Smart Internz

TEAM MEMBERS

1.	**DHARSHANALA NAVEEN**	**20MID0162**
2.	**THANIKONDA HARATHI**	**20MID0021**
3.	**NAMANI HARSHITH**	**20MID0157**
4.	**IPILLI KARTHIK KUMAR**	**20MID0148**

Age and Gender Detection Using Deep Learning

1. INTRODUCTION

1.1 OVERVIEW

Because of its wide range of applications in a variety of facial investigations, automatic age and gender prediction from face photos has recently gained a lot of attention. We can leverage the aforementioned technologies to determine a person's age and gender just on a single glimpse from a camera, image, or video. This project will outline convolutional neural network (CNN) using deep learning, methodologies, and algorithms that can be used, and how everything fits together for gender classification and age detection. Technology will also underline its importance and how it may be used to better our everyday lives. The paper's prime objective of use deep learning to develop a gender and age detector that can roughly predict the gender and age of a human face in an image. Further, the map shows how this technology might be applied to our benefit and look at the broad array of applications where it could be used: from intelligence agencies, CCTV cameras, and policing to matrimony sites.

1.2 PURPOSE

Gender and age play a significant role in interpersonal interactions among people who live in communities. The use of smart gadgets has expanded as technology has progressed, and social media has begun to draw everyone's attention. Daily studies on gender and age prediction have grown in prominence, it increases the number of apps that use such techniques. In these applications, facial photographs are commonly employed since they contain useful information that may be used to extract human interaction. For gender detection and age prediction, Image processing, feature extraction, and classification steps are usually used. These steps may change based on the objective of the study and the characteristics to be used. The face images were processed using a variety of approaches, and calculations were performed based on the results of the investigations. For image processing, there are two basic and typical which we need to follow. Image enhancement is the process of improving an image so that the resultant image is of higher quality and can be used by other applications. The most popular technique for extracting information from an image is the other technique. The image is divided into a specified number of parts or objects in order to solve the challenge and this procedure is called Segmentation. Due to the accuracy of its classification technique, deep learning techniques are a variety of tasks such as classification, feature extraction, object recognition, and so on, it helps in gender and age prediction.

Smart Internz

2. LITERATUTE SURVEY

2.1 EXISTING PROBLEM

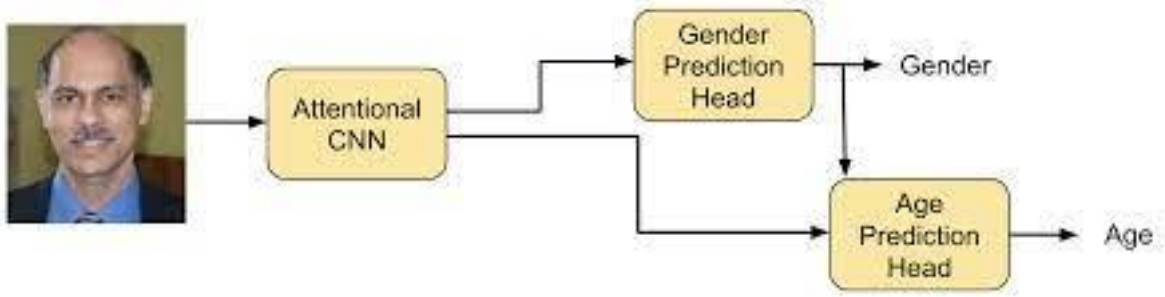
- ✓ Ecascaded Adaboost learning algorithm in face detection and achieved the age estimation mechanism using Gabor wavelets and OLPP.
- ✓ This paper is organized in the following sections. First, our presented face detection system includes histogram lighting normalization, feature selection, the cascaded Adaboost classifier and the region-based clustering algorithm.
- ✓ The age estimation process, including the feature extraction using Gabor wavelets, feature reduction and selection, and age classification, is then introduced.
- ✓ Finally, the experimental results and conclusions are provided and summarized

2.2 PROPOSED SOLUTION

- ✓ Age and Gender Detection, Deep EXpectation (DEX) – is used for age estimation which can be seen in image classification [5, 32, 47] and object detection fuelled by deep learning. From the deep learning concept we learn four key ideas that we apply to our solution:
- ✓ The deeper the open cv(by sheer increase of parameters / model complexity) the better is the capacity to model highly non-linear transformations - with some optimal depth on current architectures;
- ✓ The larger and more diverse the datasets used for training, the better the network learns to generalize and the more effective it becomes to over-fitting;
- ✓ The alignment of the object in the input image impacts the overall performance;
- ✓ When the training data is small that is when we must finetune a network pre-trained for comparable inputs and goals which would benefit us from the transferred knowledge

3. THEORITICAL ANALYSIS

3.1 BLOCK DIAGRAM



3.2 HARDWARE/ SOFTWARE DESIGNING

I. SOFTWARE PREREQUISITES

Keras (with TensorFlow backend)

OpenCV

Python 3.5 (TensorFlow not supported in higher versions)

NumPy

TensorFlow

II. HARDWARE PREREQUISITES

Intel core processor with high GPU power & frequency

4. EXPERIMENTAL INVESTIGATIONS

Age and gender detection using deep learning involves several key analyses and investigations throughout the solution development process. The initial dataset analysis examines the distribution of age and gender labels, class imbalances, and the quality and diversity of the data. This analysis helps understand the challenges and biases present in the dataset, which can impact the model's performance.

The selection of the model architecture is crucial, and its strengths and weaknesses are investigated. Factors such as complexity, parameter count, computational requirements, and performance on benchmark datasets are analyzed to choose the most suitable architecture.

Preprocessing techniques, including image resizing, normalization, data augmentation, and face alignment, are explored and analyzed to enhance the training data's quality and reliability. The impact of different preprocessing steps on the model's performance is assessed.

Performance evaluation is essential to assess the model's accuracy. Metrics like accuracy, precision, recall, F1 score, and confusion matrix are used to evaluate the model's performance on validation or test datasets. This analysis helps understand the model's strengths, limitations, and areas for improvement.

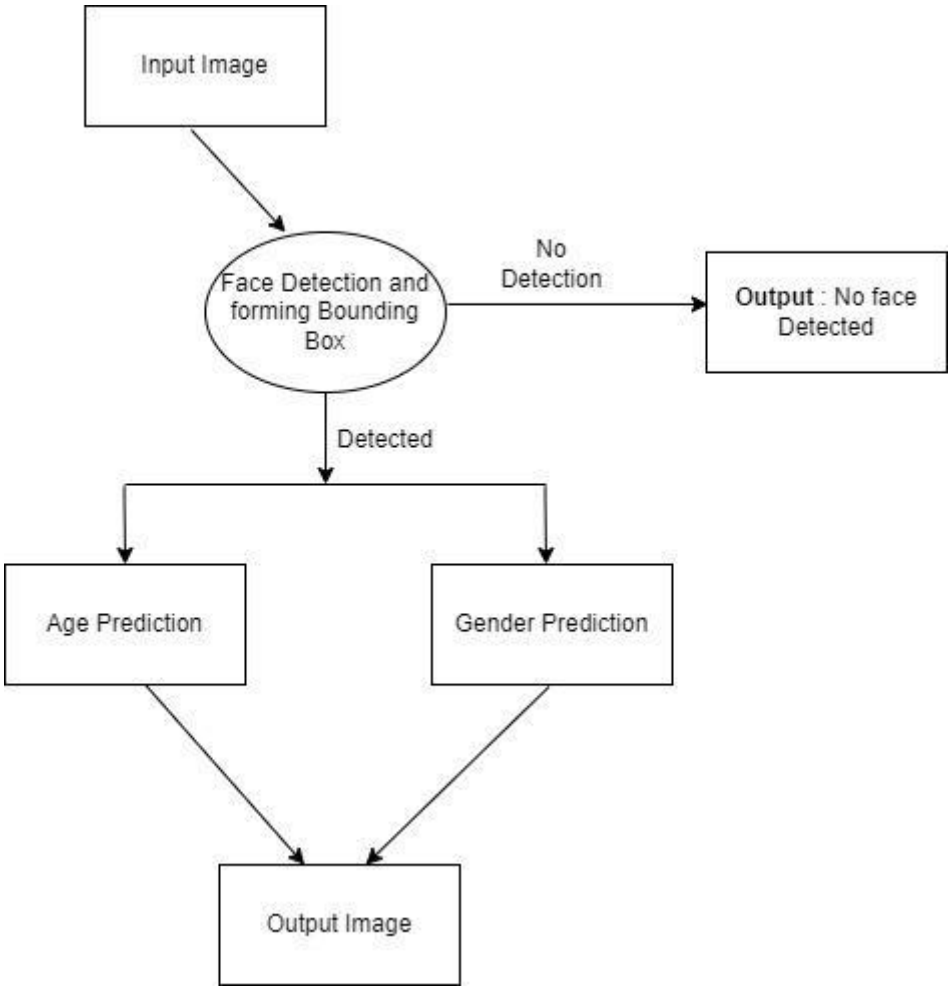
Error analysis involves examining misclassified samples to identify patterns and understand the model's failure cases. This analysis helps refine the model by addressing challenging age or gender categories, variations in pose or lighting conditions, or specific demographic biases.

Generalization and transfer learning capabilities are investigated to assess the model's performance on unseen data. This analysis involves evaluating the model on different datasets, assessing its ability to generalize across demographics, and exploring transfer learning techniques.

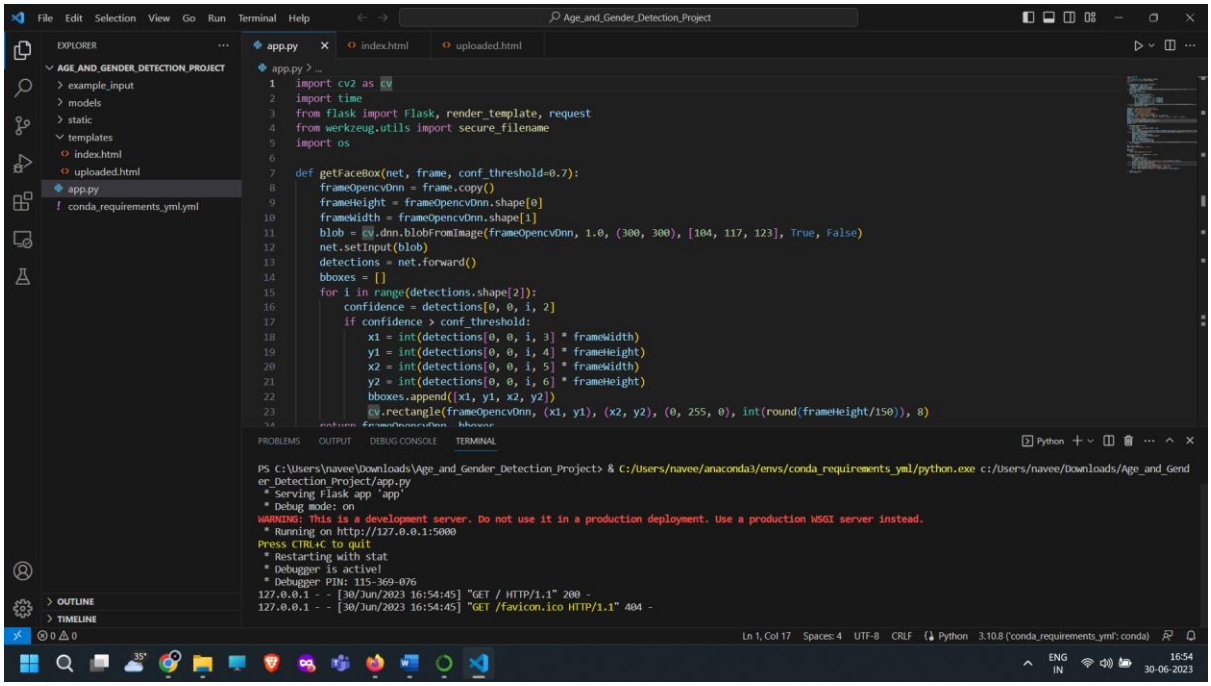
Ethical considerations are crucial in age and gender detection. Analyzing biases, fairness, and privacy implications involves evaluating the model's performance across demographic groups and assessing potential impacts on privacy and sensitive information.

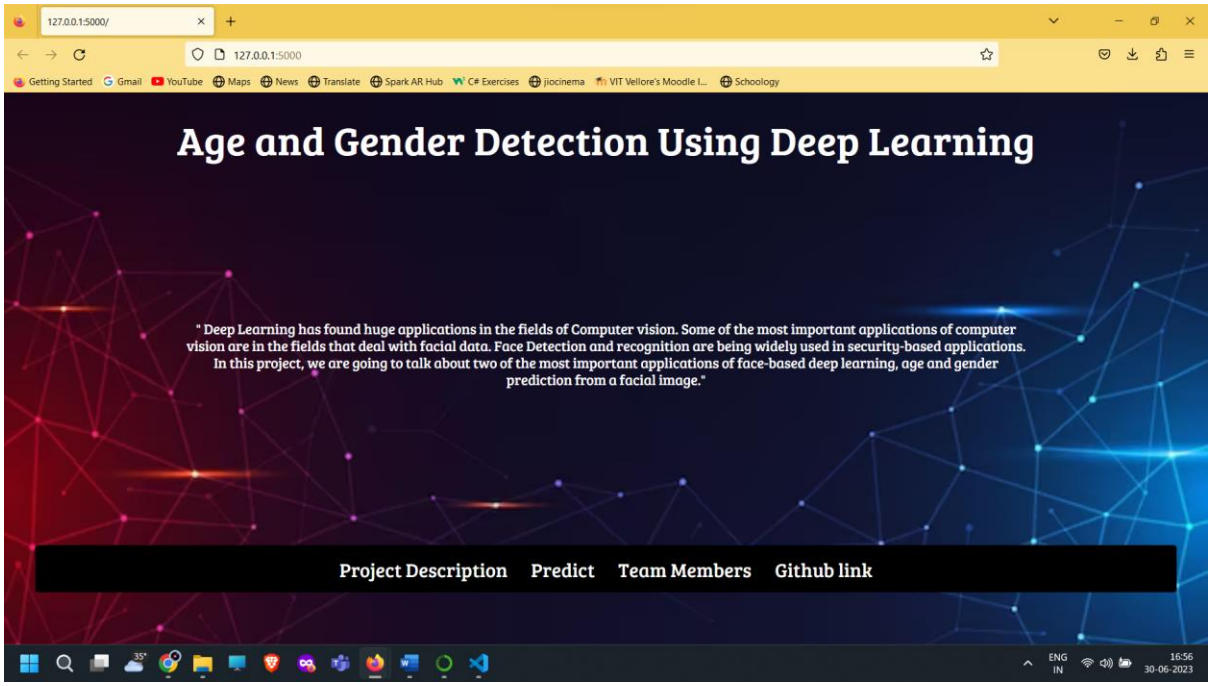
These analyses and investigations collectively refine the age and gender detection solution. They improve the model's performance, address biases and ethical concerns, and ensure responsible and unbiased deployment. By considering these factors, the accuracy, reliability, and ethical soundness of the age and gender detection system can be enhanced.

5. FLOWCHART


Input Image
Face Detection and forming Bounding Box
No Detection
Output : No face Detected
Detected
Age Prediction
Gender Prediction
Output Image

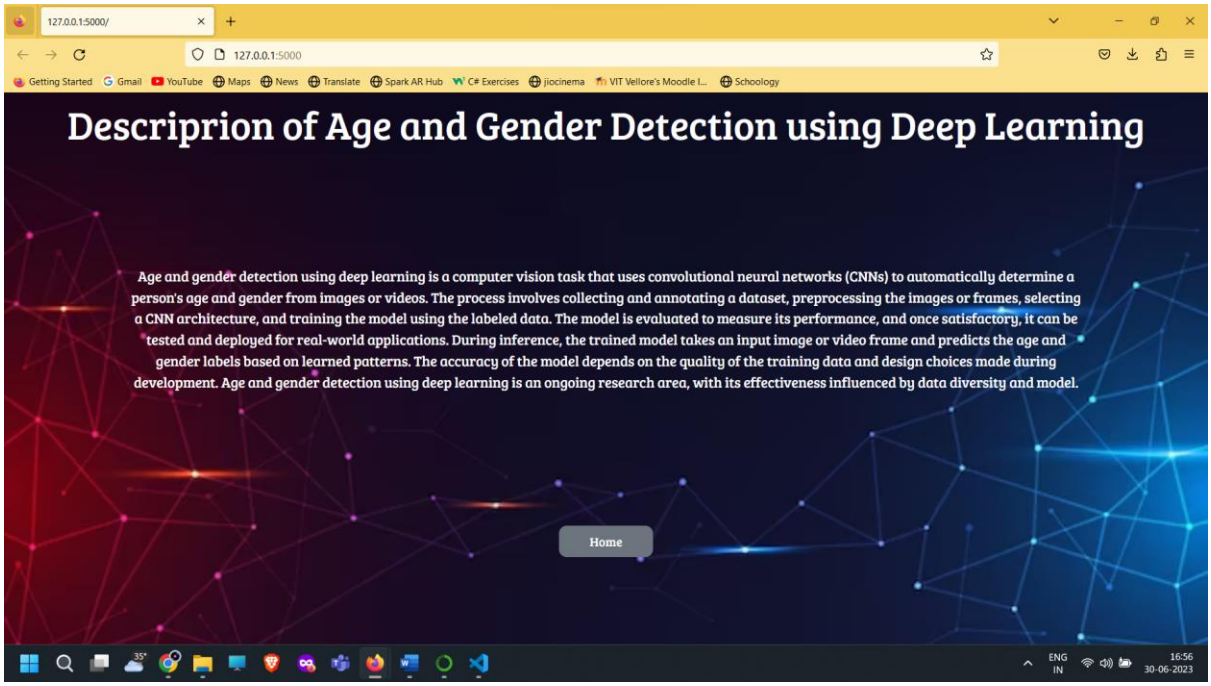
6. RESULT


File Edit Selection View Go Run Terminal Help
Age_and_Gender_Detection_Project
EXPLORER
AGE_AND_GENDER_DETECTION_PROJECT
example_input
models
static
templates
index.html
uploaded.html
app.py
conda_requirements_yml.yml
app.py
index.html
uploaded.html
import cv2 as cv
import time
from flask import Flask, render_template, request
from werkzeug.utils import secure_filename
import os
def getFaceBox(net, frame, conf_threshold=0.7):
frameOpencvDnn = frame.copy()
frameHeight = frameOpencvDnn.shape[0]
frameWidth = frameOpencvDnn.shape[1]
blob = cv.dnn.blobFromImage(frameOpencvDnn, 1.0, (300, 300), [104, 117, 123], True, False)
net.setInput(blob)
detections = net.forward()
bboxes = []
for i in range(detections.shape[2]):
confidence = detections[0, 0, i, 2]
if confidence > conf_threshold:
x1 = int(detections[0, 0, i, 3] * frameWidth)
y1 = int(detections[0, 0, i, 4] * frameHeight)
x2 = int(detections[0, 0, i, 5] * frameWidth)
y2 = int(detections[0, 0, i, 6] * frameHeight)
bboxes.append([x1, y1, x2, y2])
cv.rectangle(frameOpencvDnn, (x1, y1), (x2, y2), (0, 255, 0), int(round(frameHeight/150)), 8)
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Python
PS C:\Users\navee\Downloads\Age_and_Gender_Detection_Project> & C:/Users/navee/anaconda3/envs/conda_requirements_yml/python.exe c:/Users/navee/Downloads/Age_and_Gender_Detection_Project/app.py
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 115-369-076
127.0.0.1 - - [30/Jun/2023 16:54:45] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [30/Jun/2023 16:54:45] "GET /favicon.ico HTTP/1.1" 404 -
OUTLINE
TIMELINE
Ln 1, Col 17 Spaces: 4 UTF-8 CRLF Python 3.10.8 ('conda_requirements_yml': conda)
ENG IN 16:54 30-06-2023

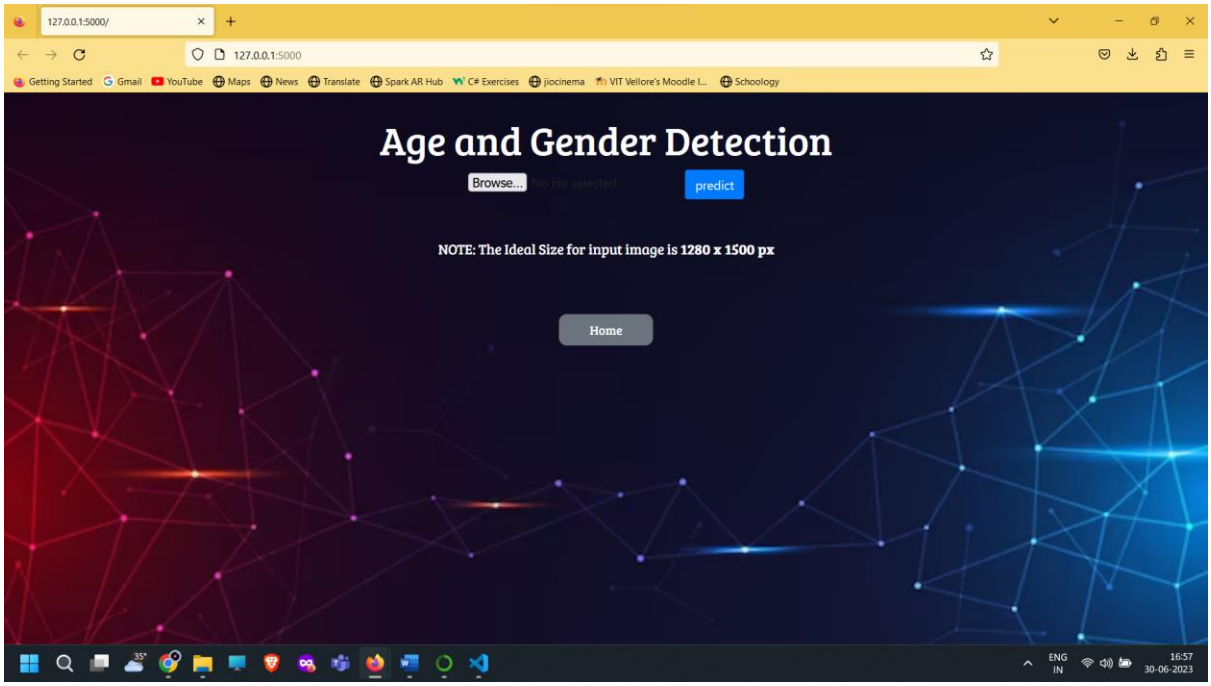

127.0.0.1:5000/
127.0.0.1:5000
Getting Started Gmail YouTube Maps News Translate Spark AR Hub C# Exercises jiocinema VIT Vellore's Moodle l... Schoology
Age and Gender Detection Using Deep Learning
" Deep Learning has found huge applications in the fields of Computer vision. Some of the most important applications of computer vision are in the fields that deal with facial data. Face Detection and recognition are being widely used in security-based applications. In this project, we are going to talk about two of the most important applications of face-based deep learning, age and gender prediction from a facial image."
Project Description Predict Team Members Github link
ENG IN 16:56 30-06-2023

Smart Internz

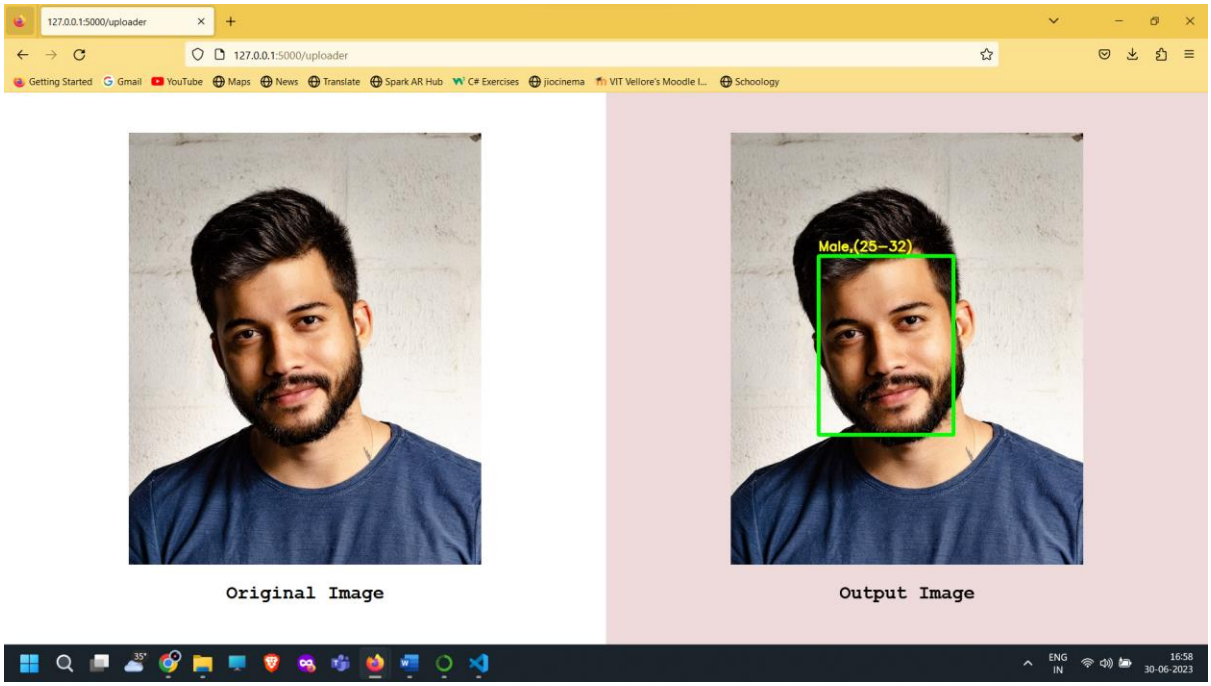
PROJECT DESCRIPTION



PREDICT

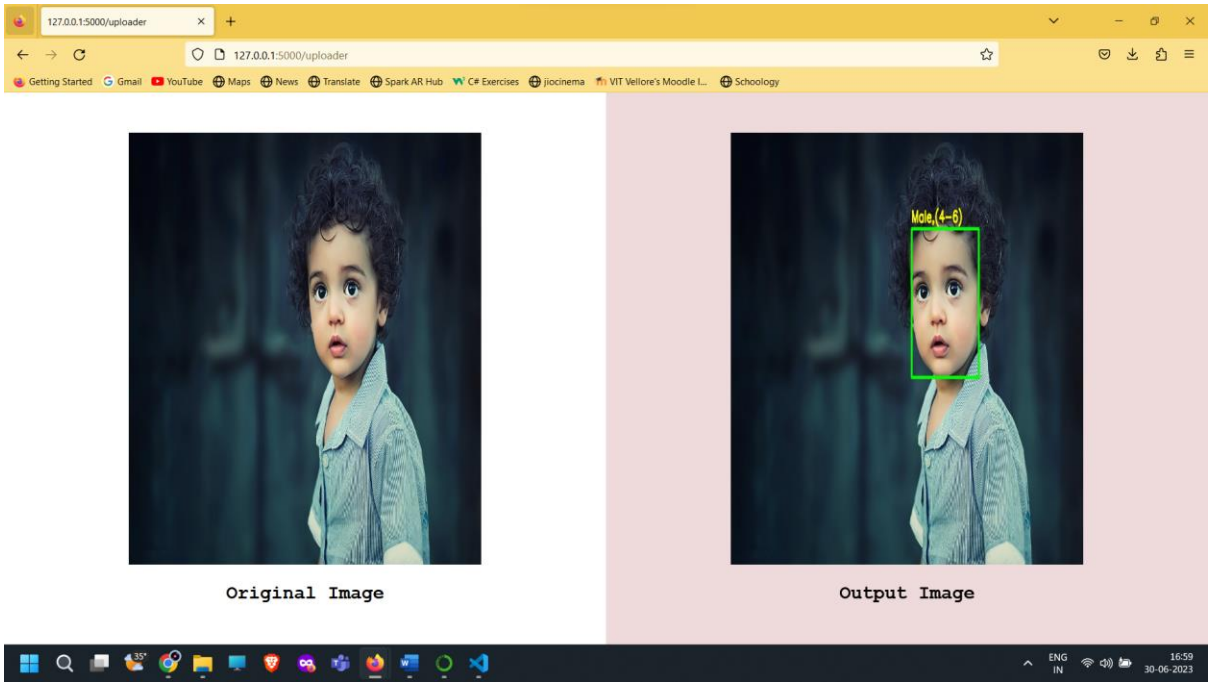


Smart Internz



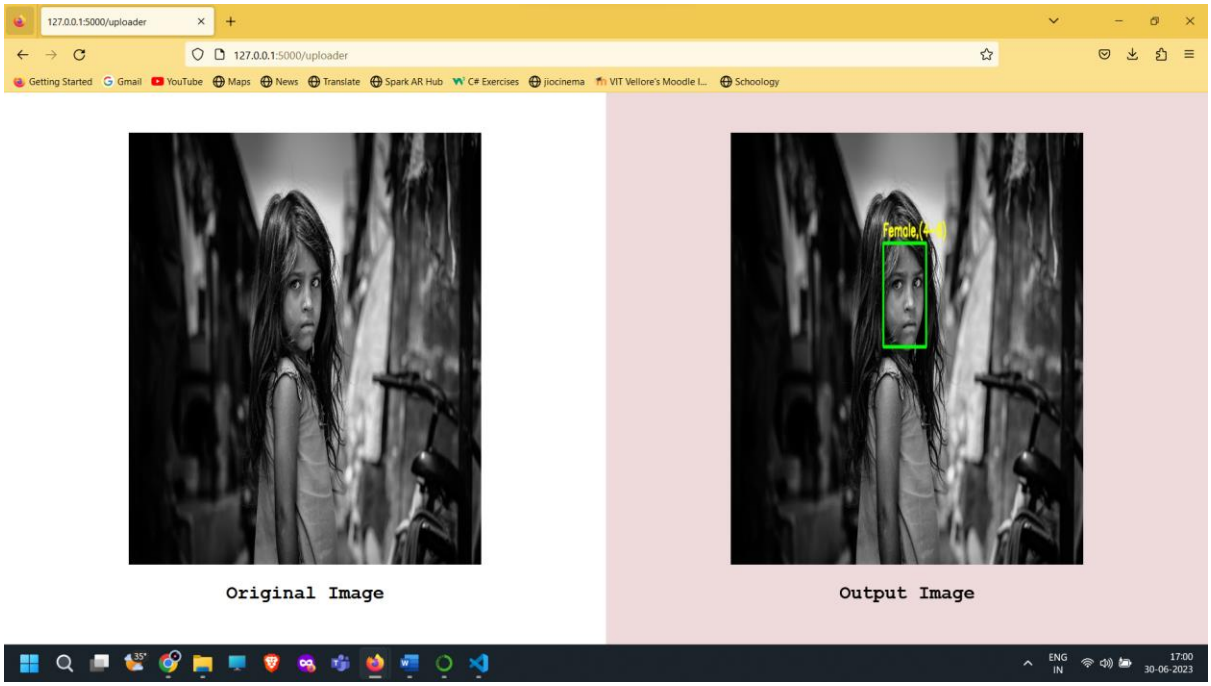
Original Image

Output Image

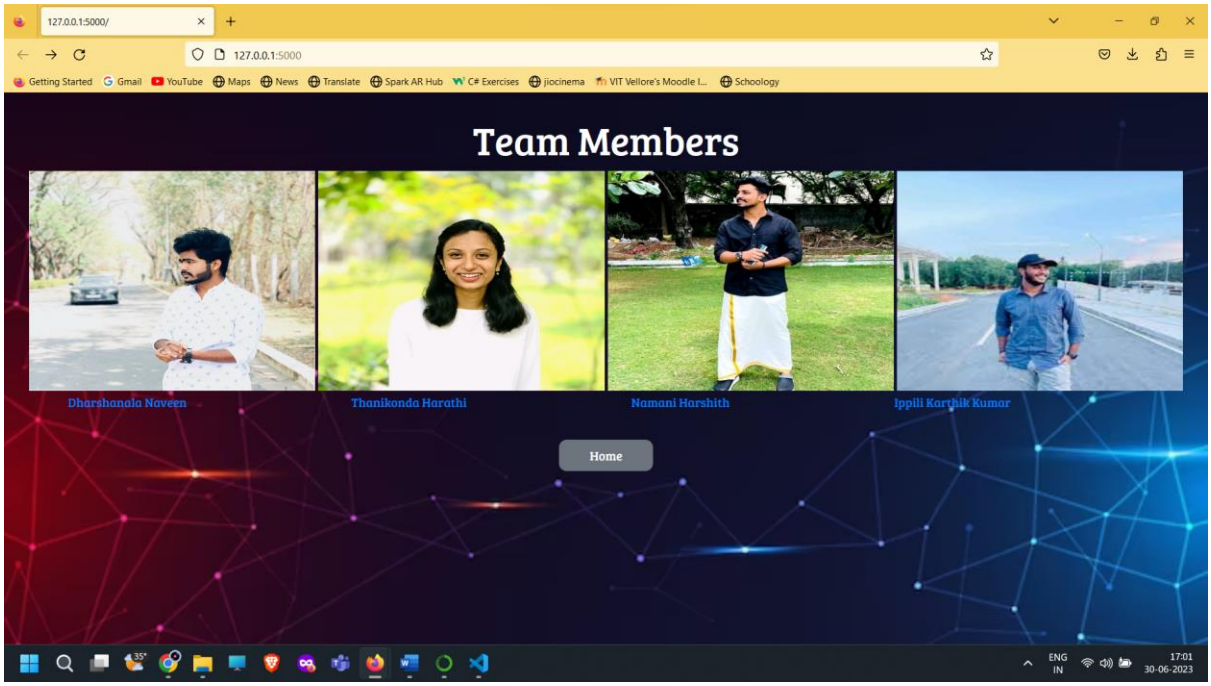


Original Image

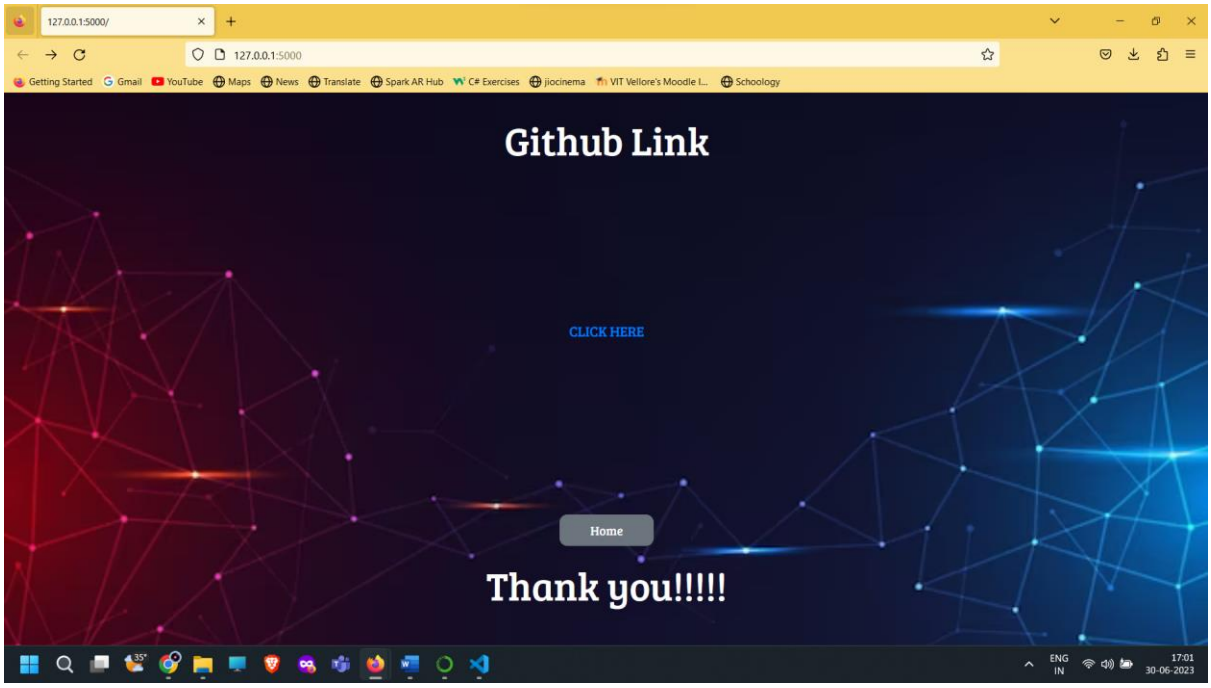
Output Image



TEAM MEMBERS



GITHUB LINK



7. ADVANTAGES AND DISADVANTAGES

ADVANTAGES OF AGE AND GENDER DETECTION USING DEEP LEARNING:

1. Automation and Efficiency: Deep learning-based age and gender detection systems automate the process of determining age and gender from visual data. This eliminates the need for manual analysis, saving time and increasing efficiency in various applications.

2. Accurate and Objective: Deep learning models have shown impressive accuracy in age and gender detection tasks. They can learn complex patterns and features from large datasets, resulting in more accurate and objective predictions compared to manual assessments.

3. Scalability: Deep learning models can handle large-scale datasets and process images or videos in real-time, making them scalable for applications that require high-speed analysis, such as surveillance systems or real-time marketing campaigns.

4. Non-Invasive: Age and gender detection using deep learning can be performed non-invasively, requiring only visual data. This makes it suitable for applications where obtaining age or gender information through direct interaction or surveys may not be feasible or desirable.

5. Personalization and Targeting: The information obtained from age and gender detection can be used for personalization and targeted marketing. It enables businesses to deliver tailored experiences, recommendations, or advertisements to specific demographic groups, improving customer engagement and satisfaction.

DISADVANTAGES OF AGE AND GENDER DETECTION USING DEEP LEARNING:

1. Data Bias and Generalization: Deep learning models are sensitive to biases present in the training data. If the training dataset is unbalanced or contains inherent biases, the model's predictions may be skewed or unfair towards certain demographics. Generalizing the model's performance across diverse populations or cultural contexts can also be challenging.

2. Privacy Concerns: Age and gender detection from visual data raises privacy concerns, as it involves analyzing personal characteristics. Ensuring the ethical collection, storage, and usage of such data is crucial to maintain individuals' privacy and prevent misuse.

3. Challenges in Complex Scenarios: Age and gender detection may face challenges in complex scenarios, such as occlusions, variations in lighting conditions, pose, or facial expressions. These factors can impact the accuracy of the model, requiring additional preprocessing techniques or specialized models to address them.

4. Lack of Contextual Understanding: Deep learning models for age and gender detection primarily rely on visual features. They may lack contextual understanding, leading to occasional misclassifications or inability to consider other relevant factors that could influence age or gender perception.

5. Need for Large and Representative Datasets: Deep learning models require large and representative datasets for training to achieve optimal performance. Obtaining and curating such datasets can be time-consuming and challenging, particularly when considering diverse age ranges, demographics, and cultural variations.

8. APPLICATIONS

Age and gender detection using deep learning has various applications across different industries and domains. Here are some notable applications:

1. Marketing and Advertising: Age and gender detection can assist marketers and advertisers in understanding their target audience better. This information can be used to create personalized marketing campaigns, tailor advertisements, and improve customer targeting strategies.

2. Retail and Customer Analysis: Age and gender detection enables retailers to analyze customer demographics and behavior. It helps in optimizing product displays, creating personalized shopping experiences, and making data-driven decisions about inventory management.

3. Content Recommendation: Online platforms that provide personalized content, such as video streaming services or news aggregators, can utilize age and gender detection to enhance their recommendation systems. By understanding the age and gender of users, they can deliver more relevant and personalized content.

4. User Experience Enhancement: Age and gender detection can be employed to enhance user experiences in various applications. For instance, in virtual reality (VR) or augmented reality (AR) systems, the system can adapt its content or interface based on the age and gender of the user, providing a more immersive and tailored experience.

5. Security and Surveillance: Age and gender detection can contribute to security systems and surveillance applications. It aids in identifying potential threats, monitoring crowd dynamics, and enhancing public safety measures by analyzing the age and gender of individuals in real-time.

6. Healthcare and Gerontology: In healthcare, age and gender detection can assist in patient management and diagnostics. It can be used for age-specific medical screening, identifying gender-related health risks, and supporting research in gerontology to study aging patterns and demographics.

7. Human-Computer Interaction: Age and gender detection can enhance human-computer interaction experiences. User interfaces can adapt based on the age and gender of the user, providing personalized interactions and optimizing user engagement.

8. Social Media Analysis: Age and gender detection can be used in social media platforms for user analysis and understanding social trends. It aids in demographic studies, content filtering, sentiment analysis, and identifying influencers within specific age and gender categories.

9. CONCLUSION

Age and Gender Classification are two of the most essential resources for getting information from an individual. Human faces contain enough information to be useful for a variety of purposes. Human age and gender classification are critical for reaching the right audience. We attempted to replicate the process using standard equipment. The algorithm's efficiency is determined by a number of factors, but the major goal of this study is to make it as simple and quick as possible while maintaining the highest level of accuracy. Work is being done to improve the algorithm's efficiency. Future enhancements include discarding faces for non-human objects, adding more datasets for people of other ethnic groups, and giving the computer more granular control over its workflow. Deep learning and CNN could be used to improve this prototype's ability to reliably identify a person's gender and age range out of a single image of their face. From this study, we can conclude with two important conclusions. First, despite the limited availability of age and gender-tagged photos, CNN can be used to improve age and gender detection outcomes. Second, by employing additional training data and more complex systems, the system's performance can be slightly increased.

10. FUTURE SCOPE

In the future, several enhancements can be made to further improve age and gender detection using deep learning. Here are some potential areas for enhancement:

1. Biometric Features: Deep learning models can be enhanced by incorporating additional biometric features beyond age and gender, such as facial expressions, ethnicity, or other physical characteristics. This would provide a more comprehensive understanding of individuals and enable more nuanced analysis.

2. Multimodal Analysis: Integrating multiple modalities, such as combining visual data with voice or textual information, can enhance the accuracy and robustness of age and gender detection. This multimodal approach would leverage the power of different data sources and provide a more holistic view of individuals.

3. Cross-Cultural Adaptation: Age and gender detection models can be further improved by accounting for cross-cultural variations. Training the models on diverse datasets that include different ethnicities, cultural contexts, and age-related characteristics would enhance their performance across different populations.

4. Real-Time Performance: Optimizing deep learning models to achieve real-time performance is crucial for applications that require immediate analysis, such as surveillance systems or live event monitoring. Enhancements in model architecture, hardware acceleration, or model compression techniques can contribute to faster inference times.

5. Ethical Considerations: Future enhancements should focus on addressing potential biases and fairness issues in age and gender detection systems. Fairness metrics, bias detection techniques, and continuous evaluation can be integrated into the development pipeline to ensure that the models are unbiased, transparent, and ethically sound.

By focusing on these areas of enhancement, age and gender detection using deep learning can become more accurate, inclusive, and privacy-aware. This would pave the way for its broader adoption across industries and domains while addressing ethical concerns and ensuring responsible deployment.

11. BABILOGRAPHY

https://www.ijraset.com/research-paper/detection-of-gender-and-age-using-ml

https://ieeexplore.ieee.org/document/9331995

https://www.mdpi.com/2504-2289/6/4/128

https://towardsdatascience.com/age-detection-using-facial-images-traditional-machine-learning-vs-deep-learning-2437b2feeab2

Smart Internz

APPENDIX

A. SOURCE CODE

app.py

```
import cv2 as cv
import time
from flask import Flask, render_template, request
from werkzeug.utils import secure_filename
import os

def getFaceBox(net, frame, conf_threshold=0.7):
    frameOpencvDnn = frame.copy()
    frameHeight = frameOpencvDnn.shape[0]
    frameWidth = frameOpencvDnn.shape[1]
    blob = cv.dnn.blobFromImage(frameOpencvDnn, 1.0, (300, 300), [104, 117,
123], True, False)
    net.setInput(blob)
    detections = net.forward()
    bboxes = []
    for i in range(detections.shape[2]):
        confidence = detections[0, 0, i, 2]
        if confidence > conf_threshold:
            x1 = int(detections[0, 0, i, 3] * frameWidth)
            y1 = int(detections[0, 0, i, 4] * frameHeight)
            x2 = int(detections[0, 0, i, 5] * frameWidth)
            y2 = int(detections[0, 0, i, 6] * frameHeight)
            bboxes.append([x1, y1, x2, y2])
            cv.rectangle(frameOpencvDnn, (x1, y1), (x2, y2), (0, 255, 0),
int(round(frameHeight/150)), 8)
    return frameOpencvDnn, bboxes

faceProto = "models/opencv_face_detector.pbtxt"
faceModel = "models/opencv_face_detector_uint8.pb"
ageProto = "models/age_deploy.prototxt"
ageModel = "models/age_net.caffemodel"
genderProto = "models/gender_deploy.prototxt"
genderModel = "models/gender_net.caffemodel"
MODEL_MEAN_VALUES = (78.4263377603, 87.7689143744, 114.895847746)
ageList = ['(0-2)', '(4-6)', '(8-12)', '(15-20)', '(25-32)', '(38-43)', '(48-
53)', '(60-100)']
```

```
genderList = ['Male', 'Female']
ageNet = cv.dnn.readNet(ageModel, ageProto)
genderNet = cv.dnn.readNet(genderModel, genderProto)
faceNet = cv.dnn.readNet(faceModel, faceProto)
padding = 20

def age_gender_detector(frame):
    t = time.time()
    frameFace, bboxes = getFaceBox(faceNet, frame)
    for bbox in bboxes:
        face = frame[max(0,bbox[1]-
padding):min(bbox[3]+padding,frame.shape[0]-1),max(0,bbox[0]-
padding):min(bbox[2]+padding, frame.shape[1]-1)]
        blob = cv.dnn.blobFromImage(face, 1.0, (227, 227), MODEL_MEAN_VALUES,
swapRB=False)
        genderNet.setInput(blob)
        genderPreds = genderNet.forward()
        gender = genderList[genderPreds[0].argmax()]
        ageNet.setInput(blob)
        agePreds = ageNet.forward()
        age = ageList[agePreds[0].argmax()]
        label = "{},{}".format(gender, age)
        cv.putText(frameFace, label, (bbox[0], bbox[1]-10),
cv.FONT_HERSHEY_SIMPLEX, 0.8, (0, 255, 255), 2, cv.LINE_AA)
    return frameFace

app = Flask(__name__)
app.config['UPLOAD_FOLDER'] = 'static/'

@app.route('/')
def home():
    return render_template("index.html")

@app.route("/uploader" , methods=['GET', 'POST'])
def uploader():
    if request.method=='POST':
        f = request.files['file1']
        f.filename = "image.jpg"
        f.save(os.path.join(app.config['UPLOAD_FOLDER'],
secure_filename(f.filename)))
        input = cv.imread("static/image.jpg")
        output = age_gender_detector(input)
        cv.imwrite("static/output.jpg", output)
        pic1 = os.path.join(app.config['UPLOAD_FOLDER'], 'output.jpg')
        pic2 = os.path.join(app.config['UPLOAD_FOLDER'], 'image.jpg')
```

```
        return render_template("uploaded.html", output_image=pic1,
input_image=pic2)

if __name__ == '__main__':
    app.run(debug=True)
```

index.html

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css
" integrity="sha384-
JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z"
crossorigin="anonymous">
        <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"
integrity="sha384-
DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj"
crossorigin="anonymous"></script>
        <script
src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js"
integrity="sha384-
9/reFTGAW83EW2RDu2S0VKaIzap3H66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN"
crossorigin="anonymous"></script>
        <script src="https://stackpath.bo
otstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js" integrity="sha384-
B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57BaghqFfPlYxofvL8/KUEfYiJOMMV+rV"
crossorigin="anonymous"></script>
<style>
@import
url('https://fonts.googleapis.com/css2?family=Bree+Serif&family=Caveat:wght@40
0;700&family=Lobster&family=Monoton&family=Open+Sans:ital,wght@0,400;0,700;1,4
00;1,700&family=Playfair+Display+SC:ital,wght@0,400;0,700;1,700&family=Playfai
r+Display:ital,wght@0,400;0,700;1,700&family=Roboto:ital,wght@0,400;0,700;1,40
0;1,700&family=Source+Sans+Pro:ital,wght@0,400;0,700;1,700&family=Work+Sans:it
al,wght@0,400;0,700;1,700&display=swap');
.bg-container{
background-image: url("https://img.freepik.com/free-vector/gradient-network-
connection-background_23-2148881321.jpg");
height: 100vh;
background-size: cover;
padding: 30px;
}
```

Smart Internz

```
.bg-container1{
background-image: url("https://img.freepik.com/free-vector/gradient-network-connection-background_23-2148881321.jpg");
background-size: cover;
height: 100vh;
padding: 10px;
}
.bg-container2{
background-image: url("https://img.freepik.com/free-vector/gradient-network-connection-background_23-2148881321.jpg");
height: 100vh;
background-size: cover;
padding: 30px;
}
.main-heading{
text-align: center;
color: #ffffff;
font-family: "Bree Serif";
font-size: 50px;
}
.image{
padding: 10px;
height: 500px;
width: 600px;
border-radius: 20px;
}
.button{
text-align: center;
margin: 20px;
border-radius: 10px;
width: 120px;
border-width: 0px;
height: 40px;
font-family: "Bree Serif";
}

.para{
font-family: "Bree Serif";
font-size: large;
color: #ffffff;
margin: 200px;
text-align: center;
}
.para1{
font-family: "Bree Serif";
color: #ffffff;
```

```
margin: 15px;
font-size: 25px;
}
.para2{
font-family: "Bree Serif";
font-size: large;
color: #ffffff;
margin: 150px;
text-align: center;
}
.para3{
font-family: "Bree Serif";
font-size: large;
color: #ffffff;
margin: 50px;
text-align: center;
}
.list{
font-family: "Bree Serif";
color: #ffffff;
}
.card{
background-color: black;
padding: 0px;
height: 60px;
margin: 10px;
}
.name{
font-family: "Bree Serif";
color: #ffffff;
margin: 15px;
font-size: 15px;
}
.image2{
height: 280px;
width: 370px;
margin: 2px;
}
.title{
font-family: "Bree Serif";
color: #ffffff;
margin-left: 20px;
margin-right: 190px;
}
</style>
</head>
```

Smart Internz

```
<body>
<div id="sectionhome">
<div class="bg-container">
<h1 class ="main-heading"><b>Age and Gender Detection Using Deep
Learning</b><h1>

<p class="para">" Deep Learning has found huge applications in the fields of
Computer vision. Some of the most important applications of computer vision
are in the fields that deal with facial data. Face Detection and recognition
are being widely used in security-based applications. In this project, we are
going to talk about two of the most important applications of face-based deep
learning, age and gender prediction from a facial image."</p>

<div class="card d-flex flex-row justify-content-center">
<p class="para1"onclick="display('sectionbio')">Project Description</p>
<p class="para1" onclick="display('sectionwriting')">Predict</p>
<p class="para1" onclick="display('sectionaward')">Team Members</p>
<p class="para1" onclick="display('sectionmore')">Github link</p>
</div>
<div class="d-flex flex-row justify-content-end">
</div>
</div>
</div>
<div id="sectionbio">
<div class="bg-container1">
<h1 class="main-heading">Descriprion of Age and Gender Detection using Deep
Learning</h1>
</hr>
<p class="para2">Age and gender detection using deep learning is a computer
vision task that uses convolutional neural networks (CNNs) to automatically
determine a person's age and gender from images or videos. The process
involves collecting and annotating a dataset, preprocessing the images or
frames, selecting a CNN architecture, and training the model using the labeled
data. The model is evaluated to measure its performance, and once
satisfactory, it can be tested and deployed for real-world applications.
During inference, the trained model takes an input image or video frame and
predicts the age and gender labels based on learned patterns. The accuracy of
the model depends on the quality of the training data and design choices made
during development. Age and gender detection using deep learning is an ongoing
research area, with its effectiveness influenced by data diversity and model.
</p>
<div class="d-flex flex-row justify-content-center">
<button class="bn btn-secondary button"
onclick="display('sectionhome')">Home</button>
</div>
</div>
```

Smart Internz

```
</div>
<div id="sectionwriting">
<div class="bg-container2">
<h1 class="main-heading">Age and Gender Detection</h1>
<div class="d-flex flex-row justify-content-center">
<form action="/uploader" method="post" enctype="multipart/form-data">
<input type="file" name="file1">
<button type="submit" class="btn btn-primary">predict</button>
</form>
</div>
<p class="para3">NOTE: The Ideal Size for input image is <b>1280 x 1500
px</b></p>
<div class="d-flex flex-row justify-content-center">
<button class="bn btn-secondary button"
onclick="display('sectionhome')">Home</button>
</div>
</div>
</div>
<div id="sectionaward">
<div class="bg-container">
<h1 class="main-heading">Team Members</h1>
<div class=" d-flex flex-row">
<img src="https://media.licdn.com/dms/image/D4E03AQFDGYJ_OXbrwA/profile-
displayphoto-
shrink_800_800/0/1677241881801?e=2147483647&v=beta&t=gk8QsFO63VBXTepck0pBMUSZE
01snjwoovVLftMH0TY" class="image2"/>
<img src="https://media.licdn.com/dms/image/D4E03AQHpZJ-qECYWJA/profile-
displayphoto-
shrink_800_800/0/1682360249188?e=2147483647&v=beta&t=4L8Ze40Kk9XeusY_lsq8cUhYF
XCQ3KsM6l3ytscTj1s" class="image2"/>
<img src="https://media.licdn.com/dms/image/D5603AQHXwI0YIQRXLQ/profile-
displayphoto-
shrink_800_800/0/1682054482963?e=2147483647&v=beta&t=Oy8bBBV1KQ1Dsea6lgV-
ycJBMyJX77QBF4X-2WFCrKQ" class="image2"/>
<img src="https://media.licdn.com/dms/image/C5603AQHNwN2nGmToww/profile-
displayphoto-
shrink_200_200/0/1661629146293?e=1691020800&v=beta&t=Jmwja7XGdFh0u4RLKRUU9RYfB
UYnjYwDJTlfWVO7lfQ " class="image2"/>
</div>
<div class="d-flex flex-row justify-content-center">
<p class="title"><a href="https://www.linkedin.com/in/naveen-dharshanala-
4188a0266/">Dharshanala Naveen </a></p>
<p class="title"><a href="https://www.linkedin.com/in/harathi-thanikonda-
964167203/">Thanikonda Harathi </a></p>
<p class="title"><a href="https://www.linkedin.com/in/namani-harshith-
667331202/">Namani Harshith </a></p>
```

```
<p class="title"><a href="https://www.linkedin.com/in/ippili-karthik-kumar-
a14213204/">Ippili Karthik Kumar </a></p>
</p>
</div>
<div class="d-flex flex-row justify-content-center">
<button class="bn btn-secondary button"
onclick="display('sectionhome')">Home</button>
</div>
</div>
</div>
<div id="sectionmore">
<div class="bg-container">
<h2 class="main-heading">Github Link</h2>
<div class=" para d-flex flex-row justify-content-center">
<a
href="https://github.com/naveendharshanala/Age_and_Gender_Detection_Project">C
LICK HERE</a>
</div>
<script type="text/javascript"
src="https://d1tgh8fmlzexmh.cloudfront.net/ccbp-static-website/js/ccbp-ui-
kit.js"></script>
<div class="d-flex flex-row justify-content-center">
<button class="bn btn-secondary button"
onclick="display('sectionhome')">Home</button>
</div>
<h5 class="main-heading">Thank you!!!!!</h5>
</body>
</html>
```

Uploaded.html

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
body {
  font-family: Arial;
  color: white;
}

.split {
  height: 100%;
  width: 50%;
  position: fixed;
  z-index: 1;
```

```
  top: 0;
  overflow-x: hidden;
  padding-top: 20px;
}

.left {
  left: 0;
  background-color: rgb(255, 255, 255);
}

.right {
  right: 0;
  background-color: rgb(237, 218, 218);
}

.centered {
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
  text-align: center;
}

.centered img {
  width: 450px;
  height: 550px;

}

.text_below {
 color: black;
 font-family: 'Courier New', Courier, monospace;
}
</style>
</head>
<body>

<div class="split left">
  <div class="centered">
    <img src="{{ input_image }}" alt="Avatar woman">
    <h2 class="text_below">Original Image</h2>
  </div>
</div>

<div class="split right">
  <div class="centered">
```

```
    <img src="{{ output_image }}" alt="Avatar man">
    <h2 class="text_below">Output Image</h2>
  </div>
</div>

</body>
</html>
```