

## BAB 2 TINJAUAN PUSTAKA

### 2.1 Kajian Pustaka

Penelitian ini mengenai *face mask detection*. Didalam penelitian ini memperkenalkan model *face mask detection* yang didasarkan pada *deep transfer learning* dan pengklasifikasi *classical machine learning*. Dalam penelitian ini mencakup dua komponen yaitu, pada komponen yang utama adalah deep tranfering learning (ResNet50) sebagai ekstrator fitur dan komponen kedua adalah classical machine learning seperti decision tree, SVM, dan ensemble. Percobaan yang dilakukan dalam penelitian ini memiliki jejaknya yaitu, Ada 3 pengklasifikasi yang digunakan yaitu decision tree, SVM, dan ensemble); Menggunakan tiga data berupa kumpulan data RMFD yang akan dirujuk pada DS1, kumpulan data SMFD yang akan dirujuk pada DS2, dan kumpulan data LFW yang akan dirujuk pada DS3; set data untuk pelatihan dan pengujian dibagi menjadi 70% untuk pelatihan, 10% untuk validasi, 20% untuk pengujian. Menurut Loey et al.,( 2020) dari tiga dataset yang telah diujicobakan mempunyai hasil yaitu pengklasifikasi SVM dalam RMFD mencapai akurasi 99,64%, pada SMFD mencapai 99, 49% sedangkan pada LFW mencapai akurasi pengujian 100%.

Mendeteksi masker pada wajah manusia digunakan untuk mengetahui keakuratan dan kelayakan pemakaian masker pada wajah. Dalam penelitian ini menggunakan teknik Convolutional Neural Networks (CNN). Menurut penelitian Chavda et al., (2020) yang mengusulkan dua tahap arsitektur pada CNN. Tahap pertama medeteksi wajah manusia dan tahap kedua menggunakan gambar untuk mengklasifikasikan wajah yang terdeteksi wajah dengan masker dan tanpa masker. Dari hasil penelitian ini memiliki nilai akurasi dengan tiga model yaitu NASNetMobile dengan akurasi 99.23%, DenseNet121 dengan akurasi 99.49%, dan MobileNetV2 dengan akurasi 99.23%.

Adapun penelitian dari Nuanmeesri et al., (2020) penelitian ini menyajikan pengembangan sistem peringatan kepada memakai masker wajah untuk perlindungan infeksi pernafasan menggunakan IoT. ESP32-CAM dengan chip mikrokontroler ESP32-S digunakan sebagai papan utama. Ini mengumpulkan gambar dengan mendeteksi file orang-orang yang sedang fokus dan 6melihat kamera OV264 modul. Sistem ini berfokus pada esensi wajah deteksi dan deteksi topeng di wajah orang. Semua 2.160 pengambilan sampel gambar wajah orang dengan atau tanpa topeng dan gambar dari beberapa jenis topeng dilatih teknik klasifikasi kaskade dengan Python dengan menggunakan Open CV Perpustakaan. Model ini dilatih dan diterapkan ke IoT pengembangan di Arduino IDE untuk mendeteksi dan

mengklasifikasikan file orang yang memakai topeng atau tidak. Sistem ini diimplementasikan, diuji, dan dievaluasi. Untuk efisiensi evaluasi sistem, presisi, perolehan, dan pengukuran F 95,10%, 91,51%, dan 93,27% masing-masing, dimana akurasi 91,60%. Selain itu, sistem tersebut dievaluasi oleh sembilan ahli di pengujian kotak hitam. Hasil evaluasi sistem berwarna hitam pengujian kotak menunjukkan rata-rata total rata-rata 4,51 dengan standar deviasi 0,51 oleh para ahli dan rata-rata total 4,62 dengan deviasi standar 0,49 oleh pengguna. Menurut hasil evaluasi untuk penerimaan sistem yang dapat dideteksi oleh para ahli dan pengguna saat digunakan memiliki konsensus yang tinggi. Semua dari nilai dalam rentang interkuartil (IQR) tidak lebih dari 1 dan deviasi kuartil (QD) tidak lebih dari 0,5. Bisa jadi mengatakan bahwa sistem yang dikembangkan dapat mendeteksi dan mengklasifikasikan file orang yang memakai masker wajah untuk melindungi infeksi saluran pernafasan pada tingkat tertinggi.

Terdapat penelitian deteksi menggunakan masker wajah menggunakan metode Convolutional Neural Networks (CNN) menurut Budiman, (2021) dalam penelitiannya dijelaskan tentang proses deteksi penggunaan masker wajah dengan menggunakan program python. Pada dataset yang digunakan sebanyak 700 citra pada tiap 2 kategori kelas. Dimana dalam mengaksesnya melalui media kamera webcam. Klasifikasi masker wajah dengan metode CNN dengan arsitektur menggunakan model MobileNetV2. Sebelum melakukan proses deteksi masker wajah, terlebih dahulu melakukan deteksi wajah manusia menggunakan metode Haecascade Classifier. Cara kerja sistem dengan cara mengakses kamera laptop lalu diberi kotak hijau untuk menyatakan hasil deteksi orang tersebut memakai masker dan tidak memakai masker secara real time. Dari hasil pengujian pada penelitian yang dilakukan dapat mendeteksi penggunaan masker wajah dengan rata-rata akurasi yang dimiliki 88,53%. Pengujian yang dilakukan pada penelitian ini menggunakan model CNN. Pada model yang dilatih memiliki jumlah epoch 100 epoch. Jumlah data latih yang digunakan sebanyak 1400 gambar yang dibagi menjadi dua yaitu menggunakan masker dan tidak menggunakan masker dengan tiap kelas jumlahnya sebanyak 700 gambar. Pada bagian validasi dengan jumlah 140 pada gambar tidak menggunakan masker dan 140 gambar menggunakan masker. Hasil pengujian yang didapat dengan menggunakan arsitektur MobileNetV2 pada model CNN sebesar 88,53% dan hasil klasifikasi yang tepat adalah 84,45%.

Adapun Penelitian serupa tentang deteksi masker dengan deep learning menurut Loey, Manogaran, Taha, et al., (2020) dalam penelitiannya memperkenalkan sebuah model baru untuk topeng medis deteksi wajah, dengan fokus pada objek masker medis untuk mencegah COVID-19 menyebar dari manusia ke manusia. Untuk melatih dan memvalidasi detektor dalam kondisi diawasi, mendesain kumpulan data baru berdasarkan dua kumpulan data wajah bertopeng publik. Selain itu, metrik kinerja seperti AP dan tingkat kesalahan rata-rata log skor

telah dipelajari untuk eksperimen pengoptimal SGDM dan Adam. Pada skema model yang diusulkan YOLOv2 dengan ResNet-50 adalah model yang efektif untuk mendeteksi wajah bertopeng medis. Sebagai studi masa depan, Kami berencana untuk mendeteksi jenis wajah bertopeng dalam gambar dan video berbasis Untuk mendeteksi gambar menggunakan model ResNet-50 berbasis YOLO v2 untuk menghasilkan hasil berkinerja tinggi. Sehingga hasil yang didapat dengan menggunakan metode ini adalah sekitar 81%.

Penelitian lain tentang deteksi masker yang dilakukan oleh Said, (2020) yaitu detektor yang diusulkan didasarkan pada YOLO rangka dengan tulang punggung yang ringan. 7 Model yang diusulkan, disebut Pynq-YOLO-Net, dirancang untuk diimplementasikan pada papan Pynq Z1. Untuk mencapai implementasi ini, ada beberapa model teknik kompresi yang diterapkan pada penelitian ini seperti pemangkasan dan kuantisasi. Teknik-teknik itu sangat efektif untuk mengurangi ukuran model dan kompleksitas komputasi. Itu model diimplementasikan pada perangkat keras dan perangkat lunak ke mempercepat kesimpulan. Kinerja Pynq-YOLO-Net dievaluasi set pengujian yang merupakan 30% dari data yang dikumpulkan. PynqYOLO-Net mencapai Presisi 94,6% dan Penarikan Kembali 95,8% untuk proses pelatihan pertama menggunakan representasi floatingpoint 32-bit. Setelah mengompresi model dan dilakukan kembali pada data yang sama, akurasi mencapai 90,7% Akurasi dan 92.3 dari Recall.

Pada Penelitian yang dilakukan oleh Ansor et al., (2020) memulai tahap pada penelitiannya dengan melakukan pelatihan model deteksi masker, menyusun program yang akan diimplementasikan proses pelatihan model. Pertama yang dilakukan adalah dependensi perpustakaan termasuk; tensorflow + keras, numpy, matplotlib, argparse, os, scikit-learn dan manisils. Dalam pelatihan ini model yang dibentuk tidak hanya mengandalkan dataset yang telah penulis siapkan, tetapi melalui proses transfer pembelajaran model dasar mobilenetv2 dan dataset (berat) dari ImageNet melalui fungsi fine-tuning pada tensorflow + keras.

Tahap selanjutnya adalah eksekusi program `train_deksi_masker.py` di Google Colaboratory dengan ekstensi dataset dan program yang telah disiapkan sebelumnya. Berdasarkan hasil pembahasan yang meliputi perencanaan kebutuhan, implementasi dan tahapan pengujian yang telah penulis lakukan, dapat diambil kesimpulan sebagai berikut; (1) Deteksi topeng aktif gambar dan video (realtime) dapat dilakukan dengan memanfaatkan kerangka kerja Tensorflow dan CNN terlatih model berdasarkan Raspberry Pi. Mendeteksi masker dimulai dengan membuat (melatih) model untuk dijalankan di Perangkat Raspberry Pi, proses pelatihan dilakukan dengan melatih dataset citra melalui transfer learning dan metode fine tuning. Model ini dibuat dengan arsitektur MobilenetV2 di atas model ImageNet mendasarkan. Penggunaan model deteksi wajah terlatih, yang diunduh dari Github OpenCV resmi halaman, cukup membantu sebagai filter pertama sebelum model deteksi topeng bekerja. (2) Berdasarkan hasil Dari pengujian deteksi topeng yang

telah dibahas, diketahui akurasi yang diperoleh 96% pada pengujian dengan input file gambar dan 91% pada pengujian dengan input video (realtime), sedangkan untuk presisi didapat 100% pada pengujian dengan input file gambar dan 80% pengujian dengan input video (realtime). 8 Rasio positif benar (recall) performansi yang merupakan sensitivitas model dalam mengambil informasi memiliki nilai 92% pada pengujian dengan input file gambar dan 100% pada pengujian dengan input video (realtime).

Pemakaian masker ialah salah satu protokol kesehatan untuk warga di tempat serta sarana universal dalam rangka penangkalan serta pengendalian Covid- 19. Pandemi covid- 19 di segala dunia sudah mendesak para pengambil keputusan serta bermacam elemen warga buat ambil bagian dalam bermacam bidang serta berkontribusi dalam memencet penyebaran covid- 19. Pada penelitian Ansor et al., (2020) menjelaskan *machine learning* serta *computer vision* ialah salah satu cabang dari Artificial Intelligence (AI) serta bisa dibesarkan dalam bermacam pengenalan citra. Dalam riset ini digunakan Application Program Interface (API) *machine learning* ialah *Tensorflow* serta model CNN yang sudah dilatih tadinya, dan Raspberry Pi yang ialah fitur mobile selaku sistem pendeteksi topeng. Analisis dicoba buat mengenali akurasi, presisi serta *recall* (sensitivitas) dari hasil implementasi sistem. Hasil implementasi menampilkan kalau pemakaian transfer learning serta fine- tuning lumayan menolong proses pelatihan model. Kala model dijalankan pada fitur Raspberry Pi, dihasilkan persentase akurasi sebesar 96% pada pengujian dengan input file foto serta 91% pada pengujian dengan input video (realtime), presisi 100% pada pengujian dengan input file foto serta 80% pada pengujian dengan input file foto. pengujian dengan input video ( *realtime*). Pada kinerja *true positive rate* ( *recall*), ditemui 92% pada pengujian dengan input file foto serta 100% pada pengujian dengan input video ( *realtime*). Arsitektur terdiri dari pengklasifikasi *MobileNetV2* dan Pengoptimal ADAM sebagai tulang punggung itu dapat digunakan untuk tinggi dan rendah skenario komputasi. Deteksi masker wajah kami dilatih pada model CNN dan kami menggunakan CV Oven, Tensor Flow, Keras dan python untuk mendeteksi seseorang memakai masker atau tidak. Model pada penelitian ini menggunakan waktu streaming video. Akurasi model tercapai dan, optimasi model adalah proses berkelanjutan. Model khusus ini dapat digunakan sebagai penggunaan kasus analitik tepi. epoch ke-20, memiliki akurasi 97,86% pada set pelatihan dan 99.22% pada set pengujian.

Penelitian yang berkaitan pada sistem absensi dengan pengenalan wajah menggunakan beberapa metode, diringkas dalam Tabel 2.1

**Tabel 2.1** *State Of The Art*

<b>Judul Penelitian</b>	<b>Metode Klasifikasi</b>	<b>Data Collection</b>	<b>Spesifikasi Requirement</b>	<b>Hasil</b>
<p><i>A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic</i></p> <p>Loey et al., (2020)</p>	<p><i>A hybrid deep transfer learning model with machine learning</i></p>	<p>Dataset of RMFD, SMFD, LFW dan dataset kombinasi dari DS1 dan DS2.</p>	<p>Menggunakan computer server yang memiliki spesifikasi prosesor intel Xeon (2 Ghz) dengan RAM 96 GB dan juga menggunakan paket perangkat lunak MATLAB.</p>	<p>SVM dalam RMFD mencapai akurasi 99,64%, pada SMFD mencapai 99,49% sedangkan pada LFW mencapai akurasi pengujian 100%.</p>
<p><i>Multi-Stage CNN Architecture for Face Mask Detection</i></p> <p>Chavda et al., (2020)</p>	<p><i>Convolutional Neural Networks (CNN)</i></p>	<p>7855 dataset gambar. 3440 gambar wajah memakai masker dan 4415 gambar wajah tidak menggunakan masker</p>	<p>Tidak Diketahui</p>	<p>NASNetMobile dengan akurasi 99.23%, DenseNet121 dengan akurasi 99.49%, dan MobileNetV2 dengan akurasi 99.23%</p>
<p><i>Face Mask Detection And Warning System For Preventing Respiratory</i></p>	<p><i>Internet of things</i></p>	<p>Ada 1.014 gambar untuk masker saja, 836 gambar untuk orang yang</p>	<p>Sistem ini dikembangkan di platform IoT dengan menggunakan Arduino IDE</p>	<p>Untuk efisiensi evaluasi sistem, presisi, perolehan, dan pengukuran F 95,10%,</p>

Judul Penelitian	Metode Klasifikasi	Data Collection	Spesifikasi Requirement	Hasil
<p><b><i>Infection Using The Internet Of Things</i></b></p> <p>Nuanmees ri et al., (2020)</p>		memakai masker, dan 310 gambar untuk orang yang jangan pakai masker.	versi 1.8.10 pada Windows 10 Edisi Profesional x64. Ini terdiri dari empat perangkat di IoT; ada papan ESP32-CAM, penguat audio LM386 modul, speaker 8 ohm 1,5 watt, dan baterai pack.	91,51%, dan 93,27% masing-masing, dimana akurasi 91,60%
<p><b><i>Fighting Against COVID-19: A Novel Deep Learning Model Based On YOLOV2 With Resnet-50 For Medical Face Mask Detection</i></b></p> <p>Loey, Manogara n, Taha, et al., (2020)</p>	<p><i>deep learning model based on YOLO-v2 with ResNet-50</i></p>	The first dataset is Medical Masks Dataset (MMD), The MMD dataset consists of 682 pictures with over 3k medical masked faces wearing masks and The FMD dataset consists of 853 images	The GPU used NVIDIA RTX with the CUDA with Tensorflow, MATLAB, and Deep Neural Network library (CuDNN) for GPU learning.	presisi rata-rata sama dengan 81% menggunakan YOLOv2 dengan ResNet-50 berdasarkan pengoptimal Adam. Dengan menganalisis kinerja YOLO v2
<p><b><i>Pynq-YOLO-Net: An Embedded Quantized Convolutional Neural Network for Face Mask Detection in COVID-19</i></b></p>	<p><i>Convolutional Neural Network (CNN)</i></p>	Real-World Masked Face Dataset (RMFD) Dataset berisi 5.000 gambar dari 525 orang yang memakai masker, dan 90.000	Tidak Diketahui	PynqYOLO-Net mencapai Presisi 94,6% dan Penarikan Kembali 95,8% untuk proses pelatihan pertama menggunakan

Judul Penelitian	Metode Klasifikasi	Data Collection	Spesifikasi Requirement	Hasil
<p><i>Pandemic Era</i></p> <p>Rahman et al., (2020)</p>		gambar yang sama 525 orang tanpa masker dan Masked Face Detection Dataset (MFDD) sebanyak 24471 menggunakan masker		representasi floatingpoint 32-bit. akurasi mencapai 90,7% Akurasi dan 92.3 dari Recall.
<p><i>Mask Detection Using Framework Tensorflow and Pre-Trained CNN Model Based on Raspberry Pi</i></p> <p>Ansor, Ritzkal and Afrianto, (2020)</p>	Framework tensorflow and Pre-Trained Convolutional Neural Network (CNN) Model Based on Raspberry Pi	1676 citra yang terbagi menjadi dua, 826 gambar wajah yang memakai masker dan 850 gambar wajah tanpa masker.	Raspberry Pi connected to access Point penelitian NCC WLAN Interface.	akurasi yang diperoleh 96% pada pengujian dengan input file gambar dan 91% pada pengujian dengan input video (realtime)
<p><i>A Facemask Detector Using Machine Learning And Image Processing Techniques</i></p> <p>Science, Bhadani and Sinha, (2020)</p>	<i>Convolutional Neural Network (CNN)</i>	Dataset gambar wajah memakai masker sebanyak 800 dan 750 tidak memakai masker.	Tidak Diketahui	epoch ke-20, memiliki akurasi 97,86% pada set pelatihan dan 99.22% pada set pengujian.

## 2.2 Dasar Teori

Pada subbab ini dibahas mengenai dasar teori yang mendukung teknik yang digunakan dalam penelitian ini, serta informasi umum mengenai objek penelitian.

### 2.2.1 Deep Learning

*Deep learning* sangat penting pada bidang AI yang menggunakan banyak lapisan dalam penanganan data nonlinier untuk dilakukan pada proses ekstraksi fitur, proses pengenalan pola, dan klasifikasi. *Deep learning* adalah metodologi dalam menangani masalah dalam kerangka pembelajaran komputer yang memanfaatkan gagasan rantai kepentingan. Ide rantai yang penting dapat memperdayakan komputer untuk mempelajari ide-ide kompleks dengan menggabungkan ide-ide yang kurang kompleks. Jika diumpamakan pada sebuah graf tentang konsep yang dibangun atau konsep lainnya yang berhubungan dengan konsep tersebut. Maka hal ini dapat memperdalam pengertian dari proses *deep learning* itu sendiri berdasarkan banyak *layer* yang telah digambarkan pada graf tersebut.

Seperti pada mendeteksi objek, pekerjaan awal dalam deteksi wajah adalah mengambil paradigma *Sliding Window*, di mana *classifier* diterapkan pada grid gambar yang padat. Mengikuti paradigma ini, kemajuan awal dalam deteksi wajah terkait erat dengan penggunaan kombinasi fitur buatan tangan dan pengklasifikasi, misalnya, fitur seperti *Histogram dari Gradien Berorientasi* (HOG) (Loy, 2020). HOG memiliki efek rapuh pada masalah cahaya rendah dibandingkan dengan sistem deteksi wajah lainnya. Setelah mendeteksi, gambar memerlukan beberapa *preprocessing* yang disebut normalisasi. *Face Landmark Estimation* dapat digunakan untuk normalisasi (Ara, Simul and Islam, 2018).

Deteksi wajah yang kotemporer didasarkan pada metode *Convolutional Neural Network* (CNN), dimana ekstraksi fitur dan klasifikasi yang dilatih bersama secara *end-to-end*. Deteksi wajah menggunakan metode CNN ini memiliki performa jauh lebih baik karena kemampuan CNN dalam mengekstraksi fitur yang diskriminatif pada data. Detektor wajah berdasarkan CNN dapat dengan mudah mendeteksi wajah berbagai variasi dalam pose wajah, skala, ekspresi wajah, oklusi, dan kondisi pencahayaan. Akibatnya detektor wajah berbasis *Convolutional Neural Network* sekarang banyak digunakan dalam berbagai produk konsumen, misalnya, sistem pengawasan video, kamera digital, dan jejaring sosial (Loy, 2020).

### 2.2.2 Deteksi Wajah

Pengenalan wajah merupakan salah satu tahap yang sangat penting dimana sistem biometrik digunakan pada sistem pengenalan wajah. Proses deteksi wajah (*face detection*) merupakan tahapan awal yang dilakukan, karena memiliki fungsi



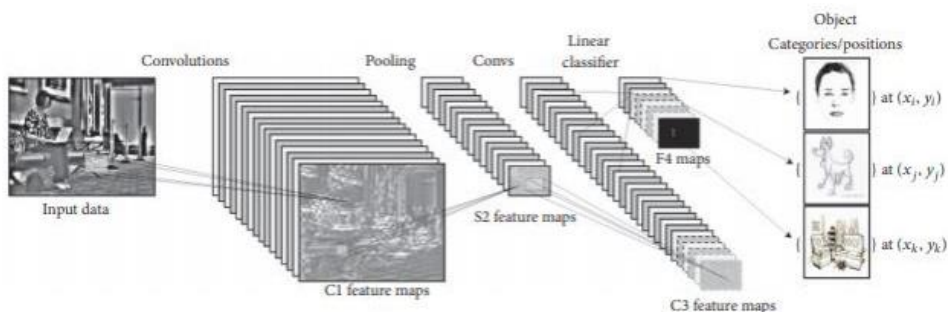
yang sangat penting sebelum dilakukan pada proses pengenalan masker wajah yang dipakai.

Hal menantang pada proses deteksi wajah adalah dengan adanya faktor-faktor yang mempengaruhi pada proses mendeteksi obyek wajah yang ada, salah satunya adalah posisi pada wajah. Pada bagian posisi wajah ini banyak hal yang perlu diperhatikan, yaitu arah wajah yang dimiliki dengan berbagai posisi wajah yang menghadap kesamping, posisi tegak, wajah yang menoleh ataupun miring dan wajah yang dilihat dari arah samping wajah. Bagian-bagian yang dimiliki pada area wajah juga menjadi salah satu pengaruhnya, seperti wajah terdapat jenggot, kumis, dan aksesoris kacamata pada wajah tersebut. Tidak hanya itu saja ekspresi yang dimiliki pada wajah seseorang, seperti tertawa, tersenyum, berbicara, sedih, dan ekspresi lainnya. Proses deteksi wajah juga dipengaruhi oleh kondisi lingkungan yang ada disekitarnya, baik itu didalam ruangan maupun diluar ruangan, bisa ditempat umum atau tempat keramaian yang dapat menutupi obyek wajah. Sehingga wajah tidak dapat terlihat dengan sempurna. Pengaruh cahaya juga mempengaruhi yang didapat dari cahaya ruangan, arah sumber cahaya, intensitas cahaya dan karakteristik dari sensor kamera dan lensa kamera yang digunakan dalam proses deteksi wajah berlangsung.

### 2.2.3 Convolutional Neural Networks

*Convolutional Neural Network (CNN)* adalah merupakan sebuah metode yang telah dilakukam pengembangan dari sebuah *Multilayer Perceptron (MLP)* yang telah dirancang untuk melakukan proses pengolahan data secara dua dimensi. *Convolutional Neural Network* bagian dari *Deep Neural Network*, yang dimana memiliki sebuah jaringan yang tinggi dan banyak dipraktekkan pada sebuah citra. Pada MLP yang ada pada sebuah kasus klasifikasi citra tidak dapat menyimpan sebuah informasi spasial dari data citra, karena hanya membaca pada setiap piksel citra tersebut sebagai fitur yang independen sehingga mendapatkan hasil akhir yang kurang tepat Eka Putra, (2016).

CNN terdiri dari tiga jenis lapisan saraf utama, yaitu, *convolutional layers*, *pooling layers*, dan *fully connected layers*. Setiap jenis lapisan memainkan peran yang berbeda, gambar dibawah menunjukkan arsitektur CNN untuk deteksi objek dalam tugas gambar. Setiap lapisan 14 CNN mengubah input volume ke volume keluaran aktivasi neuron, akhirnya mengarah ke lapisan terakhir yang sepenuhnya terhubung, menghasilkan a pemetaan data masukan ke vektor fitur 1D. CNN memiliki sangat sukses dalam aplikasi *computer vision*, seperti pengenalan wajah, deteksi objek, memperkuat penglihatan robotika, dan mobil yang bisa mengemudi sendiri (Deng and Yu, 2013).

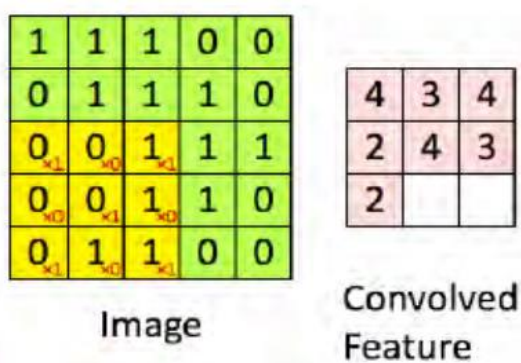


**Gambar 2.1** Arsitektur *Convolutional Neural Networks* (Deng and Yu, 2013)

### 2.2.3.1 *Convolution Layer*

Lapisan Konvolusional. Di lapisan konvolusional, CNN menggunakan berbagai kernel untuk menggabungkan seluruh gambar serta peta fitur perantara, menghasilkan berbagai peta fitur *convolution*. *Convolution Layer* merupakan proses dasar atau utama yang dilakukan pada metode CNN. Pada proses operasinya melakukan operasi konvolusi yang terletak pada output dari layer sebelumnya.

Konvolusi merupakan sebuah sebutan matematis yang berarti mengaplikasikan suatu fungsi secara berulang, fungsi yang dilakukan pada output atau pada fungsi lainnya. Dalam pengolahan citra, konvolusi diartikan sebagai kernel (kotak kuning) yang dapat dilihat pada semua *offset* pada citra yang ditunjukkan pada Gambar 2.2 pada kotak hijau yang berarti akan dijalankan konvolusi pada sebuah citra. Pergerakan kernel yaitu dengan cara dimulai dari sudut kiri bagian atas lalu ke kanan bagian bawah. Sehingga hasil akhir yang didapat pada konvolusi dari sebuah citra tersebut terlihat pada gambar disebelah kanan (Eka Putra, 2016).

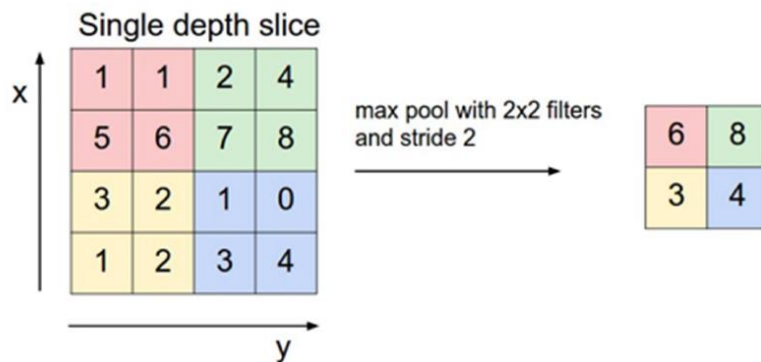


**Gambar 2.2** Proses *Convolution Layer* (Eka Putra, 2016)

Konvolusi itu sendiri memiliki tujuan dari proses yang telah dilakukan yaitu bertugas sebagai mengekstraksi sebuah fitur yang diambil dari input sebuah citra. Hasil konvolusi itu sendiri berupa transformasi *linear* yang didapat dari data input sesuai dengan informasi spasial yang terletak pada data inputnya. Berat pada layer tersebut akan menspesifikasikan sebuah kernel konvolusi yang dipakai, sehingga pada kernel konvolusi tersebut dapat dilakukan pelatihan berdasarkan input yang ada pada CNN.

### 2.2.3.2 *Subsampling Layer*

*Subsampling layer* merupakan proses mereduksi sebuah dimensi suatu informasi citra. Dalam pengolahan citra, *subsampling* bertujuan untuk meningkatkan invariansi posisi dari fitur yang dimiliki. Sebagian besar CNN memiliki sebuah tata cara *subsampling layer* yang digunakan yaitu *max pooling*. *Max pooling* memiliki peran sebagai output dari *convolution layer* menjadi sebagian grid kecil kemudian mengambil nilai optimal dari setiap grid untuk menyusun sebuah matriks citra yang sudah direduksi semacam yang ditunjukkan pada Gambar 2.3 grid yang berwarna merah, hijau, kuning, serta biru ialah kelompok grid yang hendak diseleksi nilai maksimumnya. Sehingga hasil dari proses tersebut bisa dilihat pada kumpulan grid disebelah kanannya. Proses tersebut membenarkan fitur yang didapatkan hendak sama walaupun obyek citra menghadapi translasi atau perpindahan.

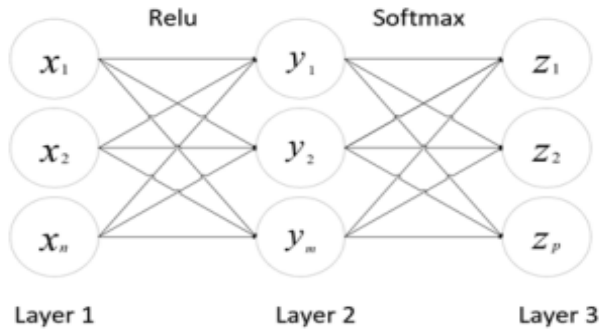


**Gambar 2.3** Proses *Max Pooling* (Eka Putra, 2016)

Pada Gambar 2.3 merupakan penggunaan *max pooling layer* yang ada pada CNN. Tujuan yang dimiliki dari proses ini untuk melakukan reduksi ukuran dari citra dan dapat digantikan sebagai *convolution layer* menggunakan stride yang sama berdasarkan pooling layer yang dimiliki. Hal ini dapat memberi dampak, yaitu proses yang mudah untuk dilakukan.

### 2.2.3.3 Fully Connected Layer

*Fully Connected Layer* merupakan sebuah proses yang memiliki fungsi sebagai klasifikasi dengan cara menggunakan *softmax* yang sama dengan proses yang diumpakan pada gambar dibawah ini.

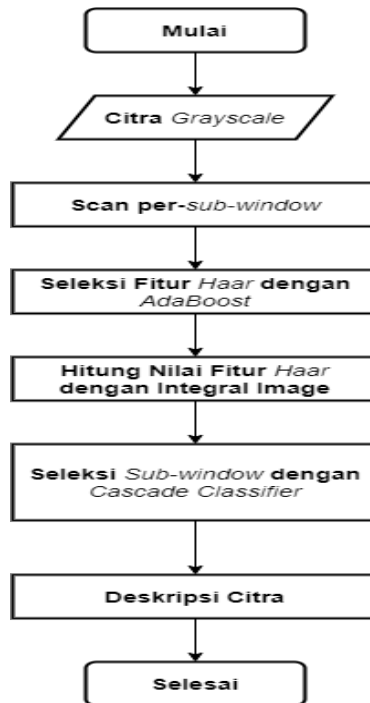


**Gambar 2.4** Lapisan Klasifikasi (*fully connected layer*) (Prasmatio, Rahmat and Yuniar, 2020)

Dari Gambar 2.4 pada bagian layer 1 akan melakukan proses *feedforwarding* yang selanjutnya akan diarahkan pada layer 2 dengan memakai fungsi dari aktivasi ReLU. Bagian layer 2 akan dilakukan proses klasifikasi dengan *softmax*. *Fully connected layer* ini sering digunakan pada multi lapisan *perceptron* yang memiliki tujuan sebagai klasifikasi. Ada perbedaan yang dimiliki antara *fully connected layer* dengan lapisan konvolusi yaitu pada neuron yang terletak di konvolusi dapat terhubung pada dserah tertentu saja yaitu bagian input. Sedangkan *fully connected layer* bekerja secara luas pada neuron yang dimiliki. Kesamaan kedua lapisan ini dapat bekerja pada dot, sehingga fungsi yang dimiliki tidak begitu berbeda (Prasmatio, Rahmat and Yuniar, 2020).

### 2.2.4 Viola Jones

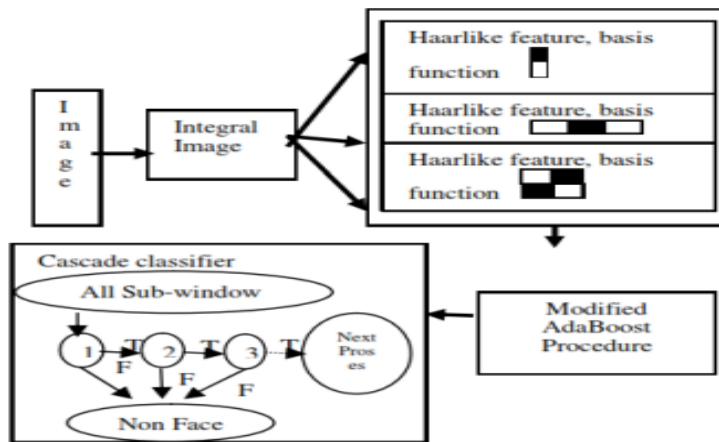
*Viola Jones* pada deteksi wajah merupakan sebuah prosedur yang mengklasifikasikan gambar atau citra berdasarkan pada nilai fitur secara sederhana. Ada banyak argumen mengenai fitur yang digunakan pada *viola jones* yang dimana fitur tersebut lebih dipakai daripada piksel secara langsung. Argumen yang paling umum adalah bahwa fitur tersebut digunakan buat mengkodekan pengetahuan *domain ad-hoc* yang susah dalam proses pembelajaran terhadap jumlah data latih yang terbatas. Argumen yang kedua dimiliki oleh *viola jones* ini memakai fitur dengan operasi jauh lebih efisien dan cepat daripada sistem berbasis *pixel* Dwisnanto et al., (2012).



**Gambar 2.5** Blog Diagram Metode Viola Jones (Dedi Ary Prasetya, 2012)

Flattening obyek pada wajah yang dicari menggunakan *Viola Jones* yang telah digambarkan pada Gambar 2.5 yang dimana sebuah citra yang dimiliki berupa grayscale akan discan secara per-sub-window untuk menemukan fitur positif menggunakan *adaboost* dan *Cascade Classifier*. Bila wajah ditemukan, maka selanjutnya akan melakukan proses penggambaran garis dengan membentuk sebuah persegi pada wajah tersebut. Ada operasi dasar yang dimiliki pada fitur jauh lebih cepat daripada pengolahan pixel. Banyaknya jumlah *haar* mewakili bentuk persegi yang telah digambarkan pada citra dan menjumlahkan seluruh piksel pada wilayah tersebut. *Viola Jones* memiliki tiga metode, metode pertama yaitu *Haar-Like Feature* yang digunakan untuk ekstraksi fitur sebagai konsep pertama deteksi wajah pada gambar. Selanjutnya jika *Haar-Like Feature* ini ditemukan pada sebuah wajah, akan diteruskan ke metode yang kedua yaitu, metode *AdaBoost* yang digunakan untuk pemilihan pengklasifikasi-pengklasifikasi lemah yang dapat digabungkan menjadi pengklasifikasi kuat. Metode ketiga adalah metode *Cascade Classifier*. Algoritma yang ada pada *Viola Jones* diperkenalkan oleh Paul Viola dan M. J. Jones yang telah memberi usulan untuk mengkombinasikan tiga metode dalam melakukan proses deteksi wajah (Viola and Jones, (2001). Tiga kontribusi algoritma *viola jones* adalah:

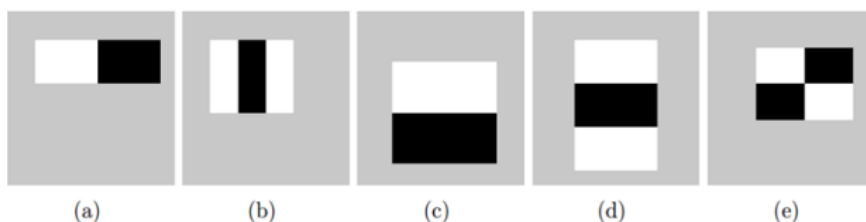
1. Menggambarkan sebuah citra baru yang disebut sebagai citra integral. Dari citra integral tersebut diharapkan dapat mengevaluasi sebuah fitur menjadi lebih cepat
2. Sebuah metode yang simple dan efisien dalam mengatur sebuah fitur serta pemilihan dengan menggunakan *adaboost*
3. Pada metode ini menggabungkan classifier secara kompleks dengan terus menerus pada suatu *Cascade Classifier*.



**Gambar 2.6** Struktur Viola Jones (Wahyusari and Haryoko, 2014)

### 2.2.4.1 Haar Like Feature

Pengkategorian pada sebuah citra berdasarkan nilai yang dimiliki fitur. Dengan adanya ini memiliki tujuan untuk melakukan proses pemisah pada sebuah citra yang tidak diperlukan, salah satunya *background* yang dimiliki pada sebuah citra tidak dihitung. Fitur ini memiliki 3 macam jenis berdasarkan jumlah yang dimiliki, yaitu persegi panjang yang terdapat terang dan gelap didalamnya. Persegi panjang yang dimaksud telah digambarkan pada gambar dibawah ini:



**Gambar 2.7** Fitur Haar Like (Suharso, 2017)

Pada Gambar 2.7 merupakan sebuah fitur dengan jumlah lima, yaitu dari bagian kiri ke kanan, pertama dan kedua memiliki duap persegi panjang, dan pada

fitur ketiga dan keempat adalah persegi panjang, sedangkan fitur yang terakhir yaitu fitur kelima terdapat empat persegi panjang. Proses menghitung sebuah nilai yang dimiliki pada fitur dengan cara mengurangi jumlah pixel pada bagian area yang berwarna hitam dengan daerah pixel berwarna putih.

Mempunyai rumus seperti dibawah ini untuk mendapatkan nilai fitur yang tepat dengan bagian jumlah kotak, B, W: Black (hitam), White (putih) berikut :

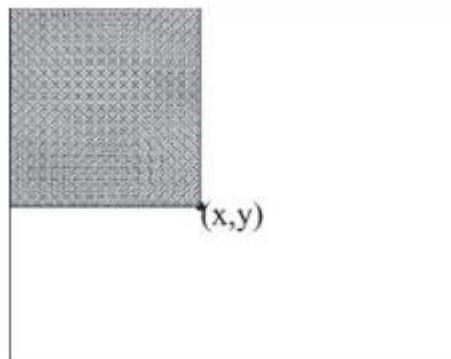
$$\text{Dua Kotak} = W - B \quad (2.1)$$

$$\text{Tiga Kotak} = W_1 + W_2 - B \quad (2.2)$$

$$\text{Empat Kotak} = (W_1 + W_2) - (B_1 + B_2) \quad (2.3)$$

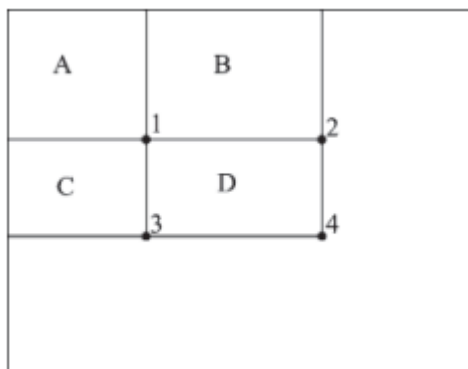
Dari proses diatas dapat memnetukan sebuah tingkat *luminance* dari citra yang akan dilakukan proses deteksi. Dari hal ini juga dapat membedakan obyek yang ada pada citra, salah satunya adalah dapat mendeskripsikan wajah pada manusia. Pada proses perhitungan ini agar lebih mudah dalam menentukan nilai fiturnya, maka metode *Viola Jones* memakai media sebagai citra integral (Septian and Fitriyani, 2014).

Pada *integral image* merupakan citra yang memiliki nilai pada setiap pikselnya hasil dari penjumlahan nilai piksel kiri atas hingga bagian kanan bawah. Pada Gambar 2.8 merupakan salah contoh yang dapat dilihat dibawah ini :



**Gambar 2.8** *Fitur Haar-Like* (Dwisnanto, Teguh and Winduratna.B, 2012)

Dalam menentukan nilai rata-rata dari sebuah piksel di bagian daerah segi empat atau daerah yang terdapat arsirannya dapat dilakukan dengan cara membagi nilai  $(x, y)$  oleh daerah segiempat Untuk menentukan nilai rata-rata piksel pada area segiempat (daerah yang diarsir) dapat dilakukan hanya dengan membagi nilai pada  $(x, y)$  oleh area segiempat. Pada Gambar 2.9 merupakan perhitungan integral salah satu daerah segiempat.



**Gambar 2.9** Fitur Haar-Like (Dwisnanto, Teguh and Winduratna.B, 2012)

Pada integral image dapat mengetahui nilai piksel pada segiempat yang dimiliki misalnya, segiempat D yang terdapat pada Gambar 2.9 dapat dilakukan dengan cara melakukan penjumlahan di area segiempat yaitu  $A+B+C$ , selanjutnya dikurangi jumlah dalam segiempat  $A+B$  dan  $A+C$ , ditambah dengan jumlah piksel didalam A. Dengan  $A+B+C+D$  adalah nilai dari integral image pada lokasi 4,  $A+B$  adalah nilai pada lokasi 2,  $A+C$  adalah nilai pada lokasi 3, dan A pada lokasi 1. Sehingga hasil dari D dapat dikomputasikan  $D = (A+B+C+D)-(A+B)-(A+C)+A$ .

#### 2.2.4.2 Ada Boost

Algoritma *adaboost* merupakan algoritma yang digunakan pada metode *Viola Jones*. *Adaboost* memiliki sebuah singkatan yaitu *Adaptive Boosting*. Algoritma ini memiliki tujuan untuk membentuk sebuah template pada obyek yang akan dilakukan proses deteksi wajah.

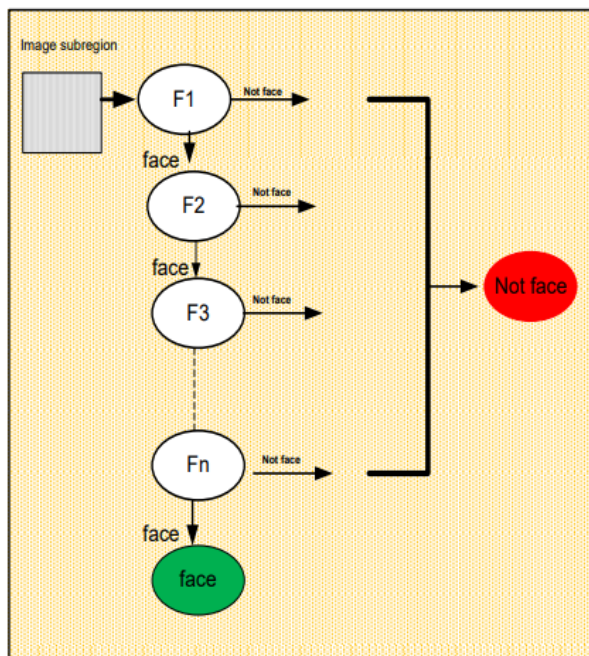
*Adaboost* merupakan gabungan dari banyaknya *classifier* lemah untuk menciptakan sebuah *classifier* yang kuat. Dengan cara menggabungkan ini *adaboost classifier* merupakan sebuah rangkaian *filter* yang efisien untuk mengklasifikasi daerah citra. Pada setiap filter terdapat satu *adaboost classifier* yang telah terpisah dari *classifier* lemah atau *filter haar*. Proses pemfilteran ini, jika ada salah satu *filter* yang gagal untuk melewati sebuah citra, maka daerah itu akan diklasifikasikan menjadi daerah bukan wajah. Tetapi ketika filter mampu melewati sebuah daerah citra hingga melewati semua proses filter yang telah dilakukan dalam rangkaian filter tersebut, dalam hal itu dapat digolongkan sebagai wajah (Suharso, 2017).

Adanya algoritma *adaboost* dapat membangun sebuah *strong classifier* dengan menkombinasikan jumlah *simple* atau *weak classifier* secara *linear*. Prinsip yang digunakan pada algoritma *adaboost* adalah *decision tree* yang dimana prosesnya melau tingkat percabangan satu ke tingkat percabangan lainnya (Wahyusari and Haryoko, 2014).



### 2.2.4.3 Cascade Classifier

Selanjutnya adalah *cascade*. Proses urutan yang dimiliki oleh *cascade* ditentukan oleh berat yang diberikan dari *adaboost*. Jumlah *filter* yang paling besar diletakkan pada proses pertama, hal ini bertujuan untuk mengapus daerah pada citra yang bukan bagian obyek wajah dengan harapan secepat mungkin pada pendeteksiannya. Algoritma Viola Jones memiliki karakteristik yaitu proses klasifikasi yang dilakukan secara bertingkat. Terdapat tiga klasifikasi pada algoritma ini yang dimana pada tiap tingkalan mengeluarkan sebuah subcitra yang bukan bagian dari wajah. Dengan dilakuan proses ini dapat mempermudah dalam menilai subcitra yang bukan wajah daripada menilai subcitra tersebut merupakan bagian dari wajah (Suharso, 2017).



**Gambar 2.10** Rangkaian Cascade Classifier (Syafira, 2017)

Gambar 2.10 merupakan rangkaian dari *cascade classifier* adalah metode dari klasifikasi bertingkat yang memiliki peran dalam menolak area pada citra yang tidak terdeteksi bagian wajah dengan *classifier* yang telah dilatih dengan algoritma *adaboost* sesuai dengan tingkat klasifikasinya. Kalsifikasi tingkat pertama, setiap inputan yang ada berupa *sub window* lalu akan dilakukan proses klasifikasi secara sederhana. Semakin bertambahnya tingkat klasifikasi akan memerlukan sebuah syarat yang lebih spesifik, maka dari itu *classifier* yang digunakan semakin kompleks. Proses ini ditujukan supaya dapat mengurangi sebuah input atau *sub*

*window* yang telah gagal pada salah satu filter yang dilewatkan. Pada area *sub window* dapat digolongkan sebagai area bukan wajah. Jika seluruh *filter* yang ada didalam *cascade classifier* terlewatkan, maka area *sub window* dianggap dapat mendeteksi area bagian wajah.

## 2.2.5 Training Cascade Object Detector

Pada proses training *Cascade Object Detector* hanya berisi beberapa yang sudah terlatih pengklasifikasi, tidak cukup untuk aplikasi deteksi wajah, *cascade layer* membutuhkan pelatihan untuk setiap pengguna, menggunakan fungsi *train CascadeObjectDetector*. *Cascade Layer* pelatihan dilakukan dengan menggunakan satu set sampel positif (jendela dengan wajah) dan satu set gambar negatif. Contoh negatifnya adalah dihasilkan secara otomatis dari kumpulan gambar negatif. Untuk mendapatkan detektor yang lebih akurat jumlah lapisan *cascade*, jenis fitur (Haar dalam penelitian ini) dan fungsinya parameter harus ditentukan. Pelatihan *Cascade Layer* (Alionte and Lazar, 2015).

Sayangnya untuk menemukan hasil deteksi yang optimal, sampai saat ini masih menjadi masalah yang sangat sulit. Dalam praktiknya, proses kerja yang sangat sederhana digunakan untuk menghasilkan klasifikasi yang sangat efisien. Setiap tahapan dalam *Cascade* dilakukan pengurangan pada tingkat positif palsu dan menurunkan tingkat deteksi. Target dipilih untuk mengurangi rasio positif palsu dan menurunkan deteksi maksimum. Pada setiap tahap dilakukan proses data latih dengan menambahkan fitur hingga deteksi target dan tingkat positif palsu terpenuhi (tingkatan ini ditentukan dengan menguji detektor pada set validasi). Selanjutnya tahapan ditambahkan sampai target keseluruhan untuk positif palsu dan tingkat deteksi terpenuhi (Viola and Jones, 2001).