

## DETEKSI DEEPFAKE DALAM CITRA MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK (CNN)

Steven Adventino Gulo, Ayu Amelia Pertiwi, Sarah Putri Syaifullah Nasution, Hermawan Syahputra

Ilmu Komputer, Universitas Negeri Medan  
Jl. William Iskandar Ps. V Medan, Indonesia  
stevenadventgulo@gmail.com

### ABSTRAK

Dewasa ini, banyak sekali citra yang bertebaran dimana-mana, baik di dunia nyata, maupun di dunia maya. Di dunia nyata misalnya di majalah, koran, poster, dan lainnya, sedangkan di dunia maya biasanya terdapat pada berita di internet dan media sosial. Salah satu citra adalah gambar. Salah satu tantangan untuk menghadapi konten-konten yang mengandung deepfake termasuk wajah, ialah sulitnya mendeteksinya secara visual atau dengan mata langsung. Diperlukan metode yang berbasis komputasi dan otomatis untuk mengetahui atau landmark wajah yang telah dimanipulasi deepfake. Ini sangat berguna untuk menyaring media-media yang sudah dimanipulasi deepfake. Convolutional Neural Network (CNN) merupakan salah satu algoritma dalam deep learning yang dirancang untuk mengolah data citra sebagai masukan, digunakan salah satu arsitekturnya yaitu EfficientNetB0. Penelitian ini juga menggunakan OpenCV sebagai landmark wajah. Metode dimulai dari pengumpulan dataset, preprocessing gambar, pengekstrasian landmark, klasifikasi, dan pengujian aplikasi. Hasil dari pelatihan dataset dengan akurasi sebesar 0,91. Setelah itu, diuji dalam aplikasi dengan antarmuka gradio sebagai kemudahan pembuatan aplikasi dengan confidence threshold deepfake  $>0.6$  dan real  $<0.6$ , dihasilkan gambar deepfake sebesar 0,94 dan gambar real sebesar 0,04. Penelitian ini sangat berguna tentang pendeteksian dan pengenalan wajah terutama deteksi deepfake. Saran penelitian selanjutnya dengan mendeteksi video dan suara, serta menggunakan arsitektur CNN lain.

**Kata kunci :** *deepfake, OpenCV, Convolutional Neural Network (CNN), gradio, EfficientNetB0, landmark*

### 1. PENDAHULUAN

Dewasa ini, banyak sekali citra yang bertebaran dimana-mana, baik di dunia nyata, maupun di dunia maya. Di dunia nyata misalnya di majalah, koran, poster, dan lainnya, sedangkan di dunia maya biasanya terdapat pada berita di internet dan media sosial. Salah satu citra adalah gambar. Gambar salah satunya dapat berupa wajah seseorang, digunakan untuk memberikan detail informasi seseorang pada berita, baik di koran, poster, atau majalah, ataupun pada media sosial. Tapi dengan teknologi yang berkembang, wajah pada gambar dapat dimanipulasi dengan mengeditnya menjadi suatu deepfake. Deepfake ini dapat digunakan sebagai hal yang negatif, seperti penipuan. Kebanyakan orang awam, terutama orang tua, dapat terkecoh dengan hal ini. Jadi perlu ketelitian yang tinggi apakah wajah pada suatu gambar termasuk real atau deepfake.

Citra digital adalah gambar diam maupun gambar bergerak yang pengolahannya dilakukan menggunakan komputer secara digital [1]. Salah satu tantangan untuk menghadapi konten-konten yang mengandung deepfake termasuk wajah, ialah sulitnya mendeteksinya secara visual atau dengan mata langsung. Diperlukan metode yang berbasis komputasi dan otomatis untuk mengetahui pola-pola atau landmark wajah yang telah dimanipulasi deepfake. Ini sangat berguna untuk menyaring media-media yang sudah dimanipulasi deepfake. Penggunaan teknik deteksi deepfake dapat menggunakan Convolutional Neural Network (CNN) yang dikhususkan dalam pendeteksian wajah dan terbukti efektif dalam ekstraksi

informasi dan artefak gambar dengan tingkat akurasi yang cukup tinggi.

Penelitian ini bertujuan untuk mendeteksi gambar yang sudah dimanipulasi deepfake dengan menggunakan Convolutional Neural Network (CNN). CNN memiliki keunggulan dalam mengenali fitur-fitur spasial yang kompleks dalam gambar, yang menjadikannya sangat potensial untuk digunakan dalam mendeteksi artefak atau manipulasi pada citra yang sudah terkena deepfake. CNN yang akan digunakan oleh peneliti adalah EfficientNetB0 dimana merupakan salah satu arsitektur CNN dengan efisiensi dan akurasi yang tinggi. Tujuan menggunakan EfficientNetB0 adalah untuk mendapatkan keseimbangan optimal antara akurasi dan efisiensi komputasi, serta deep learning pertama dari varian EfficientNet lainnya sehingga hanya memerlukan daya komputasi yang tidak begitu tinggi.

Ada banyak artikel-artikel yang berhubungan dengan EfficientNetB0. Salah satu penelitian terdahulu mencoba data training pada dua dataset yang berbeda yaitu DFDC (DeepfakeDetection) dan Celeb-DF. Akurasinya menunjukkan bahwa DFDC menghasilkan 84.44% sedangkan Celeb-DF menghasilkan 30.81% [2]. Sehingga peneliti mengambil dataset yang berasal dari DFDC

Penulis berharap jurnal ini dapat membantu orang-orang bagaimana cara mendeteksi deepfake dengan menggunakan deep learning, terutama CNN. Penulis juga berharap agar jurnal ini dapat menjadi referensi oleh penulis lain dalam pendeteksian citra yang menggunakan metode deep learning, serta dapat

dikembangkan lagi yang efisiensi dan keakuratan yang lebih tinggi lagi.

## 2. TINJAUAN PUSTAKA

### 2.1. Deepfake

Deepfake merupakan teknologi AI, termasuk deep learning, yang dapat memanipulasi fitur-fitur wajah pada manusia [3]. Deepfake dapat berupa gambar/ foto, video, ataupun suara. Pada gambar, deepfake dapat mengubah wajah orang pada gambar tersebut menjadi wajah yang baru (fitur wajahnya terganti), praktik ini sering digunakan dalam modus penipuan dan pencemaran nama baik.

Pada video, deepfake ini biasanya sering digunakan dalam industri layar lebar, seperti saat aktor/aktris yang sudah tua ingin memerankan karakter yang lebih muda dari mereka, maka deepfake digunakan untuk mengurangi beberapa fitur wajah mereka. Deepfake pada video juga berbahaya karena maraknya iklan-iklan di media sosial yang menggunakan video-video orang ternama dan diubah fitur wajahnya agar orang dapat percaya.

Pada suara, ini sangat terkait pada video yang sudah diubah deepfake-nya tadi. Suara yang ada pada video tersebut, kemudian diubah agar mirip dengan wajah yang fiturnya sudah diubah tadi. Ada juga modus penipuan dengan menggunakan suara dari orang-orang terkenal dengan permintaan yang bermacam-macam. Deepfake dapat dilihat dengan adanya kejanggalan pada gerakan mata, mulut, hidung, serta blur dari hasil deepfake tersebut. Namun dengan teknologi yang canggih ini, banyak juga deepfake yang tidak bisa dilihat hanya dengan mata manusia, diperlukan proses komputasi untuk mendeteksi apakah suatu citra itu real atau deepfake.



Gambar 1. Perbandingan gambar real dan deepfake [4]

Dari gambar di atas bisa diartikan bahwa perbandingan gambar real dan deepfake sangat signifikan. Deepfake di atas tidak bisa dilihat dengan hanya visual dari mata manusia saja. Oleh karena itu, sangat dibutuhkan model komputasi untuk mendeteksi gambar tersebut.

### 2.2. Deep Learning

Deep learning pada komputasi merupakan cabang dari machine learning dengan algoritma yang mirip dengan struktur sistem saraf pusat manusia yaitu otak, struktur otak tersebut dinamakan Jaringan Saraf Tiruan atau disebut juga Artificial Neural Networks (ANN) [5]. Deep learning sangat cocok untuk mengenali data dengan pola besar dan tidak terstruktur, seperti citra dan suara. Ini sangat cocok untuk mendeteksi artefak pada sebuah citra.

Deep learning membutuhkan data yang banyak dan komputasi yang besar, dalam artinya dibutuhkan sistem komputasi yang bagus dan dataset dengan skala besar agar performa dalam pengenalan wajah mencapai akurasi dan efisiensi yang optimal. Salah satu arsitektur deep learning yang akan dipakai dalam penelitian ini adalah Convolutional Neural Network (CNN).

### 2.3. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) merupakan salah satu algoritma dalam deep learning yang dirancang untuk mengolah data citra sebagai masukan. CNN bekerja dengan mempelajari bobot dan bias untuk mengidentifikasi bagian-bagian penting dalam gambar, sehingga mampu mengenali serta membedakan berbagai objek yang ada di dalamnya [6]. Dengan kemampuan tersebut, CNN menjadi fondasi utama dalam berbagai tugas visual seperti klasifikasi gambar, deteksi objek, dan pengenalan pola.

CNN memiliki sejumlah komponen utama, seperti lapisan konvolusi, aktivasi (ReLU), pooling, dan fully connected, yang disusun secara berlapis untuk memproses citra digital secara mendalam. Proses dimulai dari lapisan konvolusi yang menggeser filter atau kernel di atas gambar guna menangkap fitur lokal seperti tepi atau pola sederhana, lalu menghasilkan feature map sebagai representasi awal. Filter ini akan terus disesuaikan melalui proses pelatihan menggunakan metode backpropagation, sebagaimana bobot pada lapisan fully connected. Selanjutnya, fungsi aktivasi ReLU digunakan untuk menambahkan kemampuan non-linear pada jaringan, sehingga model dapat mengenali pola yang lebih kompleks dalam data.

Setelah fitur-fitur utama berhasil diekstraksi, lapisan pooling seperti *max pooling* digunakan untuk mengecilkan ukuran data sambil tetap mempertahankan informasi penting, sehingga proses komputasi menjadi lebih ringan dan tetap akurat meskipun terjadi pergeseran posisi objek dalam gambar. Selanjutnya, hasil dari *feature map* diratakan (*flatten*) dan diteruskan ke lapisan fully connected, yang berfungsi menyatukan seluruh fitur tersebut guna melakukan proses klasifikasi atau prediksi, biasanya dengan menggunakan fungsi aktivasi seperti softmax.

## 2.4. Confusion Matrix

Confusion Matrix adalah metode yang digunakan untuk mengukur akurasi hasil klasifikasi dengan membandingkan prediksi model terhadap data aktual dalam konsep data mining [7]. Confusion Matrix digambarkan sebagai tabel dengan empat komponen utama, yaitu True Negative (TN), True Positive (TP), False Positive (FP), dan False Negative (FN).

True Positive (TP) ialah ketika model memprediksi citra deepfake dengan benar, ini berarti bahwa model mengerti bahwa gambar yang sudah dimanipulasi oleh deepfake yang setelah diprediksi bahwa gambar tersebut adalah benar-benar deepfake. True Negative (TN) ialah ketika model memprediksi citra deepfake dengan benar, ini berarti bahwa model mengerti bahwa gambar itu benar-benar real.

False Positive (FP) ialah ketika model gagal memprediksi gambar real dengan benar, ini berarti bahwa model mengira bahwa gambar real yang telah diprediksi adalah gambar real. False Negative (FN) ialah ketika model gagal memprediksi gambar real, ini berarti bahwa model mengira bahwa gambar real yang diprediksi adalah gambar deepfake.

|                  |              | Actual Values |              |
|------------------|--------------|---------------|--------------|
|                  |              | Positive (1)  | Negative (0) |
| Predicted Values | Positive (1) | TP            | FP           |
|                  | Negative (0) | FN            | TN           |

Gambar 2. Tabel confusion matrix

Setelah ditemukan, confusion matrix akan dipakai untuk menghitung nilai accuracy, precision, recall, dan f1-score [8]. Nilai accuracy dihitung berdasarkan seberapa banyak prediksi yang benar pada seluruh data. Precision mengukur ketepatan model dalam memprediksi kelas positif, baik True Positive (TP) dan False Negative (FN). Hasil recall menunjukkan kemampuan model yang tersedia dalam mengenali semua data positif. F-1 merupakan hasil dari rata-rata harmonis antara precision dan recall. Untuk persamaan dalam menghitung accuracy, precision, recall, dan f-1 score dapat dilihat sebagai berikut:

a. Accuracy

$$\frac{TP + TN}{TP + TN + FP + FN}$$

b. Precision

$$\frac{TP}{TP + FP}$$

c. Recall

$$\frac{TP}{TP + FN}$$

d. F-1 Score

$$2 \left( \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \right)$$

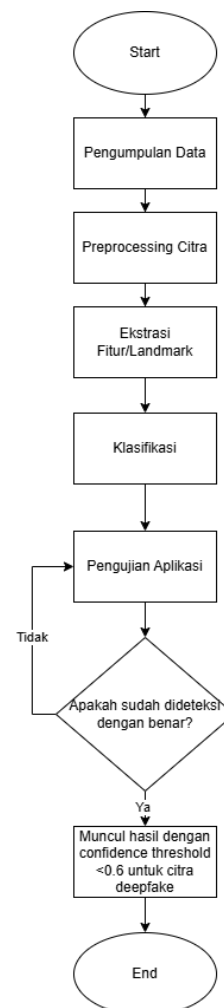
## 2.5. OpenCV (Open Source Computer Vision)

OpenCV (Open Source Computer Vision Library) merupakan sebuah *library open source* yang dikembangkan oleh Intel dengan tujuan untuk menyederhanakan dan mempercepat proses pemrograman dalam bidang pengolahan citra digital [9]. OpenCV dibuat hanya dikhususkan untuk pengenalan citra, dimana agar komputer dapat mengenali tentang pengenalan visual dari wajah manusia dengan lebih akurat dengan menggunakan lebih dari 2500 algoritma optimasi [10]. OpenCV sudah membantu banyak orang-orang tentang pengenalan dan pendeteksian wajah manusia dan pendeteksian objek.

OpenCV dikembangkan oleh Intel pada 2019 dengan tujuan menyediakan library pengenalan citra digital dan computer vision yang efisien. Dikembangkan juga untuk mempercepat aplikasi berbasis machine learning dan real-time vision agar lebih optimal. OpenCV dirilis pada tahun 2020, sampai sekarang OpenCV membantu sangat banyak orang dalam hal akademik maupun komersial.

## 3. METODE PENELITIAN

### 3.1. Pemilihan Arsitektur CNN dan Flowchart



Gambar 3. Flowchart alur penelitian

Pada penelitian ini akan menggunakan salah satu metode deep learning, yaitu Convolutional Neural Network (CNN) dan memakai salah satu arsitekturnya yaitu EfficientNetB0 adalah model dasar dari series EfficientNet yang mempunyai parameter yang kecil dan harga komputasi yang sedang. Walaupun dengan simplisitasnya, EfficientNetB0 tetap dapat diandalkan dan pilihan yang bagus untuk beberapa tugas yang berhubungan dengan pengenalan citra digital [11]. Setelah data sudah dilatih, maka akan disimpan dalam bentuk format .keras (tensorflow). Sebelum melatih data maupun menjalankan aplikasi, alangkah baiknya untuk mengunduh beberapa library yaitu gradio, tensorflow, opencv-contrib-python, numpy, scikit-learn, matplotlib, dan seaborn.

Pada gambar 3 merupakan flowchart penelitian ini, dimulai dari pengumpulan data untuk melatih data, preprocessing citra dengan mengekstrak frame citra dengan OpenCV, ekstrasi fitur dengan memakai CNN, klasifikasi dengan menemukan confusion matrix, pengujian aplikasi berbasis gradio. Jika ya maka akan muncul hasil yang diharapkan, jika tidak kembali ke pengujian aplikasi.

### 3.2. Pengumpulan Data

Dataset diambil dari DeepfakeDetection di situs HuggingFace [12] dengan jumlah 70.000 gambar real dan 70.000 gambar fake. Namun peneliti hanya menggunakan 10.000 gambar real dan 10.000 gambar deepfake untuk data training. Data yang sudah diambil akan dilatih dengan menggunakan EfficientNetB0 yang sudah dibahas sebelumnya. Data akan dibagi menjadi data latih dan data uji dengan rasio 4:1. Data ini dibuat dengan seimbang agar model tidak bias selama menjalani masa pelatihan.

### 3.3. Preprocessing Data

Setelah dataset citra telah dikumpulkan, akan diproses dengan menggunakan CNN. Dengan tahapan preprocessing, yaitu:

- Mengatur ukuran citra menjadi ke ukuran tetap (224x224 pixel)
- Menormalisasi pixel pada citra dari rentang 0-1 atau -1 sampai 1.
- Mendeteksi wajah dengan metode dari OpenCV yaitu Haar Cascade.

Tujuan dari tahap ini adalah agar model dipastikan memiliki format data yang seragam dan meningkatkan akurasi model tersebut agar pendeteksian deepfake lebih mudah.

### 3.4. Ekstrasi Fitur/Landmark

Menggunakan landmark yang sudah diambil dari OpenCV untuk mengekstraksi fitur wajah. Fitur atau landmark dari OpenCV ini dapat memperkuat model CNN untuk klasifikasi wajah yang akan mau dideteksi. Fitur ini juga dipilih karena tidak mudah terpengaruh oleh cahaya, fitur ini juga memakai ciri geometris yaitu lebar hidung, tebal alis, jarak antara alis dan

mata, lebar pangkal hidung, tinggi bibir atas dan tinggi bibir bawah [13].

### 3.5. Klasifikasi

Setelah fitur diperoleh, tahap ini dilakukan menggunakan arsitektur CNN. Model CNN terdiri dari beberapa lapisan konvolusi, pooling, dan fully connected yang mampu mempelajari pola visual kompleks dalam citra untuk membedakan wajah real dan palsu. Model ini dilatih menggunakan dataset yang sudah diproses, dan hasil pelatiahannya akan digunakan untuk mendeteksi citra baru yang belum pernah dilihat sebelumnya. Confusion Matrix akan dihasilkan beserta hasil dari accuracy, precision, recall, dan F-1 score.

### 3.6. Pengujian Aplikasi

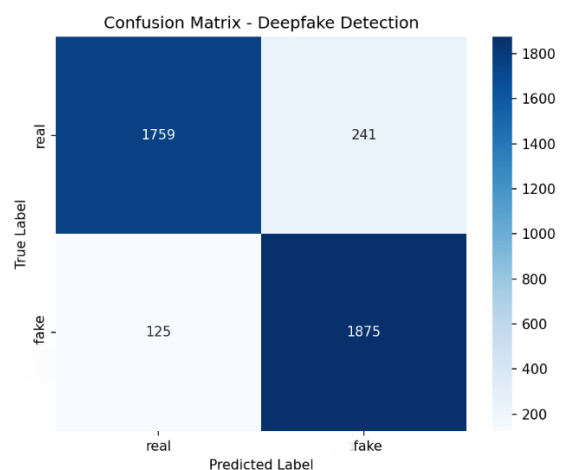
Model yang sudah dibuat disimpan ke format tensorflow keras dan dibuat suatu program dengan menggunakan antarmuka gradio. Gradio dipakai agar proses input gambar lebih cepat dan tidak perlu memakai kode program yang panjang lagi. Di aplikasi ini akan memakai confidence threshold yang artinya ambang nilai suatu objek yang akan dideteksi atau diklasifikasikan. Peneliti memakai confidence threshold di mana jika  $>0,6$  maka gambar akan diprediksi sebagai gambar yang sudah dimanipulasi oleh deepfake, sedangkan jika  $<0,6$  maka gambar akan diprediksi sebagai gambar real.

## 4. HASIL DAN PEMBAHASAN

Model dilatih dengan menggunakan salah satu arsitektur Convolutional Neural Network (CNN) yaitu EfficientNetB0 dan parameter yang diatur agar dapat menghindari overfitting dan juga data yang seimbang agar model tidak mengalami bias. Dataset terbagi dari data latih dan data uji dengan rasio 4:1.

### 4.1. Confusion Matrix

Setelah dataset telah dilatih dan diuji, maka berikutnya menentukan evaluasi pelatihan data dengan confusion matrix untuk mengetahui apa saja yang sudah dideteksi masuk ke kategori apa.



Gambar 4. Hasil confusion matrix

Perhatikan Gambar 4. di atas, jika dihitung keseluruhan maka data uji ada 2000 gambar real dan 2000 gambar deepfake. Ada nilai 1759 pada kategori True Positive (TP), artinya bahwa ada 1759 gambar real yang berhasil diidentifikasi oleh model. Untuk nilai 241 di False Positive (FP), maka artinya model mengira bahwa 241 gambar real tersebut keliru, setelah diidentifikasi malah menganggap citra tersebut telah dimanipulasi oleh deepfake. Lanjut untuk nilai 125 di False Negative (FN) artinya bahwa gambar yang sudah dimanipulasi deepfake dikira real oleh model. Terakhir untuk nilai 1875 menunjukkan bahwa ada 1875 di True Negative (TN) adalah citra yang sudah berhasil diidentifikasi oleh model bahwa citra tersebut benar-benar deepfake.

#### 4.2. Evaluasi Precision, Recall, dan F1-Score

Setelah ditemukannya confusion matrix, maka sistem selanjutnya menghitung evaluasi dari hasil pelatihan data sebelumnya. Berikut metrik evaluasi diambil dari sistem akan masuk ke dalam tabel berikut:

Tabel 1. Evaluasi Klasifikasi Deepfake

| Kelas            | Precision | Recall | F1-Score | Support |
|------------------|-----------|--------|----------|---------|
| Real             | 0,93      | 0,88   | 0,91     | 2000    |
| Fake             | 0,89      | 0,94   | 0,91     | 2000    |
| Accuracy         |           |        | 0,91     | 4000    |
| Macro Average    | 0,91      | 0,91   | 0,91     | 4000    |
| Weighted Average | 0,91      | 0,91   | 0,91     | 4000    |

Diketahui bahwa akurasi model setelah pelatihan data mencapai 0,91 di mana model sudah dilatih dengan baik dan memiliki bias yang sangat kecil atau tidak berpengaruh sama sekali.

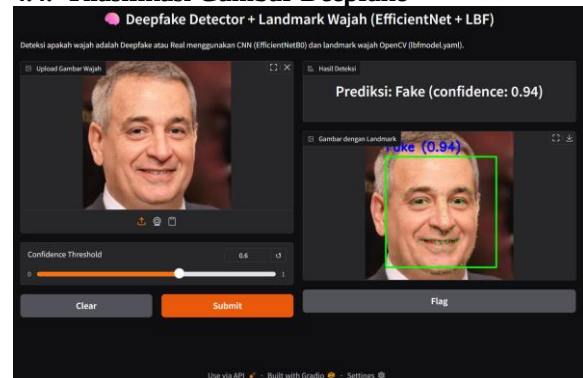
Untuk pembahasan spesifiknya, kelas *Real* menghasilkan nilai precision sebesar 0,93, recall sebesar 0,88, f1-score sebesar 0,91 dari total data uji sebanyak 2000. Untuk kelas *Fake* menghasilkan nilai precision sebesar 0,89, recall sebesar 0,94, dan f1-score sebesar 0,91 dari total data uji sebanyak 2000. Terlihat juga nilai dari macro average dan weighted average yang sama persis yang artinya bahwa model memiliki kondisi yang ideal dan memiliki distribusi data merata dan performa model yang bagus.

#### 4.3. Mendeteksi Deepfake dengan Antarmuka Gradio

Pengujian model yang sudah dilatih sebelumnya akan dimasukkan ke sebuah aplikasi dengan antarmuka gradio. Gradio dipakai agar dapat membuat aplikasi dengan mudah tanpa harus menambahkan kode tambahan untuk membuat aplikasi baru. Implementasi GUI dibuat semenarik mungkin agar lebih interaktif dan mudah dipahami. Pengujian akan dilakukan dengan memasukkan 2 gambar dimana masing-masing merupakan gambar real dan deepfake. Gambar juga diambil yang tidak pernah dilihat oleh sistem sebelumnya selama data training. Setelah

gambar diinput, maka akan dibantu oleh landmark OpenCV yang sudah disisip ke dalam sistem. Gambar juga diambil dari dataset yang sama, cuma yang tidak masuk ke dalam data training sebelumnya. Juga menggunakan confidence threshold dimana  $>0,6$  untuk deepfake dan  $<0,6$  untuk real.

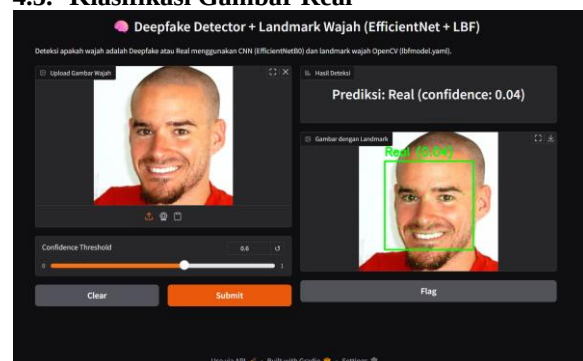
#### 4.4. Klasifikasi Gambar Deepfake



Gambar 5. Klasifikasi gambar deepfake

Hasil pada Gambar 5. menunjukkan bahwa sistem berhasil memprediksi citra yang dideteksi adalah deepfake dengan confidence threshold sebesar 0,94. Terlihat juga beberapa titik disekitar wajah yang artinya model OpenCV mendeteksi landmark wajah dan model EfficientNetB0 mendeteksi citranya.

#### 4.5. Klasifikasi Gambar Real



Gambar 6. Klasifikasi gambar real

Hasil pada Gambar 6. menunjukkan bahwa sistem berhasil mendeteksi bahwa gambar real tersebut benar-benar real dengan confidence threshold sebesar 0.04 (hampir mendekati nol), yang artinya model CNN tersebut hampir sangat akurat.

### 5. KESIMPULAN DAN SARAN

Berdasarkan penelitian yang sudah dibahas, pendeteksian deepfake sangat berguna untuk mengetahui apakah sebuah citra, termasuk gambar, masuk ke dalam kategori real atau deepfake. Apalagi jika gambar yang sudah dimanipulasi deepfake tidak dapat dilihat dengan visual dari manusia, jadi butuh metode yang dalam untuk mendeteksi sebuah citra. Setelah sistem sudah dimasukkan ke dalam aplikasi, maka model landmark OpenCV akan menandai



landmark-landmark yang penting untuk wajah pada citra, sedangkan model CNN akan memprediksi apakah gambar yang sudah dimasukkan merupakan deepfake atau real. Hasil dari confusion matrix dengan 1759 di TP, 241 di FP, 125 di FN. dan 1875 di TN. Setelah itu diketahui accuracy menunjukkan nilai dengan rate yang tinggi yaitu 0,91 atau 91%. Lanjut untuk pendeteksian deepfake di aplikasi antarmuka gradio dengan confidence threshold >0.6 untuk deepfake, di mana pada gambar deepfake terdeteksi dengan CT 0.94 dan pada gambar real dengan CT 0.04. Ini menunjukkan tingginya keakurasian model dan juga bantuan dari landmark wajah dari OpenCV. Disarankan untuk penelitian selanjutnya agar juga mendeteksi deepfake pada video dan suara agar pembelajaran tentang deteksi deepfake ini sangat bermanfaat bagi semua orang. Dan juga memakai arsitektur CNN lainnya seperti ResNet, VGGNet, DenseNet, dan lain-lain untuk menghasilkan output dan evaluasi yang berbeda.

#### DAFTAR PUSTAKA

- [1] Ratna, S., "Pengolahan Citra Digital dan Histogram dengan Python dan Text Editor Pycharm," *Technologia: Jurnal Ilmiah*, vol. 11, no. 3, pp. 181-186, 2020.
- [2] Sahar, N. M., M. F. S. M. Rozi, N. S. Suriani, S. Sari, S. Ismail, A. A. Jamal, & A. M. Alessa, "Advances in DeepFake Detection: Leveraging InceptionResNetV2 for Reliable Video Authentication," *Journal of Advanced Research in Applied Sciences and Engineering Technology*, vol. 52, no. 1, pp. 90-105, 2025.
- [3] Noerman, C. T., "Kriminalisasi Deepfake Di Indonesia Sebagai Bentuk Pelindungan Negara," *Jurnal USM Law Review*, vol. 7, no. 2, pp. 603-621, 2024.
- [4] Sistem Informasi IIBN, "Apa itu Deepfake dan Bagaimana Mengenali Penipuan Berbasis Deepfake," *Sistem Informasi IIBN – Mastering Digital Ecosystem*, February 2025. [Online]. Available: <https://si.indobarunasional.ac.id/apa-itu-deepfake-dan-bagaimana-mengenali-penipuan-berbasis-deepfake/>. [Accessed: Jun. 23, 2025].
- [5] Alfari, M. R. S., M. Z. Al-farish, M. Taufiqurrahman, G. Ardiansah, & M. Elgar, "Penggunaan Python Sebagai Bahasa Pemrograman untuk Machine Learning dan Deep Learning," *Karimah Tauhid*, vol. 2, no. 1, pp. 1-6, 2023.
- [6] Azmi, K., S. Defit, & Sumijan, "Implementasi Convolutional Neural Network (CNN) Untuk Klasifikasi Batik Tanah Liat Sumatera Barat," *Jurnal Unitek*, vol. 16, no. 1, pp. 28-40, 2023.
- [7] Pratiwi, B. P., A. S. Handayani, & Sarjana, "Pengukuran Kinerja Sistem Kualitas Udara Dengan Teknologi WSN Menggunakan Confusion Matrix," *Jurnal Informatika UPRIS*, vol. 6, no. 2, pp. 66-75, 2020.
- [8] Hendriyanto, M. D., A. A. Ridha, & U. Enri, "Analisis Sentimen Ulasan Aplikasi Mola pada Google Play Store Menggunakan Algoritma Support Vector Machine," *Journal of Information Technology and Computer Science*, vol. 5, no. 1, pp. 1-7, Juni 2022.
- [9] Susim, T. & C. Darujati, "Pengolahan Citra Untuk Pengenalan Wajah (Face Recognition) Menggunakan OpenCV," *Jurnal Syntax Admiration*, vol. 2, no. 3, pp. 534-545, Maret 2021.
- [10] Santoso, B. & R. P. Kristianto, "Implementasi Penggunaan OpenCV pada Face Recognition Untuk Sistem Presensi Perkuliahan Mahasiswa," *Sistemasi: Jurnal Sistem Informasi*, vol. 9, no. 2, pp. 352-361, Mei 2020.
- [11] Khairunnisa, P., W. E. Putra, W. Yitong, A. Jufrizal, & M. N. A. Makmum, "Convolutional Neural Networks Using EfficientNetB0 Architecture and Hyperparameters on Skin Disease Classification," *Public Research Journal of Engineering, Data Technology and Computer Science*, vol. 2, no. 2, pp. 127-137, Januari 2025.
- [12] [11] Y. Duhan, "DeepFakeDetection," Hugging Face, 2023. [Online]. Available: <https://huggingface.co/datasets/yashduhan/DeepFakeDetection>.
- [13] Sejati, R. R. H. P. & R. Mardhiyyah, "Deteksi Wajah Berbasis Facial Landmark Menggunakan OpenCV dan Dlib," *Jurnal Teknologi Informasi*, vol. 5, no. 2, pp. 144-148, Desember 2021.