Research Publication and Awards

Year

2022

Shahid Virpatni Laxmi Mahavidyalaya
Affl No. 452
Titave

FACE SENSE WEB CAMERA-BASED ATTENDANCE MONITORING WITH PYTHON AND ML

1.Prof. Prashant M. Palkar, 2. Prof. Sagar V. Shetage 3. Prof. Shashikant Patil, 4. Prof. Shrivijay A. Chavan, 5. Prof. Sanyogita S. Warake (Assistant Professor, Shahid Virpatni Laxmi Mahavidyalaya, Titave)

Abstract:

In an era where having a computer sing of your operations is a matter of prestige for some and necessity for others. Management of these projects is still an area which is ignored or done in traditional manner. We in a system called Face Sense Web Camera-based Attendance Monitoring with Python and ML is integrated and automation software for cable operators. Cable operators will provide distributed channels to their customers.

Introduction:

The main objective of this project is to develop face recognition based automated student attendance system. To achieve better performance, the test images and training images of this proposed approach are limited to frontal and upright facial images that consist of a single face only. The test images and training images must be captured by using the same device to ensure no quality difference. In addition, the students must register in the database to be recognized. The enrolment can be done on the spot through the user-friendly interface.

Language Learning:

Python: Python is the primary programming language used for developing the system. You'll need to be familiar with basic Python syntax, data structures, and libraries.

Machine Learning: You'll need to learn the basics of machine learning, including data preprocessing, model training, and evaluation. Specifically, you'll use machine learning.

CASE study of Face Sense Web Camera-based Attendance Monitoring with Python and ML:

Problem Statement:

A university wants to automate its attendance tracking system to improve accuracy and efficiency. They currently use manual methods that are prone to errors and time-consuming. The university wants a solution that can automatically recognize students' faces using a webcam and mark their attendance accordingly.

Solution Overview:

A face recognition system will be developed using Python and machine learning. The system will capture images from a webcam, detect faces in the images, and match them against a database of known students to mark attendance. The system will be trained on a dataset of students' faces to recognize them accurately.

Technologies Used:

Python: For developing the face recognition system and integrating it with the webcam.

OpenCV: For capturing images from the webcam and detecting faces.

Dlib: For facial landmark detection and face recognition.

Machine Learning: For training the face recognition model on a dataset of students' faces.

Implementation Steps:

Data Collection: Collect a dataset of students' faces. Each student should have multiple images taken from different angles and lighting conditions.

Preprocessing: Preprocess the images to improve the quality of the data. This may include resizing, normalization, and grayscale conversion.

Face Detection: Use OpenCV to detect faces in the images. This step is necessary to isolate the faces from the rest of the image.

Facial Landmark Detection: Use Dlib to detect facial landmarks (e.g., eyes, nose, mouth) in the detected faces. This step helps in aligning the faces for better recognition accuracy.

Feature Extraction: Extract features from the detected faces. This step converts the faces into a format that can be used for training the machine learning model.

Model Training: Train a machine learning model (e.g., SVM, KNN) on the extracted features. The model should be trained to recognize the faces of known students.

Attendance Marking: In real-time, capture images from the webcam, detect faces, and use the trained model to recognize the faces. Mark the attendance of recognized students.

Integration: Integrate the face recognition system with the university's attendance tracking system. This may involve developing APIs or other interfaces for data exchange.

Results:

The face recognition system achieved an accuracy of over 90% in recognizing students' faces.

The automated attendance system reduced the time taken to mark attendance and improved accuracy compared to manual methods.

Future Enhancements:

Implementing a user interface for administrators to monitor attendance and manage the system.

Adding support for real-time notifications or alerts for attendance-related issues.

Integrating the system with other university systems for seamless data exchange.

By implementing this solution, the university was able to automate its attendance tracking system, leading to increased efficiency and accuracy.

Conclusion:

In this approach, a face recognition based automated student attendance system is thoroughly described. The proposed approach provides a method to identify the individuals by comparing their input image obtained from recording video frame with respect to train image. This proposed approach able to detect and localize face from an input facial image, which is obtained from the recording video frame. Besides, it provides a method in pre-processing stage to enhance the image contrast and reduce the illumination effect. Extraction of features from the facial image is performed by applying both LBP and PCA. The algorithm designed to combine LBP and PCA able to stabilize the system by giving consistent results. The accuracy of this proposed approach is 100 % for high-quality images, 92.31 % for low-quality images and 95.76 % of Yale face database when two images per person are trained.

As a conclusion for analysis, the extraction of facial feature could be challenging especially in different lighting. In pre-processing stage, Contrast Limited Adaptive Histogram Equalization (CLAHE) able to reduce the illumination effect. CLAHE perform better compared to histogram equalization in terms of contrast improvement. Enhanced LBP with larger radius sizes specifically, radius size two, perform better compared to original LBP operator, with less affected by illumination and more consistent compared to other radius sizes.

References:

1. Fussell, S. (2018). NEWS Facebook's new face recognition features: What we do (and don't) know. Gizmodo. Retrieved from https://gizmodo.com/facebooks-new-face-recognition-features-what-we-do-an-1823359911
2. DuVal, A. (2012). Face recognition software - History of forensic psychology. Forensic Psychology. Retrieved from http://forensicpsych.umwblogs.org/research/criminal-justice/face-recognition-software/
3. G.R, P., et al. (2010). An automated attendance system using image processing. International Journal of Advanced Networking & Applications.
4. Devan, P. A. M., et al. (2017). Smart attendance system using face recognition. Advances in Natural and Applied Sciences, 11 (7), 139-144.
5. Shirodkar, M. (2015). Automated attendance management system using face recognition. International Journal of Computer Applications and International Conference and Workshop on Emerging Trends in Technology.

6. Paul, L. C., & Al Sumam, A. (2012). Face recognition using principal component analysis method. IJARCET, V1(9).

FACE SENSE WEB CAMERA-BASED ATTENDANCE MONITORING WITH PYTHON AND ML

1.Prof. Prashant M. Palkar, 2. Prof. Sagar V. Shetage 3. Prof. Shashikant Patil, 4. Prof. Shrivijay A. Chavan, 5. Prof. Sanyogita S. Warake (Assistant Professor, Shahid Virpatni Laxmi Mahavidyalaya, Titave)

Abstract:

In an era where having a computer sing of your operations is a matter of prestige for some and necessity for others. Management of these projects is still an area which is ignored or done in traditional manner. We in a system called Face Sense Web Camera-based Attendance Monitoring with Python and ML is integrated and automation software for cable operators. Cable operators will provide distributed channels to their customers.

Introduction:

The main objective of this project is to develop face recognition based automated student attendance system. To achieve better performance, the test images and training images of this proposed approach are limited to frontal and upright facial images that consist of a single face only. The test images and training images must be captured by using the same device to ensure no quality difference. In addition, the students must register in the database to be recognized. The enrolment can be done on the spot through the user-friendly interface.

Language Learning:

Python: Python is the primary programming language used for developing the system. You'll need to be familiar with basic Python syntax, data structures, and libraries.

Machine Learning: You'll need to learn the basics of machine learning, including data preprocessing, model training, and evaluation. Specifically, you'll use machine learning.

CASE study of Face Sense Web Camera-based Attendance Monitoring with Python and ML:

Problem Statement:

A university wants to automate its attendance tracking system to improve accuracy and efficiency. They currently use manual methods that are prone to errors and time-consuming. The university wants a solution that can automatically recognize students' faces using a webcam and mark their attendance accordingly.

Solution Overview:

A face recognition system will be developed using Python and machine learning. The system will capture images from a webcam, detect faces in the images, and match them against a database of known students to mark attendance. The system will be trained on a dataset of students' faces to recognize them accurately.

Technologies Used:

Python: For developing the face recognition system and integrating it with the webcam.

OpenCV: For capturing images from the webcam and detecting faces.

Dlib: For facial landmark detection and face recognition.

Machine Learning: For training the face recognition model on a dataset of students' faces.

Implementation Steps:

Data Collection: Collect a dataset of students' faces. Each student should have multiple images taken from different angles and lighting conditions.

Preprocessing: Preprocess the images to improve the quality of the data. This may include resizing, normalization, and grayscale conversion.

Face Detection: Use OpenCV to detect faces in the images. This step is necessary to isolate the faces from the rest of the image.

Facial Landmark Detection: Use Dlib to detect facial landmarks (e.g., eyes, nose, mouth) in the detected faces. This step helps in aligning the faces for better recognition accuracy.

Feature Extraction: Extract features from the detected faces. This step converts the faces into a format that can be used for training the machine learning model.

Model Training: Train a machine learning model (e.g., SVM, KNN) on the extracted features. The model should be trained to recognize the faces of known students.

Attendance Marking: In real-time, capture images from the webcam, detect faces, and use the trained model to recognize the faces. Mark the attendance of recognized students.

Integration: Integrate the face recognition system with the university's attendance tracking system. This may involve developing APIs or other interfaces for data exchange.

Results:

The face recognition system achieved an accuracy of over 90% in recognizing students' faces.

The automated attendance system reduced the time taken to mark attendance and improved accuracy compared to manual methods.

Future Enhancements:

Implementing a user interface for administrators to monitor attendance and manage the system.

Adding support for real-time notifications or alerts for attendance-related issues.

Integrating the system with other university systems for seamless data exchange.

By implementing this solution, the university was able to automate its attendance tracking system, leading to increased efficiency and accuracy.

Conclusion:

In this approach, a face recognition based automated student attendance system is thoroughly described. The proposed approach provides a method to identify the individuals by comparing their input image obtained from recording video frame with respect to train image. This proposed approach able to detect and localize face from an input facial image, which is obtained from the recording video frame. Besides, it provides a method in pre-processing stage to enhance the image contrast and reduce the illumination effect. Extraction of features from the facial image is performed by applying both LBP and PCA. The algorithm designed to combine LBP and PCA able to stabilize the system by giving consistent results. The accuracy of this proposed approach is 100 % for high-quality images, 92.31 % for low-quality images and 95.76 % of Yale face database when two images per person are trained.

As a conclusion for analysis, the extraction of facial feature could be challenging especially in different lighting. In pre-processing stage, Contrast Limited Adaptive Histogram Equalization (CLAHE) able to reduce the illumination effect. CLAHE perform better compared to histogram equalization in terms of contrast improvement. Enhanced LBP with larger radius sizes specifically, radius size two, perform better compared to original LBP operator, with less affected by illumination and more consistent compared to other radius sizes.

References:

1. Fussell, S. (2018). NEWS Facebook's new face recognition features: What we do (and don't) know. Gizmodo. Retrieved from https://gizmodo.com/facebooks-new-face-recognition-features-what-we-do-an-1823359911
2. DuVal, A. (2012). Face recognition software - History of forensic psychology. Forensic Psychology. Retrieved from http://forensicpsych.umwblogs.org/research/criminal-justice/face-recognition-software/
3. G.R, P., et al. (2010). An automated attendance system using image processing. International Journal of Advanced Networking & Applications.
4. Devan, P. A. M., et al. (2017). Smart attendance system using face recognition. Advances in Natural and Applied Sciences, 11 (7), 139-144.
5. Shirodkar, M. (2015). Automated attendance management system using face recognition. International Journal of Computer Applications and International Conference and Workshop on Emerging Trends in Technology.

6. Paul, L. C., & Al Sumam, A. (2012). Face recognition using principal component analysis method. IJARCET, V1(9).

FACE SENSE WEB CAMERA-BASED ATTENDANCE MONITORING WITH PYTHON AND ML

1.Prof. Prashant M. Palkar, 2. Prof. Sagar V. Shetage 3. Prof. Shashikant Patil, 4. Prof. Shrivijay A. Chavan, 5. Prof. Sanyogita S. Warake (Assistant Professor, Shahid Virpatni Laxmi Mahavidyalaya, Titave)

Abstract:

In an era where having a computer sing of your operations is a matter of prestige for some and necessity for others. Management of these projects is still an area which is ignored or done in traditional manner. We in a system called Face Sense Web Camera-based Attendance Monitoring with Python and ML is integrated and automation software for cable operators. Cable operators will provide distributed channels to their customers.

Introduction:

The main objective of this project is to develop face recognition based automated student attendance system. To achieve better performance, the test images and training images of this proposed approach are limited to frontal and upright facial images that consist of a single face only. The test images and training images must be captured by using the same device to ensure no quality difference. In addition, the students must register in the database to be recognized. The enrolment can be done on the spot through the user-friendly interface.

Language Learning:

Python: Python is the primary programming language used for developing the system. You'll need to be familiar with basic Python syntax, data structures, and libraries.

Machine Learning: You'll need to learn the basics of machine learning, including data preprocessing, model training, and evaluation. Specifically, you'll use machine learning.

CASE study of Face Sense Web Camera-based Attendance Monitoring with Python and ML:

Problem Statement:

A university wants to automate its attendance tracking system to improve accuracy and efficiency. They currently use manual methods that are prone to errors and time-consuming. The university wants a solution that can automatically recognize students' faces using a webcam and mark their attendance accordingly.

Solution Overview:

A face recognition system will be developed using Python and machine learning. The system will capture images from a webcam, detect faces in the images, and match them against a database of known students to mark attendance. The system will be trained on a dataset of students' faces to recognize them accurately.

Technologies Used:

Python: For developing the face recognition system and integrating it with the webcam.

OpenCV: For capturing images from the webcam and detecting faces.

Dlib: For facial landmark detection and face recognition.

Machine Learning: For training the face recognition model on a dataset of students' faces.

Implementation Steps:

Data Collection: Collect a dataset of students' faces. Each student should have multiple images taken from different angles and lighting conditions.

Preprocessing: Preprocess the images to improve the quality of the data. This may include resizing, normalization, and grayscale conversion.

Face Detection: Use OpenCV to detect faces in the images. This step is necessary to isolate the faces from the rest of the image.

Facial Landmark Detection: Use Dlib to detect facial landmarks (e.g., eyes, nose, mouth) in the detected faces. This step helps in aligning the faces for better recognition accuracy.

Feature Extraction: Extract features from the detected faces. This step converts the faces into a format that can be used for training the machine learning model.

Model Training: Train a machine learning model (e.g., SVM, KNN) on the extracted features. The model should be trained to recognize the faces of known students.

Attendance Marking: In real-time, capture images from the webcam, detect faces, and use the trained model to recognize the faces. Mark the attendance of recognized students.

Integration: Integrate the face recognition system with the university's attendance tracking system. This may involve developing APIs or other interfaces for data exchange.

Results:

The face recognition system achieved an accuracy of over 90% in recognizing students' faces.

The automated attendance system reduced the time taken to mark attendance and improved accuracy compared to manual methods.

Future Enhancements:

Implementing a user interface for administrators to monitor attendance and manage the system.

Adding support for real-time notifications or alerts for attendance-related issues.

Integrating the system with other university systems for seamless data exchange.

By implementing this solution, the university was able to automate its attendance tracking system, leading to increased efficiency and accuracy.

Conclusion:

In this approach, a face recognition based automated student attendance system is thoroughly described. The proposed approach provides a method to identify the individuals by comparing their input image obtained from recording video frame with respect to train image. This proposed approach able to detect and localize face from an input facial image, which is obtained from the recording video frame. Besides, it provides a method in pre-processing stage to enhance the image contrast and reduce the illumination effect. Extraction of features from the facial image is performed by applying both LBP and PCA. The algorithm designed to combine LBP and PCA able to stabilize the system by giving consistent results. The accuracy of this proposed approach is 100 % for high-quality images, 92.31 % for low-quality images and 95.76 % of Yale face database when two images per person are trained.

As a conclusion for analysis, the extraction of facial feature could be challenging especially in different lighting. In pre-processing stage, Contrast Limited Adaptive Histogram Equalization (CLAHE) able to reduce the illumination effect. CLAHE perform better compared to histogram equalization in terms of contrast improvement. Enhanced LBP with larger radius sizes specifically, radius size two, perform better compared to original LBP operator, with less affected by illumination and more consistent compared to other radius sizes.

References:

1. Fussell, S. (2018). NEWS Facebook's new face recognition features: What we do (and don't) know. Gizmodo. Retrieved from https://gizmodo.com/facebooks-new-face-recognition-features-what-we-do-an-1823359911
2. DuVal, A. (2012). Face recognition software - History of forensic psychology. Forensic Psychology. Retrieved from http://forensicpsych.umwblogs.org/research/criminal-justice/face-recognition-software/
3. G.R, P., et al. (2010). An automated attendance system using image processing. International Journal of Advanced Networking & Applications.
4. Devan, P. A. M., et al. (2017). Smart attendance system using face recognition. Advances in Natural and Applied Sciences, 11 (7), 139-144.
5. Shirodkar, M. (2015). Automated attendance management system using face recognition. International Journal of Computer Applications and International Conference and Workshop on Emerging Trends in Technology.

6. Paul, L. C., & Al Sumam, A. (2012). Face recognition using principal component analysis method. IJARCET, V1(9).

FACE SENSE WEB CAMERA-BASED ATTENDANCE MONITORING WITH PYTHON AND ML

1.Prof. Prashant M. Palkar, 2. Prof. Sagar V. Shetage 3. Prof. Shashikant Patil, 4. Prof. Shrivijay A. Chavan, 5. Prof. Sanyogita S. Warake (Assistant Professor, Shahid Virpatni Laxmi Mahavidyalaya, Titave)

Abstract:

In an era where having a computer sing of your operations is a matter of prestige for some and necessity for others. Management of these projects is still an area which is ignored or done in traditional manner. We in a system called Face Sense Web Camera-based Attendance Monitoring with Python and ML is integrated and automation software for cable operators. Cable operators will provide distributed channels to their customers.

Introduction:

The main objective of this project is to develop face recognition based automated student attendance system. To achieve better performance, the test images and training images of this proposed approach are limited to frontal and upright facial images that consist of a single face only. The test images and training images must be captured by using the same device to ensure no quality difference. In addition, the students must register in the database to be recognized. The enrolment can be done on the spot through the user-friendly interface.

Language Learning:

Python: Python is the primary programming language used for developing the system. You'll need to be familiar with basic Python syntax, data structures, and libraries.

Machine Learning: You'll need to learn the basics of machine learning, including data preprocessing, model training, and evaluation. Specifically, you'll use machine learning.

CASE study of Face Sense Web Camera-based Attendance Monitoring with Python and ML:

Problem Statement:

A university wants to automate its attendance tracking system to improve accuracy and efficiency. They currently use manual methods that are prone to errors and time-consuming. The university wants a solution that can automatically recognize students' faces using a webcam and mark their attendance accordingly.

Solution Overview:

A face recognition system will be developed using Python and machine learning. The system will capture images from a webcam, detect faces in the images, and match them against a database of known students to mark attendance. The system will be trained on a dataset of students' faces to recognize them accurately.

Technologies Used:

Python: For developing the face recognition system and integrating it with the webcam.

OpenCV: For capturing images from the webcam and detecting faces.

Dlib: For facial landmark detection and face recognition.

Machine Learning: For training the face recognition model on a dataset of students' faces.

Implementation Steps:

Data Collection: Collect a dataset of students' faces. Each student should have multiple images taken from different angles and lighting conditions.

Preprocessing: Preprocess the images to improve the quality of the data. This may include resizing, normalization, and grayscale conversion.

Face Detection: Use OpenCV to detect faces in the images. This step is necessary to isolate the faces from the rest of the image.

Facial Landmark Detection: Use Dlib to detect facial landmarks (e.g., eyes, nose, mouth) in the detected faces. This step helps in aligning the faces for better recognition accuracy.

Feature Extraction: Extract features from the detected faces. This step converts the faces into a format that can be used for training the machine learning model.

Model Training: Train a machine learning model (e.g., SVM, KNN) on the extracted features. The model should be trained to recognize the faces of known students.

Attendance Marking: In real-time, capture images from the webcam, detect faces, and use the trained model to recognize the faces. Mark the attendance of recognized students.

Integration: Integrate the face recognition system with the university's attendance tracking system. This may involve developing APIs or other interfaces for data exchange.

Results:

The face recognition system achieved an accuracy of over 90% in recognizing students' faces.

The automated attendance system reduced the time taken to mark attendance and improved accuracy compared to manual methods.

Future Enhancements:

Implementing a user interface for administrators to monitor attendance and manage the system.

Adding support for real-time notifications or alerts for attendance-related issues.

Integrating the system with other university systems for seamless data exchange.

By implementing this solution, the university was able to automate its attendance tracking system, leading to increased efficiency and accuracy.

Conclusion:

In this approach, a face recognition based automated student attendance system is thoroughly described. The proposed approach provides a method to identify the individuals by comparing their input image obtained from recording video frame with respect to train image. This proposed approach able to detect and localize face from an input facial image, which is obtained from the recording video frame. Besides, it provides a method in pre-processing stage to enhance the image contrast and reduce the illumination effect. Extraction of features from the facial image is performed by applying both LBP and PCA. The algorithm designed to combine LBP and PCA able to stabilize the system by giving consistent results. The accuracy of this proposed approach is 100 % for high-quality images, 92.31 % for low-quality images and 95.76 % of Yale face database when two images per person are trained.

As a conclusion for analysis, the extraction of facial feature could be challenging especially in different lighting. In pre-processing stage, Contrast Limited Adaptive Histogram Equalization (CLAHE) able to reduce the illumination effect. CLAHE perform better compared to histogram equalization in terms of contrast improvement. Enhanced LBP with larger radius sizes specifically, radius size two, perform better compared to original LBP operator, with less affected by illumination and more consistent compared to other radius sizes.

References:

1. Fussell, S. (2018). NEWS Facebook's new face recognition features: What we do (and don't) know. Gizmodo. Retrieved from https://gizmodo.com/facebooks-new-face-recognition-features-what-we-do-an-1823359911
2. DuVal, A. (2012). Face recognition software - History of forensic psychology. Forensic Psychology. Retrieved from http://forensicpsych.umwblogs.org/research/criminal-justice/face-recognition-software/
3. G.R, P., et al. (2010). An automated attendance system using image processing. International Journal of Advanced Networking & Applications.
4. Devan, P. A. M., et al. (2017). Smart attendance system using face recognition. Advances in Natural and Applied Sciences, 11 (7), 139-144.
5. Shirodkar, M. (2015). Automated attendance management system using face recognition. International Journal of Computer Applications and International Conference and Workshop on Emerging Trends in Technology.

6. Paul, L. C., & Al Sumam, A. (2012). Face recognition using principal component analysis method. IJARCET, V1(9).

FACE SENSE WEB CAMERA-BASED ATTENDANCE MONITORING WITH PYTHON AND ML

1.Prof. Prashant M. Palkar, 2. Prof. Sagar V. Shetage 3. Prof. Shashikant Patil, 4. Prof. Shrivijay A. Chavan, 5. Prof. Sanyogita S. Warake (Assistant Professor, Shahid Virpatni Laxmi Mahavidyalaya, Titave)

Abstract:

In an era where having a computer sing of your operations is a matter of prestige for some and necessity for others. Management of these projects is still an area which is ignored or done in traditional manner. We in a system called Face Sense Web Camera-based Attendance Monitoring with Python and ML is integrated and automation software for cable operators. Cable operators will provide distributed channels to their customers.

Introduction:

The main objective of this project is to develop face recognition based automated student attendance system. To achieve better performance, the test images and training images of this proposed approach are limited to frontal and upright facial images that consist of a single face only. The test images and training images must be captured by using the same device to ensure no quality difference. In addition, the students must register in the database to be recognized. The enrolment can be done on the spot through the user-friendly interface.

Language Learning:

Python: Python is the primary programming language used for developing the system. You'll need to be familiar with basic Python syntax, data structures, and libraries.

Machine Learning: You'll need to learn the basics of machine learning, including data preprocessing, model training, and evaluation. Specifically, you'll use machine learning.

CASE study of Face Sense Web Camera-based Attendance Monitoring with Python and ML:

Problem Statement:

A university wants to automate its attendance tracking system to improve accuracy and efficiency. They currently use manual methods that are prone to errors and time-consuming. The university wants a solution that can automatically recognize students' faces using a webcam and mark their attendance accordingly.

Solution Overview:

A face recognition system will be developed using Python and machine learning. The system will capture images from a webcam, detect faces in the images, and match them against a database of known students to mark attendance. The system will be trained on a dataset of students' faces to recognize them accurately.

Technologies Used:

Python: For developing the face recognition system and integrating it with the webcam.

OpenCV: For capturing images from the webcam and detecting faces.

Dlib: For facial landmark detection and face recognition.

Machine Learning: For training the face recognition model on a dataset of students' faces.

Implementation Steps:

Data Collection: Collect a dataset of students' faces. Each student should have multiple images taken from different angles and lighting conditions.

Preprocessing: Preprocess the images to improve the quality of the data. This may include resizing, normalization, and grayscale conversion.

Face Detection: Use OpenCV to detect faces in the images. This step is necessary to isolate the faces from the rest of the image.

Facial Landmark Detection: Use Dlib to detect facial landmarks (e.g., eyes, nose, mouth) in the detected faces. This step helps in aligning the faces for better recognition accuracy.

Feature Extraction: Extract features from the detected faces. This step converts the faces into a format that can be used for training the machine learning model.

Model Training: Train a machine learning model (e.g., SVM, KNN) on the extracted features. The model should be trained to recognize the faces of known students.

Attendance Marking: In real-time, capture images from the webcam, detect faces, and use the trained model to recognize the faces. Mark the attendance of recognized students.

Integration: Integrate the face recognition system with the university's attendance tracking system. This may involve developing APIs or other interfaces for data exchange.

Results:

The face recognition system achieved an accuracy of over 90% in recognizing students' faces.

The automated attendance system reduced the time taken to mark attendance and improved accuracy compared to manual methods.

Future Enhancements:

Implementing a user interface for administrators to monitor attendance and manage the system.

Adding support for real-time notifications or alerts for attendance-related issues.

Integrating the system with other university systems for seamless data exchange.

By implementing this solution, the university was able to automate its attendance tracking system, leading to increased efficiency and accuracy.

Conclusion:

In this approach, a face recognition based automated student attendance system is thoroughly described. The proposed approach provides a method to identify the individuals by comparing their input image obtained from recording video frame with respect to train image. This proposed approach able to detect and localize face from an input facial image, which is obtained from the recording video frame. Besides, it provides a method in pre-processing stage to enhance the image contrast and reduce the illumination effect. Extraction of features from the facial image is performed by applying both LBP and PCA. The algorithm designed to combine LBP and PCA able to stabilize the system by giving consistent results. The accuracy of this proposed approach is 100 % for high-quality images, 92.31 % for low-quality images and 95.76 % of Yale face database when two images per person are trained.

As a conclusion for analysis, the extraction of facial feature could be challenging especially in different lighting. In pre-processing stage, Contrast Limited Adaptive Histogram Equalization (CLAHE) able to reduce the illumination effect. CLAHE perform better compared to histogram equalization in terms of contrast improvement. Enhanced LBP with larger radius sizes specifically, radius size two, perform better compared to original LBP operator, with less affected by illumination and more consistent compared to other radius sizes.

References:

1. Fussell, S. (2018). NEWS Facebook's new face recognition features: What we do (and don't) know. Gizmodo. Retrieved from https://gizmodo.com/facebooks-new-face-recognition-features-what-we-do-an-1823359911

2. DuVal, A. (2012). Face recognition software - History of forensic psychology. Forensic Psychology. Retrieved from http://forensicpsych.umwblogs.org/research/criminal-justice/face-recognition-software/

3. G.R, P., et al. (2010). An automated attendance system using image processing. International Journal of Advanced Networking & Applications.

4. Devan, P. A. M., et al. (2017). Smart attendance system using face recognition. Advances in Natural and Applied Sciences, 11 (7), 139-144.

5. Shirodkar, M. (2015). Automated attendance management system using face recognition. International Journal of Computer Applications and International Conference and Workshop on Emerging Trends in Technology.

6. Paul, L. C., & Al Sumam, A. (2012). Face recognition using principal component analysis method. IJARCET, V1(9).