masukan gambar atau ini merupakan ekstraksi peta fitur, kemudian menambahkan bias  $b_x$  maka akan didapatkan lapisan konvolusi  $C_x$ . Kemudian proses *sampling* yaitu  $n_{piksel}$  dari setiap tetangga melalui langkah-langkah *pooling*, menjadi piksel.



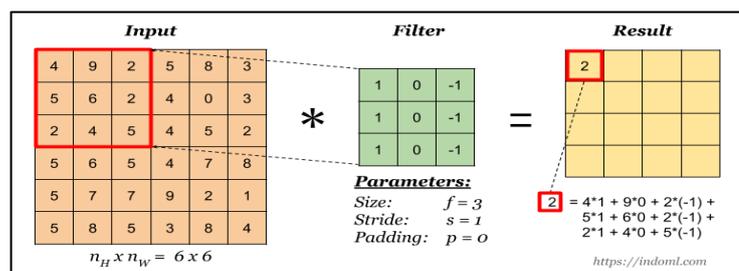
Gambar 2.6 Proses Utama CNN  
(Sumber : medium.com)



Gambar 2.7 Ilustrasi Konvolusi  
(Sumber: medium.com)

### 2.5.1. Convolution

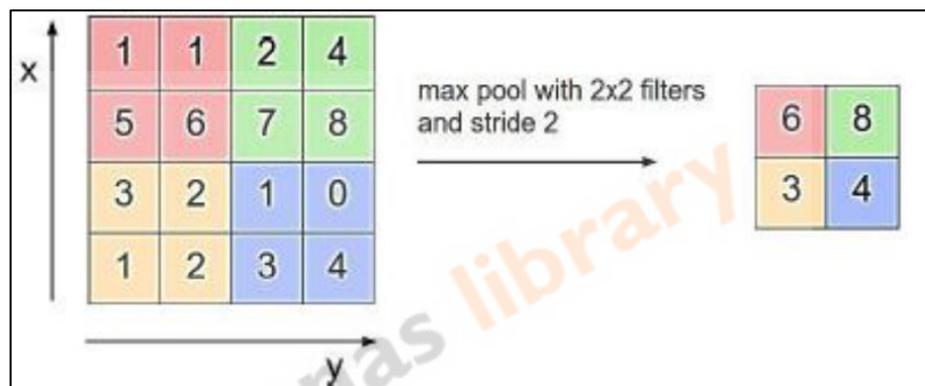
*Convolution* atau konvolusi adalah sebuah fungsi pada keluaran fungsi lain yang dilakukan berulang. Gambar 2.8 menunjukkan masukan, *filter* dan keluaran secara umum dari proses konvolusi pada suatu citra. *Filter* akan bergerak sesuai ukuran citra awal dan ukuran *filter*. Kemudian didapatkan keluaran dari hasil perhitungan *filter* dengan citra awal.



Gambar 2.8 Ilustrasi Konvolusi

### 2.5.2. Pooling Layer

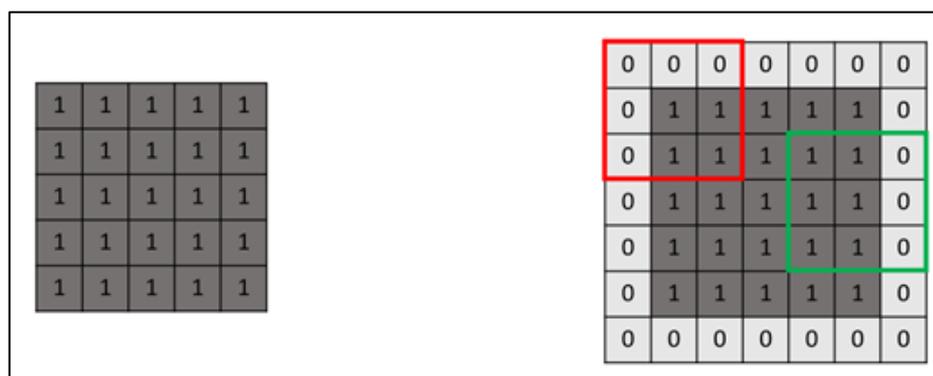
*Pooling layer* berfungsi untuk mengurangi dimensi citra dengan begitu mengurangi parameter dan kompleksitas komputasi yang model. *Pooling* yang banyak digunakan dalam metode CNN adalah *max pooling*. *Max pooling* membagi keluaran dari *layer* konvolusi menjadi beberapa bingkai kecil lalu mencari nilai maksimum nya dari setiap bingkai tersebut untuk Menyusun matriks citra yang telah direduksi sebelumnya oleh konvolusi. Proses ini dilakukan untuk memperbanyak jaringan atau *neuron* yang akan diambil dari citra. Sehingga menambah variasi dan jaringan yang dibuat.



Gambar 2.9 Ilustrasi Max Pooling

### 2.5.3. Zero Padding

*Zero padding* merupakan proses yang penting dalam konvolusi karena ketika *filter* melampaui batas dari citra maka *zero padding* akan berperan agar *filter* dan citra dapat melakukan konvolusi. Sehingga tidak ada kehilangan ekstraksi ciri pada piksel citra pada posisi terluar piksel citra.



Gambar 2.10 Ilustrasi Zero Padding

#### 2.5.4. *ReLu Activation*

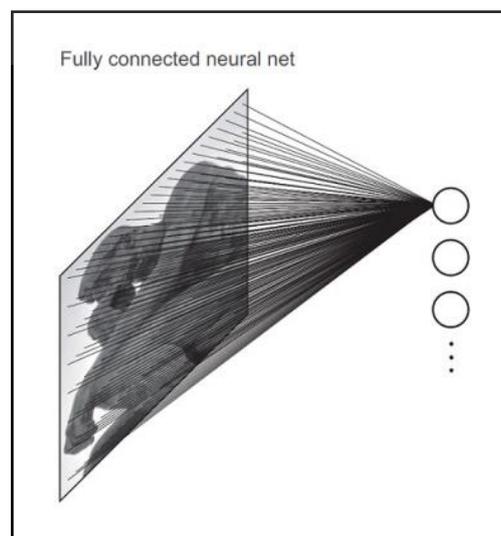
Relu merupakan singkatan dari *Rectife Linear Unit* yang berfungsi untuk mengubah nilai  $x$  menjadi 0 jika nilai  $x$  adalah negatif. Jika nilai  $x$  positif atau lebih dari 0 maka akan dipertahankan.

1,2	2,5	2		1,2	2,5	2
-2,1	2	-1,2		0	2	0
-1,1	4	5		0	4	5

Gambar 2.11 Ilustrasi ReLU Activation

#### 2.5.5. *Fully Connected Layer*

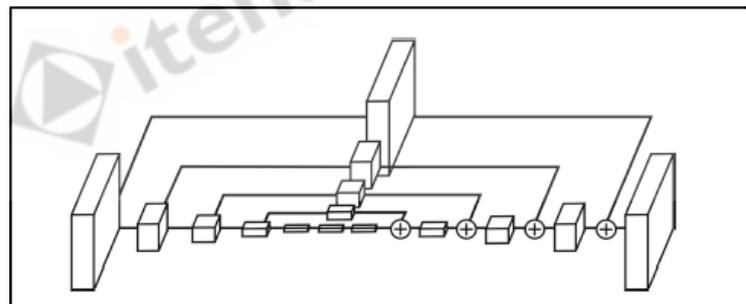
*Fully connected layer* merupakan lapisan terakhir dari algoritma *Convolutional Neural Network* atau CNN. Lapisan ini merupakan lapisan yang sepenuhnya terhubung. Lapisan ini yang menerjemahkan masukan citra dan memberi label sesuai kategorinya. Lapisan ini mendapatkan input dari lapisan sebelumnya untuk menentukan ekstraksi fitur mana yang paling berhubungan dengan kelas atau klasifikasi tertentu. Lapisan ini menyatukan semua node menjadi satu dimensi seperti ilustrasi gambar 2.12.

Gambar 2.12 *Fully Connected Layer*

(Sumber : Elgendy, 2019)

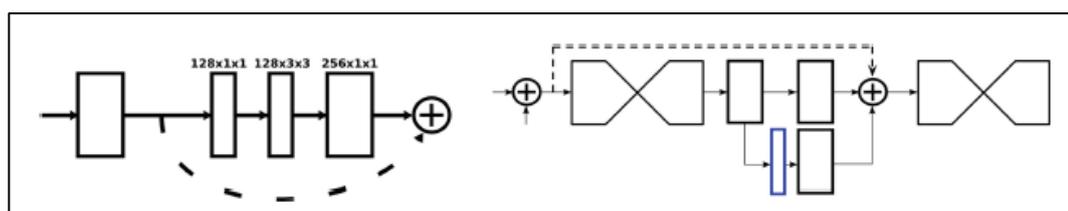
## 2.6. *Stacked Hourglass Network*

Arsitektur *Stacked Hourglass Network* yang diusulkan (B & B, 2016) digunakan untuk memperkirakan lokasi setiap *landmark*. Komponen pada Gambar 2.13 adalah *hourglass network*, seperti yang ditunjukkan pada Gambar 2.14. Ini adalah struktur empat tingkat. Modul residual dapat mengekstraksi fitur tingkat tinggi berdasarkan operasi konvolusional, pada saat yang sama, dapat mempertahankan informasi asli dengan rute lewati. Itu hanya mengubah kedalaman data tanpa mengubah ukuran data. Oleh karena itu, dapat dilihat sebagai lapisan konvolusi. Struktur *hourglass* pada Gambar 2.14 adalah topologi simetris, sehingga mampu menangkap dan mengkonsolidasikan informasi dari skala dan resolusi yang berbeda. Sebelum operasi pengambilan sampel, ini memisahkan satu rute untuk mempertahankan informasi dalam ukuran saat ini. Sebelum operasi *unsampling*, ia menambahkan peta dengan ukuran yang sama dari lapisan asli. Antara pemrosesan dua operasi *sampling*, ia menggunakan tiga modul residual. Setelah menambahkan dua peta, modul residual lain digunakan untuk mengekstrak fitur.



Gambar 2.13 *Stacked Hourglass Network Method*

(Sumber : B and B, 2016)

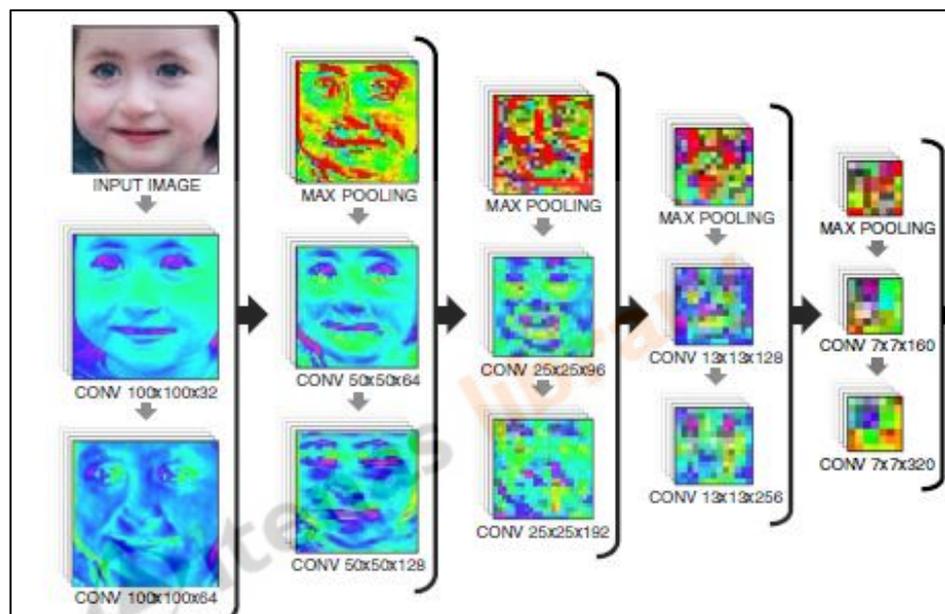


Gambar 2.14 Residual Module *Stacked Hourglass Network*

(Sumber : B and B, 2016)

### 2.6.1. Top-Down

Proses pada metode Stacked Hourglass Network secara umum dapat dibagi dua yaitu *top-down* dan *bottom up*. Pada proses *top-down* dilakukan ekstraksi citra dengan melakukan konvolusi dan *max pooling* (B and B, 2016). Konvolusi menggunakan *filter* 5x5 dan filter 3x3 untuk mencari ekstraksi fitur yang terdapat pada citra.

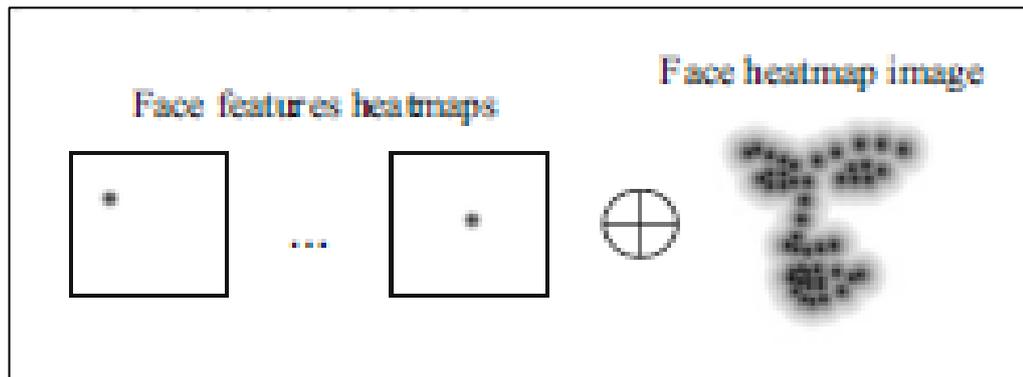


Gambar 2.15 Ilustrasi Top-Down

(Sumber: medium.com)

### 2.6.2. Bottom-Up

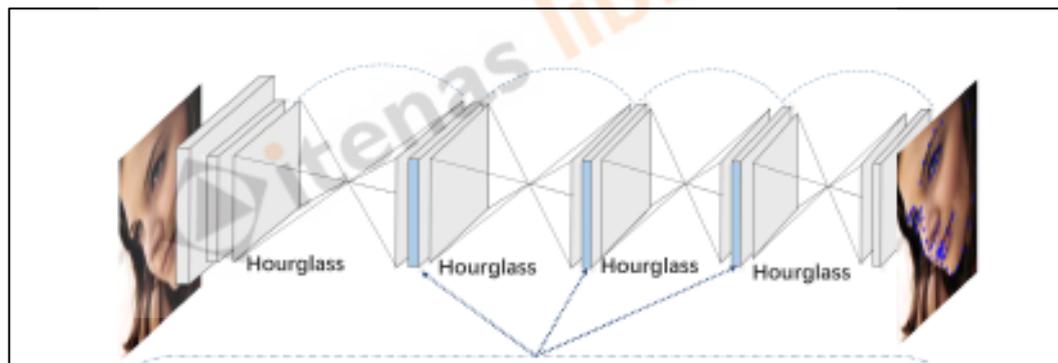
Setelah proses *top-down* untuk mendapatkan fitur dari ekstraksi citra. Selanjutnya dilakukan proses *bottom-up* atau *upsampling* dengan mengumpulkan informasi fitur yang telah diekstrak sebelumnya oleh proses *bottom-up* (B and B, 2016).



Gambar 2.16 Ilustrasi Bottom-Up

(Sumber: medium.com)

Maka setelah proses *top-down* dan *bottom-up* dihasilkan citra dengan *key-point* atau *landmark* yang telah didapatkan dari ekstraksi fitur dan pengumpulan informasi fitur tersebut.



Gambar 2.17 Ilustrasi Hasil Stacked Hourglass Network

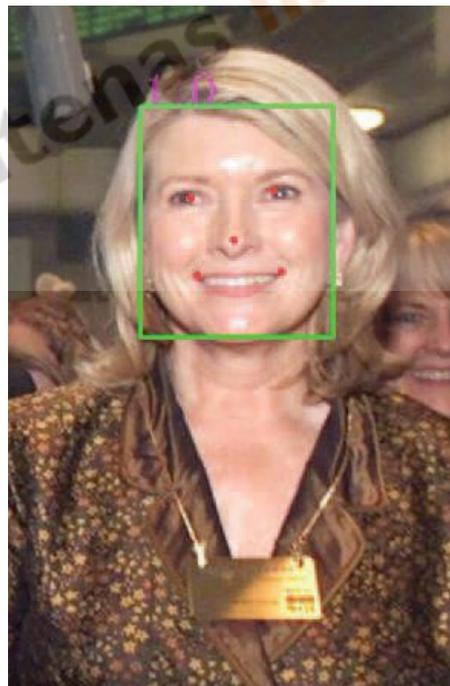
### 2.7. Multi-View Face Detection

Deteksi wajah merupakan langkah awal dari banyak pemrosesan tingkat lanjut seperti pengenalan wajah, analisis wajah, estimasi pose wajah dan lainnya. Tingkat pemrosesan yang lebih tinggi dapat dilakukan hanya jika wajah ditandai secara akurat dalam gambar atau video dan *landmark* wajah secara akurat berada. Sejak munculnya deteksi wajah dan teknologi lokalisasi wajah, banyak metode telah dicoba untuk menyelesaikan dua tugas tersebut, dan ada juga banyak algoritma yang sangat baik. Namun, dengan semakin mendalamnya

pemrosesan/penelitian, dapat diketahui bahwa masalah seperti sikap sudut besar, pencahayaan ekstrem, dan oklusi menambah banyak kesulitan pada wajah dan proses pendeteksian *landmark* nya. Oleh karena itu, cara mengatasi faktor-faktor ini dan meningkatkan akurasi deteksi dengan meningkatkan akurasi deteksi wajah menggunakan *multi-view*. *Multi-view* ini dapat mendeteksi dari berbagai sudut pandang bagian wajah.

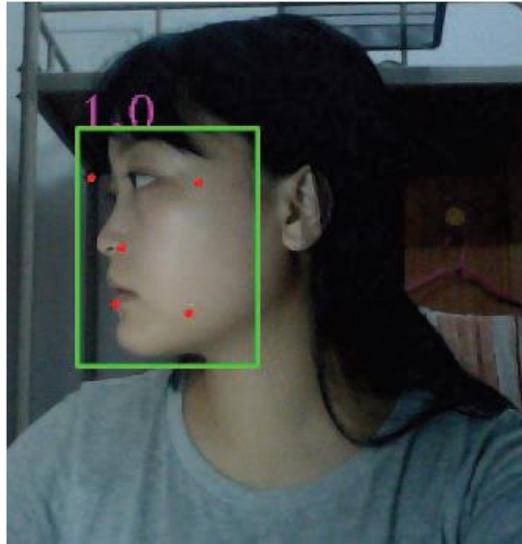
$$\arg \min_R \sum_{i=1}^N \|V_i - RP_i\|_2^2, \dots\dots\dots(1)$$

Dimana  $V_i$  merupakan status tampilan.  $P_i \in \mathbb{R}^{32 \times 1}$  adalah lokasi dari delapan bagian *DPM* yang cacat.  $R$  adalah matriks regresi, yang dapat diselesaikan dengan metode kuadrat terkecil. Untuk pengelompokan tampilan wajah frontal memiliki sudut ( $-15^\circ - 15^\circ$ ), kiri ( $-30^\circ - 0^\circ$ ), dan kanan ( $0^\circ - 30^\circ$ ), yang mencakup semua pose wajah dari dataset pelatihan 300-W.



Gambar 2.18 Contoh *Frontal Face Detection*

(Sumber : Ma, Wang, 2018)



Gambar 2.19 Contoh *Non-Frontal Face Detection*  
(Sumber : Ma, Wang, 2018)

## 2.8. Pengujian Kinerja Sistem

Dalam mengukur kinerja dari sebuah sistem deteksi dan lokalisasi wajah digunakan pengujian dengan perhitungan utama NME (Normalised Mean Error). Normalize Mean Error merupakan perhitungan yang sangat sering dipakai dalam deteksi dan lokalisasi wajah. Perhitungan ini membandingkan nilai ground truth dengan nilai prediksi pada citra uji. Nilai ground truth didapatkan dari citra yang sudah diberi label.

$$NME = \frac{1}{N} \sum_{i=1}^N \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{d},$$

Dimana :

$N$  = banyak landmark

$\Delta x_i$  = perbedaan nilai x ground truth and prediction

$\Delta y_i$  = perbedaan nilai y ground truth and prediction

$d$  = image height/width

## 2.9. Studi Kasus

Pada studi kasus dijelaskan mengenai studi kasus dari proses digitalisasi gambar, normalisasi, konvolusi, *max pooling*, *padding*, *ReLU activation*, *view angle* yang dilakukan terhadap citra 5x5.



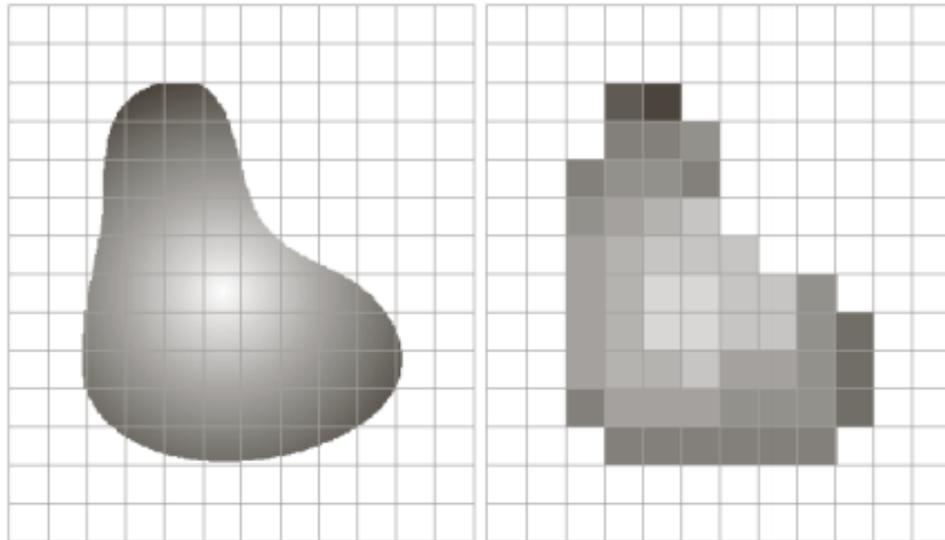
Gambar 2.20 Studi Kasus

### 2.9.1. *Digital Image Quantization*

*Digital image quantization* adalah mengubah nilai analog pada citra menjadi nilai digital sehingga dapat diproses oleh komputer (Paiva, 2010). Terdapat dua domain dalam *digital image quantization* yaitu *sampling* dan *quantization*. *Sampling* merupakan diskritisasi atau mencari nilai diskrit yang ada pada ruang citra. Domain sampling mempunyai persamaan berikut.

$$f : [1, \dots, N] \times [1, \dots, M] \rightarrow R^m$$

Ilustrasi sampling dapat dilihat pada gambar 2.21.

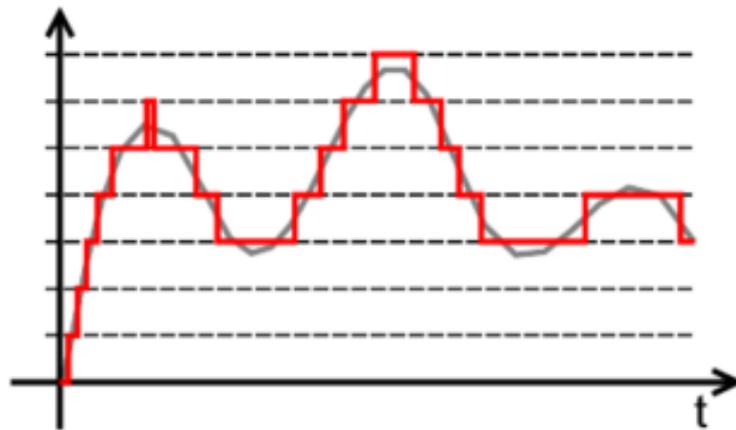
Gambar 2.21 *Sampling DIQ*

Sampling akan menghasilkan nilai matriks seperti ilustrasi gambar 2.22.

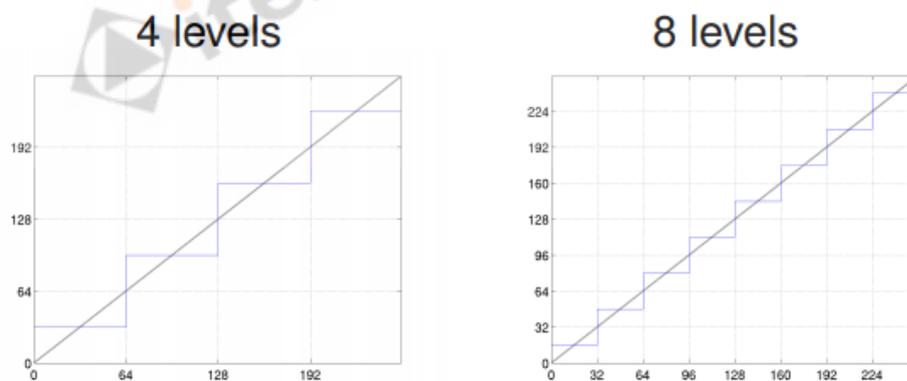
$$f = \begin{bmatrix} f(1, 1) & f(1, 2) & \cdots & f(1, M) \\ f(2, 1) & f(2, 2) & \cdots & f(2, M) \\ \vdots & \vdots & \ddots & \vdots \\ f(N, 1) & f(N, 2) & \cdots & f(N, M) \end{bmatrix} .$$

Gambar 2.22 Matriks hasil *sampling*

Elemen terkecil pada hasil diskritisasi ruang pada citra disebut piksel. Sedangkan *quantization* merupakan diskritisasi pada nilai intensitas yang ada pada citra. Domain *quantization* mempunyai ilustrasi pada gambar 2.23.

Gambar 2.23 *Quantization DIQ*

*Quantization* menentukan nilai intensitas pada citra. Intensitas citra ini mempengaruhi kualitas kejelasan gambar. Biasanya intensitas citra mempunyai nilai 256 levels (8bits/piksel) sedangkan pada gambar berwarna digunakan 8bits/piksel pada tiap warnanya. Ilustrasi intensitas gambar dapat dilihat pada gambar 2.24.

Gambar 2.24 *Quantization Intensity*

### 2.9.2. *Normalization*

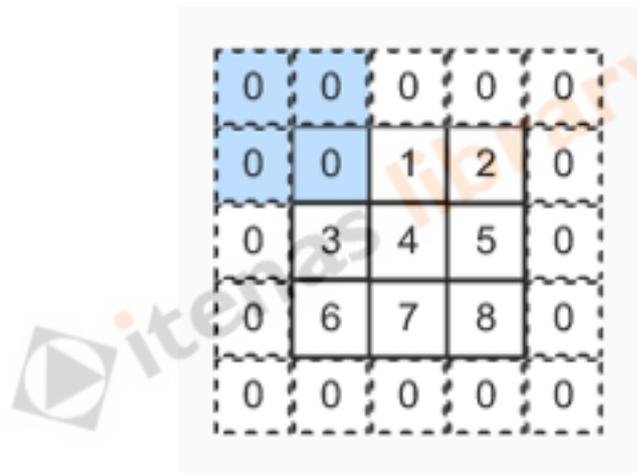
Normalisasi merupakan proses membagi nilai citra dengan 255. 255 merupakan nilai maksimal sebuah histogram dari sebuah citra. Ilustrasi normalisasi dapat dilihat pada gambar 2.25

16	216	162	59	171	178	/255	=	0,062745	0,847059	0,635294	0,231373	0,670588	0,698039
44	235	213	78	154	21			0,172549	0,921569	0,835294	0,305882	0,603922	0,082353
96	200	39	176	212	34			0,376471	0,784314	0,152941	0,690196	0,831373	0,133333
227	41	139	157	228	117			0,890196	0,160784	0,545098	0,615686	0,894118	0,458824
104	74	243	171	225	64			0,407843	0,290196	0,952941	0,670588	0,882353	0,25098
135	168	190	132	97	83			0,529412	0,658824	0,745098	0,517647	0,380392	0,32549

Gambar 2.25 Normalization

### 2.9.3. Padding

*Padding* atau *zero padding* merupakan teknik menambahkan piksel pada setiap sisi luar input. Tujuan *padding* atau *zero padding* ini yaitu memanipulasi dimensi output agar ekstraksi fitur yang dilakukan oleh konvolusi tidak banyak yang hilang. Ilustrasi *padding* atau *zero padding* dapat dilihat pada gambar 2.26



Gambar 2.26 Zero Padding

### 2.9.4. Konvolusi

Proses konvolusi dilakukan menggunakan kernel 5x5 dan 3x3 pada citra yang akan dideteksi wajah. Gambar 20 filter 5x5 dan gambar 19 filter 3x3

0	0	0
0	1	0
0	0	-1

Gambar 2.27 Filter 3x3

0	0	0	0	0
0	1	1	1	0
0	1	4	1	0
0	1	1	1	0
0	0	0	0	0

Gambar 2.28 Filter 5x5

Contoh hasil Proses Konvolusi matriks 10x10 dengan filter 7x7

Nilai Citra									
445,7	260,5	1001	639,9	66,5	819,1	718,1	620,8	868,3	151,3
486,5	652,8	450,3	439	313,9	489,4	592,1	861,9	554	307,3
796,9	600,6	1023,7	499,6	827,9	151,3	136,5	463,5	9,5	798,8
1023,2	168,7	404,7	808,3	661,8	1022,1	307,6	14,6	493,9	170,2
488,1	742,9	293,8	714,8	543,2	494,6	962,9	174,1	1	461,7
611,4	757,5	732,6	65,4	334,4	1022,4	984,3	325,3	47,5	272
600,8	782,9	847,2	979,4	184,3	851,9	447,9	686,6	476,3	286,1
499,6	830,7	282,6	294,2	325,6	704,6	331,8	124,2	10,6	935,6
771,2	403,2	302,7	148,4	447,9	229,5	143,6	176,3	73	973,1
561,4	535,5	32	638,3	576,3	517,7	519,4	115	626,2	552,1
885	564,4	779,4	739,5	553,8	846,8	396,1	862,4	105,4	526,4

Gambar 2.29 Ilustrasi Nilai Citra

Filter 7x7						
0	0	0	0	0	0	0
0	1	1	1	1	1	0
0	1	1	1	1	1	0
0	1	1	3	1	1	0
0	1	1	1	1	1	0
0	1	1	1	1	1	0
0	0	0	0	0	0	0

Gambar 2.30 Ilustrasi nilai filter

HASIL KONVOLUSI			
296,6	401,5	334,3	440,6
403,5	284,2	346,3	354,9
225,5	388	189,1	440,7
434	307,3	423,6	377,8
177,7	295,6	232,4	497,5

Gambar 2.31 Hasil Konvolusi

Proses konvolusi dilakukan dengan melakukan perkalian antara citra input dengan kernel atau filter, berikut merupakan contoh perhitungan yang dilakukan konvolusi dengan nilai citra input

1. Konvolusi dilakukan dari sudut kiri atas, untuk mendapatkan nilai baris 1 kolom 1 dilakukan perhitungan seperti pada gambar 24 maka hasil perhitungan :

$$\begin{aligned}
 &= (0*81,7)+(0*1,8)+(0*82,3)+(0*98,8)+(0*32,8)+(0*88,2)+(0*81)+ \\
 &\quad (0*0,7)+(1*48)+(0*59,4)+(1*65,2)+(0*29,8)+(1*84,8)+(0*53,7)+ \\
 &\quad (0*31,7)+(0*22,4)+(0*70)+(0*45,3)+(0*63,6)+(0*70,4)+(0*24,5)+ \\
 &\quad (0*36,3)+(0*50,9)+(0*22)+(1*27,1)+(0*89)+(0*31,5)+(0*48)+ \\
 &\quad (0*69,1)+(0*39,8)+(0*14,8)+(0*84)+(0*1,4)+(0*19,7)+(0*62,3)+ \\
 &\quad (0*62,4)+(1*17,1)+(0*7,2)+(1*44,5)+(0*88,7)+(1*9,9)+(0*73,7)+ \\
 &\quad (0*9,1)+(0*79,7)+(0*22,7)+(0*38)+(0*5,3)+(0*63,7)+(0*96,7) \\
 &= 296,6
 \end{aligned}$$

Nilai Citra										Filter 7x7							
81,7	1,8	82,3	98,8	32,8	88,2	81	73,2	43,6	83,5	0	0	0	0	0	0	0	0
0,7	48	59,4	65,2	29,8	84,8	53,7	47,7	81,6	65,8	0	1	0	1	0	1	0	0
31,7	22,4	70	45,3	63,6	70,4	24,5	27,4	34,9	55	0	0	0	0	0	0	0	0
36,3	50,9	22	27,1	89	31,5	48	34,9	38,2	64,3	0	0	0	1	0	0	0	0
69,1	39,8	14,8	84	1,4	19,7	62,3	39,9	37,2	43,1	0	0	0	0	0	0	0	0
62,4	17,1	7,2	44,5	88,7	9,9	73,7	50,7	65,1	88,9	0	0	0	0	0	0	0	0
9,1	79,7	22,7	38	5,3	63,7	96,7	81,8	67,6	19,8	0	1	0	1	0	1	0	0
57,3	57,6	29,5	8,2	21	5,7	89,8	71,8	81	36,2	0	0	0	0	0	0	0	0
92,1	87,4	72,1	68	90,8	97,1	60,6	51,2	28,8	19,8	0	0	0	0	0	0	0	0
80,8	47,1	16,5	11,2	77,2	39,7	11,3	70,7	91,7	66,1	0	0	0	0	0	0	0	0
61,4	80,2	0,9	58,9	15	92	90	52,5	32,3	71,4								

Gambar 2.32 Ilustrasi Perhitungan Konvolusi

2. Untuk mendapatkan nilai baris 1 kolom 2 dilakukan pergeseran 1 stride. Lakukan perkalian sama seperti langkah 1

Nilai Citra										Filter 7x7						
81,7	1,8	82,3	98,8	32,8	88,2	81	73,2	43,6	83,5	0	0	0	0	0	0	0
0,7	48	59,4	65,2	29,8	84,8	53,7	47,7	81,6	65,8	0	1	0	1	0	1	0
31,7	22,4	70	45,3	63,6	70,4	24,5	27,4	34,9	55	0	0	0	0	0	0	0
36,3	50,9	22	27,1	89	31,5	48	34,9	38,2	64,3	0	0	0	0	0	0	0
69,1	39,8	14,8	84	1,4	19,7	62,3	39,9	37,2	43,1	0	0	0	1	0	0	0
62,4	17,1	7,2	44,5	88,7	9,9	73,7	50,7	65,1	88,9	0	0	0	0	0	0	0
9,1	79,7	22,7	38	5,3	63,7	96,7	81,8	67,6	19,8	0	1	0	1	0	1	0
57,3	57,6	29,5	8,2	21	5,7	89,8	71,8	81	36,2	0	0	0	0	0	0	0
92,1	87,4	72,1	68	90,8	97,1	60,6	51,2	28,8	19,8	0	0	0	0	0	0	0
80,8	47,1	16,5	11,2	77,2	39,7	11,3	70,7	91,7	66,1							
61,4	80,2	0,9	58,9	15	92	90	52,5	32,3	71,4							

Gambar 2.33 Ilustrasi Perhitungan Konvolusi

$$\begin{aligned}
&= (0*1,8)+(0*82,3)+(0*98,8)+(0*32,8)+(0*88,2)+(0*81)+(0*73,2)+ \\
&\quad (0*48)+(1*59,4)+(0*65,2)+(1)29,8+(0*84,8)+(0*53,7)+(0*47,7)+ \\
&\quad (0*22,4)+(0*70)+(0*45,3)+(0*63,6)+(0*70,4)+(0*24,5)+(0*27,4)+ \\
&\quad (0*50,9)+(0*22)+(0*27,1)+(1*89)+(0*31,5)+(0*48)+(0*34,9)+ \\
&\quad (0*39,8)+(0*14,8)+(0*84)+(0*1,4)+(0*19,7)+(0*62,3)+(0*39,9)+ \\
&\quad (0*17,1)+(1*7,2)+(0*44,5)+(1*88,7)+(0*9,9)+(1*73,7)+(0*50,7)+ \\
&\quad (0*79,7)+(0*22,7)+(0*38)+(0*5,3)+(0*63,7)+(0*96,7)+(0*81,8) \\
&= 403,5
\end{aligned}$$

- Untuk mendapatkan nilai baris 1 kolom 3 dan seterusnya dilakukan pergeseran 1 stride dan melakukan perhitungan seperti pada langkah 1 dan 2.

### 2.9.5. Max Pooling

296,6	401,5	334,3	440,6
403,5	284,2	346,3	354,9
225,5	388	189,1	440,7
434	307,3	423,6	377,8

Gambar 2.34 Ilustrasi Max Pooling

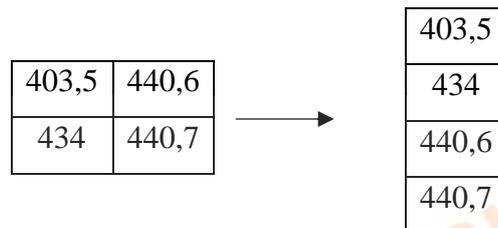
Operasi *max pooling* dilakukan untuk memilih nilai terbesar pada matriks, *max pooling* menggunakan matriks 2x2 sehingga terbagi menjadi 4 bagian dan hasil *max pooling* dapat dilihat pada gambar 2.34

403,5	440,6
434	440,7

Gambar 2.35 Ilustrasi Hasil Max Pooling

### 2.9.6. Flattening

Operasi *flattening* merupakan proses mengubah bentuk dimensi matriks menjadi 1 dimensi. Dilakukan pada hasil max pooling seperti pada gambar 2.35.



Gambar 2.36 Ilustrasi Flattening

### 2.9.7. View Angle

Nilai *view angle* atau posisi wajah pada citra dapat dilihat pada nilai citra atau koordinat citra yang telah dilatih atau diuji. Contoh view angle pada citra dengan posisi menghadap kanan, kiri atau frontal dapat dilihat pada table 2.1.

Tabel 2.1 Contoh *View-Angle* pada citra

Citra	Nilai Koordinat Landmark Citra
	102.250000 144.750000
	130.750000 100.250000
	143.250000 113.250000
	112.750000 138.750000
	156.250000 155.750000

	107.250000 147.750000 126.250000 106.250000 140.750000 108.750000 113.250000 143.750000 158.750000 162.750000
	99.750000 139.250000 113.750000 112.750000 141.250000 113.750000 107.750000 133.750000 159.250000 152.750000

Jika dianalisa nilai koordinat *landmark* pada setiap perbedaan posisi wajah pada citra dapat dilihat pada koordinat letak bagian wajah seperti mata, hidung dan mulut.