Deteksi Wajah Palsu Berbasis Transformer Ringan Dengan Analisis Perbandingan Terhadap Vision Transformer

Ahmad Amirudin[1]
Fakultas Informatik
Universitas Telkom
Bandung
amirlabs@students.telkomuniversity.ac.id

Gamma Kosala[2]
Fakultas Informatika
Universitas Telkom
Bandung
gammakosala@telkomuniversity.ac.id

Abstrak

Penelitian terkait sistem deteksi wajah palsu berbasis kecerdasan buatan telah meningkat dalam lima tahun terakhir. Hal ini sejalan dengan kebutuhan yang terus meningkat dan jenis serangan baru. Berbagai metode deep learning telah digunakan untuk mengenali wajah palsu. Dalam lima tahun tersebut penelitian didominasi menggunakan model berbasis konvolusi. Oleh karena itu, peneliti mulai mencoba menggunakan model lain yaitu *Vision Transformer* (*ViT*) yang dapat mengenali wajah palsu diluar domain. Namun, ViT memiliki biaya kecepatan inferensi dan memori yang tinggi sehingga diperlukan optimasi pada model tersebut. *EfficientViT* merupakan pengembangan arsitektur *ViT* yang telah dioptimasi dengan cara mengurangi redundansi komputasi dan ketergantungan memori. *EfficientViT-M2* 5.8 kali lebih cepat dengan peningkatan akurasi 1.9% lebih tinggi dibandingkan MobileViT-XXS pada dataset ImageNet. Hasil eksperimen pada kasus pengenalan wajah palsu menunjukkan *EfficientViT-M5* mencapai performa klasifikasi terbaik dengan ACER 11,39% mengungguli ViT-B/16 sebesar 6,13% dan ViT-B/32 sebesar 9,74%. Dari sisi komputasi EfficientViT-M5 mendapatkan waktu inferensi 3,89 ms per citra menjadikannya 3,19 kali lebih cepat dibandingkan ViT-B/16 yang hanya 12,39 ms Temuan ini mengkonfirmasi bahwa EfficientViT merupakan alternatif yang lebih ringan, lebih cepat, sekaligus lebih akurat dibandingkan ViT standar untuk sistem deteksi spoofing wajah berbasis transformer.

Kata kunci: wajah palsu, efficientvit, vision transformer.

Abstract

Research related to fake face detection systems has increased in the last five years. This is in line with the increasing needs and new types of attacks. Various deep learning methods have been used to recognize fake faces. In those five years, research has been dominated by convolution-based models. Therefore, researchers began to try using another model, namely the Vision Transformer (ViT), which can recognize fake faces outside the domain. However, ViT has a high inference speed and memory costs, so optimization of the model is needed. EfficientViT is an optimized development of the ViT architecture by reducing computational redundancy and memory dependence. EfficientViT-M2 is 5.8 times faster with a 1.9% increase in accuracy compared to MobileViT-XXS on the ImageNet dataset. Experimental results in the case of fake face recognition show that EfficientViT-M5 achieves the best classification performance with an ACER of 11.39%, outperforming ViT-B/16 by 6.13% and ViT-B/32 by 9.74%. Computationally, EfficientViT-M5 achieves an inference time of 3.89 ms per image, making it 3.19 times faster than ViT-B/16 that only 12,39 ms. These findings confirm that EfficientViT is a lighter, faster, and more accurate alternative to standard ViT for transformer-based face spoofing detection systems.

Keywords: face spoofing, efficientvit, vision transformer

1. Pendahuluan

Latar Belakang

Biometrik merupakan metode autentikasi yang mulai banyak digunakan akhir akhir ini. Teknologi ini banyak diterapkan dalam sistem keamanan yang membutuhkan verifikasi identitas. Berbagai macam gawai seperti smartphone, tablet, dan laptop sudah mulai menerapkan biometrik seperti sidik jari atau pengenalan wajah untuk autentikasi. Selain itu berbagai jenis modalitas biometrik juga telah dikembangkan, mulai dari wajah, iris, telinga [1], [2], [3]. Pengenalan wajah merupakan salah satu bentuk autentikasi biometrik yang paling banyak dikembangkan dan digunakan. Hal ini berbanding lurus dengan jumlah publikasi karya ilmiah tentang pengenalan wajah yang meningkat setiap tahunnya. Survei oleh *Ali et al* [4] menunjukkan jumlah riset terkait pengenalan wajah pada tahun 2020 meningkat sebanyak sepuluh kali lipat dibandingkan pada tahun 2000. Hal ini disebabkan oleh kebutuhan industri yang tinggi mulai dari sektor keamanan hingga sektor hiburan di berbagai platform. Hal tersebut juga didukung dengan keluarnya metode *deep learning*. Sehingga sistem pengenalan wajah mendapatkan akurasi prediksi yang sangat tinggi dibandingkan metode-metode sebelumnya.

Kesuksesan sistem pengenalan wajah merupakan hal yang sangat luar biasa. Meski demikian, sistem pengenalan wajah pada umumnya rentan terhadap serangan pemalsuan wajah atau istilah lainnya adalah *spoofing*. *Spoofing* merupakan proses pemalsuan wajah seseorang untuk mendapatkan akses ilegal ke sistem biometrik yang dapat dilakukan dengan menampilkan wajah orang lain di depan kamera atau lebih dikenal dengan istilah *presentation attack*. *Presentation attack* sendiri bervariasi mulai dari foto cetak, rekaman, rias wajah, masker 3D, dll[5].

Presentation Attack Detection sendiri telah berkembang secara pesat. Perkembangan ini dimulai dari pendekatan awal dengan menggunakan pembuatan fitur secara manual menggunakan *LBP*, *HOG*, dan *DoG*[4]. Pendekatan ini berfokus pada tekstur permukaan wajah. Setelah itu baru dilanjutkan dengan metode pembelajaran mesin seperti *SVM* dan *Random Forest*. Transformasi besar terjadi pada saat

kemunculan jaringan saraf tiruan dalam khususnya *Convolutional Neural Network*. Hal ini memungkinkan pembelajaran otomatis dari data dan mampu menangkap informasi spasial secara komplek. Pendekatan modern mengintegrasikan modalitas lain seperti *RGB*, *IR*, *depth* dan teknik lanjutan seperti *domain generalization*, *adversarial training*, serta *transfer learning* untuk menghadapi serangan yang belum terlihat sebelumnya menjadikan *PAD* semakin adaptif dan kuat terhadap tantangan keamanan biometrik yang terus berkembang[5].

Dalam beberapa tahun terakhir pendekatan berbasis *deep learning* telah menjadi standar dalam sistem *PAD*. *Vision Transformer* menandai fase baru dalam evolusi teknologi ini. Tidak seperti *CNN* yang bergantung pada fitur lokal untuk ekstraksi fitur spasial *ViT* memanfaatkan mekanisme *self-attention* untuk menangkap hubungan global antar patch gambar. Hal ini terbukti sangat berguna dalam mendeteksi pola-pola *spoofing* yang tidak eksplisit. Beberapa penelitian seperti [6], [7], [8] menunjukkan bahwa *ViT* memiliki keunggulan signifikan dalam hal generalisasi terutama ketika diuji pada protokol *cross-dataset* atau *unknown attack scenarios*. Arsitektur seperti *FM-ViT* [9] berhasil mendapatkan kinerja yang unggul dibandingkan pendekatan *CNN* konvensional khususnya dalam mengintegrasikan input multimodal seperti *RGB*, *IR*, dan *depth*. Oleh karena itu, pemanfaatan *Vision Transformer* dapat memperluas area penelitian yang baru dalam *PAD*.

Disamping keunggulan dari *Vision Transformer* terdapat harga yang perlu dibayar berupa kecepatan komputasi dan kebutuhan memori yang tinggi terutama pada tahap inferensi. Penelitian seperti *EfficientViT* [10] melakukan optimasi pada arsitektur *transformer* dengan memperkenalkan strategi desain yang lebih efisien secara memori dan komputasi. *EfficientViT-M2* mendapatkan peningkatan throughpout 5.8 kali lebih banyak serta akurasi 1.9% lebih tinggi dibandingkan *MobileViT-XXS*[11] pada dataset *ImageNet*. Sehingga hal ini perlu dilakukan investigasi apakah keunggulan efisiensi *EfficientViT* dapat dipertahankan pada kasus deteksi *spoofing* wajah dan sejauh mana pertukaran antara akurasi dan efisiensinya.

Topik dan Batasannya

Penelitian ini mengangkat topik deteksi wajah palsu yaitu upaya membedakan apakah citra wajah yang ditangkap oleh sistem berasal dari wajah asli seseorang atau merupakan hasil rekayasa presentasi. Contohnya foto cetak, tampilan pada layar yang direkam ulang, atau masker yang menyerupai wajah asli. Sebagai contoh sebuah foto wajah yang diambil dari kamera akan diklasifikasikan sebagai asli sedangkan foto cetak yang dihadapkan ke kamera akan diklasifikasikan sebagai palsu.

Untuk mengerjakan persoalan ini, penelitian menggunakan arsitektur turunan *Vision Transformer* bernama *EfficientViT*. Arsitektur tersebut dibandingkan dengan *Vision Transformer* standar (*ViT-B/16* dan *ViT-B/32*). Perbandingan dilakukan dari performa klasifikasi, efisiensi komputasi, dan pola perhatian visual yang terbentuk saat model mengambil keputusan.

Beberapa batasan diterapkan pada penelitian ini. Pertama, penelitian hanya menangani citra dua dimensi dan tidak melibatkan data video atau informasi pergerakan wajah, karena keterbatasan waktu untuk memproses data *temporal* yang jauh lebih besar volumenya serta memerlukan arsitektur tambahan di luar cakupan penelitian ini. Kedua, jenis serangan pada dataset tidak dibedakan berdasarkan kategori seperti foto cetak, rekaman wajah, masker 3D melainkan diperlakukan sebagai satu kelas "palsu" saja. Hal ini disebabkan pengenalan setiap kategori serangan akan memerlukan skenario eksperimen dan evaluasi yang jauh lebih kompleks dan di luar cakupan waktu yang tersedia. Ketiga, penelitian hanya menggunakan satu *dataset* yaitu *CelebA-Spoof* tanpa pengujian lintas *dataset*. Oleh karena itu hasil yang diperoleh mencerminkan performa model pada karakteristik data tersebut saja. Keempat, model yang dibandingkan dibatasi pada *EfficientViT M1, M3, M5* sebagai model utama serta *ViT-B/16* dan *ViT-B/32* sebagai pembanding tanpa melibatkan arsitektur lainnya. Kelima, konfigurasi *hyperparameter* pelatihan ditetapkan pada satu skenario tertentu tanpa pencarian kombinasi *hyperparameter* secara menyeluruh, mengingat pencarian tersebut membutuhkan sumber daya komputasi dan waktu yang jauh lebih besar dari yang tersedia. Terakhir, penelitian ini berhenti pada tahap pengembangan dan evaluasi model dalam lingkungan eksperimental dan tidak mencakup implementasi maupun pengujian pada perangkat produksi. Hal tersebut memerlukan tahapan rekayasa perangkat lunak tambahan yang berada di luar cakupan tujuan akademis penelitian ini.

Tujuan

Penelitian ini bertujuan untuk mengembangkan model deteksi wajah palsu menggunakan *EfficientViT* kemudian mengukur sejauh mana performa klasifikasi dan komputasi jika dibandingkan dengan model *Vision Transformer* (*ViT-B/16* dan *ViT-B/32*) sehingga dapat diketahui besarnya pertukaran antara beban komputasi dan akurasi ketika digunakan untuk kasus deteksi *spoofing* wajah. Selain itu, penelitian ini bertujuan menghasilkan visualisasi peta aktivasi dari kedua jenis model untuk mengetahui bagian wajah mana yang menjadi perhatian model dalam menentukan hasil prediksinya sehingga mendapatkan pemahaman yang lebih interpretatif mengenai cara kerja *EfficientViT* dan *Vision Transformer* dalam mengenali indikasi wajah palsu.

2. Studi Terkait

Bab ini membahas penelitian-penelitian terdahulu yang berkaitan dengan deteksi spoofing wajah serta menempatkan berbagai arsitektur dan metode yang telah digunakan secara efektif dalam bidang tersebut. Dengan meninjau dan merangkum perkembangan serta hasil-hasil penting dari literatur yang ada, bab ini menyajikan gambaran menyeluruh mengenai pendekatan dan teknik terkini yang menjadi rujukan dalam penelitian deteksi spoofing wajah.

2.1 Evolusi Spoofing Wajah

Perkembangan *PAD* menunjukkan pergeseran yang signifikan dari arsitektur berbasis *Convolutional Neural Network* menuju *Vision Transformer* dalam lima tahun terakhir. Survei oleh *Yu et al* [5] menunjukan pendekatan berbasis *CNN* mendominasi penelitian karena kemampuannya dalam mengenali wajah asli dan palsu dengan sangat baik dibandingkan *handcrafted features*. Sebagai contoh, pada penelitian [12] didapatkan akurasi yang sangat tinggi. Pendekatan ini efektif dalam menangkap pola tekstur lokal seperti pantulan layar secara lokal namun cenderung terbatas dalam memahami informasi global serta rentan terhadap *overfitting* ketika menghadapi variasi *domain* dan kondisi pencahayaan yang komplek. Upaya lanjutan seperti *background subtraction*, *generative reconstruction* [13], [14] membantu meningkatkan *interpretabilitas* model tetapi *CNN* tetap memiliki keterbatasan dalam generalisasi pada serangan baru seperti masker 3D dan serangan parsial.

Vision Transformer mencoba mengatasi ini dengan memanfaatkan mekanisme *self-attention* untuk menangkap ketergantungan spasial dan hubungan antar *patch* dalam citra wajah secara menyeluruh. Model seperti *FM-ViT*, *Adaptive Transformer*, *CNN-ViT* dan *ViTransPAD* [8], [9], [15], [16]

memperlihatkan peningkatan kinerja dalam mengenali serangan yang lebih kompleks melalui pemahaman fitur global yang lebih dalam. Namun, ViT memiliki keterbatasan dalam efisiensi karena jumlah parameter yang besar dan kebutuhan komputasi yang tinggi. Hal ini menjadi tantangan utama ketika diterapkan pada sistem autentikasi berbasis perangkat bergerak yang menuntut model dengan efisiensi memori dan kecepatan inferensi yang lebih baik.

Menanggapi permasalahan tersebut penelitian terkini mengarah pada pengembangan *Vision Transformer* ringan yang dirancang untuk menyeimbangkan kinerja dan efisiensi sumber daya. Beberapa model berupaya mengintegrasikan keunggulan representasi global dari *ViT* dan efisiensi lokal *CNN*. Sebagai contoh pendekatan seperti *MobileViT* [11], *Lite Vision Transformer* [17] menunjukkan peningkatan signifikan dalam hal efisiensi komputasi dan kecepatan inferensi dibandingkan *ViT*.

2.2 Dataset spoofing wajah

Dataset memegang peranan penting dalam penelitian berbasis deep learning baik dalam fase pelatihan maupun evaluasi. Hal ini menentukan sejauh mana model mampu mempelajari karakteristik wajah asli dan wajah palsu serta mengukur kemampuan generalisasi sistem terhadap berbagai skenario serangan. Survei oleh *Yu et al* [5] menjelaskan perkembangan dataset deteksi spoofing wajah dalam beberapa tahun terakhir dapat dirangkum ke dalam tiga keunggulan utama yaitu peningkatan skala data, keberagaman jenis serangan dan kondisi perekaman serta pemanfaatan berbagai *modalitas* dan sensor. Dari survei tersebut *dataset spoofing* wajah generasi awal umumnya masih memiliki keterbatasan dari sisi jumlah data, subjek, variasi serangan yang sempit, serta kondisi perekaman yang relatif terkontrol sehingga kurang mendukung pelatihan dan evaluasi model *deep learning* yang bersifat general. Ringkasan karakteristik dataset *PAD* yang umum digunakan dalam literatur ditunjukkan pada Tabel 2.1.

TABEL 1
Menyajikan perbandingan beberapa dataset PAD yang banyak digunakan, mencakup dataset generasi awal hingga dataset yang lebih baru sebagaimana dibahas dalam survei [5].

Dataset	Tahun	Data	Jenis Serangan	Subjek	Modalitas
Replay-Attack	2012	1.200 (Video)	Print, Replay	50	Visual
CASIA-MFSD	2012	600 (Video)	Print, Replay	50	Visual
3DMAD	2014	255 (Video)	3D mask	14	Visual
Msspoof	2015	4.704 (Image)	Print	21	Visual
MSU-USSA	2016	10.260 (Image)	Print, Replay	1,140	Visual
OULU-NPU	2017	5.940 (Video)	Print, Replay	55	Visual
CASIA-SURF	2018	21.000 (Video)	Paper Cut	1,000	Visual
SiW-M	2019	1.628 (Video)	Print, replay, mask, makeup,	493	Visual
HQ-WMCA	2020	2904 (Video)	partial, resin, paper cut	51	Visual, Depth, NIR, SWIR, Thermal
CelebA-Spoof	2020	625.537 (Image)	Print, replay, partial, mask, makeup	10,177	Visual
PADISI-Face	2021	2029 (Video)	Print, Replay, 3D, Paper Cut	360	Visual, Depth, NIR, SWIR, Thermal

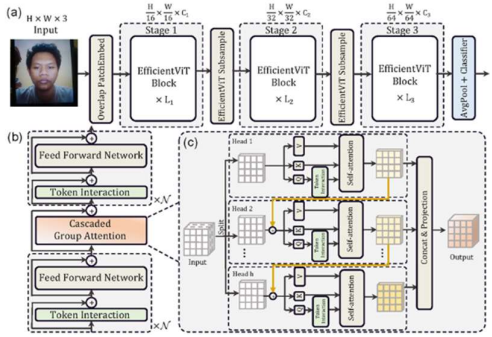
Berdasarkan perbandingan pada Tabel 2.1 *CelebA-Spoof* dipilih sebagai *dataset* utama dalam tugas akhir ini dikarenakan memiliki jumlah subjek yang banyak. *Dataset* ini menyediakan skala data yang luas dengan lebih dari sepuluh ribu subjek sehingga mendukung pembelajaran representasi yang lebih stabil dan mengurangi potensi bias identitas. Terutama apabila yang diterima model hanya berupa wajah tanpa latar belakang.

2.3 Metode EfficientViT

Berbagai arsitektur *Vision Transformer* ringan seperti *MobileViT* [11] dan *Lite Vision Transformer* [17] telah berhasil menurunkan kompleksitas model melalui pengurangan jumlah parameter dan *floating-point operations* (*FLOPs*). Hal tersebut berhasil menurunkan beban komputasi secara signifikan. Namun menurut *Liu et al* melalui "*EfficientViT: Memory Efficient Vision Transformer with Cascaded Group Attention*" [10] mengidentifikasi bahwa kendala utama pada *Vision Transformer* bukan hanya terletak pada ukuran model tetapi pada inefisiensi memori akibat operasi *multi-head self-attention* (*MHSA*) terutama pada proses *tensor reshaping* dan fungsi *element-wise* yang bersifat *memory-bound*.

Untuk mengatasi permasalahan tersebut *EfficientViT* memperkenalkan tiga inovasi utama yang berfokus pada peningkatan efisiensi memori, komputasi, dan distribusi parameter. Pertama dengan *Sandwich Layout* yaitu desain blok yang menempatkan satu lapisan *MHSA* diapit oleh beberapa *feed-forward network* (*FFN*). Strategi ini secara signifikan mengurangi waktu akses memori pada *MHSA* sekaligus mempertahankan komunikasi antar kanal melalui *FFN*. Kedua

yaitu *Cascaded Group Attention* (*CGA*) yang menggantikan *MHSA* dengan mekanisme *grouped attention* di mana setiap *attention head* menerima subset fitur yang berbeda dan hasil antar *head* dihubungkan secara bersusun. Pendekatan ini tidak hanya mengurangi redundansi komputasi antar *head* tetapi juga memperkaya kapasitas representasi tanpa menambah parameter secara signifikan. Ketiga yaitu *Parameter Reallocation* yaitu redistribusi kapasitas parameter dengan memperbesar *value projection* yang paling berpengaruh terhadap kualitas representasi sembari mengurangi dimensi *query* dan *key projection* yang lebih redundan. Sebagai gambaran arsitektur *EfficientViT* sendiri dapat dilihat pada Gambar 2.1.



GAMBAR 1
Ringkasan EfficientViT. (a). Arsitektur EfficientViT; (b) Sandwich Layout block;
(c) Cascaded Group Attention.

Eksperimen yang dilakukan oleh *Liu et al[10]* pada dataset *ImageNet-1K* menunjukkan bahwa *EfficientViT-M2* mencapai akurasi *Top-1* sebesar 70,8%, lebih tinggi 1,8% dibanding *MobileViT-XXS* dengan peningkatan kecepatan inferensi sebesar 5,8× pada *GPU* dan 3,7× pada *CPU*. Setelah dikonversi ke format *ONNX* kecepatan meningkat hingga 7,4× lebih cepat. Pada pengujian di perangkat bergerak dengan *chipset Apple A13 Bionic EfficientViT-M2* berjalan 2,3× lebih cepat daripada *MobileViT-XXS*. Kombinasi efisiensi memori, kecepatan inferensi, dan akurasi yang kompetitif menjadikan *EfficientViT* salah satu pendekatan potensial untuk mendukung sistem deteksi spoofing wajah berbasis *transformer* yang membutuhkan kinerja tinggi dengan beban komputasi rendah.

2.4 Visualisasi Peta Aktivasi

Seiring meningkatnya kompleksitas arsitektur *deep learning* kebutuhan terhadap *interpretabilitas* menjadi penting dalam sistem deteksi *spoofing* wajah. Sistem juga harus mampu menjelaskan dasar keputusan model secara transparan khususnya pada aplikasi keamanan biometrik.

Dalam penelitian ini digunakan *Kernel Principal Component Analysis Class Activation Mapping* (*KPCA-CAM*) [18] sebagai metode *interpretabilitas* untuk model *ViT* dan *EfficientViT*. *KPCA-CAM* bekerja mirip *EigenCAM*[19] namun menggantikan *PCA linear* dengan *Kernel PCA* untuk menangkap hubungan tidak linear pada peta fitur. Dengan memanfaatkan fungsi *kernel* seperti *Radial Basis Function* atau *Sigmoid KPCA-CAM* dapat memproyeksikan aktivasi ke ruang berdimensi tinggi sehingga dapat mengekstraksi struktur kompleks yang lebih representatif dari data.

Metode ini menghasilkan peta aktivasi yang lebih halus dan akurat tanpa memerlukan gradien atau modifikasi arsitektur. Dengan memproyeksikan hasil dekomposisi *Kernel PCA* kembali ke domain spasial. *KPCA-CAM* akan menyoroti area wajah yang paling relevan terhadap keputusan klasifikasi berdasar *layer* yang dipilih. Penggunaan metode ini memungkinkan penjelasan yang lebih mendalam terhadap area fokus perhatian model sehingga meningkatkan keyakinan terhadap hasil deteksi *spoofing* wajah.

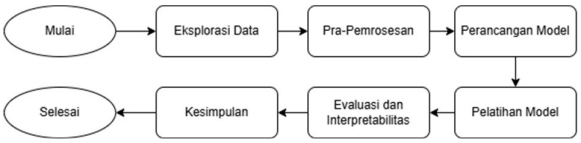
2.5 Ringkasan Studi Terkait

Penelitian deteksi *spoofing* wajah menunjukkan pergeseran dari arsitektur berbasis *CNN* menuju *ViT*. Model berbasis *CNN* efektif dalam menangkap pola tekstur lokal namun terbatas dalam memahami informasi global serta rentan terhadap *overfitting* ketika menghadapi variasi *domain* dan kondisi pencahayaan yang kompleks. *ViT* dapat memperluas representasi global melalui mekanisme *self-attention* meski menggunakan biaya yang mahal. Model *ViT* ringan seperti *MobileViT* [11], *Lite Vision Transformer* [17] berupaya menyeimbangkan akurasi dan efisiensi dengan menggabungkan kekuatan global *ViT* dan efisiensi lokal *CNN*. Selain itu kebutuhan interperatif model menjadi aspek penting untuk menjamin transparansi dan keandalan sistem. Pendekatan seperti *KPCA-CAM* [18] dapat memberikan visualisasi keputusan model secara lebih stabil tanpa menambah beban komputasi maupun perubahan arsitektur.

3. Sistem yang Dibangun

3.1 Desain Perancangan Sistem

Sistem *EfficientViT* dibangun menggunakan bahasa pemrograman *Python* dan *framework PyTorch* sebagai platform utama untuk pelatihan dan evaluasi model. Alur kerja sistem dimulai dari eksplorasi data, pra-pemrosesan, perancangan dan pelatihan model, evaluasi performa, hingga visualisasi dan analisa. Proses alur penelitian dapat dilihat pada Gambar 3.1, yang menggambarkan tahapan utama mulai dari pengumpulan data hingga evaluasi performa dan analisis model.



GAMBAR 2
Menampilkan *flowchart* alur kerja penelitian.

3.2 Dataset

CelebA-Spoof merupakan salah satu dataset deteksi *spoofing* wajah terbesar serta memiliki banyak subjek. Keseluruhan data terdiri dari 625.537 citra dan setelah dilakukan pembersihan didapatkan 561.573 gambar dengan 43 jenis anotasi yang mencakup berbagai bentuk serangan *2D* dan *3D* seperti serangan cetak, *replay*, *3D mask*, potongan kertas pada beragam kondisi pencahayaan serta lingkungan perekaman. Dataset ini dirancang untuk mengevaluasi ketahanan model terhadap variasi *domain* dan kondisi nyata autentikasi wajah. Anotasi pada dataset *CelebA-Spoof* mencakup berbagai kategori serangan dan kondisi pencahayaan yang digunakan dalam proses pelatihan dan evaluasi model.

3.3 Pra-pemrosesan

3.3.1 Keseluruhan Data

Setiap citra terlebih dahulu melalui tahap ekstraksi wajah menggunakan model *RetinaFace* yang berfungsi memberikan *bounding box* dan memotong area wajah. Tahap ini bertujuan agar data seragam sehingga evaluasi kemampuan kedua jenis model dalam melakukan klasifikasi dapat telihat perbedaanya.

Data dibagi menjadi tiga subset yaitu data latih, validasi, dan uji dengan rasio 8:1:1,2. Pembagian ini dilakukan dengan memastikan tidak ada subjek yang tumpang tindih di lebih dari satu subset. Distribusi kelas pada masing-masing subset

ditunjukkan pada Tabel 3.1.

TABEL 2
Distribusi kelas dataset CelebA-Spoof pada setiap subset.

Subset	Asli/Bonafide	Palsu/Spoof	Total	Rasio B:S
Latih	146.326	298.637	444.963	1:2,04
Validasi	16.136	33.304	49.440	1: 2,06
Uji	19.923	47.247	67.170	1:2,37
Total	182.385	379.188	561.573	1:2,07

Sebaran tersebut menunjukan bahwa sebaran kelas *CelebA-Spoof* tidak seimbang dengan jumlah sampel palsu yang lebih besar dibandingkan sampel asli. Kondisi ini dapat dimaklumi mengingat serangan *spoofing* lebih bervariasi dibandingkan sampel wajah asli. Namun tetap perlu diperhatikan karena dapat mempengaruhi hasil pelatihan yang terlalu condong ke kelas mayoritas.

Untuk mengatasi permasalahan tersebut digunakan teknik *class weighting* pada fungsi *loss* selama proses pelatihan. Pendekatan ini dipilih karena tidak memerlukan duplikasi maupun pengurangan data secara fisik seperti *oversampling* atau *undersampling*. Hal ini tidak akan menambah durasi latih secara signifikan sehingga distribusi data asli tetap terjaga sementara model tetap dapat untuk memberi perhatian yang proporsional terhadap kelas minoritas (wajah asli). Kelas *spoof* didefinisikan sebagai kelas positif (label 1) dan kelas bonafit sebagai kelas negatif (label 0). Bobot kelas dihitung berdasarkan rasio jumlah sampel kelas negatif terhadap kelas positif pada data latih, yaitu:

$$\omega_{pos} = \frac{N_{bonafide}}{N_{spoof}} = \frac{146.326}{298.637} = 0{,}49 \qquad \textbf{(3.1)}$$

Nilai pada (3.1) kemudian digunakan sebagai parameter *pos_weight* pada fungsi *loss BCEWithLogitsLoss*. Karena kelas wajah palsu merupakan kelas mayoritas, nilai *pos_weight* yang lebih kecil dari satu ini berfungsi menurunkan bobot kontribusi kesalahan pada kelas spoof. Sehingga model diarahkan untuk memberi perhatian lebih besar terhadap kesalahan klasifikasi pada kelas wajah asli.

3.3.2 Data Latih

Pada data latih terdapat pra-pemrosesan tambahan yang berfungsi agar model kecerdasan buatan dapat mempelajari fitur-fitur wajah secara maksimal dan tidak terpengaruh oleh posisi, rotasi, dan kualitas kamera. Setiap metode pra-pemrosesan, probabilitas, dan fungsinya dapat dilihat pada Tabel 3.2.

TABEL 3
Nama, probabilitas, fungsi setiap jenis pra-pemrosesan pada data latih.

Nama	Probabilitas (%)	Parameter Tambahan	Deskripsi
RandomResizedCrop	20	Skala 90%-100%	Memotong sebagian area gambar secara acak lalu mengubah ukurannya kembali, agar model tidak bergantung pada posisi objek yang selalu di tengah.
RandomHorizontalFlip	50	-	Membalik gambar secara horizontal agar model mengenali objek dari kedua arah (kiri-kanan).
RandomRotation	20	Derajat perputaran 1-5	Memutar gambar dalam sudut kecil agar model tahan terhadap kemiringan gambar.
RandomPerspective	10	Skala distorsi 8%	Mendistorsi sudut pandang gambar agar model tahan terhadap variasi sudut pengambilan foto.
ColorJitter	20	Kecerahan 15% Kontras 15% Saturasi 15% Hue 5%	Mengubah kecerahan, kontras, saturasi, dan hue agar model tidak sensitif terhadap variasi pencahayaan.
GaussianBlur	15	Ukuran kernel 3x3 Tingkat keburaman 10-100%	Mengaburkan gambar agar model tetap mengenali objek meski gambar kurang fokus.
RandomErasing	20	Skala 2-8%	Menghapus sebagian

			kecil area gambar agar model tetap mengenali objek meski sebagian tertutup/ter halang.

Setelah pra-pemrosesan tersebut dilakukan didapatkan sampel hasil seperti yang terlihat pada Gambar 3.2



GAMBAR 3
Sampel hasil pra-pemrosesan pada data latih tiap metode.

3.4 Perancangan Model

Sistem dirancang untuk menerima citra wajah sebagai masukan dan menghasilkan keluaran berupa klasifikasi dua kelas yaitu asli dan palsu. Model dibangun menggunakan dua jenis arsitektur utama yaitu *Vision Transformer* varian *ViT-B/16* dan *ViT-B/32* sebagai model acuan dan *EfficientViT* varian *M1*, *M3*, dan *M5* sebagai model utama yang dipilih. Model tersebut dipilih dengan mempertimbangkan kemiripan akurasi dan kecepatan paling ringan dari hasil yang didapatkan pada *ImageNet-1K*.

Kedua jenis model dilatih dari awal tanpa menggunakan bobot *pretrained* dari dataset lain. Hal ini untuk memastikan bahwa representasi fitur yang dipelajari oleh model sepenuhnya berasal dari dataset sehingga performa yang dihasilkan murni mencerminkan kemampuan masing-masing arsitektur dalam mempelajari pola *spoofing* wajah tanpa pengaruh dari pengetahuan domain lain. Pendekatan ini juga memungkinkan perbandingan yang lebih adil antar model karena seluruh model berangkat dari kondisi inisialisasi bobot yang setara.

Alur sistem dimulai dari input citra wajah hasil ekstraksi dari dataset *CelebA-Spoof*. Seluruh citra terlebih dahulu dilakukan pra-pemrosesan yang dijelaskan pada 3.3. Setelah itu, citra diubah menjadi potongan-potongan kecil yang disebut *patch embedding* untuk kemudian direpresentasikan dalam bentuk vektor fitur. Ukuran *patch* tiap model berbeda, pada model *ViT-B/32* citra dibagi menjadi *patch* berukuran 32×32 piksel sedangkan pada *EfficientViT* dan *ViT-B/16* menggunakan ukuran *patch* yang lebih kecil yaitu 16×16 piksel. Perbedaan ukuran *patch* ini memengaruhi tingkat detail informasi spasial yang dapat dipelajari oleh model. *Patch* yang lebih kecil memungkinkan penangkapan detail visual yang lebih halus.

Representasi fitur hasil *patch embedding* kemudian diproses oleh kedua model untuk mengenali hubungan antarbagian wajah dan mengekstraksi pola yang relevan terhadap deteksi serangan. Hasil akhir berupa nilai probabilitas untuk dua kelas. Nilai yang mendekati 0 akan dianggap sebagai asli sementara nilai yang mendekati 1 akan dianggap sebagai palsu.

3.5 Pelatihan Model

Mengingat jumlah data yang banyak, proses pelatihan hanya akan dilakukan selama 10 *epoch* dengan ukuran *batch* sebesar 32. Optimizer *AdamW* dipilih karena kemampuan konvergensi dan generalisasi yang lebih baik daripada *Adam* seperti yang disebutkan dalam penelitian [20]. Selain itu penjadwalan *CosineAnnealingLR* digunakan untuk penyesuaian laju pembelajaran secara bertahap. Yang sedikit berbeda adalah dikarenakan jumlah data latih yang banyak dan *epoch* yang sedikit maka perlu perubahan pada cara penjadwalan yang biasanya dilakukan setiap *epoch* menjadi setiap *batch*. Fungsi *loss* yang diterapkan adalah *BCEWithLogitsLoss* dengan penyesuaian bobot berdasar rasio distribusi kelas sesuai yang ada pada (3.1). Nilai *loss* dan akurasi divalidasi pada setiap *epoch* untuk memastikan kestabilan proses pembelajaran dan mencegah *overfitting*. Model terbaik disimpan secara otomatis melalui mekanisme cegatan untuk evaluasi lebih lanjut. Setelah pelatihan selesai model diuji menggunakan data uji yang terpisah dari data latih. Hasil inferensi berupa probabilitas kemudian dikonversi menjadi label biner dan evaluasi dengan berbagai metrik.

3.6 Metrik Evaluasi

Evaluasi performa model dilakukan menggunakan empat metrik utama yang umum digunakan dalam penelitian deteksi *spoofing* wajah yaitu *Attack Presentation Classification Error Rate* (*APCER*), *Bonafide Presentation Classification Error Rate* (*BPCER*), *Average Classification Error Rate* (*ACER*), dan akurasi. Metrik ini memiliki formula sebagai berikut:

$$APCER = \frac{FP}{FP + TN} \quad \textbf{(3.2)}$$

$$BPCER = \frac{FN}{FN + TP} \quad \textbf{(3.3)}$$

$$ACER = \frac{APCER + BPCER}{2} \quad \textbf{(3.4)}$$

$$Akurasi = \frac{TP + TN}{TP + FP + TN + FN} \quad \textbf{(3.5)}$$

TP: Sampel *Spoof* yang benar diprediksi sebagai *Spoof*.

TN: Sampel *Bonafide* yang benar diprediksi sebagai *Bonafide*.

FP: Sampel *Spoof* yang salah diprediksi sebagai *Bonafide*.

FN: Sampel *Bonafide* yang salah diprediksi sebagai *Spoof*.

Akurasi **(3.5)** menunjukkan persentase keseluruhan prediksi yang benar terhadap total data uji. Nilai *APCER* (3.2) menunjukkan seberapa sering model gagal mendeteksi serangan sedangkan *BPCER* (3.3) mengukur seberapa sering wajah asli salah diklasifikasikan sebagai serangan. *ACER* (3.4) digunakan sebagai indikator gabungan yang menunjukkan rata-rata tingkat kesalahan antara deteksi serangan yang gagal dan

kesalahan identifikasi wajah asli.

Evaluasi beban komputasi dilakukan menggunakan metrik *Multiply-Accumulate Operations* (*MACs*), Jumlah parameter yang dilatih, dan kecepatan inferensi tiap gambar untuk mengetahui kompleksitas model.

3.7 Interpretabilitas

Model yang telah dilatih kemudian digunakan untuk menghasilkan peta perhatian guna mengetahui bagian wajah yang menjadi fokus utama dalam proses klasifikasi. Fitur yang digunakan diekstraksi dari blok terakhir masing-masing model. Pada *ViT* menggunakan layer terakhir pada encoder sementara *EfficientViT* menggunakan blok *EfficientViT* terakhir. Dikarenakan fitur *ViT* berbentuk vektor satu dimensi maka perlu dilakukan proses *reshape* agar kembali ke bentuk spasial dua dimensi sesuai posisi *patch* pada citra asli. Sebaliknya pada *EfficientViT* proses ini tidak diperlukan karena arsitekturnya menggunakan mekanisme *local window attention* yang secara alami mempertahankan struktur spasial fitur sepanjang proses ekstraksi.

Setelah representasi fitur diperoleh dilakukan dekomposisi menggunakan *KPCA-CAM* untuk mengekstraksi komponen utama *non-linear* yang paling berpengaruh terhadap keputusan model. Komponen tersebut kemudian diproyeksikan kembali ke citra masukan untuk menghasilkan *heatmap* yang memperlihatkan area wajah dengan kontribusi terbesar terhadap prediksi kelas asli maupun palsu. Hasil visualisasi ini memberikan pemahaman tambahan terhadap perilaku model khususnya dalam menilai konsistensi fokus perhatian terhadap area wajah yang relevan dalam proses deteksi *spoofing* wajah.

4. Evaluasi

4.1 Skenario Percobaan

Pengujian dilakukan dalam tiga skenario untuk menjawab tujuan penelitian secara sistematis. Seluruh skenario menggunakan data uji dari dataset *CelebA-Spoof* yang telah dipisahkan dari data latih dan validasi dengan rasio 8:1:1,2. Model yang diuji terdiri dari dua varian *Vision Transformer* standar yaitu *ViT-B/16* dan *ViT-B/32* serta tiga varian *EfficientViT* yaitu *M1*, *M3*, dan *M5*.

4.1.1 Evaluasi Performa Klasifikasi

Skenario pertama bertujuan mengukur dan membandingkan performa klasifikasi *spoofing* wajah antara model *EfficientViT* dan *ViT* yang telah dilatih. Evaluasi dilakukan menggunakan metrik pada 3.6 yaitu akurasi, *APCER*, *BPCER*, dan *ACER*. Semakin rendah nilai *APCER, BPCER, ACER* semakin baik performa model. Sebaliknya semakin tinggi nilai akurasi semakin baik performa model.

4.1.2 Evaluasi Efisiensi Komputasi

Skenario kedua bertujuan mengukur dan membandingkan beban komputasi seluruh model yang diuji. Efisiensi diukur menggunakan dua metrik: *M*ultiply-Accumulate Operations (*MACs*) sebagai ukuran kompleksitas secara teori dan waktu inferensi rata-rata per citra sebagai ukuran kecepatan pemrosesan secara asli. Jumlah parameter yang dapat dilatih juga dicatat sebagai indikator kebutuhan memori model. Seluruh pengukuran dilakukan pada perangkat dan kondisi yang identik untuk memastikan perbandingan yang adil antar model.

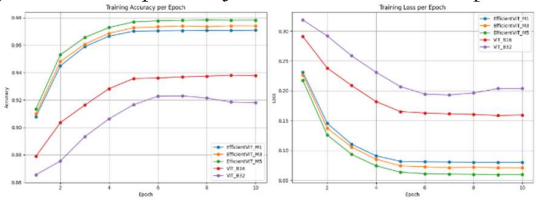
4.1.3 Visualisasi dan Analisa Peta Aktivasi

Skenario ketiga bertujuan membuat visualisasi dan menganalisis pola perhatian visual kedua model dalam proses klasifikasi. Visualisasi dilakukan pada model terbaik dari masing-masing kelompok dan menggunakan sampel representatif dari empat kondisi prediksi: *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN). Sehingga area yang menjadi fokus perhatian model untuk mengambil keputusan dapat diketahui pada setiap jenis sampel.

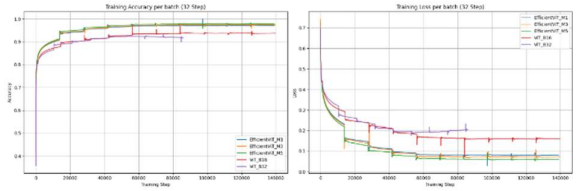
4.2 Hasil Percobaan

4.2.1 Hasil Klasifikasi

Proses pelatihan masing-masing model dapat dilihat melalui kurva akurasi dan *loss* per *epoch* yang terdapat pada Gambar 4.1 dan per *batch* pada Gambar 4.2. Kurva per *epoch* menunjukkan akurasi pelatihan yang konsisten di atas 90% sejak awal. Hal ini tidak cukup untuk membuktikan pembelajaran yang stabil karena *dataset* memiliki ketidakseimbangan kelas yang signifikan karena rasio asli:palsu sebesar 1:2,07. Sehingga dibuatkan kurva per *batch* yang memberikan gambaran yang lebih informatif dengan menunjukkan bahwa seluruh model mengalami penurunan *loss* yang stabil dan peningkatan akurasi yang bertahap serta mengindikasikan bahwa proses pembelajaran berlangsung dengan baik meskipun hanya dilakukan selama 10 *epoch*.

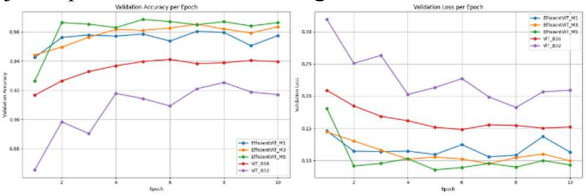


GAMBAR 4
Kurva akurasi dan loss setiap epoch pada tahap pelatihan model.



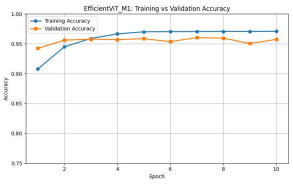
GAMBAR 5
Kurva akurasi dan loss setiap batch pada tahap pelatihan model.

Selain itu, hasil evaluasi pada data validasi tiap epoch ditunjukan pada kurva validasi sebagai berikut:

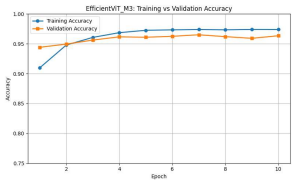


GAMBAR 6
Kurva akurasi dan loss setiap epoch pada tahap validasi model.

Pada Gambar 4.3 menunjukan bahwa model memperoleh perkembangan hasil yang yang mirip antara data latih dan data validasi. Selain itu dari kurva tersebut dapat diketahui tidak ada indikasi *overfitting* antara hasil pada pelatihan dan hasil pada validasi hal ini dapat diketahui dengan selisih akurasi pada saat pelatihan yang kecil. Visuaisasi selisih antara hasil pelatihan dan hasil validasi dapat dilihat pada Gambar 4.4.



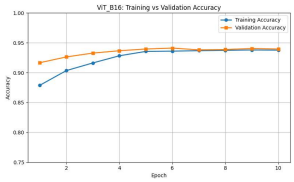
(A)



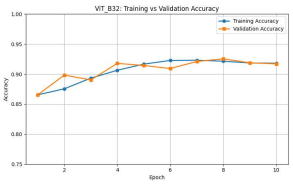
(B)



(C)



(D)



(E)

GAMBAR 7
Visualisasi perbandingan antara akurasi hasil pelatihan dan validasi tiap epoch pada model EfficientViT-M1 (A), EfficientViT-M13(B), EfficientViT-M5 (C), ViT-B/16 (D), dan ViT-B/32 (E).

Selain itu hasil evaluasi pada data uji ditunjukkan pada Tabel 4.1. Nilai *APCER*, *BPCER*, dan *ACER* yang lebih rendah menunjukkan performa yang lebih baik, sedangkan *ACC* yang lebih tinggi menunjukkan performa yang lebih baik.

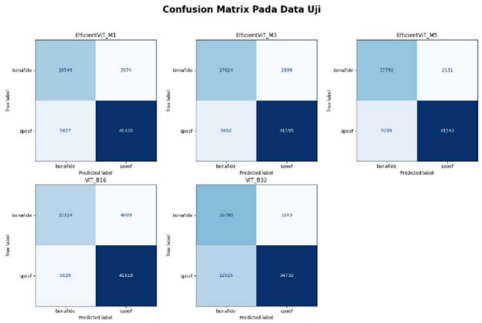
TABEL 4
Menyajikan perbandingan nilai performa berbagai metrik pengukuran pada data uji.

Model	ACC (%)	APCER (%)	BPCER (%)	ACER (%)
EfficientViT-M1	86,35	16,94	12,33	14,63
EfficientViT-M3	87,27	14,55	11,96	13,26
EfficientViT-M5	**88,34**	**10,70**	12,07	**11,39**
ViT-B/16	84,76	23,13	**11,91**	17,52
ViT-B/32	76,69	15,78	26,49	21,13

Dari hasil tersebut *EfficientViT-M5* memperoleh *ACER* terendah sebesar 11,39% dan akurasi tertinggi sebesar 88,34%.

Dalam kelompok *EfficientViT* terdapat tren peningkatan performa seiring bertambahnya kapasitas model: *ACER* turun dari 14,63% (*M1*) ke 13,26% (*M3*) ke 11,39% (*M5*).

Distribusi prediksi tiap model ditampilkan melalui *confusion matrix* pada Gambar 4.5. Terlihat bahwa kedua jenis model dapat melakukan sebagian besar klasifikasi pada sampel asli dan palsu secara benar. Semua model *EfficientViT* dan *ViT-B/16* mendapatkan hasil yang serupa sedangkan *ViT-B/32* mengalami kesulitan dalam mengetahui wajah palsu. Hal ini dapat diartikan bahwa ukuran *patch* yang lebih besar menyebabkan pengurangan informasi yang dapat diterima model dikarenakan fitur-fitur yang berdekatan digabung menjadi informasi yang sama.



GAMBAR 8
Confusion matrix setiap model pada data uji.

4.2.2 Hasil Komputasi

Pengukuran efisiensi komputasi dilakukan pada perangkat pengujian yang sama untuk seluruh model guna memastikan perbandingan yang adil. Spesifikasi perangkat pengujian yang digunakan ditampilkan pada Tabel 4.2.

TABEL 5
Spesifikasi perangkat pengujian.

Komponen	Spesifikasi
CPU	Intel Core i7 13620H 14 cores
GPU	RTX4060 Laptop 8GB DDR6
RAM	16GB DDR4 3200mhz Dual channel
CUDA	12.4
Framework	Pytorch 2.4.1
Bahasa Pemrograman	Python 3.9.13

Hasil pengukuran efisiensi komputasi seluruh model ditunjukkan pada Tabel 4.3. Nilai *MACs* yang lebih rendah menunjukkan model yang lebih ringan secara teoritis, sementara waktu inferensi yang lebih rendah menunjukkan kecepatan pemrosesan yang lebih tinggi secara praktis.

TABEL 6
Perbandingan biaya komputasi yang diperlukan setiap model.

Model	MACs (G)	Jumlah Parameter (M)	Waktu Inferensi per Citra (ms)
EfficientViT-M1	**0.17**	**2.79**	**3.67**
EfficientViT-M3	0.26	6.58	3.88

EfficientViT-M5	0.52	10.69	3.89
ViT-B/16	11.29	57.30	12.39
ViT-B/32	2.95	59.07	5.34

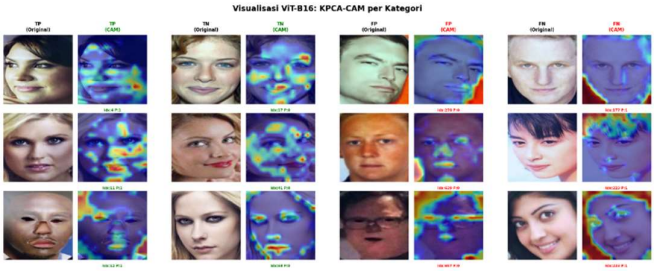
Model *EfficientViT* secara keseluruhan menunjukkan kebutuhan *MACs* yang lebih rendah dibandingkan model *ViT* standar. *EfficientViT-M1* merupakan model paling ringan dengan 0,17 *GMACs* diikuti oleh *EfficientViT-M3* dengan 0,26 *GMACs* dan *EfficientViT-M5* dengan 0,52 *GMACs*. Sementara itu *ViT-B/32* membutuhkan 2,95 *GMACs* dan *ViT-B/16* membutuhkan 11,29 *GMACs*. Dari sisi waktu inferensi model *EfficientViT-M1* memproses satu citra paling cepat dengan rata-rata 3,67 milidetik.

4.2.3 Hasil Visualisasi Peta Aktivasi

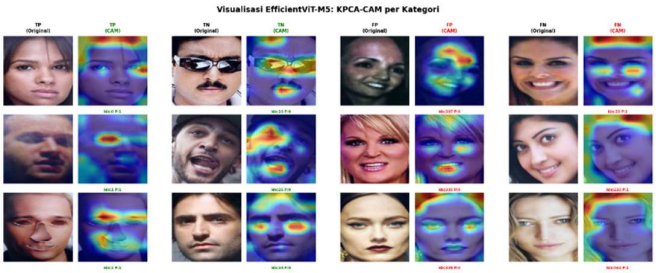
Visualisasi peta aktivasi menggunakan *KPCA-CAM* dilakukan pada sampel dari data uji. Visualisasi dilakukan pada model dengan performa terbaik dari masing-masing kelompok yaitu *ViT-B/16* dan *EfficientViT-M5* dengan sampel yang mencakup empat kondisi prediksi yaitu:

- Wajah palsu yang diprediksi palsu (TP)
- Wajah palsu yang diprediksi asli (FP)
- Wajah asli yang diprediksi asli (TN)
- Wajah asli yang diprediksi palsu (FN)

Hasil visualisasi peta suhu *KPCA-CAM* ditampilkan pada Gambar 4.6 dan Gambar 4.7.



GAMBAR 9
Peta aktivasi KPCA-CAM model ViT-B/16 pada sampel TP, TN, FP, FN.



GAMBAR 10
Peta aktivasi KPCA-CAM model EfficientViT-M5 pada sampel TP, TN, FP, FN.

Gambar tersebut memperlihatkan kedua model cenderung memfokuskan perhatiannya pada area yang lebih gelap pada sampel bonafide maupun spoof. Pada model *Vision Transformer* titik perhatian lebih tersebar dibandingkan dengan *EfficientViT* yang lebih terpusat pada beberapa titik. Meskipun begitu dapat dilihat pada sampel *TP* dan *TN* kedua model dominan memberikan perhatian ke fitur wajah seperti mata, mulut, dan area wajah. Sementara pada sampel *FP* dan *FN* kedua model cenderung menaruh perhatian pada area tepi wajah yang kurang relevan seperti rambut dan latar belakang.

4.3 Analisis Hasil Pengujian

4.3.1 Analisis Performa Klasifikasi

Hasil pada Tabel 4.1 menunjukan *EfficientViT-M5* mencapai *ACER* lebih rendah sebesar 11,39% dengan akurasi 88,34% dibandingkan *ViT-B/16* yang mendapatkan ACER 17.52%. Temuan ini menunjukkan bahwa *EfficientViT-M5* lebih cepat dalam mengekstraksi fitur yang relevan untuk deteksi *spoofing* wajah.

Pada *EfficientViT* terdapat tren peningkatan performa seiring bertambahnya kapasitas model. *ACER* turun dari 14,63% (M1) ke 13,26% (M3) ke 11,39% (M5). Pola ini menunjukkan bahwa peningkatan parameter dari *M1* ke *M5* tidak memberikan perbedaan secara signifikan pada hasil prediksi model. Performa model hanya meningkat sebesar 3,24%. Meskipun begitu tetap dapat dijadikan pertimbangan tambahan mengingat komputasi hanya bertambah sebesar 0,22 detik. Perlu diingat bahwa interpretasi ini perlu dibatasi karena hanya tiga varian yang diuji sehingga tidak cukup untuk menyimpulkan saturasi performa secara umum pada seri *EfficientViT*. Pada *ViT* justru terdapat peningkatan performa yang signifikan. *ACER* turun dari 21,13% (*ViT-B/32*) ke 17,52% (*ViT-B/16*). Hasil pada *ViT* menunjukan bahwa ukuran *patch* sangat berpengaruh terhadap hasil prediksi dikarenakan jumlah fitur yang terbatas.

Jika dianalisa lebih lanjut pada metrik *APCER* dan *BPCER* mengungkapkan pola kesalahan yang berbeda. *ViT-B/16* memiliki *BPCER* terendah (11,91%) namun *APCER* tertinggi (23,13%). Ini berarti model cenderung bias memprediksi sampel sebagai asli sehingga secara proporsional lebih sering gagal mendeteksi serangan yang masuk. Sebaliknya *ViT-B/32* memiliki *BPCER* sebesar 26,49% yang berarti model terlalu agresif mengklasifikasikan wajah asli sebagai serangan. Pola ini juga terlihat pada *confusion matrix* Gambar 4.5 di mana *ViT-B/32* menghasilkan 12.515 prediksi bonafide pada sampel spoof jauh lebih tinggi dibandingkan model lain. Kondisi ini dapat disebabkan oleh dua faktor yaitu ukuran *patch* 32×32 yang lebih besar membatasi informasi spasial halus yang tersedia untuk model. Dari kelima model tersebit *EfficientViT-M5* merupakan model paling seimbang dengan selisih *APCER* dan *BPCER* hanya 1,37%.

4.3.2 Analisis Efisiensi Komputasi

Tabel 4.3 menunjukan varian *EfficientViT* membutuhkan *MACs* yang jauh lebih rendah dibandingkan kedua model *ViT* standar. *EfficientViT-M5* sebagai varian terberat dalam kelompoknya hanya membutuhkan 0,52 *GMACs* sedangkan *ViT-B/32* membutuhkan 2,95 *GMACs* dan ViT-B/16 membutuhkan 11,29 *GMACs*. Artinya *EfficientViT-M5* secara teoritis 5,67 kali lebih ringan dari *ViT-B/32* dan 21,71 kali lebih ringan dari *ViT-B/16*.

Perbedaan ini terkonfirmasi secara empiris pada waktu inferensi. *EfficientViT-M5* memproses satu citra selama 3.89 milidetik sehingga 1,37 kali lebih cepat dari *ViT-B/32* (5,34 ms) dan 3,19 kali lebih cepat dari *ViT-B/16* (12,39 ms). Perlu dicatat bahwa rasio percepatan empiris ini lebih kecil dari rasio *MACs* teoritis. Hal ini wajar karena *MACs* hanya mengukur operasi perkalian-akumulasi dan tidak memperhitungkan faktor lain seperti *memory bandwidth*, efisiensi paralelisasi *GPU*, dan *overhead* operasi pada perangkat keras yang digunakan. Dengan demikian waktu inferensi merupakan ukuran yang lebih representatif terhadap performa nyata dibandingkan *MACs* semata.

Dari sisi jumlah parameter *EfficientViT-M5* hanya memiliki 10,69 juta parameter dibandingkan 57,30 juta (*ViT-B/16*) dan 59,07 juta (*ViT-B/32*), menjadikannya sekitar 5,5 kali

lebih hemat memori. Pada *ViT-B/32* ada kasus menarik dimana jumlah parameter pelatihan jauh lebih tinggi dibandingkan *ViT-B/16* sementara jumlah *MACs* nya lebih rendah. Peningkatan parameter tersebut disebabkan oleh ukuran parameter filter pada *patch* 32×32 lebih besar dibandingkan filter pada ukuran *patch* 16×16.

Apabila performa klasifikasi dan efisiensi komputasi dipertimbangkan secara bersamaan sebagaimana ditunjukkan pada Tabel 4.4 *EfficientViT-M5* menjadi model terbaik pada ketiga dimensi sekaligus apabila dibandingkan dengan kedua model *ViT* standar. Temuan ini mengkonfirmasi bahwa optimasi arsitektur *EfficientViT* berhasil meningkatkan efisiensi sekaligus performa. Keunggulan ini menjadikan *EfficientViT-M5* kandidat yang lebih layak untuk skenario *deployment* yang membutuhkan kecepatan dan efisiensi memori.

TABEL 7
Perbandingan trade-off antara ACER dan MACs model terbaik tiap kelompok.

Model	ACER (%)	MACs (G)	Waktu Inferensi per Citra (ms)
EfficientViT-M1	14,63	**0,17**	**3,67**
EfficientViT-M3	13,26	0,26	3,88
EfficientViT-M5	**11,39**	0,52	3,89
ViT-B/16	17,52	11,29	12,39
ViT-B/32	21,13	2,95	5.34

4.3.3 Analisis Visualisasi Peta Aktivasi

Berdasarkan visualisasi *heatmap KPCA-CAM* pada Gambar 4.6 dan Gambar 4.7 terdapat pola perhatian yang berbeda secara konsisten antara kondisi prediksi yang benar (*TP* dan *TN*) dan kondisi prediksi yang salah (*FP* dan *FN*) pada kedua model.

Pada sampel *TP* dan *TN* kedua model memusatkan perhatian pada area fitur wajah sentral seperti mata, mulut, dan area pipi. Pada sampel *TP* yaitu wajah asli yang benar diprediksi sebagai asli fokus pada area sentral wajah konsisten. Pada sampel *TN* yaitu wajah palsu yang benar terdeteksi perhatian pada area yang sama mengindikasikan bahwa kedua model mampu mendeteksi tanda serangan yang muncul di area wajah seperti tekstur cetak, gradasi kecerahan tidak alami, atau refleksi layar. Pola ini secara semantik relevan terhadap karakteristik *spoofing* dan mendukung keandalan keputusan model pada kondisi prediksi yang benar.

Sebaliknya pada sampel *FP* dan *FN* perhatian kedua model bergeser ke area tepi wajah seperti rambut dan latar belakang. Pada kasus *FP* yaitu sampel *spoof* yang salah diprediksi sebagai *bonafide* pergeseran perhatian ke area non-wajah mengindikasikan bahwa model gagal menangkap artefak serangan dan sebagai gantinya bergantung pada informasi kontekstual yang tidak informatif. Pada kasus *FN* yaitu wajah asli yang salah diprediksi sebagai *spoof* perhatian pada area tepi mengindikasikan model kemungkinan merespons variasi pencahayaan alami atau bayangan sebagai indikator serangan.

Hasil *interpretabilitas* ini perlu dibatasi karena *KPCA-CAM* mengekstraksi komponen non-linear yang paling dominan dari fitur lapisan terakhir tanpa gradien sehingga peta yang dihasilkan merepresentasikan fitur yang paling menonjol secara struktural bukan fitur yang paling diskriminatif secara langsung terhadap keputusan kelas. Oleh karena itu interpretasi hasil ini bersifat indikatif dan perlu divalidasi menggunakan metode lain berbasis gradien pada penelitian selanjutnya.

5. Kesimpulan

Berdasarkan hasil percobaan dan analisis yang telah dilakukan diperoleh kesimpulan sebagai berikut. Model EfficientViT menunjukan pertukaran efisiensi dan akurasi yang sangat baik dalam kasus deteksi *spoofing* wajah. *EfficientViT-M5* mencapai performa klasifikasi tertinggi dengan *ACER* sebesar 11,39% dan akurasi 88,34% mengungguli seluruh model yang diuji termasuk *ViT-B/16* (17,52%) dan *ViT-B/32* (21,13%). *EfficientViT-M5* juga menunjukkan keseimbangan antara *APCER* dan *BPCER* dengan selisih hanya 1,37% menjadikannya model paling seimbang dalam mendeteksi serangan maupun mengidentifikasi wajah asli. Tren peningkatan performa juga terlihat seiring bertambahnya kapasitas pada seri *EfficientViT* meskipun peningkatan nyata baru terjadi dari M3 ke M5.

Seluruh varian *EfficientViT* menunjukkan efisiensi komputasi yang jauh lebih baik dibandingkan *ViT* standar. Hal ini disebabkan karena ukuran parameter yang sangat kecil. *EfficientViT-M5* hanya membutuhkan 0,52 *GMACs* dan memproses satu citra dalam 3,89 milidetik menjadikannya 21,71 kali lebih ringan secara teoritis dan 3,19 kali lebih cepat secara empiris dibandingkan *ViT-B/16*. Optimasi arsitektur yang dilakukan oleh EfficientViT terbukti efektif menekan kebutuhan memori dan komputasi tanpa mengorbankan performa klasifikasi.

Visualisasi menggunakan *KPCA-CAM* menunjukkan pola perhatian yang konsisten pada kedua model. Pada prediksi yang benar (*TP* dan *TN*) kedua model memusatkan perhatian pada area fitur wajah seperti mata dan mulut yang relevan secara semantik terhadap deteksi *spoofing*. Sebaliknya pada prediksi yang salah (*FP* dan *FN*) perhatian bergeser ke area tepi wajah seperti rambut dan latar belakang. Hal ini mengindikasikan bahwa kegagalan klasifikasi berkaitan dengan ketidakmampuan model menangkap artefak serangan yang halus.

Berdasarkan keterbatasan penelitian ini terdapat beberapa hal yang dapat dikembangkan pada penelitian selanjutnya. Penelitian ini tidak melibatkan evaluasi lintas dataset sehingga kemampuan generalisasi model terhadap distribusi data yang berbeda belum dapat dikonfirmasi. Penelitian selanjutnya disarankan untuk melakukan evaluasi protokol *cross-dataset* agar kemampuan generalisasi *EfficientViT* dalam skenario domain shift dapat diukur secara lebih komprehensif.

Varian model *Vision Transformer* yang digunakan adalah variasi *ViT* yang memiliki kecepatan tertinggi dan dilatih tanpa bobot awal. Penelitian berikutnya dapat mengeksplorasi varian *ViT* lainnya agar dapat meningkatkan akurasi model *ViT* serta menggunakan bobot *pretrained* dengan kasus yang serupa.

Interpretabilitas menggunakan *KPCA-CAM* bersifat indikatif karena tidak berbasis gradien. Penelitian selanjutnya disarankan untuk memvalidasi temuan interpretabilitas menggunakan metode berbasis gradien seperti *GradCAM* atau *Attention Rollout*, serta memperluas jumlah sampel visualisasi agar generalisasi pola perhatian dapat dibuktikan secara statistik.

REFERENSI

[1] I. Adjabi, A. Ouahabi, A. Benzaoui, dan S. Jacques, “Multi-block color-binarized statistical images for single-sam-ple face recognition,” *Sensors (Switzerland)*, vol. 21, no. 3, hlm. 1–21, Feb 2021, doi: 10.3390/s21030728.

[2] S. El Morabit, A. Rivenq, M. E. N. Zighem, A. Hadid, A. Ouahabi, dan A. Taleb-Ahmed, “Automatic pain estimation from facial expressions: A comparative

analysis using off-the-shelf cnn architectures," *Electronics (Switzerland)*, vol. 10, no. 16, Agu 2021, doi: 10.3390/electronics10161926.

[3] Y. Khaldi, A. Benzaoui, A. Ouahabi, S. Jacques, A. Taleb-Ahmed, dan S. Jacques, "Ear Recognition Based on Deep Unsupervised Active Learning," *IEEE Sens. J.*, vol. 2021, no. 18, hlm. 20704–20713, 2021, doi: 10.1109/JSEN.2021.3100151ï.

[4] W. Ali, W. Tian, S. U. Din, D. Iradukunda, dan A. A. Khan, "Classical and modern face recognition approaches: a complete review," *Multimed. Tools Appl.*, vol. 80, no. 3, hlm. 4825–4880, Jan 2021, doi: 10.1007/s11042-020-09850-1.

[5] Z. Yu, Y. Qin, X. Li, C. Zhao, Z. Lei, dan G. Zhao, "Deep Learning for Face Anti-Spoofing: A Survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 5, hlm. 5609–5631, Mei 2023, doi: 10.1109/TPAMI.2022.3215850.

[6] A. George dan S. Marcel, "On the Effectiveness of Vision Transformers for Zero-shot Face Anti-Spoofing," dalam *2021 IEEE International Joint Conference on Biometrics (IJCB)*, Shenzhen, China: IEEE, Nov 2021, hlm. 1–8. [Daring]. Tersedia pada: http://arxiv.org/abs/2011.08019

[7] M. Mendes Silva dan D. Jorge Manuel Moreira de Campos Pereira Batista, "VISION TRANSFORMERS FOR FACE ANTI-SPOOFING," Master's Thesis, Universidade de Coimbra, Coimbra, Portugal, 2023.

[8] H.-P. Huang *dkk.*, "Adaptive Transformers for Robust Few-shot Cross-domain Face Anti-spoofing," *European conference on computer vision*, Jul 2023, [Daring]. Tersedia pada: http://arxiv.org/abs/2203.12175

[9] A. Liu *dkk.*, "FM-ViT: Flexible Modal Vision Transformers for Face Anti-Spoofing," *IEEE Transactions on Information Forensics and Security*, Mei 2023, [Daring]. Tersedia pada: http://arxiv.org/abs/2305.03277

[10] X. Liu, H. Peng, N. Zheng, Y. Yang, H. Hu, dan Y. Yuan, "EfficientViT: Memory Efficient Vision Transformer with Cascaded Group Attention," dalam *IEEE/CVF conference on computer vision and pattern recognition*, Mei 2023. [Daring]. Tersedia pada: http://arxiv.org/abs/2305.07027

[11] S. Mehta dan M. Rastegari, "MobileViT: Light-weight, General-purpose, and Mobile-friendly Vision Transformer," *ArXiv*, Okt 2021, [Daring]. Tersedia pada: http://arxiv.org/abs/2110.02178

[12] R. Budiarto Hadiprakoso dan I. K. S. Buana, "Deteksi Serangan Spoofing Wajah Menggunakan Convolutional Neural Network," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 7, no. 3, Des 2021, doi: 10.28932/jutisi.v7i3.4001.

[13] A. Benlamoudi *dkk.*, "Face Presentation Attack Detection Using Deep Background Subtraction," *Sensors*, vol. 22, no. 10, Mei 2022, doi: 10.3390/s22103760.

[14] Y. Qin, Z. Yu, L. Yan, Z. Wang, C. Zhao, dan Z. Lei, "Meta-Teacher For Face Anti-Spoofing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, hlm. 6311–6326, Nov 2021, [Daring]. Tersedia pada: http://arxiv.org/abs/2111.06638

[15] Y. Lee, Y. Kwak, dan J. Shin, "Robust face anti-spoofing framework with Convolutional Vision Transformer," dalam *2023 IEEE International Conference on Image Processing (ICIP)*, IEEE, Jul 2023. [Daring]. Tersedia pada: http://arxiv.org/abs/2307.12459

[16] Z. Ming, Z. Yu, M. Al-Ghadi, M. Visani, M. MuzzamilLuqman, dan J.-C. Burie, "ViTransPAD: Video Transformer using convolution and self-attention for Face Presentation Attack Detection," *2022 IEEE International Conference on Image Processing (ICIP)*, Mar 2022, [Daring]. Tersedia pada: http://arxiv.org/abs/2203.01562

[17] C. Yang *dkk.*, "Lite Vision Transformer with Enhanced Self-Attention," *CoRR*, vol. abs/2112.10809, 2021, Diakses: 15 Mei 2026. [Daring]. Tersedia pada: http://openaccess.thecvf.com/content/CVPR2022/html/Yang_Lite_Vision_Transformer_With_Enhanced_Self-Attention_CVPR_2022_paper.html

[18] S. Karmani, T. Sivakaran, G. Prasad, M. Ali, W. Yang, dan S. Tang, "KPCA-CAM: Visual Explainability of Deep Computer Vision Models using Kernel PCA," *2024 IEEE 26th International Workshop on Multimedia Signal Processing (MMSP)*, Sep 2024, [Daring]. Tersedia pada: http://arxiv.org/abs/2410.00267

[19] Mohammed Bany Muhammad dan Mohammed Yeasin, "Eigen-CAM: Class Activation Map using Principal Components," dalam *international joint conference on neural networks (IJCNN)*, IEEE, 2021.

[20] P. Zhou, X. XIE Zhouchen LIN Shuicheng YAN, X. Xie, Z. Lin, dan S. Yan, "Towards understanding convergence and generalization of AdamW," *IEEE Trans. Pattern Anal. Mach. Intell.*, hlm. 1–8, 2024, [Daring]. Tersedia pada: https://ink.library.smu.edu.sg/sis_research

Lampiran

Tautan paper terkait, kode sumber, dan model yang dilatih: https://drive.google.com/drive/folders/1cbocVjWisuJQNkfG7RLrvCcFBIfc5LnL?usp=sharing