

Deteksi Wajah Tersamar Menggunakan Metode VGGFace dan SVM

*¹Sri Desy Siswanti, ²Heni Puspita, ³Huda Ubaya, ⁴Selly, ⁵Dina Herdiana
^{1,2,3,5}Universitas Nurtanio Bandung, ⁴Universitas Sriwijaya Palembang

desy0712unnur@gmail.com

*Penulis Korespondensi

Diajukan : 05/03/2025

Diterima : 07/04/2025

Dipublikasi : 09/04/2025

ABSTRAK

Teknologi pengenalan wajah merupakan metode identifikasi atau verifikasi seseorang melalui gambar atau video, yang banyak digunakan di bidang keamanan, pengawasan, verifikasi identitas, serta sistem peradilan. Namun, tantangan muncul ketika seseorang, seperti DPO (Daftar Pencarian Orang), menggunakan penyamaran, baik disengaja maupun tidak. Penyamaran ini bisa berupa perubahan gaya rambut, penggunaan aksesoris seperti topi, kacamata, masker, atau bahkan riasan tebal dan operasi plastik yang mengubah struktur wajah, sehingga menyulitkan sistem dalam mengenali individu tersebut. Penelitian ini mengusulkan pengembangan sistem pengenalan wajah tersamar dengan memodifikasi arsitektur VGG pada tahap klasifikasi. Perubahan dilakukan dengan menambahkan lapisan flatten dan mengganti lapisan klasifikasi akhir dengan metode Support Vector Machine (SVM). Arsitektur VGG digunakan untuk mengekstraksi fitur wajah, sedangkan SVM berperan dalam tahap klasifikasi. Sistem ini terdiri dari empat tahap utama, yaitu pengambilan data, pengolahan data, ekstraksi fitur, dan klasifikasi. Data wajah dikumpulkan langsung melalui kamera, mencakup wajah tanpa dan dengan penyamaran dalam lima posisi sudut wajah (kanan, kiri, depan, atas, bawah). Implementasi dilakukan menggunakan library Keras, Sklearn, dan Numpy. Penyesuaian parameter SVM seperti nilai Cost (C) dan gamma (γ) turut dilakukan untuk meningkatkan akurasi. Hasil penelitian menunjukkan bahwa sistem yang dikembangkan memberikan akurasi yang lebih tinggi dibandingkan metode sebelumnya, meskipun masih terdapat beberapa kekurangan yang perlu diperbaiki di penelitian mendatang.

Kata Kunci: SVM, tersamar, VGG, wajah

I. PENDAHULUAN

Wajah adalah salah satu indikator fisiologis yang paling sederhana dan sering digunakan untuk membedakan identitas individu, karena wajah merupakan bagian tubuh manusia yang unik dan tidak bisa diduplikasi (Rizal et al., 2019). Algoritma pengenalan wajah tradisional rentan terhadap variabel situasional dan kondisi wajah seperti pose, pencahayaan, ekspresi, penuaan, heterogenitas, dan penyamaran. Meskipun beberapa variabel ini telah diteliti dengan baik, variasi penyamaran masih kurang mendapat perhatian. Penelitian dalam pengenalan wajah telah berkembang pesat dalam beberapa tahun terakhir. Akibatnya, pengenalan wajah kini digunakan dalam berbagai aplikasi, mulai dari penegakan hukum seperti kontrol perbatasan hingga aplikasi komersial seperti membuka kunci ponsel cerdas dan penandaan wajah di media social (Singh et al., 2019). Dalam skenario dunia nyata, seorang DPO (Daftar Pencarian Orang) mungkin menggunakan penyamaran, baik secara sengaja maupun tidak sengaja, untuk menyembunyikan diri atau berpura-pura menjadi orang lain. Misalnya, jenggot, kumis, dan gaya rambut yang berubah secara alami dapat menyebabkan kebingungan. Selain itu, aksesoris penyamaran seperti wig, topi, syal, helm, kerudung, kacamata hitam, atau masker dapat membuat bagian wajah terlihat berbeda. Riasan tebal

atau prosedur eksternal seperti operasi plastik juga dapat mengubah bentuk, tekstur, dan warna wajah, sehingga menyulitkan pengenalan (Razzaq et al., 2022).

CNN adalah salah satu jenis arsitektur jaringan saraf tiruan yang mampu memproses data dalam jumlah besar, termasuk data wajah. Model CNN semacam ini disebut CNN Face atau dikenal dengan metode VGGFace (Parkhi et al., 2015). Metode ini mengadaptasi arsitektur VGG16, yang terdiri dari lapisan ekstraksi fitur dan lapisan *Fully Connected* (FC) untuk klasifikasi. Penelitian tersebut menunjukkan bahwa VGGFace mampu menangani dataset yang besar dalam hal ukuran dan kompleksitas, serta menghasilkan kinerja yang sebanding dengan teknologi pengenalan wajah lainnya. Dalam penelitian tersebut, kombinasi VGG dan SVM juga diterapkan (Chen & Haoyu, 2019), metode ini mencapai tingkat akurasi sebesar 93% dan 97,47% pada dataset celebA dan LFW dengan 400 jenis wajah. Penelitian tersebut berhasil mengatasi masalah dimensi fitur dalam pengenalan wajah menggunakan Deep Learning (DL), sedangkan dalam penelitian (Nuruddin Pk et al., 2020) membahas pengenalan wajah dengan menggabungkan metode Multi-Class SVM dan HOG, yang mencapai tingkat akurasi 96% pada dataset wajah ORL dan YALE. Pada penelitian (Ch. Ramya Pravallika, n.d.) penggabungan VGG16 dengan SVM untuk klasifikasi tumor otak menghasilkan akurasi sebesar 88,59%. Pada penelitian (Margolang et al., 2023) berfokus pada deteksi penggunaan masker wajah di masa pandemi Covid-19. Penelitian ini menggunakan VGG-16 untuk ekstraksi fitur sebanyak 4.096 fitur dan dikombinasikan dengan model Multilayer Perceptron (MLP) untuk proses klasifikasi dan evaluasinya dilakukan menggunakan *10-fold Cross Validation*, dengan hasil akurasi terbaik sebesar 98,1%. Sementara itu pada penelitian (Anggraini & Yamasari, 2023) VGG16 digunakan untuk ekstraksi fitur wajah dalam klasifikasi citra wajah berdasarkan rentang usia dengan klasifikasi menggunakan metode *Artificial Neural Network* (ANN). Meskipun optimasi telah dilakukan menggunakan algoritma *Stochastic Gradient Descent* (SGD), akurasi yang dicapai sebesar 87%. Disisi lain, pada penelitian (Hidayatulloh et al., 2020) menerapkan pendekatan yang berbeda dengan menggunakan kombinasi metode *Sobel Edge Detection* dan SVM untuk pengenalan wajah berbasis deteksi tepi. Dataset yang digunakan terdiri dari 309 gambar yang diambil dari dua sumber, yaitu galeri dan foto langsung. Dengan menggunakan metode Holdout Validation dan klasifikasi berbasis One Against All, penelitian ini berhasil mencapai akurasi **100%** pada data latih. Namun, ketika diuji dengan citra baru, akurasi menurun menjadi **89%**.

Berdasarkan hal-hal tersebut maka metode dalam tulisan ini yaitu metode VGGFace dimodifikasi hanya menggunakan lapisan ekstraksi fitur yang ditambahkan 1 lapisan *flatten* untuk menggabungkan fitur yang ada dalam 1 vektor yang kemudian akan di gabungkan dengan SVM. Sistem pengenalan wajah memerlukan akurasi tinggi untuk memastikan keefektifan dan keandalannya, sehingga perlu dilakukan evaluasi kinerja dasar terhadap model yang dirancang. Berdasarkan (Chen & Haoyu, 2019; Dar & Palanivel, 2021; Deng & Zafeririou, 2019; Sudiana et al., 2021) parameter evaluasi yang digunakan dalam tulisan ini yaitu mencakup *Confusion-Matrix* dan *cross-validation* untuk SVM.

II. STUDI LITERATUR

Bidang penelitian pengenalan wajah terus berkembang, terutama dalam hal keamanan dan identifikasi individu. Beberapa teknik telah dikembangkan untuk meningkatkan keakuratan pengenalan wajah, baik dalam situasi normal maupun saat wajah mengalami perubahan karena penyamaran. Studi sebelumnya telah menunjukkan bahwa metode berbasis deep learning, seperti Convolutional Neural Network (CNN), dapat membantu dalam pengenalan wajah. Namun, ada masalah ketika wajah disamarkan, yang dapat mengaburkan karakteristik unik seseorang. Pada penelitian sebelumnya (J. Kim et al., 2019) menggunakan model CNN seperti Xception, VGG-16, dan ShallowNet untuk melakukan penelitian tentang klasifikasi wajah asli dan wajah tersamar. Penelitian tersebut menggunakan dataset Disguised Faces in the Wild (DFW) 2018, yang mencakup berbagai jenis penyamaran, termasuk perubahan aksesoris dan riasan wajah. Hasil penelitian menunjukkan bahwa metode Xception memiliki akurasi terbaik di antara model yang diuji, tetapi hanya mencapai 62%, yang masih dianggap kurang untuk aplikasi yang efektif untuk mengidentifikasi orang yang mencoba menyamarkan identitas mereka.

Sementara itu, penelitian (Dar & Palanivel, 2021) menggunakan transfer learning untuk mengevaluasi kinerja CNN dan VGG-16 dalam sistem pengenalan wajah real-time. Penelitian ini menunjukkan bahwa menggunakan transfer learning pada model VGG-16 dapat meningkatkan akurasi sistem pengenalan wajah hingga 99.37%. Namun, penelitian ini tidak memperhitungkan kasus di mana wajah disamarkan atau mengalami perubahan yang signifikan karena pengaruh faktor eksternal seperti pencahayaan dan sudut pengambilan. Dengan demikian, teknik ini belum terbukti efektif dalam mengenali wajah yang menyamar, meskipun hasilnya sangat bagus. Penelitian yang dilakukan oleh (Anggraini & Yamasari, 2023; Hidayatulloh et al., 2020; Margolang et al., 2023) membahas berbagai pendekatan dalam sistem pengenalan wajah dengan fokus dan metode yang berbeda. Meskipun keempat penelitian menggunakan algoritma klasifikasi berbasis pembelajaran mesin, terdapat perbedaan signifikan dalam tujuan, metode ekstraksi fitur, serta hasil yang diperoleh. Pada penelitian (Margolang et al., 2023) berfokus pada deteksi penggunaan masker wajah selama pandemi Covid-19. Metode yang diterapkan menggunakan VGG-16 untuk mengekstraksi 4.096 fitur dari setiap citra, yang kemudian diklasifikasikan menggunakan Multilayer Perceptron (MLP). Evaluasi dengan 10-fold Cross Validation menunjukkan akurasi tertinggi sebesar 98,1%, membuktikan bahwa metode ini sederhana namun sangat efektif dalam membedakan wajah bermasker dan tidak bermasker, tapi bukan identifikasi atau pengenalan individu secara spesifik. Berbeda dengan penelitian sebelumnya (Anggraini & Yamasari, 2023) meneliti klasifikasi usia berdasarkan citra wajah menggunakan dataset UTKFace yang berisi lebih dari 23.708 gambar dengan rentang usia 0 hingga 116 tahun. Proses ekstraksi fitur dilakukan menggunakan arsitektur VGG-16, sementara klasifikasi dilakukan dengan metode Artificial Neural Network (ANN). Optimasi menggunakan algoritma Stochastic Gradient Descent (SGD) menghasilkan akurasi sebesar 87%, yang lebih rendah dibandingkan dengan penelitian lainnya. Di sisi lain, penelitian yang dilakukan oleh (Hidayatulloh et al., 2020) menerapkan pendekatan yang berbeda dengan menggunakan kombinasi metode Sobel Edge Detection dan SVM untuk pengenalan wajah berbasis deteksi tepi. Dataset yang digunakan terdiri dari 309 gambar yang diambil dari dua sumber, yaitu galeri dan foto langsung. Dengan menggunakan metode Holdout Validation dan klasifikasi berbasis One Against All, penelitian ini berhasil mencapai akurasi 100% pada data latih. Namun, ketika diuji dengan citra baru, akurasi menurun menjadi 89%, yang disebabkan oleh variasi kualitas citra seperti pencahayaan dan jarak pengambilan gambar.

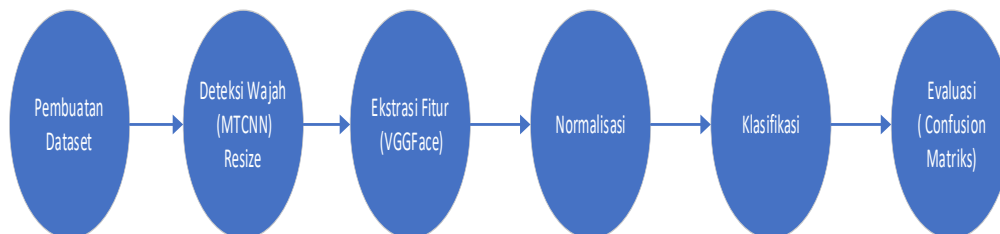
Dalam penelitian yang dilakukan oleh (Damer et al., 2022), algoritma VGGFace digunakan sebagai salah satu metode untuk mengevaluasi performa sistem pengenalan wajah dalam kondisi wajah bermasker. VGGFace adalah model pengenalan wajah yang dikembangkan oleh Visual Geometry Group (VGG) di Universitas Oxford, yang memanfaatkan jaringan saraf konvolusi (Convolutional Neural Network/CNN) untuk mengekstraksi fitur dari citra wajah. Dalam penelitian tersebut, VGGFace digunakan secara langsung tanpa kombinasi dengan metode klasifikasi tambahan, sehingga lebih rentan terhadap penurunan performa akibat tertutupnya sebagian besar fitur wajah oleh masker. Hal ini menunjukkan bahwa meskipun VGGFace memiliki kemampuan ekstraksi fitur yang baik, perlu adanya metode klasifikasi tambahan atau optimasi lebih lanjut agar model ini lebih robust terhadap kondisi wajah yang tertutup masker. Dengan demikian, penelitian Naser Damer et al. menyoroti keterbatasan VGGFace dalam menghadapi kondisi wajah bermasker, sekaligus memperkuat temuan dari penelitian lain yang menunjukkan bahwa penggabungan VGG dengan metode klasifikasi tambahan, seperti SVM atau MLP, dapat meningkatkan ketahanan model terhadap gangguan visual yang disebabkan oleh masker. Dalam penelitian (Jantarasorn et al., 2023), penggunaan arsitektur VGG16 terbukti sangat efektif dalam mengenali wajah yang tersamar. Keberhasilan ini didukung oleh beberapa faktor utama. Pertama, VGG16 memiliki arsitektur yang dalam dengan 13 lapisan konvolusi yang memungkinkan model mengekstraksi fitur mendalam dari area wajah yang tidak tertutup, seperti mata, alis, dan dahi. Hal ini sangat penting dalam kondisi penyamaran, di mana fitur utama wajah yang terlihat menjadi sangat terbatas. Kedua, penerapan teknik cropping area bawah wajah yang dilakukan dalam penelitian ini membuat model fokus pada bagian atas wajah, yang tidak tertutup oleh masker atau aksesori penyamaran lainnya. Dengan pendekatan ini, VGG16 mampu memperkuat pola-pola penting di area mata dan dahi, yang menjadi kunci utama dalam proses identifikasi wajah tersamar. Selain itu, kemampuan generalisasi VGG16 yang dilatih menggunakan dataset besar seperti ImageNet membuatnya mampu

beradaptasi dengan berbagai bentuk penyamaran, pencahayaan, dan sudut pengambilan gambar. Model ini juga menunjukkan stabilitas yang tinggi, dengan akurasi yang mencapai 100% pada rasio dataset 80:10:10, serta tetap konsisten tinggi pada rasio lainnya, yaitu 99,57% dan 98,65%. Hasil penelitian ini membuktikan bahwa VGG16 mampu menjaga performa yang optimal bahkan saat menghadapi penyamaran kompleks, seperti penggunaan kacamata, wig, masker, hingga topi. Dengan keunggulan dalam menyoroti fitur-fitur penting di area wajah yang tetap terlihat, VGG16 menjadi pilihan yang sangat tepat untuk diterapkan dalam sistem keamanan dan pengawasan yang membutuhkan identifikasi individu di tengah kondisi penyamaran. Berdasarkan hasil penelitian menunjukkan bahwa menggunakan kombinasi CNN dan SVM lebih akurat daripada menggunakan CNN secara mandiri. Namun, penelitian ini tidak memeriksa teknik yang digunakan dalam kondisi real-time, dan juga tidak mempertimbangkan masalah yang muncul dalam situasi di mana wajah tersamar, misalnya, ketika seseorang mengenakan aksesoris atau ketika karakteristik wajah berubah karena faktor eksternal.

Beberapa penelitian sebelumnya menunjukkan bahwa penelitian masih menunjukkan kekurangan dalam pengenalan wajah tersamar, terutama dalam hal meningkatkan keakuratan klasifikasi dalam kasus di mana wajah disamar dengan cara yang kompleks. Akibatnya, penelitian ini menyarankan metode baru yang menggabungkan arsitektur VGGFace yang dimodifikasi dengan tambahan lapisan flatten sebelum diklasifikasikan menggunakan SVM. Berbeda dengan penelitian sebelumnya, pendekatan yang dikembangkan dalam penelitian ini juga menggunakan keunggulan ekstraksi fitur CNN serta menggunakan SVM sebagai classifier yang lebih sesuai dengan variasi data wajah yang sangat berubah.

III. METODE

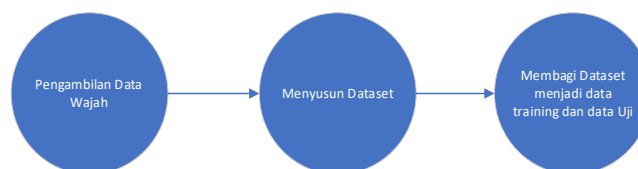
Penjelasan menyeluruh dari metode yang dilaksanakan ditunjukkan pada gambar 1, terdapat 4 langkah yaitu pembuatan dataset, deteksi wajah+resize, ekstraksi fitur, normalisasi, klasifikasi dan evaluasi.



Gambar 1. Metode Penelitian

3.1 Pembuatan Dataset

Pembuatan dataset adalah proses pembuatan data input berupa data wajah yang akan digunakan oleh sistem pengenalan wajah. Pembuatan dataset ini yang terdiri dari 3 langkah yaitu pengambilan data wajah, menyusun dataset dan membagi dataset menjadi data latihan dan data uji. Hal tersebut ditunjukkan pada gambar 2.



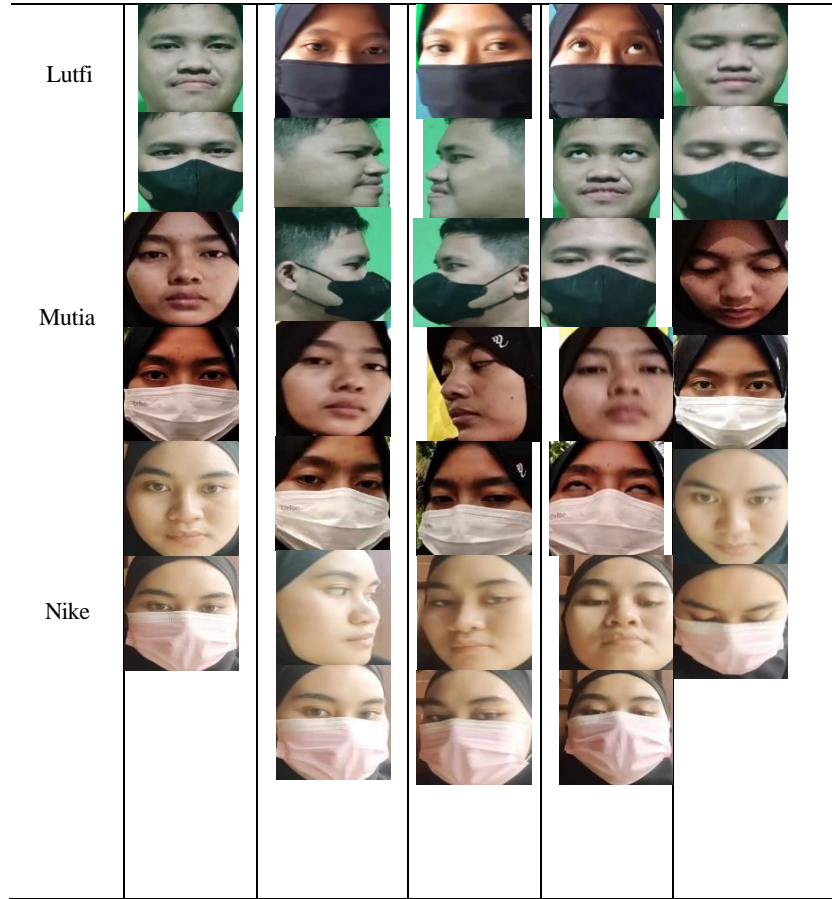
Gambar 2. Langkah Pembuatan Dataset

1. Pengambilan Data Wajah. Dalam langkah membuat dataset wajah dengan cara mengambil data berupa video dengan format .mp4 dimana data wajah dalam video tersebut terdiri dari wajah laki-laki dan perempuan tanpa tersamar dan tersamar dengan 5 posisi wajah yaitu wajah menghadap ke kanan, ke kiri, ke depan, ke atas dan ke bawah. Data tanpa tersamar

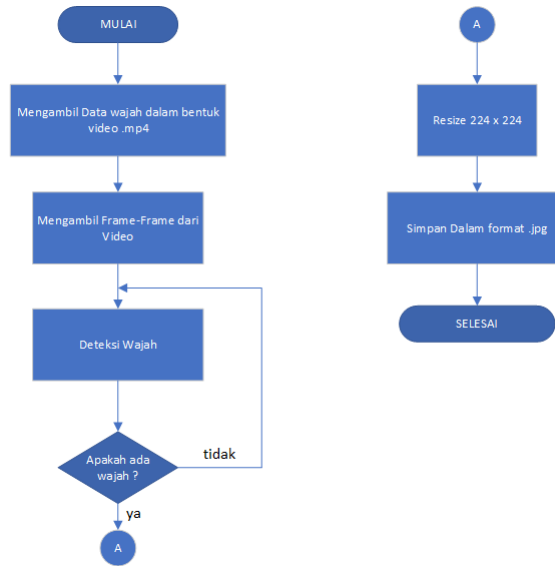
berupa data wajah tanpa ada halangan, sedangkan data tersamar berupa data wajah yang terhalang menggunakan masker dengan latar belakang dan pencahayaan yang berbeda. Data video tersebut diambil dari 10 orang yang terdiri dari 5 laki-laki dan 5 perempuan, sehingga terdapat 10 kelas. Adapun jumlah data video yang akan digunakan sebanyak 50 buah. Setiap videonya diambil 20 frame per video, sehingga di dapat total data wajah yang digunakan 1000 gambar yang ditunjukkan pada tabel 1, dimana pada tabel tersebut terdapat 10 kelas, yaitu kelas-1 (Aji), kelas-2 (Alif), kelas-3 (Asri), kelas-4 (Dela), kelas-5 (Iqbal), kelas-6 (Leni), kelas-7 (Lisa), kelas-8 (Lutfi), kelas-9 (Mutia), kelas-10 (Nike).

Tabel 1. Video dalam Bentuk .mp4

Kelas	Frame / Pose data wajah				
	1	2	3	4	5
Aji					
Alif					
Asri					
Dela					
Iqbal					
Leni					
Lisa					



2. Penyusunan Dataset. Proses menyeluruh dari penyusunan dataset ditunjukkan flowchart pada gambar 3. Dalam langkah ini mengambil frame dari video dan deteksi wajah menggunakan metode *Multi-task Cascaded Convolutional Network (MTCNN)* sehingga menghasilkan frame-frame data wajah. *Resize frame* akan dilakukan jika dalam frame tersebut terdeteksi wajah menjadi 224 x 224 dan disimpan dalam format gambar .jpg.
3. Pembagian Dataset. Dari 50 video untuk setiap videonya diambil 20 frame per video menggunakan MTCNN sehingga di dapat total data 1000 gambar wajah. Data tersebut dibagi menjadi 800 data latih dan 200 data uji yang terdiri dari data wajah tanpa tersamar dan wajah yang tersamar, dimana pembagian data tersebut secara manual. Contoh data video ditunjukkan pada gambar 4.1 dan 4.2 untuk kelas Ajim,yang diolah menjadi frame.



Gambar 3. Proses Penyusunan Dataset



Gambar 4.1 Data Video Ajim3.mp4



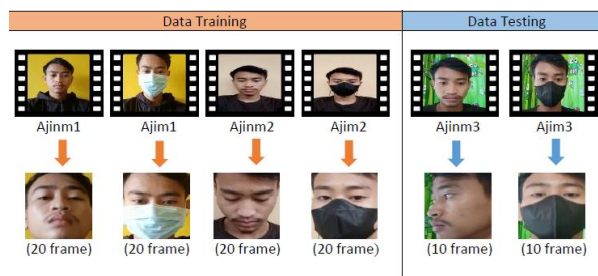
Gambar 4.2 Frame dari Video Ajim3.mp4

Pada tabel 2 ditunjukkan dataset yang dipakai dalam penelitian, dimana total data tersebut 1000 gambar yang dibagi menjadi 800 data latih dan 200 data uji.

Tabel 2 Rincian 1000 Dataset

Data Gabungan (Data Latih) Video ke-1 dan Video ke-2	800
Data Uji Video Ke-3	200
Total	1000

Contoh pembagian data latih dan data uji ditunjukkan pada gambar 5, dimana kelas Aji dengan video Ajinm1.mp4 yang diubah menjadi frame, kemudian dipilih dan dibagi menjadi data latih dan data uji.



Gambar 5. Pembagian Dataset Menjadi Data Latih dan Data Uji.

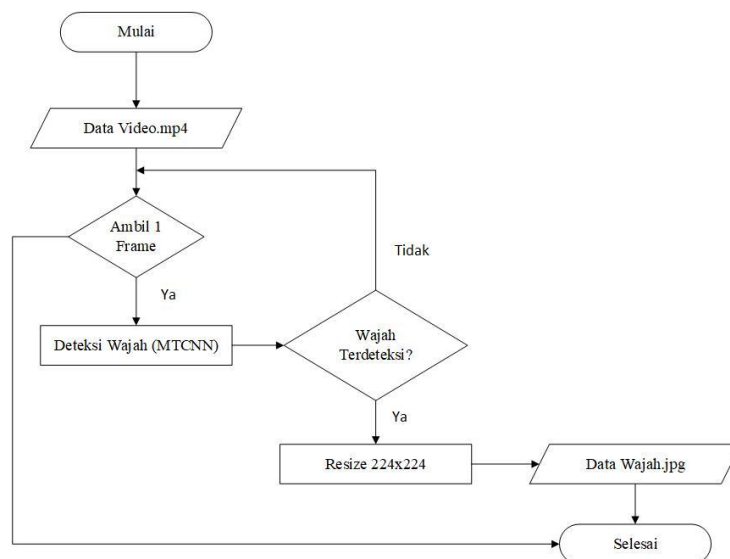
Tabel 3. Jumlah Data Latih dan Data Uji

Kelas	Label	Pose Wajah	Jumlah Data Latih	Jumlah Data Uji
Aji	0	5	80	20
Alif	1	5	80	20
Asri	2	5	80	20
Dela	3	5	80	20
Iqbal	4	5	80	20
Leni	5	5	80	20
Lisa	6	5	80	20
Lutfi	7	5	80	20
Mutia	8	5	80	20
Nike	9	5	80	20
Jumlah Data			800	200

Pada tabel 3 menunjukkan jumlah pose wajah sebanyak 5 pose yaitu ke depan, kekanan, ke kiri, ke atas dan ke bawah. Dataset tersebut terdiri dari 10 kelas yang dibagi menjadi data latih masing-masing kelas 80 gambar sehingga jumlah data latih 800 gambar dan data uji masing-masing kelas 20 gambar sehingga jumlah data uji 200 gambar. Kelas masing-masing diberi label 0 sampai 9 yang disimpan pada database dalam bentuk array dengan format *.npz*.

3.2 Deteksi Wajah

Deteksi wajah yang digunakan adalah *Multi-Task Cascaded Convolutional Network* (MTCNN), langkah yang dilakukan dalam proses ini yaitu mendapatkan piramida gambar dengan mengolah setiap frame dalam video menjadi skala yang berbeda. Gambar tersebut di ubah ukurannya menjadi 224 x 224, kemudian disimpan dalam format *.jpg* sehingga menghasilkan berupa gambar wajah dalam format *.jpg*. Langkah proses deteksi wajah menggunakan MTCNN dan resize ditunjukkan pada gambar 6. Proses deteksi wajah menggunakan metode CNN terdapat 3 tahap yaitu PNet, RNet dan ONet ((Jose et al., 2019; Ma & Wang, 2018; Zhang et al., 2016).

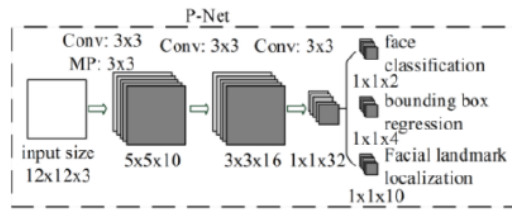


Gambar 6 . Diagram Alir Proses Deteksi Wajah Menggunakan MTCNN

P-Net (Proposal Network)

Probabilitas wajah disetiap boundingbox dan kalibrasi vektor koordinat dari boundingbox wajah tersebut dilaksanakan pada *Fully Convolutional Network*. Kemudian *Non-Maximum Supression* (NMS digabungkan dengan kandidat boundingbox yang tumpang tindih

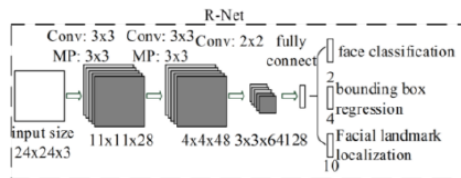
tersebut(Salscheider, 2020) sehingga menghasilkan gambar yang tumpang tindih untuk diproses pada lapisan berikutnya seperti yang ditunjukkan pada gambar 7.



Gambar 7. Struktur P-Net

R-Net (Refine Network)

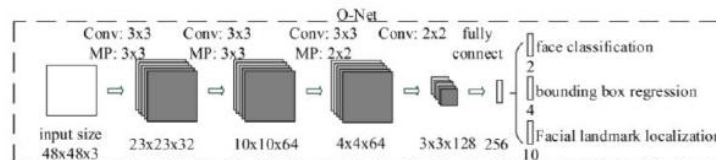
CNN memeriksa setiap kandidat boundingbox wajah yang dikumpulkan dan menyingkirkan yang dianggap salah. Setelah kalibrasi boundingbox, kandidat tumpang tindih digabungkan dengan NMS seperti yang dilakukan proses sebelumnya. Hal tersebut ditunjukkan pada gambar 8 menunjukkan hal ini.



Gambar 8 Struktur R-Net

O-Net (Output Network)

Tahap ketiga hampir identik dengan tahap kedua. Pada tahap ketiga, *landmark* wajah (posisi mata, hidung, dan mulut) dibuat dan vektor koordinat boundingbox wajah dibuat untuk menghasilkan gambar wajah yang lebih rinci, seperti yang ditunjukkan pada gambar 9.



Gambar 9. Struktur O-Net

Proses untuk menghasilkan piramida gambar menggunakan algoritma seperti yang ditunjukkan pada gambar 10 sehingga resolusi gambar minimal 12 pixel. Nilai tersebut di konvolusi dengan filter 12 x 12 pixel untuk mencari wajah, sehingga menghasilkan piramida gambar berupa gambar dengan 10 skala yang berbeda seperti yang ditunjukkan pada gambar 11.

```
width, height = image.size
# if this value is too low the algorithm will us
min_face_size = 20.0

min_length = min(height, width)

min_detection_size = 12
factor = 0.707 # sqrt(0.5)

# scales for scaling the image
scales = []

# scales the image so that
# minimum size that we can detect equals to
# minimum face size that we want to detect
m = min_detection_size/min_face_size
min_length *= m

factor_count = 0
while min_length > min_detection_size:
    scales.append(m**factor**factor_count)
    min_length *= factor
    factor_count += 1
```

Gambar 10. Fungsi Untuk Mendapatkan Piramida Gambar



Gambar 11. Piramida Gambar

Setelah mendapatkan piramida gambar gambar tersebut di normalisasikan yang menggunakan rumus 1 untuk mendapatkan range nilai -1 sampai 1.

$$\text{Normalisasi data} = (\text{data} - 127,5) \times 0.0078125 \quad \dots(1)$$

Setiap gambar piramida mengalami konvolusi pada tahap awal P-Net. Ini dilakukan dengan menggunakan kernel 12x12 dengan dua langkah yang bergerak mengelilingi gambar untuk menentukan apakah ada wajah atau tidak.



Gambar 12. Proses Filter Terhadap Frame Wajah

Hasil dari proses tersebut ditunjukkan pada gambar 12, dimana akan menghasilkan nilai probabilitas 180 bagian yang di peroleh dari nilai threshold 0,6. Setelah itu menentukan koordinat boundingbox berdasarkan posisi data kandidat wajah. Hasil tersebut kemudian diolah menggunakan NMS dengan nilai ambang batasnya 0.6 sehingga akan menghasilkan boundingbox yang saling tumpang tindih. Hal tersebut ditunjukkan pada gambar 13.



Gambar 13. BoundingBox Wajah dan Hasil NMS

Setelah itu, gambar akan di proses ke stage kedua yaitu R-Net, yang sebelumnya ukurannya diubah menjadi 24 x 24 p. Pada stage R-Net untuk mendapatkan nilai probabilitasnya dengan nilai threshold 0.7 sehingga di dapatkan hasil gambar wajah pada gambar 14.



Gambar 14. Kandidat Wajah R-Net Setelah Threshold

Hasil dari gambar 14 akan diproses ke stage selanjutnya yaitu O-Net, dimana sebelumnya gambar tersebut di resize menjadi 48 x 48 p. Nilai threshold pada stage ini adalah 0.7 untuk mendapatkan

nilai probabilitas sehingga mendapat kandidat wajah seperti yang ditunjukkan pada gambar 15a dan 15b.



Gambar 15.

a. BoundingBox wajah setelah O-Net b. Kandidat wajah O-Net setelah NMS

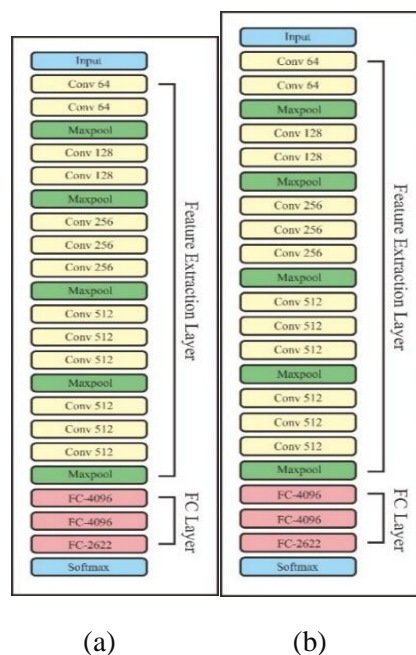
3.3 Ekstraksi Fitur Wajah

Ekstraksi fitur yang digunakan dalam tulisan ini adalah VGG Face yang merupakan pengembangan dari arsitektur VGG16. Sebenarnya arsitektur ini merupakan arsitektur CNN (Parkhi et al., 2015) yang terdiri dari lapisan ekstraksi fitur dan lapisan *fully connected* dengan total keseluruhan 23 lapisan. Tetapi pada (Chen & Haoyu, 2019; Guo et al., 2017; Parkhi et al., 2015) bahwa VGGFace dapat dimodifikasi pada bagian klasifikasi sehingga dalam tulisan ini lapisan VGGFace yang digunakan hanya lapisan ekstraksi fitur, sedangkan bagian klasifikasi di ganti menggunakan *Support Vector Machine (SVM)*. Hal ini ditunjukkan pada gambar 16a dan gambar 16b, dimana di lapisan akhir ditambahkan lapisan *flatten* untuk menggabungkan fitur menjadi satu nilai buah vektor. Data wajah yang diperoleh dari MTCNN diproses menggunakan VGGFace dengan langkah-langkah seperti dalam tabel 4.

3.4 Normalisasi

Sebelum proses klasifikasi, fitur yang telah didapat di normaliasi supaya semua fitur berada pada skala yang sama, sehingga dalam proses klasifikasi tidak ada satu fitur yang mendominasi fitur lainnya Dalam tulisan ini normaliasi yang digunakan adalah normaliasi L2 (Shepley, 2019) yang ditunjukkan pada gambar 17. Pada gambar tersebut dijelaskan bahwa fitur wajah yang telah di dapat akan di normalisasi, karena fitur yang diperoleh 25088 maka persamaan untuk mendapatkan nilai normalisasinya adalah rumus 2 (Fuad et al., 2021) .

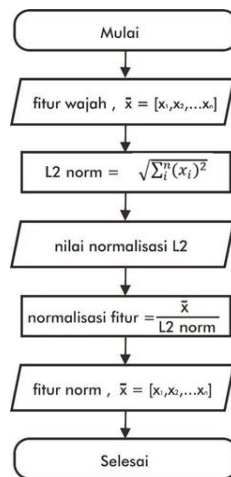
$$L2\ norm = \sqrt{\sum^{25088} (x_i)^2} = \sqrt{(x_1)^2 + (x_2)^2 + \dots + (x_{25088})^2} \quad \dots\dots(2)$$



Gambar 16. Arsitektur VGGFace a. Arsitektur VGGFace asli b. Arsitektur VGGFace Modifikasi

Tabel 4. Arsitektur VGGFace Yang digunakan

Layer	Nama	Tipe	Hasil
1	Input_1	Input Layer	224x224x3
2	Conv1_1	Conv 2D	224x224x64
3	Conv1_2	Conv 2D	224x224x64
4	Pool1	Max Pooling 2D	112x112x64
5	Conv2_1	Conv 2D	112x112x128
6	Conv2_2	Conv 2D	112x112x128
7	Pool2	Max Pooling 2D	56x56x128
8	Conv3_1	Conv 2D	56x56x256
9	Conv3_2	Conv 2D	56x56x256
10	Conv3_3	Conv 2D	56x56x256
11	Pool3	Max Pooling 2D	28x28x256
12	Conv4_1	Conv 2D	28x28x512
13	Conv4_2	Conv 2D	28x28x512
14	Conv4_3	Conv 2D	28x28x512
15	Pool4	Max Pooling 2D	14x14x512
16	Conv5_1	Conv 2D	14x14x512
17	Conv5_2	Conv 2D	14x14x512
18	Conv5_3	Conv 2D	14x14x512
19	Pool5	Max Pooling 2D	7x7x512
20	Flatten	Flatten	25088



Gambar 17. Diagram Alir Normaliasi Fitur Wajah

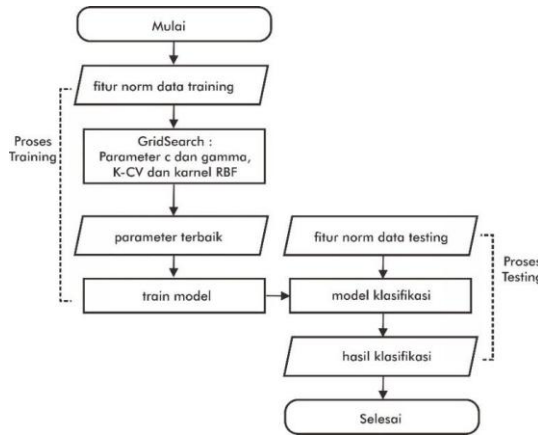
3.5 Klasifikasi Wajah

Pada tahap ini, nilai fitur yang telah dinormalisasi diklasifikasikan menurut kelasnya dengan menggunakan kernel RBF dan metode SVM.(S. K. Kim et al., 2010). Kernel RBF digunakan dalam tulisan ini karena memungkinkan klasifikasi data fitur wajah yang tidak linier dan memiliki lebih dari dua kelas. Kernel RBF memerlukan pengaturan dua parameter untuk mencapai tingkat akurasi terbaik. Parameter C dan gamma ditentukan melalui *Grid Search*, nilai parameter yang digunakan dalam klasifikasi ini yaitu C =1, gamma = 1 karena nilai tersebut memperoleh 97,64 % seperti yang ditunjukkan pada tabel 5.

Tabel 5. Cross Validation dengan C=1 dan gamma = 1

	Split 1	Split 2	Split 3	Split 4	Split 5	Rata - Rata
Score	99,4%	98,1%	97,5%	98,8%	94,4%	97,64%
Training time (s)	21.1	21	20.7	20.6	19	20.48

Proses klasifikasi terdiri dari dua bagian yaitu proses latih dan uji seperti yang ditunjukkan pada gambar 18. Pada proses latih, tahap awal dilakukan memberikan input berupa fitur normalisasi, kemudian dilakukan *Grid Search* dengan parameter C dan gamma dari tabel 5. Hasil dari proses latih ini akan dijadikan model klasifikasi yang nantinya akan digunakan untuk proses uji. Proses uji dilakukan menggunakan input fitur normalisasi yang berasal dari data latih yang selanjutnya akan diprediksi data uji sesuai dengan kelasnya masing-masing.



Gambar 18. Diagram Alir Klasifikasi Wajah

IV. HASIL DAN PEMBAHASAN

Pengujian dalam kasus ini pun dilakukan dua tahapan yaitu proses latih dan proses uji. Pada proses latih, model yang akan diuji dilatih terlebih dahulu, jumlah data latih yang sebanyak 800 data gambar. Proses uji, mengujikan data yang sudah di latih dan mengujikan data yang tidak dilatih. Hal ini dilakukan untuk mendapatkan nilai akurasi dari sistem pengenalan wajah tersamar yang menggunakan tabel evaluasi kinerja model klasifikasi yaitu *confusion matrix*.

Pada tabel 6 menunjukkan tabel confusion matrix hasil pengujian untuk evaluasi hasil dari klasifikasi SVM yang menggunakan kernel RBF. Pada tabel tersebut terlihat hasil actual-prediksi dari masing-masing kelas, sehingga didapat nilai True Positif (TP), True Negatif (TN), False Positif (FP) dan False Negatif (FN). Nilai-nilai tersebut digunakan untuk mendapat nilai akurasi dari masing-masing kelas seperti ditunjukkan pada tabel 7.

Tabel 6. Confusion Matrix Hasil Uji

Prediksi	Aktual									
	0	1	2	3	4	5	6	7	8	9
0	20	0	0	0	0	0	0	0	0	0
1	0	20	0	0	0	0	0	0	0	0
2	0	0	20	0	0	0	0	0	0	0
3	1	0	0	18	1	0	0	0	0	0
4	0	0	0	0	20	0	0	0	0	0
5	0	0	0	0	0	20	0	0	0	0
6	1	0	0	0	2	0	16	0	1	0
7	0	0	0	0	0	0	0	20	0	0
8	0	0	0	0	0	0	0	0	20	0
9	2	0	0	0	0	0	0	0	0	18

Berdasarkan hasil tersebut diperoleh akurasi 98,4%, dimana hasil ini unggul dibandingkan dengan yang lain dalam pengenalan wajah tersamar, sebagaimana ditunjukkan pada tabel 8. Pada tabel tersebut menunjukkan nilai akurasi pengenalan wajah dengan metode yang digunakannya, seperti pada penelitian akurasi tinggi (Dar & Palanivel, 2021; Guo et al., 2017) tapi yang diujinya wajah tanpa tersamar, sedangkan (J. Kim et al., 2019) mendapatkan nilai akurasi yang lebih rendah

Tabel 7. Nilai Akurasi Setiap Kelas

Label	Wajah	Accuracy
0	Aji	98%
1	Alif	100%
2	Asri	100%
3	Dela	99%
4	Iqbal	98,5%
5	Leni	100%
6	Lisa	98%
7	Lutfi	100%
8	Mutia	99,5%
9	Nike	99%
Rata - Rata		98,4%

Tabel 8. Perbandingan Nilai Akurasi

No	Judul Paper	Metode	Akurasi / Keberhasilan (%)
1	Disguised Face Recognition using Deep Learning (Jantarasorn et al., 2023)	DCNN (VGG16, CNN)	87%
2	Classifying Genuine Face Images from Disguised(J. Kim et al., 2019)	CNN (custom)	83% – 88%
3	Masked Face Recognition Human_Machine(Damer et al., 2022)	VGG16 + SVM	94.2%
4	Pengenalan Masker Wajah Menggunakan VGG-16 dan Multilayer Perceptron(Margolang et al., 2023)	VGG16 + MLP	92.31%
5	Performance Evaluation of CNNs and VGG on Real-Time Face Recognition(Dar & Palanivel, 2021)	VGG-16, CNN, VGG-Face	VGG16: 91%
6	Pengenalan Wajah dengan Algoritma Support Vector Machine (SVM) (Hidayatulloh et al., 2020)	Ekstraksi fitur + SVM	92%
7	Deteksi Wajah Tersamar Menggunakan Metode VGGFace dan SVM	VGG16+SVM	98,4%

V. KESIMPULAN

Sistem pengenalan wajah tersamar yang menggabungkan metode VGGFace dan Support Vector Machine (SVM) telah menunjukkan kinerja yang lebih unggul dibandingkan metode lainnya. Meskipun demikian, sistem ini masih menghadapi beberapa kekurangan. Data uji yang digunakan belum cukup kompleks, terutama untuk kasus wajah yang tertutupi oleh jenggot, kumis, atau alat penyamaran lainnya yang menutupi sebagian dari wajah. Untuk meningkatkan performa

sistem dalam mengenali wajah tersamar, diperlukan data uji yang lebih beragam dan kompleks. Selain itu, penambahan algoritma yang lebih canggih dan pengaturan optimal parameter SVM seperti C dan gamma sangat dibutuhkan. Dengan langkah-langkah ini, diharapkan sistem pengenalan wajah tersamar dapat mencapai nilai akurasi yang lebih tinggi dan menjadi lebih andal dalam berbagai kondisi penyamaran wajah.

VII. REFERENSI

- Anggraini, L., & Yamasari, Y. (2023). Klasifikasi Citra Wajah Untuk Rentang Usia Menggunakan Metode Artificial Neural Network. *Journal of Informatics and Computer Science (JINACS)*, 5(02), 185–192. <https://doi.org/10.26740/jinacs.v5n02.p185-192>
- Ch. Ramya Pravallika, R. B. (n.d.). *Image Processing Based Brain Tumor Classification Using VGG16 Compared with SVM to Improve Accuracy*.
- Chen, H., & Haoyu, C. (2019). Face Recognition Algorithm Based on VGG Network Model and SVM. *Journal of Physics: Conference Series*, 1229(1). <https://doi.org/10.1088/1742-6596/1229/1/012015>
- Damer, N., Boutros, F., Süßmilch, M., Fang, M., Kirchbuchner, F., & Kuijper, A. (2022). Masked face recognition: Human versus machine. *IET Biometrics*, 11(5), 512–528. <https://doi.org/10.1049/bme2.12077>
- Dar, S. A., & Palanivel, S. (2021). Performance Evaluation of Convolutional Neural Networks (CNNs) And VGG on Real Time Face Recognition System. *Advances in Science, Technology and Engineering Systems Journal*, 6(2), 956–964. <https://doi.org/10.25046/aj0602109>
- Deng, J., & Zafeririou, S. (2019). Arcface for disguised face recognition. *Proceedings - 2019 International Conference on Computer Vision Workshop, ICCVW 2019*, 485–493. <https://doi.org/10.1109/ICCVW.2019.00061>
- Fuad, M. T. H., Fime, A. A., Sikder, D., Iftae, M. A. R., Rabbi, J., Al-Rakhmi, M. S., Gumaei, A., Sen, O., Fuad, M., & Islam, M. N. (2021). Recent advances in deep learning techniques for face recognition. *IEEE Access*, 9(July), 99112–99142. <https://doi.org/10.1109/ACCESS.2021.3096136>
- Guo, S., Chen, S., & Li, Y. (2017). Face recognition based on convolutional neural network & support vector machine. *2016 IEEE International Conference on Information and Automation, IEEE ICIA 2016, August*, 1787–1792. <https://doi.org/10.1109/ICInfA.2016.7832107>
- Hidayatulloh, M. S., Permana, A. Y., Kristanto, H., & Wajah, P. (2020). Pengenalan Wajah dengan Algoritma Support Vector Machine dan Sobel Edge Detection Berbasis Computer Vision dan Caffe Framework. *Jurnal Ilmiah Komputasi*, 19(4), 535–544. <https://doi.org/10.32409/jikstik.19.4.372>
- Jantarasorn, N., Whasphuttisit, J., & Jitsakul, W. (2023). Face Recognition using Deep Learning. *7th International Conference on Information Technology, InCIT 2023*, 11(V), 470–474. <https://doi.org/10.1109/InCIT60207.2023.10413069>
- Jose, E., Greeshma, M., Mithun Haridas, T. P., & Supriya, M. H. (2019). Face Recognition based Surveillance System Using FaceNet and MTCNN on Jetson TX2. *2019 5th International Conference on Advanced Computing and Communication Systems, ICACCS 2019*, 608–613. <https://doi.org/10.1109/ICACCS.2019.8728466>

- Kim, J., Han, S., & Woo, S. S. (2019). Classifying Genuine Face images from Disguised Face Images. *Proceedings - 2019 IEEE International Conference on Big Data, Big Data 2019*, 6248–6250. <https://doi.org/10.1109/BigData47090.2019.9005683>
- Kim, S. K., Park, Y. J., Toh, K. A., & Lee, S. (2010). SVM-based feature extraction for face recognition. *Pattern Recognition*, 43(8), 2871–2881. <https://doi.org/10.1016/j.patcog.2010.03.008>
- Ma, M., & Wang, J. (2018). Multi-View Face Detection and Landmark Localization Based on MTCNN. *Proceedings 2018 Chinese Automation Congress, CAC 2018*, 4200–4205. <https://doi.org/10.1109/CAC.2018.8623535>
- Margolang, K. F., Riyadi, S., Rosnelly, R., & -, W. (2023). Pengenalan Masker Wajah Menggunakan VGG-16 dan Multilayer Perceptron. *Jurnal Telematika*, 17(2), 80–87. <https://doi.org/10.61769/telematika.v17i2.519>
- Nuruddin Pk, A. M., Ding, X., & Page, T. (2020). An Integrated Approach for Face Recognition Using Multi-class SVM. *2020 IEEE 5th International Conference on Cloud Computing and Big Data Analytics, ICCCBDA 2020*, 398–402. <https://doi.org/10.1109/ICCCBDA49378.2020.9095692>
- Parkhi, O. M., Vedaldi, A., & Zisserman, A. (2015). Deep Face Recognition. *Proceedings of the British Machine Vision Conference 2015, Section 3*, 41.1–41.12. <https://doi.org/10.5244/C.29.41>
- Razzaq, A. N., Ghazali, R., El Abbadi, N. K., & Dosh, M. (2022). A Comprehensive Survey on Face Detection Techniques. *Webology*, 19(1), 613–628. <https://doi.org/10.14704/web/v19i1/web19044>
- Rizal, R. A., Girsang, I. S., & Prasetyo, S. A. (2019). Klasifikasi Wajah Menggunakan Support Vector Machine (SVM). *REMIK (Riset Dan E-Jurnal Manajemen Informatika Komputer)*, 3(2), 1. <https://doi.org/10.33395/remik.v3i2.10080>
- Salscheider, N. O. (2020). FeatureNms: Non-maximum suppression by learning feature embeddings. *Proceedings - International Conference on Pattern Recognition*, 7848–7854. <https://doi.org/10.1109/ICPR48806.2021.9412930>
- Shepley, A. J. (2019). *Deep Learning For Face Recognition: A Critical Analysis*. <http://arxiv.org/abs/1907.12739>
- Singh, M., Chawla, M., Singh, R., Vatsa, M., & Chellappa, R. (2019). Disguised faces in the wild 2019. *Proceedings - 2019 International Conference on Computer Vision Workshop, ICCVW 2019*, 542–550. <https://doi.org/10.1109/ICCVW.2019.00067>
- Sudiana, D., Rizkinia, M., & Alamsyah, F. (2021). Performance Evaluation of Machine Learning Classifiers for Face Recognition. *2021 17th International Conference on Quality in Research (QIR): International Symposium on Electrical and Computer Engineering*, 71–75. <https://doi.org/10.1109/QIR54354.2021.9716171>
- Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016). Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks. *IEEE Signal Processing Letters*, 23(10), 1499–1503. <https://doi.org/10.1109/LSP.2016.2603342>