A

Project Report

On

GUARD SCREEN

Submitted to

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE AND TECHNOLOGIES

RK VALLEY

In partial fulfilment of the requirement for the award of the Degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE & ENGINEERING

Submitted by

SINGAMSETTI NANDINI (R180018)

ILOUR SHAIK RIZWAN RAJA (R180006)

JHANSYSREENIVAS MANCHURI (R180996)

Under the Guidance of

Mr. A.MAHENDRA, Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

(Catering the Educational Needs of Gifted Rural Youth of AP)

R.K Valley, Vempalli (M), Kadapa (Dist) – 516330, 2020 - 2024

RAJIV GANDHI UNIVERSITY OF KNOWELDEGE TECHNOLOGIES

(Government Act 18 of 2008)

RGUKT -RK Valley

Vempalli, Kadapa, Andhra Pradesh -516330

CERTIFICATE OF PROJECT COMPLETION

This is to certify that the work entitled **"Guard Screen"** is bonafide work of Singamsetti **Nandini(R180018), Ilour Shaik Rizwan Raja(R180006) ,JhansySreenivas Manchuri(R180996)** carried out under our guidance and supervision for the partial fulfilment for the degree of Bachelor of Technology in Computer Science and Engineering during the academic session August 2023-December 2023 at RGUKT-RK VALLEY.

Project Guide
Mr A Mahendra
Asst.Prof. in Dept of CSE,
RGUKT-RK Valley.

Head of the Department
Mr. N.Satyanandaram,
Lecturer in Dept of CSE,
RGUKT-RK Valley

RAJIV GANDHI UNIVERSITY OF KNOWELDEGE TECHNOLOGIES

(Government Act 18 of 2008)

RGUKT -RK Valley

Vempalli, Kadapa, Andhra Pradesh -516330

CERTIFICATE OF PROJECT COMPLETION

This is to certify that the work entitled **“Guard Screen”** is bonafide work of Singamsetti **Nandini(R180018), Ilour Shaik Rizwan Raja(R180006)** ,**JhansySreenivas Manchuri(R180996)** carried out under our guidance and supervision for the partial fulfilment for the degree of Bachelor of Technology in Computer Science and Engineering during the academic session August 2023-December 2023 at RGUKT-RK VALLEY.

Submitted for the Practical Examination on ………………

External Examiner

RAJIV GANDHI UNIVERSITY OF KNOWELDEGE TECHNOLOGIES

(A.P.Government Act 18 of 2008)

RGUKT -RK Valley

Vempalli, Kadapa, Andhra Pradesh -516330

DECLARATION

We, **Singamsetti Nandini (R180018), Ilour Shaik Rizwan Raja (R180006), JhansySreenivas Manchuri (R180996)** hereby declare that the project report entitled "**Guard Screen**" done under guidance of **Mr. A.Mahendra** is submitted in partial fulfilment for the degree of Bachelor of Technology in Computer Science and Engineering during the academic session February 2023 – July 2023 at RGUKT-RK Valley. we also declare that this project is a result of our own effort and has not been copied or imitated from any source. Citations from any websites are mentioned in the references. To the best of my knowledge, the results embodied in this dissertation work have not been submitted to any university or institute for the award of any degree or diploma.

Date:

Place:

Singamsetti Nandini - R180018

Ilour Shaik Rizwan Raja - R180006

JhansySreenivas Manchuri - R180996

ACKNOWLEDGEMENT

I would like to express my deep sense of gratitude & respect to all those people behind the screen who guided, inspired and helped me crown all my efforts with success. I wish to express my gratitude to **A.Mahendra** for his valuable guidance at all stages of study, advice, constructive suggestions, supportive attitude and continuous encouragement, without which it would not be possible to complete this project.

I would also like to extend our deepest gratitude & reverence to the Director of RGUKT, RK Valley **Dr.A V S S Kumar Swami Gupta** and HOD of Computer Science and Engineering **Mr. N. Satyanandaram** for their constant support and encouragement.

Last but not least I express my gratitude to my parents for their constant source of encouragement and inspiration for me to keep my morals high

With Sincere Regards,

Singamsetti Nandini – R180018

Ilour Shaik Rizwan Raja-R180006

JhansySreenivas Manchuri-R180996

Table of Content

ABSTRACT

Child safety is a paramount concern in various settings, from homes to public spaces. This research introduces a novel Kid Proximity Detection System (KPDS) designed to enhance child safety by providing real-time monitoring of a child's proximity to potential hazards. Utilizing a combination of advanced sensors and wireless communication technology, the KPDS aims to alert caregivers or guardians when a child is in close proximity to danger.

The system employs state-of-the-art proximity sensors strategically placed on the child and within the surrounding environment. These sensors continuously monitor the distance between the child and potential hazards, such as busy roads or restricted areas. In the event of a breach of predefined safety zones, the system triggers instant alerts through a user-friendly mobile application or wearable device.

The research outlines the development and implementation of the KPDS, emphasizing its adaptability to various contexts, including home environments, schools, and public spaces. A comprehensive evaluation of the system's effectiveness in simulated scenarios demonstrates its potential to significantly reduce the risk of child-related accidents.

This research contributes to the field of child safety technology by providing an innovative solution that combines reliable proximity detection with user-friendly interfaces. As child safety remains a critical societal concern, the KPDS represents a promising advancement in proactive measures to safeguard children in diverse environments.

Chapter 1

Introduction

1.1. INTRODUCTION

In an era dominated by digital screens, children are increasingly exposed to television as a primary source of entertainment and education. However, the manner in which children engage with this technology is crucial to their overall well-being. This research explores the significance of maintaining an optimal viewing distance when children watch television and investigates the potential effects of prolonged exposure to screens from close quarters.

Television, with its diverse content and educational programs, undoubtedly plays a pivotal role in shaping a child's cognitive and emotional development. Yet, the physical distance at which children position themselves in relation to the screen remains an aspect that warrants careful consideration. As technology seamlessly integrates into the fabric of daily life, understanding the implications of proximity to the television becomes imperative for parents, educators, and policymakers alike.

This study delves into the effects of watching television from near distances on children's eyesight, attention span, and overall health. By examining existing literature, conducting empirical research, and drawing insights from medical and educational perspectives, this investigation aims to shed light on the potential risks associated with prolonged close-range television viewing.

As we navigate the complexities of screen time guidelines for children, it is essential to equip caregivers with evidence-based information on how to foster a healthy viewing environment. This research not only addresses the issue at hand but also seeks to provide practical recommendations for ensuring that children can enjoy the benefits of television without compromising their well-being. In doing so, we contribute to a broader conversation about responsible screen usage and the promotion of holistic child development in the digital age.

1.2. Purpose

- Child safety is a paramount concern in various settings, from homes to public spaces.
- This research introduces a novel Kid Proximity Detection System (KPDS) designed to enhance child safety by providing real-time monitoring of a child's proximity to potential hazards.
- Utilizing a combination of advanced sensors and wireless communication technology, the KPDS aims to alert caregivers or guardians when a child is in close proximity to danger
- The system employs state-of-the-art proximity sensors strategically placed on the child and within the surrounding environment.
- These sensors continuously monitor the distance between the child and potential hazards, such as busy roads or restricted areas.
- In the event of a breach of predefined safety zones, the system triggers instant alerts through a user-friendly mobile application or wearable device.
- This research contributes to the field of child safety technology by providing an innovative solution that combines reliable proximity detection with user-friendly interfaces.
- As child safety remains a critical societal concern, the KPDS represents a promising advancement in proactive measures to safeguard children in diverse environments.

1.3. Technologies used

Python:

Python is a versatile, high-level programming language known for its readability, simplicity, and flexibility. Developed by Guido van Rossum and first released in 1991, Python has gained immense popularity in various domains, from web development to scientific computing.

Key Characteristics:

<u>Readable Syntax:</u>

Python's syntax emphasizes readability, making it easy for developers to express concepts in fewer lines of code.

<u>Interpreted and Interactive:</u>

Python is an interpreted language, allowing developers to run code line by line, facilitating interactive development and debugging.

<u>Extensive Standard Library:</u>

Python comes with a comprehensive standard library, providing modules and packages for a wide range of functionalities, reducing the need for external libraries.

<u>Community and Ecosystem:</u>

Python boasts a vibrant community and a vast ecosystem of third-party libraries and frameworks, fostering collaboration and innovation.

<u>Cross-Platform Compatibility:</u>

Python is platform-independent, enabling code to run seamlessly on different operating systems without modification.

<u>Object-Oriented Paradigm:</u>

Python supports object-oriented programming, encouraging the use of classes and objects for modular and reusable code.

Machine Learning:

Definition:

Machine Learning (ML) is a subset of artificial intelligence (AI) that focuses on developing algorithms and models capable of learning from and making predictions or decisions based on data. The essence of machine learning is to enable systems to automatically improve and adapt without being explicitly programmed.

Key Concepts:

Data-driven Learning:

Machine learning algorithms rely on data to identify patterns, relationships, and trends. The more diverse and relevant the data, the better the model's performance.

Training and Inference:

ML models go through a training phase where they learn patterns from labeled data. After training, the model can make predictions or decisions when presented with new, unseen data (inference).

Types of Learning:

Supervised Learning: The model is trained on labeled data, with input-output pairs provided. It learns to map inputs to corresponding outputs.

Unsupervised Learning: The model explores data patterns without labeled outputs. Clustering and dimensionality reduction are common tasks.

Reinforcement Learning: The model learns through interaction with an environment, receiving rewards or penalties based on its actions.

Algorithms:

ML employs various algorithms, including linear regression, decision trees, support vector machines, neural networks, and more. The choice of algorithm depends on the problem at hand.

Feature Engineering:

Feature selection and engineering involve choosing or creating relevant input features to improve the model's predictive capabilities.

Computer Vision Toolbox:

The Computer Vision Toolbox is a comprehensive set of tools and functions in MATLAB that facilitates the development and implementation of computer vision algorithms. MATLAB, a high-level programming language widely used in engineering and scientific applications, provides this toolbox to support image and video processing, as well as the development of computer vision systems. Here's a concise overview:

Key Features:

Image Processing Functions:

The toolbox includes a wide range of image processing functions for tasks such as filtering, morphological operations, and transformations.

Image Analysis:

Tools for image segmentation, object detection, and feature extraction enable the analysis of images to identify regions of interest and extract relevant information.

Object Recognition:

Computer Vision Toolbox provides functionality for object recognition and tracking, allowing users to design algorithms that can identify and follow objects in images or video streams.

Camera Calibration:

For tasks involving camera calibration and geometric camera parameters, the toolbox provides functions to estimate camera intrinsics and extrinsics.

Deep Learning Integration:

Integration with deep learning frameworks allows users to incorporate pre-trained deep neural networks for tasks like image classification and object detection.

Neural Network Architecture:

Neural Network Architecture refers to the structure and design of a neural network, a key component of artificial intelligence and machine learning. Neural networks are computational models inspired by the structure and functioning of the human brain, and they excel at learning complex patterns and representations from data. Here's an overview of neural network architecture:

Components of Neural Network Architecture:

Input Layer:

The first layer of the neural network, where input data is fed into the network. Each node in this layer represents a feature of the input data.

Hidden Layers:

Layers between the input and output layers where the neural network learns patterns and representations. Deep neural networks have multiple hidden layers, and each layer consists of nodes (neurons).

Neurons (Nodes):

Fundamental units that process information in a neural network. Neurons receive input, apply a mathematical transformation, and produce an output.

Weights and Biases:

Weights represent the strength of connections between neurons, determining the impact of one neuron's output on another. Biases introduce flexibility to the model by allowing shifts in the input space.

Activation Functions:

Functions applied to the output of neurons to introduce non-linearity, enabling the network to learn complex relationships in the data. Common activation functions include ReLU (Rectified Linear Unit) and sigmoid.

Output Layer:

The final layer that produces the network's output. The number of nodes in this layer depends on the task: one for binary classification, multiple for multiclass classification or regression.

Architecture Types:

Feedforward Neural Networks (FNN): Information flows in one direction, from the input to the output layer, without loops or cycles.

Recurrent Neural Networks (RNN): Allows information to persist by introducing connections that loop back on themselves, making them suitable for sequential data.

Convolutional Neural Networks (CNN): Specialized for image processing, using convolutional layers for feature extraction.

Generative Adversarial Networks (GAN): Comprises a generator and a discriminator network, commonly used for generating new data instances.

CNN: Convolutional Neural Network:

Convolutional Neural Networks (CNNs) are like super-smart computers designed to understand pictures and videos. They're really good at tasks like recognizing objects in photos or figuring out what's happening in a video. Imagine them as special detectives that look closely at different parts of a picture to find important details.

The way CNNs work is by using layers that can understand different parts of a picture. First, they look at basic things like edges and textures.

Then, they combine these details to understand bigger and more complex features. There's also a step where the computer simplifies the picture a bit to make it easier to understand. These networks use special math functions to make sure they can learn and understand all these details.

CNNs are super useful in many areas like telling what's in a photo, finding faces, or even helping self-driving cars see the road. They're like smart assistants for computers, making it possible for machines to see and understand the world around us in a way that wasn't possible before.

MobileNetV2:

MobileNetV2 is a specific architecture within the family of Convolutional Neural Networks (CNNs). It's designed to be lightweight and efficient, making it particularly suitable for mobile and edge devices with limited computational resources.

Key Characteristics of MobileNetV2:

Efficiency:

MobileNetV2 is known for its efficiency, meaning it can perform well in tasks like image classification while requiring fewer computational resources. This efficiency makes it a popular choice for applications on mobile devices.

Depthwise Separable Convolution:

One of the key innovations in MobileNetV2 is the use of depthwise separable convolutions. Instead of using a traditional convolution that operates on all input channels at once, depthwise separable convolutions split this operation into two steps: depthwise convolution (operating on each input channel separately) followed by pointwise convolution (combining information across channels).

Inverted Residuals:

MobileNetV2 introduces a concept called inverted residuals. Inverted residuals are used to maintain low-dimensional embeddings throughout the network, helping to capture and retain important information while efficiently utilizing resources.

Linear Bottlenecks:

The architecture employs linear bottlenecks, which means that the intermediate representations within the network have a linear activation function. This helps in better information flow and training stability.

Width Multiplier and Resolution Multiplier:

MobileNetV2 introduces parameters like width multiplier and resolution multiplier. The width multiplier adjusts the number of channels in each layer, allowing for a trade-off between accuracy and computational cost. The resolution multiplier scales down the input image size, providing another way to control computational complexity.

Suitability for Real-Time Applications:

Due to its lightweight nature, MobileNetV2 is often used in real-time applications, including tasks like object detection and image classification on devices with limited processing power, such as smartphones and embedded systems.

Haar Cascades:

Haar Cascades, named after the Hungarian computer scientist László Haar, refer to a machine learning object detection method used to identify objects or patterns in images or video. Haar Cascades are particularly popular for face detection. This technique is widely used due to its speed and efficiency.

Here's how Haar Cascades work:

<u>Haar-like Features:</u>

Haar Cascades are based on Haar-like features, which are simple rectangular filters. These features look at the intensity differences between adjacent regions of an image.

<u>Training a Classifier:</u>

A Haar Cascade classifier is trained using positive and negative images. Positive images contain instances of the object you want to detect (e.g., faces), and negative images contain backgrounds without the object.

<u>Integral Image:</u>

To speed up computations, an integral image is calculated for the input image. The integral image allows for rapid calculation of the sum of pixel values in any rectangular region.

<u>Cascade of Classifiers:</u>

The term "Cascade" in Haar Cascades refers to a series of classifiers arranged in a cascade fashion. Each stage of the cascade eliminates non-object regions, allowing subsequent stages to focus on more promising areas. This cascade structure enhances the speed of the detection process.

<u>Sliding Window Technique:</u>

Haar Cascades use a sliding window technique to move across the image at different scales. The classifier is applied at each position and scale to detect objects.

<u>Real-Time Object Detection:</u>

Haar Cascades are known for their real-time object detection capabilities. They have been successfully used in applications such as face detection in cameras and video surveillance systems.

OpenCV Implementation:

OpenCV, a popular computer vision library, provides pre-trained Haar Cascade classifiers for various objects, including faces, eyes, and hands. These pre-trained models can be easily integrated into applications.

Chapter 2

DATA COLLECTION TECHNIQUES

When implementing proximity detection systems to classify kids and adults, a combination of various data collection techniques can be employed to enhance accuracy and reliability. Here are some data collection techniques specifically tailored for classifying kids and adults in proximity detection:

Height Sensors:

Description**:** Use sensors that can measure height, such as ultrasonic or infrared sensors, to estimate the height of individuals. Kids typically have shorter heights compared to adults.

Advantages: Direct measurement, non-intrusive.

Facial Recognition:

Description: Employ facial recognition technology to analyse facial features and distinguish between kids and adults. This technique can be effective if the system has access to a database of facial profiles for comparison.

Advantages: Non-intrusive, potential for real-time identification.

Gait Analysis:

Description: Analyse the walking patterns or gait of individuals, as kids and adults often have distinct walking styles. This can be done using video analytics or specialized sensors.

Advantages: Can work in real-time, non-intrusive.

Biometric Data:

Description: Collect biometric data such as fingerprints or palm prints, which may have age-specific patterns. However, this approach may be more intrusive and less common in proximity detection scenarios.

Advantages: Potentially accurate, if applicable.

School Identification Cards:

Description: If the proximity detection is implemented in a school setting, utilize school identification cards that typically contain information about the student's age.

Advantages: Official and reliable data.

Parental Consent Forms:

Description: Obtain parental consent, which may include providing information about the child's age. This is particularly relevant in scenarios involving children.

Advantages: Ethical and legal compliance.

Behavioural Patterns:

Description: Analyze behavioural patterns, such as movement speed or interaction with the environment, which may differ between kids and adults.

Advantages: Contextual information, non-intrusive.

Voice Analysis:

Description: Analyze voice characteristics to distinguish between the higher-pitched voices of children and the lower-pitched voices of adults.

Advantages: Non-intrusive, potential for real-time analysis

Chapter 3

MEASURES AND FORMULAS

3.1. MEASURES

1. Optimal Viewing Distance:

The American Academy of Ophthalmology recommends a viewing distance of at least 5 times the diagonal size of the television screen. For example, if the TV screen is 32 inches diagonally, the optimal viewing distance would be approximately 160 inches or 4 meters.

2. Screen Time Guidelines:

Adhere to established guidelines on daily screen time for children. The American Academy of Pediatrics, for instance, suggests limiting screen time to one hour per day for children aged 2 to 5 years and encouraging consistent screen-free zones, especially during meals and before bedtime.

3. Eye Level Measurement:

Ensure that the television is positioned at or slightly below the eye level of the child to reduce strain on the neck and eyes.

3.2 FORMULAS

1. Optimal Viewing Distance Formula:

$$\text{Optimal Viewing Distance} = 5 \times \text{Diagonal Size of TV Screen}$$

This formula helps determine the ideal distance from which a child should watch television based on the size of the screen.

2. Visual Acuity Calculation:

Visual Acuity is a measure of the clarity of vision and can be calculated using the formula:

Visual Acuity = (Distance from the TV) / (Distance at which a standard eye can read the same text)

- This measure helps assess the strain on a child's eyes based on viewing distance.

3. Screen Time Percentage:

- To calculate the percentage of daily time spent on screen activities, the formula is:

- This provides a quantitative measure of a child's screen exposure relative to their total waking hours.

Screen Time Percentage = (Time spent on screen / Total waking hour) × 100

Remember, while these measures and formulas provide guidelines, individual differences and specific health conditions should also be taken into account. Regular eye check-ups and consultation with healthcare professionals are essential for personalized recommendations.

4. Size and Reading Distance:

For devices with text content, consider the relationship between font size (in points) and the recommended reading distance:

Recommended Reading Distance = Font Size / Constant Factor

Adjust font sizes to be legible at the appropriate viewing distance . The constant factor in the formula is a value that depends on various factors, including the characteristics of the display device, the type of content being viewed, and individual differences in visual acuity. This constant factor is used to estimate the recommended reading distance based on the font size.

5: Blue Light Exposure:

Some studies suggest that prolonged exposure to blue light emitted by screens may affect sleep patterns. To assess potential impact, consider:

Blue Light Exposure = Intensity of Blue Light / Duration of Exposure

Manage screen time, especially close to bedtime, to minimize potential disruptions to sleep.

CHAPTER 4

Project Code

Image Classification Program:

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.applications import MobileNetV2
from tensorflow.keras import layers, Model
from tensorflow.keras.callbacks import EarlyStopping
from sklearn.metrics import confusion_matrix, classification_report

# Define data generators with data augmentation
train_datagen = ImageDataGenerator(
    rescale=1./255,
    rotation_range=20,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True,
    fill_mode='nearest'
)

test_datagen = ImageDataGenerator(rescale=1./255)

train_generator = train_datagen.flow_from_directory(
    'archive/train',
    target_size=(224, 224),
    batch_size=32,
    class_mode='binary',
    classes=['children', 'adult']
)

test_generator = test_datagen.flow_from_directory(
    'archive/test',
    target_size=(224, 224),
    batch_size=32,
    class_mode='binary',
    classes=['children', 'adult']
)
```

```
# Load MobileNetV2 with pre-trained weights (without the top layers)
base_model = MobileNetV2(input_shape=(224, 224, 3), include_top=False)

# Create a smaller model based on MobileNetV2
x = layers.GlobalAveragePooling2D()(base_model.output)
x = layers.Dropout(0.5)(x)
x = layers.Dense(16, activation='relu')(x)
x = layers.Dense(1, activation='sigmoid')(x)

model = Model(inputs=base_model.input, outputs=x)

model.compile(optimizer='adam', loss='binary_crossentropy',
metrics=['accuracy'])

# Implement early stopping
early_stopping = EarlyStopping(monitor='val_loss', patience=10,
restore_best_weights=True)

# Train the model with data augmentation and early stopping
model.fit(train_generator, validation_data=test_generator, epochs=50,
callbacks=[early_stopping])

# Save the trained model to a file
model.save('trained_model.h5')

# Evaluate the model on the test set
test_loss, test_acc = model.evaluate(test_generator)
print(f'Test accuracy: {test_acc}')

# Predict the labels for the test set
y_pred_probs = model.predict(test_generator)
threshold = 0.5  # Adjust the threshold as needed
y_pred_binary = (y_pred_probs > threshold).astype(int)

# Get true labels
y_true = test_generator.classes

# Print confusion matrix and classification report
conf_matrix = confusion_matrix(y_true, y_pred_binary)
class_report = classification_report(y_true, y_pred_binary)

print("Confusion Matrix:")
```

```
print(conf_matrix)

print("\nClassification Report:")
print(class_report)
```

Screen Blur Program:

1. Import Packages

```
import cv2
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
from tensorflow.keras.applications.mobilenet_v2 import preprocess_input
```

2. To Identify the Faces on the Camera and Classifying the Person

```
def classify_frame(frame):
    # Convert BGR image to RGB
    rgb_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)

    # Detect faces in the frame
    faces = face_cascade.detectMultiScale(rgb_frame, scaleFactor=1.1,
minNeighbors=5, minSize=(30, 30))

    for (x, y, w, h) in faces:
        # Extract the face region
        face_roi = rgb_frame[y:y+h, x:x+w]
        # Resize the face image to the model input size
        img = cv2.resize(face_roi, (224, 224))
        img = image.img_to_array(img)
        img = np.expand_dims(img, axis=0)
        img = preprocess_input(img)

        # Predict the class
        prediction = model.predict(img)

        # Display the result on the frame
        label = "Child" if prediction < 0.5 else "Adult"
        color = (0, 255, 0) if label == "Child" else (0, 0, 255)
        cv2.rectangle(frame, (x, y), (x + w, y + h), color, 2)
        cv2.putText(frame, f'Prediction: {label}', (x, y - 10),
cv2.FONT_HERSHEY_SIMPLEX, 0.5, color, 2)

        # Check if it's a child and closer to the camera
```

```
        if label == "Child" and w > 100:
            # Apply blur effect to the entire frame
            frame = cv2.GaussianBlur(frame, (99, 99), 0)

    return frame
```

3. Main Code to Call the Function:

```
# Open a connection to the webcam
cap = cv2.VideoCapture(0)

while True:
    # Capture frame-by-frame
    ret, frame = cap.read()

    # Perform face classification
    frame = classify_frame(frame)

    # Display the resulting frame
    cv2.imshow('Webcam Classification', frame)

    # Break the loop if 'q' key is pressed
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

# When everything is done, release the capture
cap.release()
cv2.destroyAllWindows()
```

Chapter 5
FUTURE SCOPE

In the future, continuous refinement and expansion of the system could open avenues for collaboration with stakeholders and industries seeking innovative solutions for proximity awareness. Our major project not only serves as a testament to our technical acumen but also as a contribution to the ongoing discourse on advancing safety through cutting-edge technology.

In essence, the proximity detection system developed in this major project not only fulfils its primary objective but sets a precedent for the integration of intelligent sensing systems in diverse domains.

1. To Develop a Mobile App

Our future focus is directed towards crafting an intuitive mobile app, empowering parents and caregivers with seamless access to real- time insights and alerts from our Kid Proximity Detection system, fostering a safer environment for children.

2. Using IOT to apply the model to Webcams

In the future, we plan to use IoT technology to connect our Kid Proximity Detection system, making it smarter and more connected. This will give parents and care givers timely alerts and insights, creating a safer and more interconnected system for child safety

Chapter 6
CONCLUSION

6. Conclusion

In conclusion, the implementation of proximity detection in our major project represents a significant stride towards enhancing safety and efficiency in various contexts. The amalgamation of advanced sensor technologies, data analytics, and real-time monitoring has yielded a robust system capable of discerning the proximity of individuals in a dynamic environment.

Through meticulous research, development, and testing, we have addressed the critical need for reliable proximity detection, particularly focusing on the classification of kids and adults. The utilization of [specific technologies and methods employed] has enabled us to achieve accurate and timely identification, paving the way for a myriad of applications across sectors such as [mention relevant sectors, e.g., education, healthcare, public spaces].

The project's success lies not only in the technical prowess exhibited but also in its adherence to ethical and privacy considerations. The implementation ensures data security, respects user privacy, and aligns with legal standards, establishing a foundation for responsible technological deployment.

As we anticipate the broader integration of this proximity detection system, its impact on [specific use cases] is poised to revolutionize [industry or application]. The adaptability of the technology positions it as a versatile solution capable of addressing various challenges associated with [specific issues related to proximity and safety].

REFERENCES

Text Books:

[1] Nasir D. Memon , Toan Nguyen, “Kid on The Phone! Toward Automatic Detection of Children on Mobile Devices”, Computers & Security · April 2019, DOI:10.1016/j.cose.2019.04.001

[2] Murphy, K. P. (2012), "Machine Learning: A Probabilistic Perspective.", MIT Press, Cambridge, MA (2012)

Video Links:

1. https://youtu.be/JZZr0PjZsIk?feature=shared