

BAB V

SIMPULAN DAN SARAN

5.1 Simpulan

Aplikasi *Image Deepfake Detector* menggunakan algoritma *Neural Network* dengan model ResNet-50 memiliki akurasi cukup baik bila diterapkan pada *deepfake* yang menggunakan model serupa dalam pembuatannya dengan model *data training*. Aplikasi dapat mendeteksi *deepfake* dengan model ProGAN yang serupa dengan *data training* dengan ketepatan 100% dan terlihat mengalami penurunan ketepatan ketika digunakan dalam *deepfake* dengan model StyleGAN dan BigGAN. Secara keseluruhan, aplikasi mendapatkan akurasi sebesar 85%, presisi sebesar 100%, serta *recall* sebesar 65%.

5.2 Saran

- a) Perkembangan AI yang berlangsung cepat memungkinkan pengembangan *deepfake* yang mampu menipu mata manusia ataupun *deepfake detector* yang telah berkembang. Sebaliknya, perkembangan tersebut juga memungkinkan munculnya metode baru untuk mendeteksi *deepfake* tersebut.
- b) Terdapat banyak faktor yang layak dijadikan bahan penelitian selanjutnya mengenai *deepfake*, seperti faktor kompresi yang seringkali diterapkan pada media sosial. Bukan tidak mungkin karena faktor tersebut *deepfake* yang dikirim melalui media sosial akan dianggap sebagai gambar asli.
- c) Pengembangan model AI, terutama yang bersifat kompleks seperti gambar atau suara, terbilang cukup sulit bila dilakukan dari awal. Untuk mempermudah dan mempercepat proses, dapat digunakan model yang telah dilatih secara dasar sebelumnya seperti ResNet-50 yang disediakan Kaggle.

DAFTAR PUSTAKA

- Ategeka, Chris. 2021, *The Unintended Consequences of Technology: Solutions, Breakthroughs, and the Restart We Need*. New Jersey: Wiley.
- Baheti, Pragati 2023, *A Comprehensive Guide to Convolutional Neural Networks*, updated 3 Januari 2023, dilihat 12 Januari 2023, <<https://www.v7labs.com/blog/convolutional-neural-networks-guide>>.
- Bond, Allan 2021, *Supervised, Unsupervised, and Reinforcement Learning*, updated 27 Juli 2021, dilihat 7 September 2022, <<https://medium.com/mllearning-ai/supervised-unsupervised-and-reinforcement-learning-the-three-pillars-of-machine-learning-fd898f0c7c81>>.
- Butka, Peter, et. al. 2020, *Knowledge Discovery in Big Data from Astronomy and Earth Observation: Astroinformatics*. Berlin: Elsevier Science.
- Chandra, Akshay 2018, *Perceptron: The Artificial Neuron*, updated 12 Agustus 2018, dilihat 7 September 2022, <<https://towardsdatascience.com/perceptron-the-artificial-neuron-4d8c70d5cc8d>>.
- Chih-Chung Hsu, Yi-Xiu Zhuang, dan Chia-Yen Lee. 2020, 'Deep Fake Image Detection Based on Pairwise Learning', *Applied Sciences*, vol. 10 no 1, pp. 370-384.
- Cooper, Steven. 2018, *Machine Learning for Beginners: An Introduction for Beginners*. South Carolina: CreateSpace.
- Dixon, Brian E., dan Finnell, John T. 2022, *Clinical Informatics Study Guide: Text and Review*. New York: Springer International Publishing.
- Dreier, Thomas, and Andina, Tiziana. 2022, *Digital Ethics: The Issue of Images*. Massachusetts: Nomos Verlag.

- Hamsah, Anhar Rabi. 2022, *Kejahatan Berbahasa (Language Crime)*. Tasikmalaya: Langgam Pustaka.
- Heryadi, Yaya, dan Irwansyah, Edy. 2020, *Deep Learning: Aplikasinya di Bidang Geospasial*. Jakarta: AWI Technology Press.
- Holmes, Dawn. 2017, *Big Data : A Very Short Introduction*. Oxford: Oxford University Press.
- IBM. 2020, *AI vs Machine Learning vs Deep Learning vs Neural Networks*, updated 27 Mei 2020, dilihat 12 Januari 2023, <<https://www.ibm.com/cloud/blog/ai-vs-machine-learning-vs-deep-learning-vs-neural-networks>>
- Igbanufe, Obi, 2019. *Technology Made Simple for the Technical Recruiter, Second Edition*. Indiana: iUniverse.
- Irawan, M. Dedi. 2022. *Flowchart dan Pseudo-Code: Implementasi Notasi Algoritma dan Pemrograman*. Bandung: Media Sains Indonesia.
- Jadon, Shruti 2018, *Introduction to Different Activation Functions for Deep Learning*, updated 16 Maret 2018, dilihat 7 September 2022, <<https://medium.com/@shrutijadon/survey-on-activation-functions-for-deep-learning-9689331ba092>>.
- Jemadu, Liberty 2022, *Jumlah Pengguna Media Sosial Indonesia Capai 191,4 Juta per 2022*, updated 23 Februari 2022, dilihat 29 Agustus 2022, <<https://www.suara.com/tekno/2022/02/23/191809/jumlah-pengguna-media-sosial-indonesia-capai-1914-juta-per-2022>>.
- Lyu, Siwei. 2020, 'Deepfake Detection: Current Challenges And Next Steps', 2020, 'IEEE International Conference on Multimedia & Expo Workshops (ICMEW);', hh. 1-6.
- M. Almars, Abdulqader. 2021, 'Deepfakes Detection Techniques Using Deep Learning: A Survey', *Journal of Computer and Communications* vol. 9, hh. 20-35.

- Mukunthu, Deepak, Shah Parashar, dan Tok, Wee Hyong. *Practical Automated Machine Learning on Azure*. Sebastopol: O'Reilly Media.
- Mulaab. 2017. *Data Mining : Konsep dan Aplikasi*. Madura: Media Nusa Creative (MNC Publishing).
- Pane, Syafrial Fachri & Saputra, Yogi Aditya. 2020. *Big Data: Classification Behavior Menggunakan Python*. Bandung: Kreatif.
- Pertiwi, Wahyunanda Kusuma 2019, *Separuh Penduduk Indonesia Sudah "Melek" Media Sosial*, updated 4 Februari 2019, dilihat 29 Agustus 2022, <<https://tekno.kompas.com/read/2019/02/04/19140037/separuh-penduduk-indonesia-sudah-melek-media-sosial>>.
- Pertiwi, Wahyunanda Kusuma 2020, *Jumlah Hoaks di Indonesia Meningkat, Terbanyak Menyebar lewat Facebook*, updated 20 November 2020, dilihat 29 Agustus 2022, <<https://tekno.kompas.com/read/2020/11/20/07385057/jumlah-hoaks-di-indonesia-meningkat-terbanyak-menyebar-lewat-facebook>>.
- Prabhu 2018, *Understanding of Convolutional Neural Network (CNN) — Deep Learning*, updated 4 Maret 2018, dilihat 7 September 2022, <<https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148>>.
- Pulung Adi Nugroho, Indah Fenriana, dan Rudy Arijanto. 2020, *Implementasi Deep Learning Menggunakan Convolutional Neural Network (CNN) pada Ekspresi Manusia*, ALGOR vol. 2 no. 1, hh. 12-21.
- Reddy, V. Venkata, et. al., 2022. *'Fake Image Detection Using Machine Learning'*, International Journal of Advanced Research in Computer and Communication Engineering', vol. 11 no 1, pp. 138-144.

- Riyanarto, et. al., 2022, *Machine Learning dan Deep Learning-Konsep dan Pemrograman Python*. Yogyakarta: Penerbit Andi.
- Sarwindaa, Devvi, et. al., 'Deep Learning in Image Classification using Residual Network (ResNet) Variants for Detection of Colorectal Cancer', *Procedia* vol. 179, pp 423-431.
- Sholihin, Mahfud, dan Anggraini, Puspita Ghaniy. 2021, *Analisis Data Penelitian Menggunakan Software STATA*. Yogyakarta: Penerbit Andi.
- Sinarmata, Janner, et. al. 2020, *Teknologi Informasi dan Sistem Informasi Manajemen*. Medan: Yayasan Kita Menulis.
- Singh, Pramod & Manure, Avinash. 2019. *Learn TensorFlow 2.0*. Bangalore: Apress.
- Siregar, Amril Mutoi, dan Puspabhuana, Adam. 2017, *Data Mining: Pengolahan Data Menjadi Informasi dengan RapidMiner*. Surakarta: CV Kekata Group.
- Stone, James. 2019, *Artificial Intelligence Engines: A Tutorial Introduction to the Mathematics of Deep Learning*. Sheffield: Sebtel Press.
- Umer, Muhammad, et. al., 2020, 'Fake News Stance Detection Using Deep Learning Architecture (CNN-LSTM)', *IEEE Access* vol. 8, pp. 156695-156706.
- Widharma, I Gede Suputra, dan Wiranata, Lalu Febrian. 2022. *Mikrokontroler dan Aplikasi*. Banyumas: Wawasan Ilmu.
- Wundenberg, Sven-Michael. 2015. *Requirement Engineering for Knowledge-Intensive Processes*. Laupheim: Springer Fachmedien Wiesbaden.



UNIVERSITAS BUDDHI DHARMA

Jl. Imam Bonjol No. 41 Karawaci Ilir, Tangerang

021 5517853 / 021 558622 admin@buddhidharma.ac.id

KARTU BIMBINGAN TA/SKRIPSI

NIM : 20191000079
Nama Mahasiswa : ANANDA ADHICITTA WANGSADIDJAJA
Fakultas : Sains dan Teknologi
Program Studi : Teknik Informatika
Jenjang : Strata Satu
Tahun Akademik/Semester : 2022/2023 Genap
Dosen Pembimbing : Hartana Wijaya, M.Kom
Judul Skripsi : Penerapan Deep Learning Untuk Image Deepfake Detector Menggunakan Algoritma Neural Network

Tanggal	Catatan	Paraf
2023-03-13	Pengajuan Judul dan Bab I	
2023-04-10	Acc Judul dan Revisi Bab I	
2023-04-27	Acc Bab I dan Bab II	
2023-05-25	Revisi Bab III dan Bab IV	
2023-05-31	Acc Bab III dan Revisi Bab IV	
2023-06-09	Revisi Bab IV	
2023-06-15	Acc Bab IV dan Review Aplikasi	
2023-06-22	Review dan Acc Aplikasi	
2023-07-03	Acc Bab V, Maju Sidang	

Mengetahui
Ketua Program Studi

Hartana Wijaya, M.Kom

Tangerang, 8 August 2023
Pembimbing

Hartana Wijaya, M.Kom

Daftar Riwayat Hidup



Nama : Ananda Adhicitta Wangsadidjaja
Tempat, Tanggal Lahir : Tangerang, 8 Mei 2001
Jenis Kelamin : Laki-laki
Agama : Buddha
Alamat : Jl. Padjajaran Raya No. 65, Perumnas 3, Tangerang, Banten
No. Telp / HP : 081399453097
Email : nanda143735@gmail.com

PENDIDIKAN FORMAL

SD (2007-2013) : SD Bonavita
SMP (2013-2016) : SMP Bonavita
SMA (2016-2019) : SMA Perguruan Buddhi
Strata 1(2019- Sekarang) : Universitas Buddhi Dharma

Tangerang, 8 Agustus 2023

Ananda Adhicitta Wangsadidjaja

LAMPIRAN

Koding Aplikasi

Frontend

app.js

```
import React, { useState } from "react";
import { BrowserRouter, Routes, Route, Navigate } from 'react-router-dom';
import './App.css';
import Home from "./pages/home";
import About from "./pages/about";
import Detect from './pages/detect';
import Help from './pages/help';
import Hasil from './pages/hasil';
function App() {
  const [image, setImage] = useState(null);
  const [prediction, setPrediction] = useState(null);
  return (
    <BrowserRouter>
      <Routes>
        <Route path="/" element={ <Home /> } />
        <Route path="/about" element={ <About /> } />
        <Route path="/help" element={ <Help /> } />
        <Route path="/detect" element={ <Detect setImage={setImage} setPrediction={setPrediction} /> } />
        <Route path="/hasil" element={ <Hasil image={image} prediction={prediction} /> } />
        <Route path="*" element={ <Navigate to="/" /> } />
      </Routes>
    </BrowserRouter>
  );
}
export default App;
```

home.js

```
import React from 'react';
import logo from './th.png';
import profile from './profile.jpeg';
import Header from './partials/header';
import Footer from './partials/footer';
```



```

const center = {
  display: 'flex',
  alignItems: 'center',
  justifyContent: 'center'
};
function Home() {
  return (
    <div>
      <Header />
      <div id="title" className="row">
        <div className="col-12 col-md-8">
          <h1 className="heading">Uncover the truth</h1>
          <h3>with our AI based deepfake detector technology</h3>
          <a href="/detect">Get Started &gt;</a>
        </div>
        <div className="col-12 col-md-4" style={{ minHeight: 75, marginTop: 20 }}>
          <img className="title-image" src={logo} alt="title"></img>
        </div>
      </div>
      <div className="row">
        <div className="col-12 col-md-6">
          </img>
        </div>
        <div className="col-12 col-md-6">
          <h1 className="section-heading">About Deepfake Detector</h1>
          <h3 style={{ fontSize: "1.2rem" }}>Our website provides a powerful and reliable solution to the growing problem of deepfakes, images that have been manipulated or generated using artificial intelligence to show things that never actually happened. With our detection technology, you can easily verify the authenticity of your images and protect yourself from falling victim to misinformation or deception.</h3>
          <a href="/about" className="btn btn-primary" role="button" aria-pressed="true">Learn More &nbsp;<i>&gt;</i></a>
        </div>
      </div>
      <div className="row colored-section" >
        <div className="col-12">
          <h1 className="section-heading">Experimental Results</h1>
          <h3>We have performed extensive training to provide the most reliable solution. The following are our method settings</h3>

```

```

    <li className="section-paragraph">
      <a>Learning Rate: 0.0001</a>
    </li>
    <li className="section-paragraph">
      <a>Accuracy: 85%</a>
    </li>
    <li className="section-paragraph">
      <a>Precision: 100%</a>
    </li>
    <li className="section-paragraph">
      <a>Recall: 62.5%</a>
    </li>
  </div>
</div>
<div id="author">
  <div className="row" >
    <div className="col-12">
      <h1 className="section-heading" style={center}>About Author</h1>
      <h3 style={{ fontSize: "1.2rem", textAlign: "center" }}>This website managed and developed
by Ananda Adhicitta from Universitas Buddhi Dharma for his final year project.</h3>
    </div>
    <div className="col-12" style={center}>
      <img className="author-image" src={profile} alt="about"></img>
    </div>
  </div>
</div>
<Footer />
</div>
);
}
export default Home;

```

about.js

```

import React from 'react';
import image1 from './image1.png';
import image2 from './image2.png';
import image3 from './image3.png';
import curve from './ROC-AUC-CURVE.png';
import Header from './partials/header';
import Footer from './partials/footer';

```

```

const center = {
  display: 'flex',
  alignItems: 'center',
  justifyContent: 'center'
};
function About() {
  return (
    <div>
      <Header />
      <div id="title" className="row">
        <div className="col-12 col-md-10">
          <h1 className="heading">Deepfake</h1>
          <h3>Generative Adversarial Networks (GANs) are a form of deep neural network that has been
commonly used to generate deepfake.</h3>
        </div>
      </div>
      <div className="row">
        <div className="col-12 col-lg-8 text-right">
          <h1 className="section-heading">What's deepfake?</h1>
          <h3 style={{ fontSize: "1.2rem" }}>Deepfake is the result of the development of deep learning
technology, a branch of machine learning that uses neural network algorithms on large-scale data sets to
create false information. This fake information can be in the form of sound, image, or video that will look
very similar to the original, because it uses technology that continues to develop.</h3>
          <h3 style={{ fontSize: "1.2rem" }}>While deepfake open possibilities to generate stock images
easily, <b>deepfake often used to swipe faces of popular person and produce misleading information and
rumors.</b> In 2018, a fake video for Barack Obama was created to putting words he never uttered. In
addition, in the US 2020 election, deepfakes have been used to manipulate Joe Biden videos showing his
tongue out.</h3>
        </div>
        <div className="col-12 col-lg-4 d-flex align-items-center">
          <img className="about-image" src={image2} alt="about"></img>
        </div>
      </div>
      <div className="row colored-section">
        <div className="col-12 col-md-6 d-flex align-items-center">
          <img className="about-image" src={image1} alt="about"></img>
        </div>
        <div className="col-12 col-md-6">
          <h1 className="section-heading">How deepfakes generated?</h1>

```

<h3 style={{ fontSize: "1.2rem" }}>The architecture of GANs consists of two neural networks components: </h3>

<li className="section-paragraph" style={{ fontSize: "1.2rem" }}>

<a>The generator learns to generate plausible data. The generated instances become negative training examples for the discriminator.

<li className="section-paragraph" style={{ fontSize: "1.2rem" }}>

<a>The discriminator learns to distinguish the generator's fake data from real data. The discriminator penalizes the generator for producing implausible results.

</div>

</div>

<div className="row" >

<div className="col-12">

<h1 className="section-heading">How to prevent it?</h1>

</br>

<h3 style={{ fontSize: "1.2rem" }}>We have performed extensive training to provide the powerful and reliable solution to the growing problem of deepfakes. Using the power of ResNet50, neural network algorithm specialized in image processing, we build this deepfake detection website.</h3> <h3 style={{ fontSize: "1.2rem" }}></h3>

</div>

<div className="col-12" style={center}>

</div>

</div>

<div className="colored-section">

<div className="row" >

<div className="col-12">

<h1 className="section-heading" style={center}>How reliable is it?</h1>

<h3>Althought still need improvement, it offer quite good result toward our validation data set.</h3>

<li className="section-paragraph">

<a>Learning Rate: 0.0001

<li className="section-paragraph">

<a>Accuracy: 85%

<li className="section-paragraph">

<a>Precision: 100%


```

    <li className="section-paragraph">
      <a>Recall: 62.5%</a>
    </li>
  </div>
  <div className="col-12" style={center}>
    <img className="curve-image" src={curve} alt="about"></img>
  </div>
</div>
</div>
<Footer />
</div>
);
}
export default About;

```

help.js

```

import React, { useState } from "react";
import { ToastContainer, toast } from 'react-toastify';
import 'react-toastify/dist/ReactToastify.css';
import axios from './config/api';
import Header from './partials/header';
import Footer from './partials/footer';
function Help() {
  const [message, setMessage] = useState();
  const handleChange = (e) => {
    setMessage(e.target.value);
  }
  const handleClick = async (e) => {
    if ( message === "" ){
      toast("Message can't be empty!");
    }
    else{
      try {
        const response = await axios.post(
          "/message",
          JSON.stringify({ "message" : message }),
          {
            headers: { "Content-Type": "application/json" },
            withCredentials: false,
          }
        )
      }
    }
  }
}

```



```

);
console.log(response)
if(Response.status = 200){
  toast("Thank you for your message!");
}
else{
  toast("Sorry, we failed to send your message, please try again later!");
}
} catch (error) {
  console.log(error);
  toast("Oops! Something went wrong, please try again later");
}
}
}
return (
<div>
  <Header />
  <div className="row">
    <div className="col-12">
      <h1 className="section-heading">Help us to improve this website</h1>
      <h3 style={{ fontSize: "1.2rem" }}>If you have suggestion or want to ask about this website, fill
this box and send it to use. Any message would be meaningful to us, thank you!</h3>
      <br></br>
    </div>
    <div className="col-12">
      <textarea
        value={message}
        onChange = { handleChange }
        style={{ width: '100%', height: '250px', resize: 'vertical', padding: '10px' }}
      />
      <a className="btn btn-primary" role="button" aria-pressed="true" onClick={ handleClick }
style={{ marginTop: '30px', color: "white" }}>Send</a>
    </div>
    <ToastContainer />
  </div>
  <Footer />
</div>
)
}
export default Help;

```

detect.js

```
import React, { useState } from "react";
import { useNavigate } from 'react-router-dom';
import { ToastContainer, toast } from 'react-toastify';
import 'react-toastify/dist/ReactToastify.css';
import loadingImage from './loading.gif';
import axios from './config/api';
import Header from './partials/header';
import Footer from './partials/footer';
function Detect(props) {
  const [isLoading, setLoading] = useState(false);
  const navigate = useNavigate();
  var imageCheck = Boolean(true);

  const handleChange = async (e) => {
    var types;
    for (var i = 0; i < e.target.files.length; i++) {
      if (e.target.files[i].name.match(/^(.jpg|jpeg|png)$/)) {
        types = "image";
      }
      else{
        toast("File not supported!");
      }
    }
  }

  if (imageCheck == true) {
    try {
      const data = new FormData()
      data.append('files', e.target.files[0])

      setLoading(true)
      const response = await axios.post(
        "/detect", data,
        {
          headers: { "Content-Type": "multipart/form-data" },
          withCredentials: false,
        }
      );
      props.setImage(URL.createObjectURL(e.target.files[0]));
      props.setPrediction(response.data.prediction);
    }
  }
}
```

```

    navigate("/hasil");
  } catch (error) {
    console.log(error);
    toast("Oops! Something went wrong, please try again later");
    setLoading(false)
  }
}
}
return (
  <div className="App">
    <Header />
    <div className="detect-title">
      <h1 className="section-heading">A little help</h1>
      <h3 style={{ fontSize: "1.2rem" }}>Choose your images you need to know its origin. More longer
      videos might take some time to be detected.</h3>
      <h3 style={{ fontSize: "1.2rem" }}>Our website accept various image extension as jpg, jpeg, and
      png!</h3>
    </div>
    <div className="upload-container">
      <input type="file" onChange = { handleChange } />
    </div>
    <div style={{ textAlign: 'center', position: 'relative', marginTop: 0 }} >
      { isLoading ? <img className="loading" src = { loadingImage } /> : null }
    </div>
    <br></br>
    <Footer />
    <ToastContainer />
  </div>
)
}
export default Detect;

```

hasil.js

```

import React from 'react';
import Header from './partials/header';
import Footer from './partials/footer';
function Hasil(props) {
  return (
    <div>
      <Header />

```

```

<div className="row" >
  <div className="hasil col-12 col-md-4">
    <h1 className="text-hasil">Our algorithm predict that your image is most likely to be
{props.prediction}</h1>
    <br></br>
    <a href="/detect" className="btn btn-primary" role="button" aria-pressed="true">Try Another
Image</a>
  </div>
  <div className="col-12 col-md-8">
    <img className="output" src={props.image} alt="Gambar Hasil" />
  </div>
</div>
<Footer />
</div>
)
}
export default Hasil;

```

header.js

```

import React from 'react';
import { Navbar, Nav } from 'react-bootstrap';
function Header() {
  return (
    <div>
      <Navbar variant="dark" expand="lg" sticky="top">
        <Navbar.Brand href="/">Deepfake Detector</Navbar.Brand>
        <Navbar.Toggle aria-controls="basic-navbar-nav">
        </Navbar.Toggle>
        <Navbar.Collapse id="basic-navbar-nav">
          <Nav.Link href="/home">Home</Nav.Link>
          <Nav.Link href="/about">About</Nav.Link>
          <Nav.Link href="/help">Help</Nav.Link>
        </Navbar.Collapse>
      </Navbar>
    </div>
  );
}
export default Header;

```

footer.js

```

import React from 'react';
function Footer() {
  return (
    <div>
      <div style={{ textAlign: "center" }}>
        <a>© Ananda Adhicitta, Universitas Buddhi Dharma 2023</a>
      </div>
    </div>
  );
}
export default Footer;

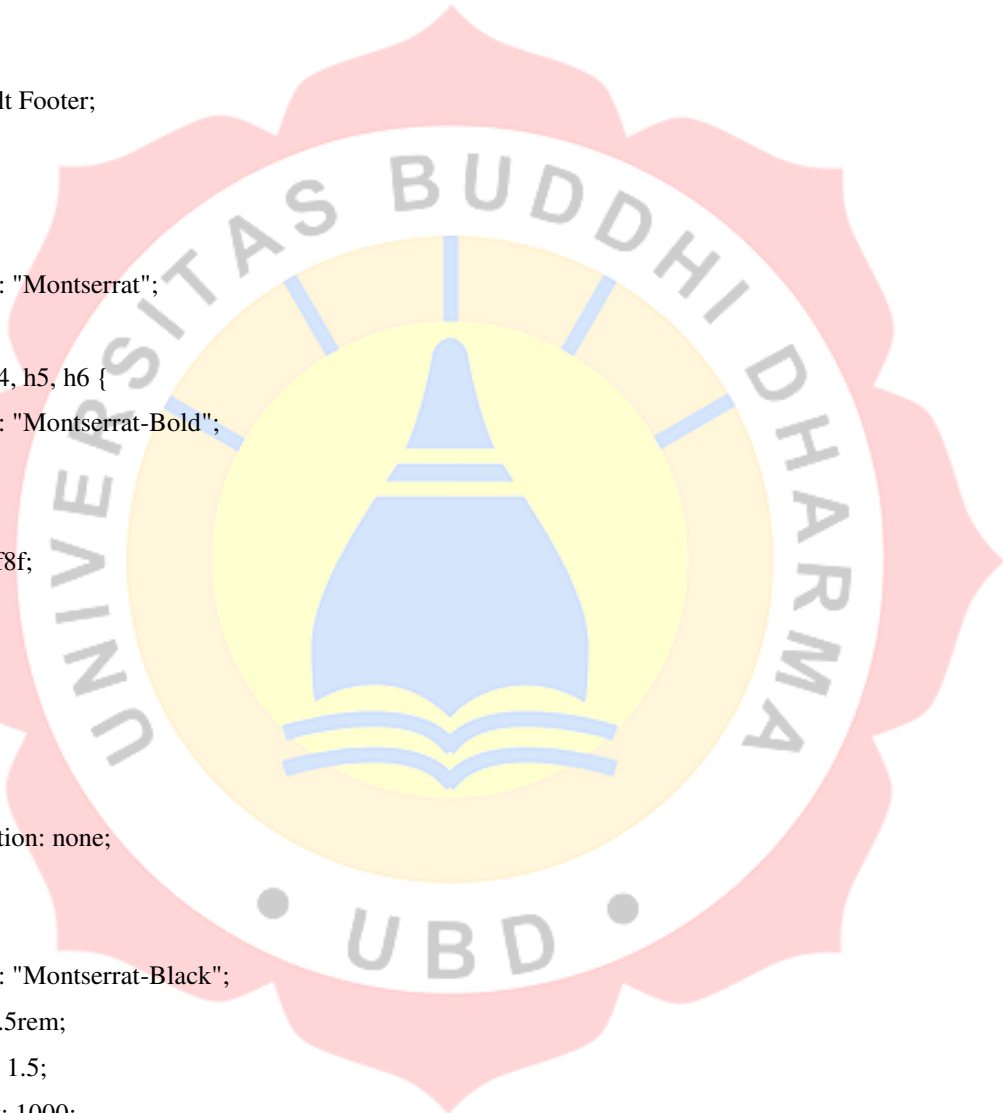
```

app.css

```

body {
  font-family: "Montserrat";
}
h1, h2, h3, h4, h5, h6 {
  font-family: "Montserrat-Bold";
}
p {
  color: #8f8f8f;
}
a {
  color: #fff;
}
a:hover {
  text-decoration: none;
}
.heading {
  font-family: "Montserrat-Black";
  font-size: 2.5rem;
  line-height: 1.5;
  font-weight: 1000;
}
.section-heading {
  font-size: 1.7rem;
  font-weight: bold;
  line-height: 1.5;
}
.section-paragraph {

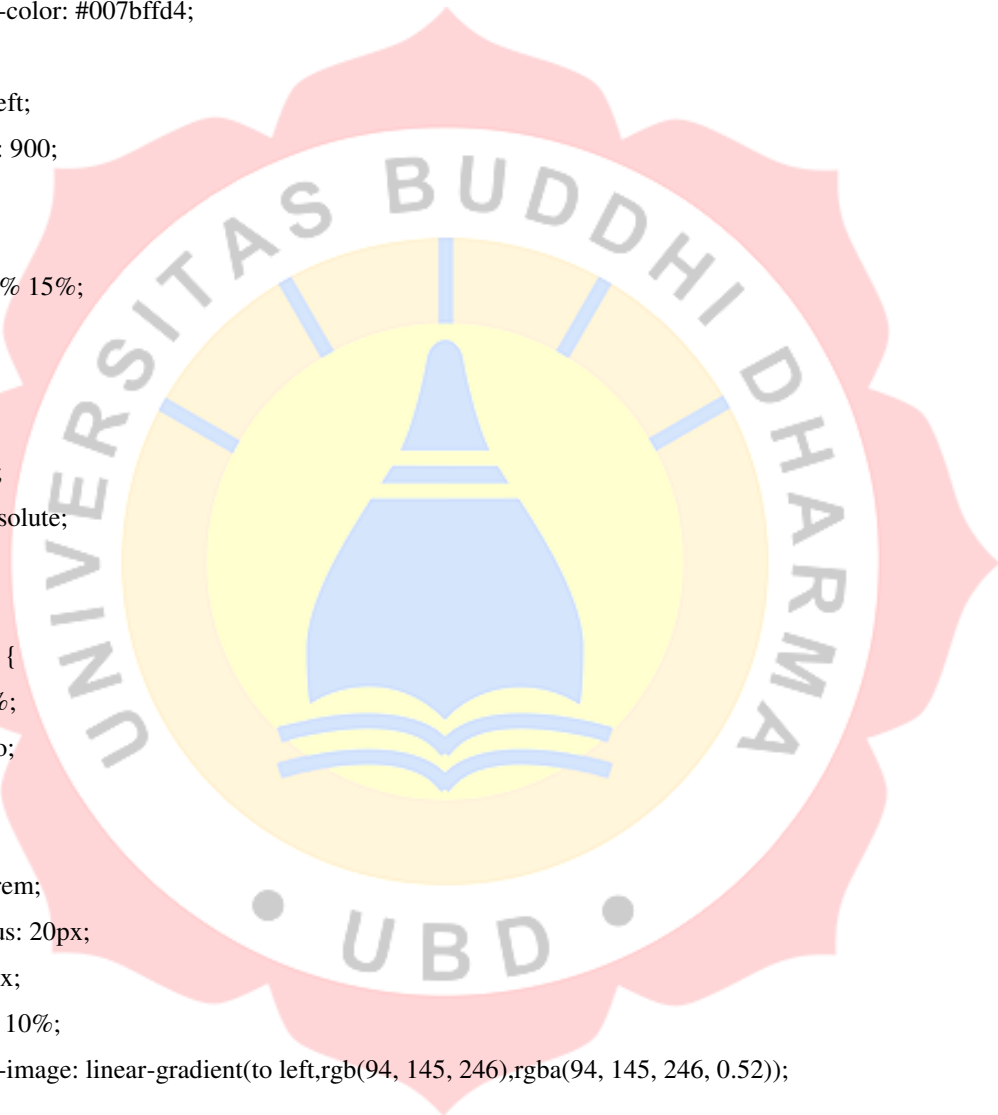
```



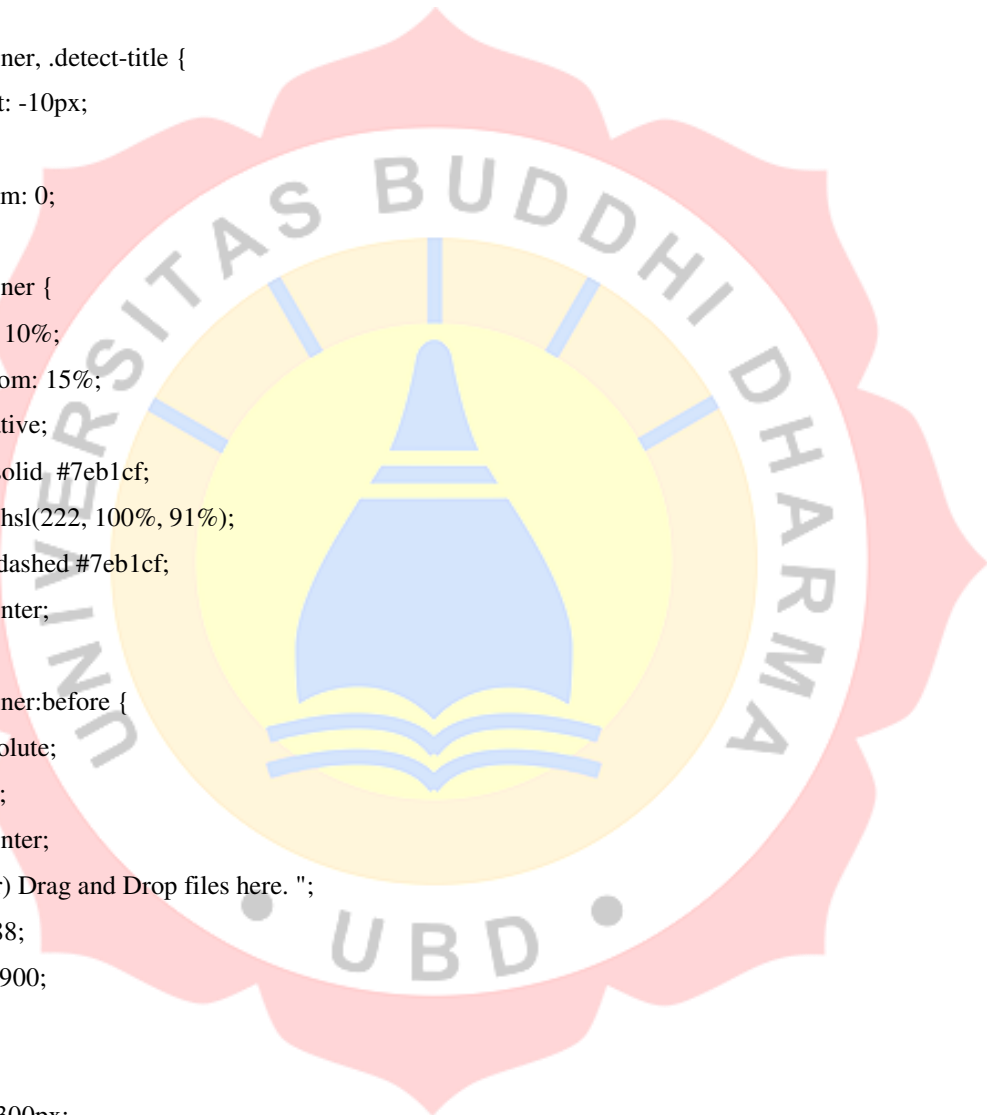

```

font-size: 1.4rem;
line-height: 1.5;
}
.colored-section, .navbar {
background-color: #4186ed;
color: #fff;
}
#title {
background-color: #007bfd4;
color: #fff;
text-align: left;
font-weight: 900;
}
.row {
padding: 10% 15%;
margin: 0;
}
.title-image {
width: 60%;
position: absolute;
left: 20%;
}
.about-image {
width: 100%;
margin: auto;
}
.btn {
font-size: 1rem;
border-radius: 20px;
width: 150px;
margin-top: 10%;
background-image: linear-gradient(to left,rgb(94, 145, 246),rgba(94, 145, 246, 0.52));
}
.btn:hover{
background-image: linear-gradient(to left, #007bff, #007bff);
box-shadow: 0 2px 2px -2px gray;
}
.author-image {
width: 25%;
margin: 10px;

```



```
border-radius: 15%;
overflow: hidden;
}
.curve-image {
width: 60%;
margin: 15px;
border-radius: 5%;
overflow: hidden;
}
.upload-container, .detect-title {
outline-offset: -10px;
margin: 5%;
margin-bottom: 0;
}
.upload-container {
padding-top: 10%;
padding-bottom: 15%;
position: relative;
border: 1px solid #7eb1cf;
background: hsl(222, 100%, 91%);
outline: 2px dashed #7eb1cf;
text-align: center;
}
.upload-container:before {
position: absolute;
bottom: 30%;
text-align: center;
content: " (or) Drag and Drop files here. ";
color: #3f4e88;
font-weight: 900;
}
.upload-btn {
margin-left: 300px;
padding: 7px 20px;
}
.loading {
max-height: 15%;
max-width: 15%;
margin-bottom: 5%;
}
```



```

.text-hasil {
  margin-bottom: 20%;
}
.hasil {
  padding: 5%;
}
.output {
  margin: 5%;
  width: 90%;
}
@media (max-width: 1000px) {
  .title-image {
    display: none;
  }
  .text-hasil {
    margin-bottom: 5%;
  }
  .loading {
    max-height: 25%;
    max-width: 25%;
  }
}

```

Backend

skripsi.py

```

from flask import Flask, jsonify, request
from werkzeug.utils import secure_filename
from flask_cors import CORS
from PIL import Image
from pymongo import MongoClient
from resnet import detect
import os
import datetime
import openpyxl
import gridfs
import io
app = Flask(__name__)
CORS(app, resources={r"/*": {"origins": "http://localhost:3000"}})

def save_data(image_path, label):

```

```

connection_string =
    "mongodb+srv://Nanda:Tester123123*@cluster0.xcajf.mongodb.net/deepfakedetectorDB?retryWrites=true&w=majority"
client = MongoClient(connection_string)
db = client['deepfakedetectorDB']
fs = gridfs.GridFS(db, collection='images')
with open(image_path, 'rb') as img_file:
    image_id = fs.put(img_file, filename=image_path)
data = {
    'image_id': image_id,
    'label': label,
    'date': datetime.datetime.now().strftime('%Y-%m-%d %H:%M:%S')
}
db.data_collection.insert_one(data)
client.close()

@app.route('/detect', methods=['POST'])
def detect_deep_fake():
    model = 'model.pth'
    file = request.files['files']
    filename = secure_filename(file.filename)
    file.save(filename)
    net = detect('cpu', model)
    img = Image.open(file).convert('RGB')
    img.load()
    result = net.apply(img)
    final_result = result.tolist()
    if (final_result > 0):
        result = {'prediction': 'fake'}
    else:
        result = {'prediction': 'real'}
    save_data(filename, result)
    os.remove(filename)
    return jsonify(result)

@app.route('/message', methods=['POST'])
def receive_message():
    json_data = request.json
    time = datetime.datetime.now().strftime('%Y-%m-%d %H:%M:%S')
    content = json_data['message']

```

```

try:
    workbook = openpyxl.load_workbook('message.xlsx')
except FileNotFoundError:
    workbook = openpyxl.Workbook()
sheet = workbook.active
next_row = sheet.max_row + 1
sheet.cell(row=next_row, column=1).value = time
sheet.cell(row=next_row, column=2).value = content
workbook.save('message.xlsx')
return 'Message received and saved'

if (__name__ == '__main__'):
    app.run()

resnet.py
import torch
import torch.nn as nn
import torchvision.transforms as transforms
import numpy as np
LIMIT_SIZE = 1536
LIMIT_SLIDE = 1024
class ChannelLinear(nn.Linear):
    def __init__(self, in_features: int, out_features: int, bias: bool = True) -> None:
        super(ChannelLinear, self).__init__(in_features, out_features, bias)
    def forward(self, x):
        out_shape = [x.shape[0], x.shape[2], x.shape[3], self.out_features]
        x = x.permute(0,2,3,1).reshape(-1,self.in_features)
        x = x.matmul(self.weight.t())
        if self.bias is not None:
            x = x + self.bias[None,:]
        x = x.view(out_shape).permute(0,3,1,2)
        return x
def conv3x3(in_planes, out_planes, stride=1):
    return nn.Conv2d(in_planes, out_planes, kernel_size=3, stride=stride, padding=1, bias=False)
def conv1x1(in_planes, out_planes, stride=1):
    return nn.Conv2d(in_planes, out_planes, kernel_size=1, stride=stride, bias=False)
class Bottleneck(nn.Module):
    expansion = 4
    def __init__(self, inplanes, planes, stride=1, downsample=None):
        super(Bottleneck, self).__init__()

```



```

self.conv1 = conv1x1(inplanes, planes)
self.bn1 = nn.BatchNorm2d(planes)
self.conv2 = conv3x3(planes, planes, stride)
self.bn2 = nn.BatchNorm2d(planes)
self.conv3 = conv1x1(planes, planes * self.expansion)
self.bn3 = nn.BatchNorm2d(planes * self.expansion)
self.relu = nn.ReLU(inplace=True)
self.downsample = downsample
self.stride = stride
def forward(self, x):
    identity = x
    out = self.conv1(x)
    out = self.bn1(out)
    out = self.relu(out)
    out = self.conv2(out)
    out = self.bn2(out)
    out = self.relu(out)
    out = self.conv3(out)
    out = self.bn3(out)
    if self.downsample is not None:
        identity = self.downsample(x)
    out += identity
    out = self.relu(out)
    return out
class ResNet(nn.Module):
def __init__(self, block, layers, num_classes=1, stride0=1):
    super(ResNet, self).__init__()
    self.inplanes = 64
    self.conv1 = nn.Conv2d(3, 64, kernel_size=7, stride=stride0, padding=3, bias=False)
    self.bn1 = nn.BatchNorm2d(64)
    self.relu = nn.ReLU(inplace=True)

```