Jurnal Algoritma
E-ISSN: 2302-7339 | P-ISSN: 1412-3622
INSTITUT TEKNOLOGI GARUT

https://jurnal.itg.ac.id/index.php/algoritma
DOI: 10.33364/algoritma/v.23-1.2982

Deteksi *Deepfake* pada Gambar Medis Menggunakan YOLOv11

Pancadrya Yashoda Pasha[1*], Ichsan Taufik[2], Aldy Rialdy Atmadja[3]
[1,2,3]UIN Sunan Gunung Djati Bandung, Indonesia

***email:** 1217050114@student.uinsgd.ac.id

Info Artikel
Dikirim: 6 Oktober 2025
Diterima: 5 Januari 2026
Diterbitkan: 31 Mei 2026

Kata kunci:
CT Scan;
Deep Learning;
Deepfake Detection;
Medical Image;
YOLO.

ABSTRAK

Kemajuan *Artificial Intelligence* memicu terlahirnya tantangan *deepfake* gambar realistis hingga sektor kesehatan. Kontribusi penelitian ini adalah implementasi dan analisis performa YOLOv11 untuk deteksi gambar *deepfake* medis pada *dataset CT scan* paru-paru yang mencakup variasi kasus *benign* dan *malign*. Lingkup penelitian dibatasi pada klasifikasi biner antara asli atau palsu yang diuji secara bertahap. Metode manipulasi CT-GAN dan *stable diffusion* (SD) dipergunakan untuk menguji performa model. Hasil penelitian menunjukkan bahwa model YOLOv11 pada manipulasi gambar buatan *stable diffusion* mencapai nilai akurasi, presisi, *recall*, dan *f1-score* 100%. Manipulasi gambar CT-GAN memiliki kendala dalam membedakan gambar *CT scan* kanker paru-paru yang asli dan palsu. Dengan perbaikan dan peningkatan lanjutan, hasil *fine tuning* YOLOv11 dapat menjadi salah satu opsi model gambar medis deep*fake* yang relatif ringan, cepat, dan akurat. Hasil ini berpotensi mendukung keamanan data pasien dan menjaga integritas diagnostik klinis di masa yang akan datang.

1. PENDAHULUAN

Belakangan ini, teknologi AI berkembang pesat. Cabangnya, *deep learning*, memungkinkan pemrosesan data di ranah *recommender system*, *Natural Language Processing* (NLP), dan *computational vision* [1]. Di era media visual yang masif, kemampuan AI membaca, memproses, dan menghasilkan gambar diaplikasikan untuk pelacakan objek, estimasi gerakan, rekonstruksi adegan, VR, pengenalan wajah, estimasi pose, dan deteksi peristiwa [2] guna meningkatkan efisiensi dan akurasi. Namun, kemajuan ini memunculkan tantangan baru yakni *deepfake*, media artifisial realistis berbasis algoritma hasil olahan AI [3]. Awalnya untuk hiburan, *deepfake* kini merambah berbagai ranah legal maupun ilegal [4]. Penyalahgunaan nya mengancam keamanan dan privasi individu maupun lembaga, serta meningkatkan ancaman sosial, ekonomi, dan politik akibat rupa yang kian realistis [5]. Data Sumsub Q1 2024 mencatat lonjakan signifikan *deepfake* (YoY) diberbagai negara, yaitu India (280%), AS (303%), Afrika Selatan dan Meksiko (500%), Singapura (1100%), Turki (1533%), Indonesia (1550%), Korea Selatan (1625%), hingga China (2800%) [6].

Deepfake kini merambah sektor kesehatan [7], [8], memicu ancaman serius seperti penipuan asuransi, serta manipulasi medis tak terdeteksi yang berisiko fatal akibat salah diagnosis dan gangguan sistem rumah sakit [9], [10]. Riset deteksi gambar medis *deepfake* mulai berkembang. Siddharth Solaiyappana dan Yuxin Wen [7] menguji keaslian gambar *CT scan* menggunakan tiga metode *machine learning* dan lima model *deep learning* secara biner dan multi kelas pada data mentah ter lokalisasi dan teraugmentsi. Penelitian ini menghasilkan model dengan akurasi yang mendekati sempurna dalam deteksi dugaan manipulasi tumor. Pada penelitian lain, Abdel Rahman Alsabbagh dan Omar Al-Kadi menguji model *Deep Convolutional Neural Network* (DCNN) dengan *dataset CT scan* paru-paru dengan jumlah 2.486 gambar [11]. Hasilnya ResNet50V2

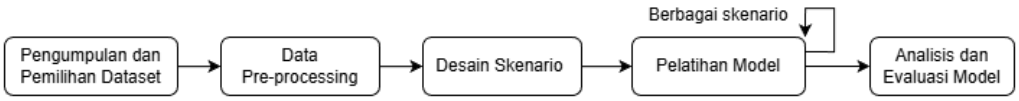
unggul dalam *precision* dan *specificity*, sedangkan DenseNet169 memimpin dalam akurasi, *recall*, dan *F1-score*. Studi lain memanfaatkan *Knee Osteoarthritis X-ray* and *Lung CT scan* untuk mengomparasi berbagai model YOLO [9]. Hasil terbaik diperoleh YoloV5su dengan *recall* 0,997 dan 60% lebih cepat dibandingkan *runner-up* YoloV8x, sedangkan hasil terburuk diperoleh YoloV5nu dengan *recall* 0,91.

Kemudian, penelitian lain memanfaatkan *Deep Neural Network* (DNN) tiga *hidden layer* yang meningkatkan efektivitas deteksi dengan akurasi 93,19%, tingkat kesalahan 6,70% dan alarm palsu 9,10% [12]. Selanjutnya, metode gabungan L2 Regularization, pra-pemrosesan LBP (*Local Binary Pattern*), SVM, dan U-Net diuji dan mencapai akurasi deteksi 93,9%, presisi 94,4%, *recall* 93,9%, *f1-score* 94%, dan AUC (*Area Under Curve*) ROC 99,2% [13]. Selain itu, penelitian lain menggunakan 2.079 gambar menunukkan MobileNetV2 sangat baik dan menjanjikan dibandingkan ResNet50 [14]. Riset lain melakukan deteksi gambar hasil generasi berbasis GAN baru bernama Jekyll. Model CSD SVM dan MesoNet diuji dan hasilnya cukup menjanjikan meskipun memiliki kerentanan terhadap taktik pengelakkan [15]. Penelitian terbaru memperkenalkan *stable diffusion* sebagai metode generasi gambar buatan yang berpotensi disalahgunakan. Setidaknya terdapat dua penelitian yang mencoba memitigasi ancaman tersebut [16], [17]. Dengan meninjau data *CT scan* paru-paru, MRI payudara, dan gambar kanker kulit berwarna, model-model yang diuji dalam kedua penelitian menunjukkan hasil yang cukup memuaskan.

Meskipun penelitian-penelitian terdahulu menunjukkan performa deteksi yang menjanjikan, terdapat celah yang dapat dieksplorasi. Metode berbasis DCNN, hibrida, dan ML konvensional cenderung membebani komputasi sehingga kurang ideal untuk sistem deteksi cepat. Sementara itu, studi menggunakan model deteksi objek yang lebih ringan masih terbatas pada arsitektur lama sehingga potensi keseimbangan antara kecepatan dan akurasi yang ditawarkan oleh arsitektur terbaru yang belum tereksplorasi. YOLOv11 membawa terobosan arsitektur baru, seperti integrasi blok C3K2, yang memungkinkan pemrosesan lebih efisien dengan akurasi yang optimal [18], masih belum diteliti dalam konteks deteksi *deepfake* medis. Oleh karena itu, kontribusi utama pada penelitian ini adalah mengimplementasikan dan menganalisis model YOLOv11 untuk mendeteksi *deepfake* pada gambar medis. Penelitian ini bertujuan untuk mengevaluasi performa model YOLOv11 dalam mendeteksi gambar medis CT-GAN dan SD serta menghasilkan model deteksi yang tervalidasi kinerjanya terhadap ancaman generatif terkini dengan efisiensi dan lebih optimal.

2. METODE PENELITIAN

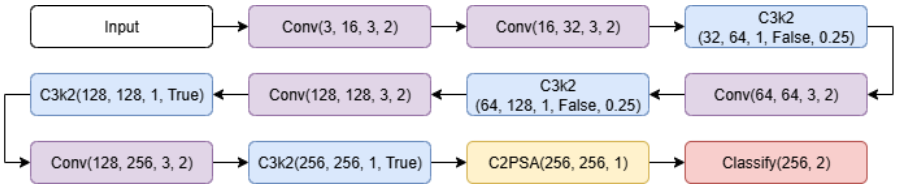
Penelitian ini menggunakan pendekatan kuantitatif dengan melakukan eksperimen komputasi secara bertahap. Metode yang digunakan dalam penelitian ini adalah metode eksperimental *deep learning* yang lazim digunakan dalam penelitian *computer vision* dan citra medis. Metode ini dimulai dari pengumpulan data dan pemilihan *dataset*. Pada langkah pertama ini, *dataset* gambar *CT scan* paru-paru asli dan palsu berhasil dihimpun dari *dataset* publik. Selanjutnya data tersebut dipilih dan dilakukan standarisasi pada tahapan *Data Pre-processing*. Tahapan selanjutnya dilakukan pelatihan model berdasarkan data yang dipilih dan dibagi untuk validasi dan tes. Pelatihan model dilakukan dengan berbagai skenario untuk mendapatkan akurasi optimal dari model yang dihasilkan, salah satunya dengan melakukan *tuning hyperparameter*. Pada akhir tahapan, proses analisis dan evaluasi model dilakukan untuk memastikan model yang optimal dari berbagai skenario pengujian yang dilakukan. Skema alur penelitian yang dilakukan digambarkan pada Gambar 1 berikut ini.



Gambar 1. Diagram Alur Metode Penelitian

Model yang dipilih untuk dilatih merupakan salah satu dari varian YOLOv11, yaitu yolo11n-cls. Nama tersebut diambil dari YOLO versi 11 nano yang dikhususkan untuk klasifikasi. Memiliki 1.6 juta parameter, yolo11n-cls menjadi varian paling kecil dan ringan. Kendati demikian, yolo11n-cls tetap memiliki performa yang

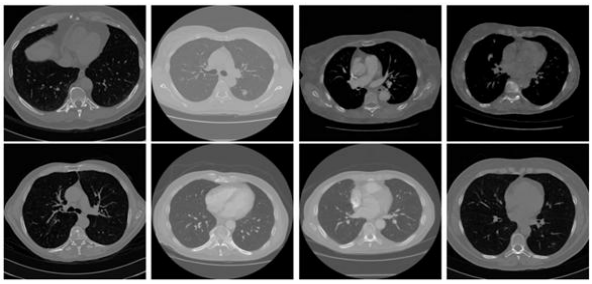
memadai dalam tugas klasifikasi gambar. Alasan dipilihnya varian YOLO klasifikasi adalah karena pada dasarnya *output* yang ingin diraih adalah sebatas membedakan gambar medis asli atau palsu sehingga penggunaan versi YOLO umum yang biasa digunakan untuk *object detection* dengan tambahan *bounding box* dirasa kurang relevan. Arsitektur model ditunjukkan yang dipergunakan pada penelitian ini diilustrasikan pada pada Gambar 2 berikut.



Gambar 2. Arsitektur Model YOLO11n-cls [19]

2.1. *Dataset*

Dataset yang digunakan pada penelitian ini merupakan *dataset* publik dengan lisensi terbuka yang berisi sekumpulan data *CT scan* paru-paru dengan dua jenis metode generasi, yaitu CT-GAN dan *stable diffusion* (SD) [17]. Data terbagi menjadi kelas *True Malign* (TM), *Fake Malign* (FM), *True Benign* (TB), dan *Fake Benign* (FB). Dalam penelitian ini, kelas TM dan TB dikategorikan sebagai label asli, sedangkan kelas FM dan FB dikelompokkan ke dalam label palsu. Contoh dari gambar asli dan palsu disajikan pada Gambar 3.



Gambar 3. Baris pertama merupakan gambar *dataset fake* serta baris kedua merupakan gambar *dataset real*

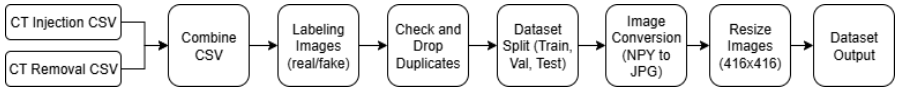
Tabel 1. Jumlah data dalam *dataset*

Split/Class	Real	Fake	Total
Train	1212	2596	3808
Val	152	324	476
Test	152	324	476

Data yang tersedia pada *dataset* terbagi menjadi beberapa *split*. Masing-masing *split* digunakan untuk berbagai tahap pembuatan model yang berbeda. Detail jumlah data dapat dilihat pada Tabel 1.

2.2. *Data Pre-Processing*

Dataset memerlukan *pre-processing* yang diawali dengan penyatuan dua *file* CSV, pelabelan sesuai tipe, dan penghapusan duplikasi. Menggunakan train_test_split Scikit-learn, data dibagi menjadi 80% *train*, 10% *validation*, dan 10% *test*. Format gambar .npy kemudian dikonversi ke JPG via library *library* Python Imaging Library (PIL) dan disesuaikan agar memiliki ukuran 416x416. Terakhir, pengecekan duplikasi ulang antar *split* dilakukan guna mencegah *data leakage* saat uji performa. Detail proses *pre-processing* data dapat dilihat pada Gambar 4.



Gambar 4. Proses *Pre-processing* Data

2.3. *Evaluation Metrics*

Evaluasi terhadap efektivitas model diukur melalui akurasi, presisi, *recall*, dan *f1-score*. Dari dua opsi akurasi Ultralytics, hanya *top-1* (*ground truth* peringkat pertama) pada *epoch* terakhir yang digunakan, sedangkan *top-5* diabaikan karena klasifikasi bersifat biner. Matriks lainnya dihitung menggunakan fungsi *classification report* Scikit-learn yang dijalankan terhadap *test set*. Mengingat keterbatasan akurasi sebagai gambaran umum [20] terhadap data *imbalance* [21] dan konsekuensi dari *error* medis [22], evaluasi dilengkapi matriks lainnya. Presisi adalah persentase alarm yang bukan merupakan alarm palsu. Semakin rendah nilainya berarti model memberikan lebih banyak alarm palsu. *Recall* adalah persentase alarm yang terlewatkan oleh model. Presisi dan *Recall* adalah metrik yang saling bersaing dan sulit untuk dioptimalkan secara bersamaan. Peningkatan pada salah satu nilai akan berdampak penurunan pada nilai yang lain. Oleh karena itu, digunakanlah satu matriks lagi, yaitu *f1-score*. Presisi dan *recall* dirata-ratakan secara *harmonic* sehingga memberikan ringkasan informasi dari kualitas model yang dibuat [20]. Cara perhitungan masing-masing matriks dapat dilihat pada rumus (1)-(4).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$\text{Presisi} = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$F1 - Score = \frac{2\ x\ precision\ x\ recall}{precision\ x\ recall} \quad (4)$$

Sebagai tambahan, pengukuran terhadap nilai *training loss*, *validation loss* dan *confusion matrix* digunakan untuk visualisasi performa model. Perbandingan *train* dan *val loss* digambarkan untuk menunjukkan optimalisasi model sehingga tidak *underfitting* atau *overfitting*. Adapun confusion matrix digunakan untuk melihat persebaran tebakan dari model terhadap kelas yang tersedia. Dalam *confusion matrix* klasifikasi biner, terdapat 4 nilai penting yang perlu diperhatikan, yaitu *True Positive* (TP), *False Positive* (FP), *True Negative* (TN), dan *False Negative* (FN). TP adalah suatu nilai yang didapatkan jika hasil prediksi positif dan memang benar seharusnya positif. FP adalah nilai dari hasil prediksi positif sedangkan aslinya adalah negatif. FP juga dikenal sebagai *Type 1 Error*. TN adalah nilai dari hasil prediksi negatif yang memang seharusnya begitu. FN adalah nilai dari hasil prediksi negatif sedangkan aslinya positif. FP sering disebut sebagai *Type 2 Error* [23].

3. HASIL DAN PEMBAHASAN

Dalam menjalankan setiap tahap penelitian, seluruh proses dilakukan pada satu perangkat laptop yang dilengkapi dengan memori 16 GB RAM, unit pengolah grafis dengan memori 4 GB, serta prosesor kelas menengah dengan enam core. Spesifikasi pengolah grafis yang mempunyai teknologi CUDA digunakan untuk proses *training* dan inferensi model, sehingga dapat berjalan dengan lebih cepat. Model yolo11n-cls dengan bobot dari ImageNet digunakan sebagai *pretrained* model karena ukurannya yang kecil mengingat daya komputasi yang dimiliki sangat terbatas. Bobot dari model yang sudah dilatih disimpan untuk kebutuhan inferensi. *Hyperparameter* yang digunakan pada percobaan awal disesuaikan dengan salah satu literatur yang menggunakan YOLO [9]. Detail masing-masing nilai diperlihatkan pada Tabel 2.

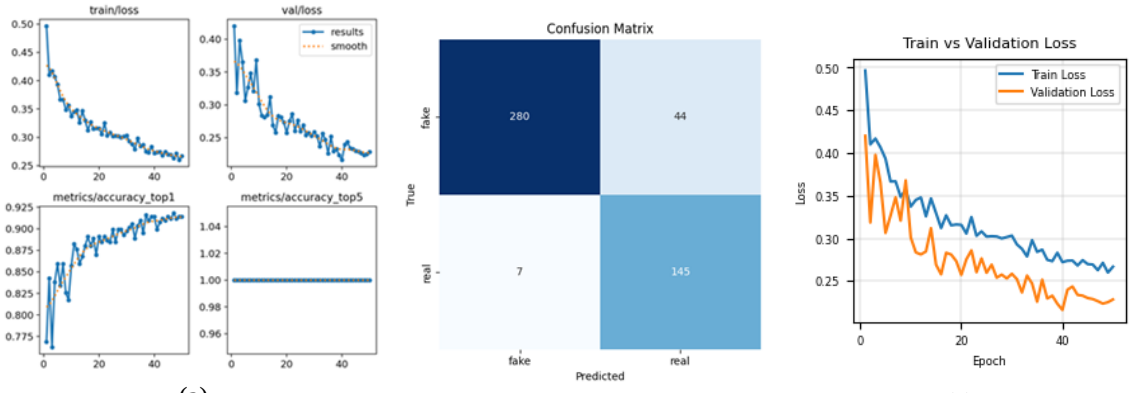
Tabel 2. *Hyperparameter* yang digunakan dalam pelatihan model

Parameter	Nilai
epochs	50
batch	16
imgsz	416
lr0	0,01
lrf (lr1)	0,01
iou	0,0005

Parameter	Nilai
momentum	0,937
warmup_epochs	3
warmup_momentum	0,8
warmup_bias_lr	0,1

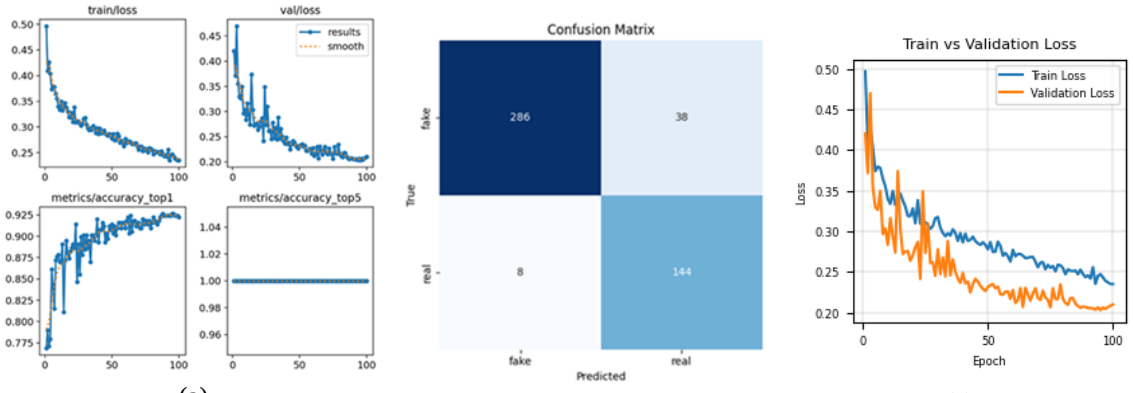
3.1 Skenario Pengujian dengan *Dataset* CT-GAN dan SD Campuran

Pada percobaan awal, model mendapatkan akurasi 0,91387. *Precision*, *recall*, dan *f1-score* kelas *fake* secara berturut-turut adalah 0,98, 0,86, dan 0,92. Sedangkan untuk *precision*, *recall*, dan *f1-score* kelas *real* secara berturut-turut adalah 0,77, 0,95, dan 0,85. Angka-angka tersebut menjadi *baseline* dari penelitian ini. *Dataset* yang digunakan pada percobaan awal ini adalah *dataset* campuran yang terdiri dari gambar CT-GAN dan SD. Karena nilai yang cukup tinggi, maka percobaan kedua dilakukan dengan menambah *epoch* menjadi 100. Hasil nilai akurasinya sebesar 0,92227. *Precision*, *recall*, dan *f1-score* kelas *fake* secara berturut-turut adalah 0,97, 0,88, dan 0,93. Sementara untuk kelas *real*, *precision*, *recall*, dan *f1-score* masing-masing bernilai 0,79, 0,95, dan 0,86. Peningkatan performa terjadi tetapi tidak signifikan sehingga upaya penambahan epoch tidak dilakukan kembali. Visualisasi performa dari kedua percobaan tersebut dapat dilihat pada Gambar 5. dan Gambar 6.



(a) (b) (c)

Gambar 5. Visualisasi performa model *mixed default*.
a) *validation loss, training loss, accuracy (accuracy top 1)*, b) *confusion matrix*, c) *Train vs Validation loss*.



(a) (b) (c)

Gambar 6. Visualisasi performa model *mixed* dengan 100 *epochs*.
a) *validation loss, training loss, accuracy (accuracy top 1)*, b) *confusion matrix*, c) *Train vs Validation loss*.

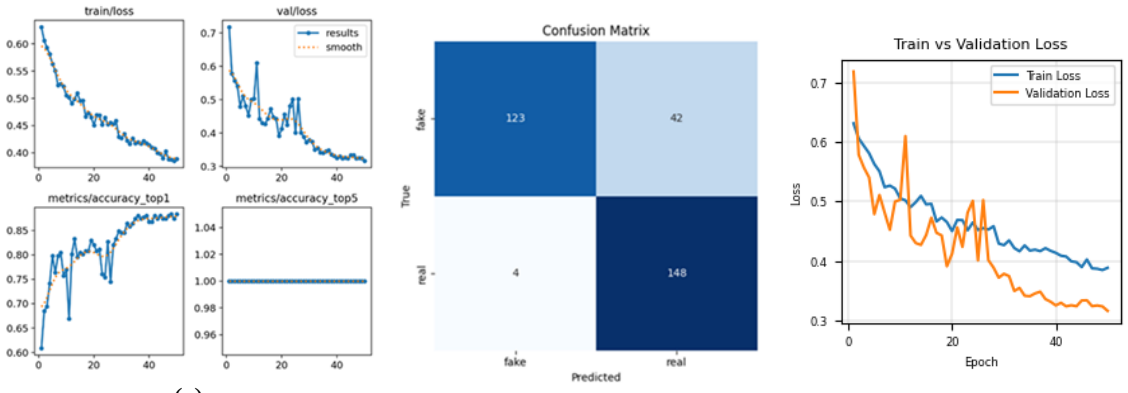
3.2 Skenario Pengujian dengan *Dataset* CT-GAN dan SD yang Terpisah

Dataset ditinjau untuk menentukan langkah lanjutan. Gambar hasil generasi CT-GAN dan SD akhirnya dipisahkan. Distribusi datanya dapat dilihat pada Tabel 3. Percobaan dilanjutkan dengan mencoba melatih model dengan konfigurasi *hyperparameter* yang sama menggunakan masing-masing *dataset*. Pada percobaan model deteksi CT-GAN, akurasi yang didapat sekitar 0,88328. *Precision*, *recall*, dan *f1-score* kelas *fake* secara berturut-turut adalah 0,97, 0,75, dan 0,84. Sementara untuk kelas *real*, *precision*, *recall*, dan *f1-score* masing-masing bernilai 0,78, 0,97, dan 0,87. Belum terlihat ada perubahan yang berarti. Hasil yang mengejutkan hadir dari percobaan menggunakan *dataset* SD. Dengan *hyperparameter* serupa, akurasi melambung menjadi 1 pada epoch terakhirnya. *Precision*, *recall*, dan *f1-score* untuk kelas *fake* dan *real* juga sama mencapai 1. Hal ini menunjukkan pentingnya penyesuaian *dataset* yang baik. Hasil yang tidak konsisten biasanya muncul saat model mendapatkan data yang bervariasi [24]. Visualisasi performa dari kedua percobaan tersebut dapat dilihat pada Gambar 7 dan Gambar 8.

Tabel 3. Jumlah data dalam *dataset* terpisah

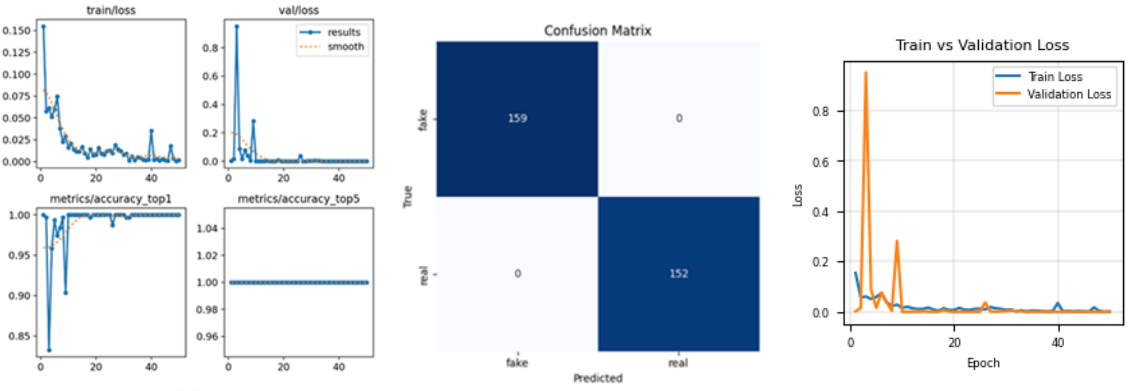
CT-GAN			
Split/Class	**Real**	**Fake**	**Total**
Train	1212	1292	2504
Val	152	165	317
Test	152	165	317

SD			
Split/Class	**Real**	**Fake**	**Total**
Train	1212	1304	2516
Val	152	159	311
Test	152	159	311



(a) (b) (c)

Gambar 7. Visualisasi performa model CT-GAN.
a) *validation loss, training loss, accuracy (accuracy top 1)*, b) *confusion matrix*, c) *Train vs Validation loss.*



(a) (b) (c)

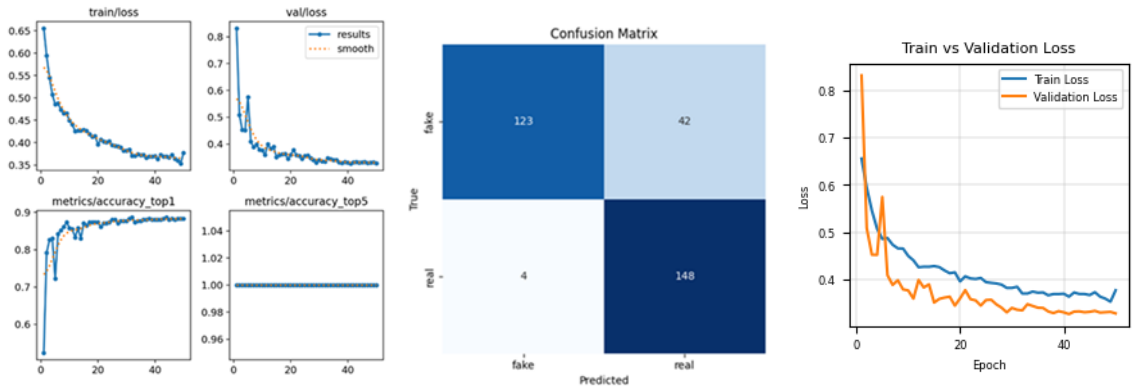
Gambar 8. Visualisasi performa model SD.
a) *validation loss, training loss, accuracy (accuracy top 1)*, b) *confusion matrix*, c) *Train vs Validation loss.*

Model CT-GAN yang belum meraih hasil maksimal di-*tuning* lebih lanjut dengan harapan dapat meningkatkan performanya. Berbagai cara dilakukan mulai dari *tuning* manual hingga otomatis. *Tuning* manual dilakukan dengan mengganti *hyperparameter batch size* dari 16 menjadi 8 dan 32, tetapi hasilnya masih kurang memuaskan. Oleh karena itu, dipilihlah metode *tuning hyperparameter* menggunakan optuna. Optuna merupakan *framework hyperparameter optimization* yang sudah banyak digunakan dalam bidang medis [25], [26]. Percobaan dilakukan sebanyak 12 kali. *Hyperparameter* lr0, momentum, weight_decay, dropout, optimizer, cos_lr, dan patience disesuaikan dengan mengatur rentang nilai, hingga variasi jenis input. Detail dari *tuning* yang dilakukan dapat dilihat pada Tabel 4.

Tabel 4. *Hyperparameter* yang digunakan dalam *tuning* menggunakan optuna

Parameter	**Nilai**
lr0	1e-5 hingga 1e-2
momentum	0,6 hingga 0,98
weight_decay	0,0 hingga 0,01
Dropout	0,0 hingga 0,05
optimizer	SGD/Adam/AdamW
cos_lr	True or False
patience	10 hingga 50

Setelah melewati berjam-jam percobaan, hasil optimasi *hyperparameter* didapatkan. Model dari hasil pelatihan terbaik mendapatkan nilai akurasi terbaik sekitar 0,88328. *Precision*, *recall*, dan *f1-score* kelas *fake* secara berturut-turut adalah 0,97, 0,75, dan 0,84. Sementara untuk kelas *real*, *precision*, *recall*, dan *f1-score* masing-masing bernilai 0,78, 0,97, dan 0,87. Visualisasi performa dari percobaan tersebut dapat dilihat pada Gambar 9. Perbaikan performa yang diinginkan belum juga tercapai. Karena *tuning hyperparameter* belum juga berhasil, maka muncul kecurigaan bahwasannya semua disebabkan *dataset* yang bermasalah. Setelah ditelaah lebih lanjut, terlihat sebagian kecil gambar memiliki karakter visual yang cukup berbeda dengan yang lainnya. Gambar-gambar tersebut kebanyakan berasal dari *scanner CT* yang berbeda. Akhirnya, data berbeda yang dianggap *outliers* dipisahkan atau *dataset* saat ini difilter. Distribusi data ditunjukkan pada Tabel 4.



(a) (b) (c)

Gambar 9. Visualisasi performa model CT-GAN Optuna.
a) *validation loss, training loss, accuracy (accuracy top 1)*, b) *confusion matrix*, c) *Train vs Validation loss.*

Setelah *dataset* sukses difilter, pelatihan model kembali dijalankan menggunakan optuna untuk mencari *hyperparameter* optimal. Kali ini, percobaan dilakukan sebanyak 15 kali dengan *hyperparameter tuning settings* seperti sebelumnya. Waktu 7,692 jam berlalu dan laporan hasil *tuning* terbuat. Nilai akurasi terbaik yang didapatkan sekitar 0,88732. *Precision*, *recall*, dan *f1-score* kelas *fake* secara berturut-turut adalah 0,95, 0,76, dan 0,84. Sementara untuk kelas *real*, *precision*, *recall*, dan *f1-score* masing-masing bernilai 0,80, 0,97, dan 0,88. Selama sejumlah percobaan, optuna mengevaluasi berbagai kombinasi *hyperparameter*

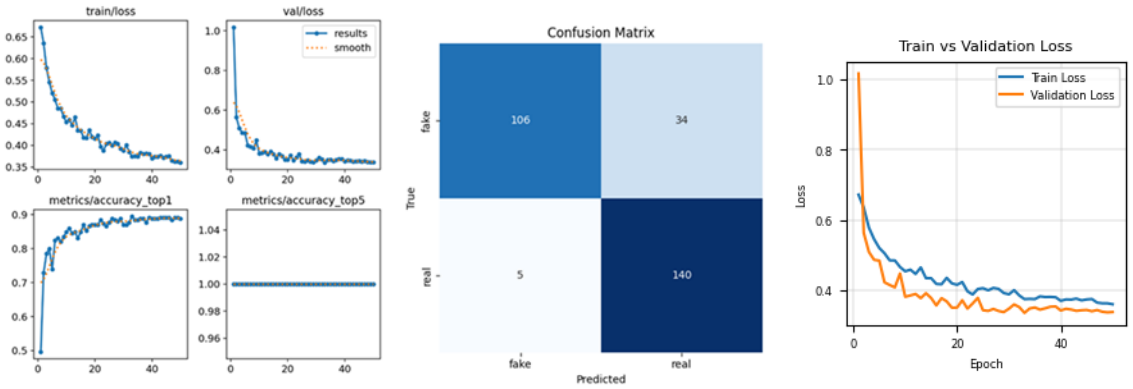
secara bertahap dan memilih konfigurasi dengan performa terbaik. Pada percobaan terbaik, optuna merekomendasikan *learning rate* (lr0) yang sangat rendah, *weight decay*, dan *dropout* tertentu karena memberikan keseimbangan optimal antara stabilitas dan pencegahan *overfitting*. *Optimizer* AdamW serta penonaktifan *cosine learning rate* (cos_lr) juga terbukti memberikan performa lebih baik berdasarkan perhitungan optuna. Selain itu, nilai *patience* yang tinggi membantu model mencapai konvergensi tanpa berhenti terlalu cepat. Dari keseluruhan proses tersebut, konfigurasi ini menghasilkan peningkatan akurasi sebesar 0,404%, sehingga dipilih sebagai *hyperparameter* untuk percobaan selanjutnya. Peningkatan yang cukup kecil ini sejalan dengan fakta bahwa meskipun sebelumnya terjadi *scanner domain shift* karena penggunaan gambar *scanner* yang berbeda-beda, tetapi untuk gambar CT, pengaruh pada performa model *deep learning* tergolong kecil [27]. Nilai-nilai *hyperparameter* terbaik disajikan pada Tabel 5, sementara visualisasi performa dari percobaan kali ini dapat dilihat pada Gambar 10.

Tabel 5. Nilai *hyperparameter* terbaik setelah *tuning*

Parameter	Nilai
lr0	0,00016273415977972064
momentum	0,608191866550641
weight_decay	0,007515928617169716
Dropout	0,3976840321350471
optimizer	AdamW
cos_lr	False
patience	28

Tabel 6. Jumlah data dalam *dataset* CT-GAN yang sudah difilter

Split/Class	Real	Fake	Total
Train	1129	1084	2213
Val	141	143	284
Test	145	140	285



(a) (b) (c)

Gambar 10. Visualisasi performa model CT-GAN optuna *filtered*.
a) *validation loss, training loss, accuracy (accuracy top 1)*, b) *confusion matrix*, c) *Train vs Validation loss.*

3.3 Skenario Pengujian dengan *Dataset* CT-GAN FB dan CT-GAN FM yang Terpisah

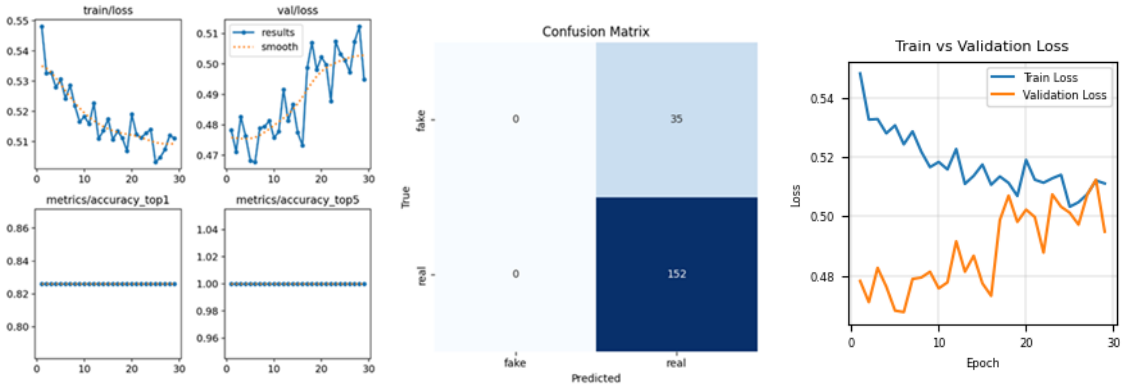
Minimnya peningkatan performa pada percobaan sebelumnya memacu dijalankannya skenario ketiga. Gambar FB tergolong sangat sulit dideteksi. Berbagai model gagal dalam membedakannya dengan gambar asli [17]. Untuk membuktikannya, *dataset* awal CT-GAN, sebelum pembersihan, dipisahkan antara gambar bertipe FB dan FM menjadi *dataset* independen. Distribusi data pada masing-masing *dataset* ditunjukkan pada Tabel 7. Pelatihan model yolo11n-cls kembali dilakukan dengan konfigurasi *hyperparameter* terbaik hasil *tuning* sebelumnya. Selama training, model FB memiliki akurasi yang stagnan 0,82609 selama 29

epoch sehingga *early stopping* terjadi. Nilai *precision*, *recall*, dan F1-scorenya menunjukkan kegagalan total model dalam memprediksi kelas *fake* dengan hasil 0 pada semua nilai tersebut. Berlawanan dengan model FB, model FM menunjukkan hasil yang superior dengan akurasi 0,99298. *Precision*, *recall*, dan *f1-score* kelas *fake* secara berturut-turut adalah 0,99, 0,98, dan 0,98. Sementara untuk kelas *real*, *precision*, *recall*, dan *f1-score* masing-masing bernilai 0,98, 0,99, dan 0,99. Visualisasi performa dari percobaan model FB dan FM dapat dilihat pada Gambar 11. dan Gambar 12. Ringkasan dari performa model pada masing-masing skenario pelatihan dan pengujian disajikan pada Tabel 7.

Tabel 7. Jumlah data dalam *dataset* CT-GAN terpisah

CT-GAN FB			
Split/Class	**Real**	**Fake**	**Total**
Train	1212	308	1520
Val	152	32	184
Test	152	35	187

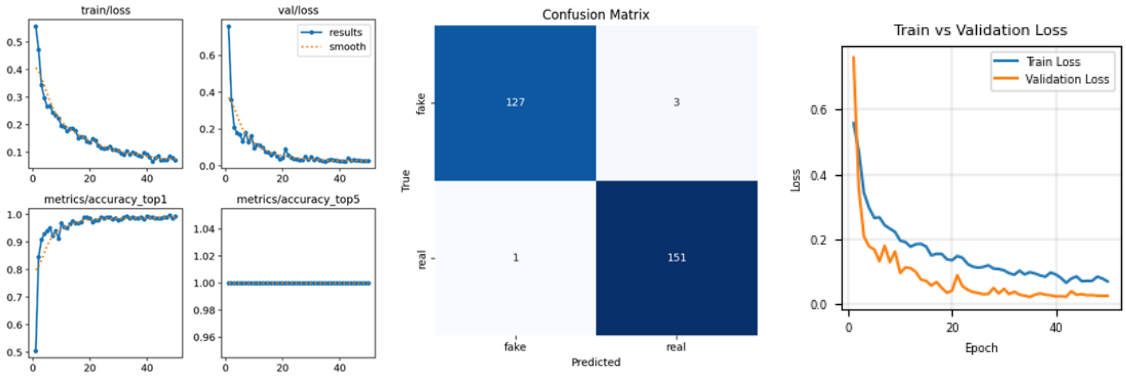
CT-GAN FM			
Split/Class	**Real**	**Fake**	**Total**
Train	1212	984	2196
Val	152	133	285
Test	152	130	282



(a) (b) (c)

Gambar 11. Visualisasi performa model CT-GAN FB

a) *validation loss, training loss, accuracy (accuracy top 1)*, b) *confusion matrix*, c) *Train vs Validation loss.*



(a) (b) (c)

Gambar 12. Visualisasi performa model CT-GAN FM.

a) *validation loss, training loss, accuracy (accuracy top 1)*, b) *confusion matrix*, c) *Train vs Validation loss.*

Tabel 8. Ringkasan performa model dalam beberapa skenario

Skenario	Akurasi	Presisi (*fake* | *real*)	*Recall* (*fake* | *real*)	*F1-Score* (*fake* | *real*)
Mixed default (baseline)	0,91387	0,98 | 0,77	0,86 | 0,95	0,92 | 0,85
Mixed 100 epochs	0,92227	0,97 | 0,79	0,88 | 0,95	0,93 | 0,86
CT-GAN	0,88328	0,97 | 0,78	0,75 | 0,97	0,84 | 0,87
SD	1,00	1,00 | 1,00	1,00 | 1,00	1,00 | 1,00
CT-GAN Optuna	0,88328	0,97 | 0,78	0,75 | 0,97	0,84 | 0,87
CT-GAN Optuna *Filtered*	0,88732	0,95 | 0,80	0,76 | 0,97	0,84 | 0,88
CT-GAN FB	0,82609	0,00 | 0,81	0,00 | 1,00	0,00 | 0,90
CT-GAN FM	0,99298	0,99 | 0,98	0,98 | 0,99	0,98 | 0,99

Untuk memastikan hasil ringkasan performa valid dan tidak hanya kebetulan karena pembagian data, pengujian lanjutan dilakukan menggunakan *stratified k-fold cross validation* dengan lima *fold* pada tiga skenario dengan akurasi tertinggi selain SD. SD tidak diikutsertakan karena hasilnya yang sempurna memerlukan perhatian khusus. Hasil pengujian mengonfirmasi bahwasanya performa ketiga model hanya menunjukkan sedikit penurunan akurasi sekitar 1,07% hingga 2,33%. Hal tersebut wajar dan memberikan gambaran performa yang lebih realistis untuk semua skenario.

Berdasarkan angka dan grafik yang didapatkan, terlihat bahwa model YOLO yang telah dilatih dapat dengan cukup baik mendeteksi gambar medis deep*fake* yang menyerupai gambar medis asli. Pada gambar manipulasi *stable diffusion*, model berhasil mendeteksi semua gambar hasil manipulasi. Meskipun terkesan fantastis, hal tersebut menimbulkan kekhawatiran mengenai potensi *data leakage* ataupun *overfitting*. Berbagai langkah pencegahan telah dilakukan, termasuk verifikasi bahwa tidak ada gambar yang muncul pada lebih dari satu *split* data serta pemeriksaan grafik *train vs validation loss* yang menunjukkan pola yang relatif stabil tanpa perbedaan signifikan. Namun demikian, interpretasi terhadap hasil tanpa celah tersebut perlu dilakukan secara hati-hati dan evaluasi tambahan diperlukan untuk memastikan keandalan model dalam situasi yang lebih beragam. Pada gambar manipulasi CT-GAN, model mampu membedakan gambar yang ditambahkan tumor kanker, tetapi tidak dengan gambar yang tumor di dalamnya dihapus. Gambar *fake benign* membuat berbagai percobaan yang melibatkannya memiliki performa yang kurang berkembang. Hal ini terjadi karena proses penghapusan tumor menyisakan jejak visual yang sangat minim. Secara spesifik, CT-GAN pada *dataset* yang digunakan memanfaatkan pola *noise* untuk menyamarkan area bekas penghapusan tersebut, sehingga manipulasi tampak identik dengan jaringan sehat asli [13]. Dengan *masking* seperti itu, model belum mampu mengidentifikasi gambar palsu tersebut. Selain itu, keterbatasan jumlah data yang hanya sepertiga dari data *fake malign* kemungkinan turut berpengaruh terhadap sulitnya model mencapai performa yang diharapkan. Secara keseluruhan, performa model yang tinggi pada sebagian besar skenario menunjukkan bahwa data yang digunakan telah cukup representatif dan memadai untuk melatih model secara optimal. Namun, performa yang lemah pada satu skenario tertentu mengindikasikan adanya potensi bias model, di mana model lebih sensitif terhadap manipulasi penambahan dibandingkan penghapusan, sejalan dengan minimnya jejak visual dan keterbatasan kuantitas data pada skenario tersebut.

Temuan-temuan yang didapatkan mendorong perlunya melihat posisi model secara lebih luas, khususnya ketika dibandingkan dengan pendekatan yang telah ada dalam literatur. Perbandingan performa YOLOv11 dengan model dari penelitian terdahulu perlu dilakukan secara konseptual, bukan angka secara langsung disebabkan perbedaan pada *dataset* yang digunakan. Model YOLOv11 menunjukkan dua keunggulan dibandingkan dengan model-model terbaik dari penelitian sebelumnya, seperti *framework* U-Net+SVM, DenseNet121, DenseNet169 dan ResNet50V2. Keunggulan pertama adalah kapabilitas baru dalam mendeteksi manipulasi *stable diffusion*. Model YOLOv11 berhasil mencapai performa yang luar biasa pada *dataset* tersebut. Keunggulan kedua adalah performa yang sangat tinggi dalam skenario spesifik injeksi tumor (CT-GAN FM). YOLOv11 mencatatkan akurasi tinggi dengan presisi dan *recall* yang seimbang untuk

tugas tersebut. Hal tersebut menunjukkan potensi YOLOv11 untuk menjadi pilihan model deteksi dengan komputasi yang efisien dan performa yang dapat bersaing dengan model-model yang lebih kompleks.

Dibalik urgensi penguatan performa model sebelum digunakan di dunia nyata, tidak bisa dipungkiri apabila YOLO, khususnya YOLOv11, dapat diandalkan untuk tugas deteksi secara cepat dan cukup akurat. Model membutuhkan input gambar yang sesuai untuk bekerja, yaitu png atau jpg, sementara data medis biasanya disimpan dalam format *Digital Imaging and Communications in Medicine* (DICOM). Maka dari itu, diperlukan semacam konverter jika ingin menggunakan model pada aplikasi kesehatan sungguhan. Dengan performa yang diperbaiki dan mekanisme *input*, proses, *output* yang baik, model YOLOv11 yang sudah dilatih diharapkan dapat menjadi salah satu instrumen pengecekan keaslian gambar medis yang andal.

4. KESIMPULAN

Penelitian ini berkontribusi pada pengembangan YOLOv11 sebagai model deteksi *deepfake* medis optimal dan efisien. Secara ilmiah, temuan penelitian mengimplikasikan bahwa YOLOv11 mampu menyeimbangkan efisiensi komputasi dan akurasi tinggi dalam ranah deteksi gambar medis, tanpa perlu bergantung pada model *deep learning* konvensional yang berat. Manipulasi yang dilakukan CT-GAN pada gambar *fake malign* dan seluruh gambar *fake stable diffusion* menunjukkan akurasi, presisi, *recall*, dan *f1-score* di atas 98%.

Kendati demikian, penelitian ini memiliki keterbatasan, khususnya pada penurunan sensitivitas model dalam mendeteksi gambar CT-GAN *fake benign*. Hal ini mengindikasikan bahwa fitur manipulasi halus pada jenis manipulasi tersebut masih menjadi tantangan bagi model saat ini. Selain itu, potensi generalisasi model masih terbatas pada *CT scan* paru-paru dan metode generasi spesifik. Penelitian ke depannya dapat dikembangkan dengan meningkatkan kemampuan model dalam mendeteksi kegagalan dalam rekognisi gambar medis bertipe CT-GAN *fake benign*. Selain itu, riset dapat dilanjutkan dengan membuat model generalisasi yang dapat mengenali tipe manipulasi yang berbeda-beda seperti CTGAN ataupun SD dalam satu model. Variasi riset lain juga perlu dilakukan dengan menggunakan jenis gambar medis yang berbeda agar deteksi gambar deep*fake* mencakup berbagai macam gambar penyakit dan jenis generasi.

REFERENSI

[1] H. Taherdoost, “Deep Learning and Neural Networks: Decision-Making Implications,” *Symmetry*, vol. 15, no. 9. 2023. doi: 10.3390/sym15091723.

[2] L. Zou, “Chapter 5 - Meta-learning for computer vision,” L. B. T.-M.-L. Zou, Ed. Academic Press, 2023, pp. 91–208. doi: https://doi.org/10.1016/B978-0-323-89931-4.00012-2.

[3] M. Mustak, J. Salminen, M. Mäntymäki, A. Rahman, and Y. K. Dwivedi, “Deepfakes: Deceptions, mitigations, and opportunities,” *J. Bus. Res.*, vol. 154, p. 113368, 2023, doi: https://doi.org/10.1016/j.jbusres.2022.113368.

[4] İ. İlhan, E. Balı, and M. Karaköse, “An Improved DeepFake Detection Approach with NASNetLarge CNN,” in *2022 International Conference on Data Analytics for Business and Industry (ICDABI)*, 2022, pp. 598–602. doi: 10.1109/ICDABI56818.2022.10041558.

[5] J. Park, L. H. Park, H. E. Ahn, and T. Kwon, “Coexistence of Deepfake Defenses: Addressing the Poisoning Challenge,” *IEEE Access*, vol. 12, no. January, pp. 11674–11687, 2024, doi: 10.1109/ACCESS.2024.3353785.

[6] Sumsub, “Deepfake Cases Surge in Countries Holding 2024 Elections, Sumsub Research Shows | Sumsub,” 2024. https://sumsub.com/newsroom/deepfake-cases-surge-in-countries-holding-2024-elections-sumsub-research-shows/ (accessed Jan. 09, 2025).

[7] S. Solaiyappan and Y. Wen, “Machine learning based medical image deepfake detection: A comparative study,” *Mach. Learn. with Appl.*, vol. 8, no. March, p. 100298, 2022, doi: 10.1016/j.mlwa.2022.100298.

[8] Y. Patel *et al.*, “Deepfake Generation and Detection: Case Study and Challenges,” *IEEE Access*, vol. 11, no. October, pp. 143296–143323, 2023, doi: 10.1109/ACCESS.2023.3342107.

[9] M. Karaköse, H. Yetiş, and M. Çeçen, “A New Approach for Effective Medical Deepfake Detection in Medical Images,” *IEEE Access*, vol. 12, no. March, pp. 52205–52214, 2024, doi:

10.1109/ACCESS.2024.3386644.

[10] Y. S. Kim, H. J. Song, and J. H. Han, "A Study on the Development of Deepfake-Based Deep Learning Algorithm for the Detection of Medical Data Manipulation," *Webology*, vol. 19, no. 1, pp. 4396–4409, 2022, doi: 10.14704/web/v19i1/web19289.

[11] A. R. Alsabbagh and O. Al-Kadi, "Comparative Analysis of Deep Convolutional Neural Networks for Detecting Medical Image Deepfakes," 2024, [Online]. Available: http://arxiv.org/abs/2406.08758

[12] K. M. A. Alheeti, A. Alzahrani, N. Khoshnaw, and D. Al-Dosary, "Intelligent Deep Detection Method for Malicious Tampering of Cancer Imagery," in *2022 7th International Conference on Data Science and Machine Learning Applications (CDMA)*, 2022, pp. 25–28. doi: 10.1109/CDMA54072.2022.00010.

[13] S. A. and S. Narayan, "Detection of GAN-manipulated Medical Images through Deep Learning Techniques," in *2024 International Conference on Advances in Modern Age Technologies for Health and Engineering Science (AMATHE)*, 2024, pp. 1–6. doi: 10.1109/AMATHE61652.2024.10582065.

[14] B. R. Reddy, M. S. Kumar, P. Neelima, C. Sushama, V. N. Sailaja, and D. Ganesh, "Medical Image Tampering Detection using Deep Learning," in *2024 5th International Conference on Smart Electronics and Communication (ICOSEC)*, 2024, pp. 1480–1485. doi: 10.1109/ICOSEC61587.2024.10722104.

[15] N. Mangaokar, J. Pu, P. Bhattacharya, C. K. Reddy, and B. Viswanath, "Jekyll : Attacking Medical Image Diagnostics using Deep Generative Models," no. 1, pp. 139–157, 2020, doi: 10.1109/EuroSP48549.2020.00017.

[16] M. A. Arshed, S. Mumtaz, Ștefan C. Gherghina, N. Urooj, S. Ahmed, and C. Dewi, "A Deep Learning Model for Detecting Fake Medical Images to Mitigate Financial Insurance Fraud," *Computation*, vol. 12, no. 9, p. 173, 2024, doi: 10.3390/computation12090173.

[17] F. Grabovski, L. Yasur, G. Amit, Y. Elovici, and Y. Mirsky, "Back-in-Time Diffusion: Unsupervised Detection of Medical Deepfakes," vol. 1, no. 1, pp. 1–18, 2024, [Online]. Available: http://arxiv.org/abs/2407.15169

[18] R. Khanam and M. Hussain, "YOLOv11: An Overview of the Key Architectural Enhancements," vol. 2024, pp. 1–9, 2024, [Online]. Available: http://arxiv.org/abs/2410.17725

[19] Ultralytics, "Image Classification - Ultralytics YOLO Docs," 2025. https://docs.ultralytics.com/tasks/classify/#models (accessed Jun. 11, 2025).

[20] M. Conciatori, A. Valletta, and A. Segalini, "Improving the quality evaluation process of machine learning algorithms applied to landslide time series analysis," *Comput. Geosci.*, vol. 184, no. November 2023, p. 105531, 2024, doi: 10.1016/j.cageo.2024.105531.

[21] A. Hernandez-Guedes, I. Santana-Perez, N. Arteaga-Marrero, H. Fabelo, G. M. Callico, and J. Ruiz-Alzola, "Performance Evaluation of Deep Learning Models for Image Classification Over Small Datasets: Diabetic Foot Case Study," *IEEE Access*, vol. 10, no. December, pp. 124373–124386, 2022, doi: 10.1109/ACCESS.2022.3225107.

[22] V. Plevris, "Assessing uncertainty in image-based monitoring: addressing false positives, false negatives, and base rate bias in structural health evaluation," *Stoch. Environ. Res. Risk Assess.*, pp. 959–972, 2025, doi: 10.1007/s00477-024-02898-7.

[23] Ž. Vujović, "Classification Model Evaluation Metrics," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 6, pp. 599–606, 2021, doi: 10.14569/IJACSA.2021.0120670.

[24] R. Kushol, A. H. Wilman, S. Kalra, and Y. H. Yang, "DSMRI: Domain Shift Analyzer for Multi-Center MRI Datasets," *Diagnostics*, vol. 13, no. 18, pp. 1–20, 2023, doi: 10.3390/diagnostics13182947.

[25] P. Lacerda, B. Barros, C. Albuquerque, and A. Conci, "Hyperparameter optimization for COVID-19 pneumonia diagnosis based on chest CT," *Sensors*, vol. 21, no. 6, pp. 1–11, 2021, doi: 10.3390/s21062174.

[26] P. Srinivas and R. Katarya, "hyOPTXg: OPTUNA hyper-parameter optimization framework for predicting cardiovascular disease using XGBoost," *Biomed. Signal Process. Control*, vol. 73, p. 103456, 2022, doi: https://doi.org/10.1016/j.bspc.2021.103456.

[27] G. Szumel *et al.*, "The Impact of Scanner Domain Shift on Deep Learning Performance in Medical Imaging: an Experimental Study," pp. 1–11, 2024, [Online]. Available: https://arxiv.org/abs/2409.04368