SMART MIRROR

Arpit Kansal, Madhav Maheshwari, Nikita Singh

School of computer science(IoT)

***Abstract*—*This paper presents the development and implementation of a smart mirror that integrates facial recognition technology to provide personalized information for each user. Built on a Raspberry Pi 4 with a two-way mirror and display, this project aims to enhance the daily user experience through the seamless interaction of IoT devices. The Magic Mirror software framework is utilized to deliver personalized content such as weather, time, and calendars, based on recognized user profiles. The system has a broad range of applications, particularly in smart homes, public places, and commercial spaces. Future development includes expanding functionality through mood detection and further customization for diverse applications.*

1. Introduction

Smart mirrors represent a growing field of innovation within the Internet of Things (IoT) ecosystem. They transform a traditionally passive device into an interactive system capable of providing real-time information to users. By integrating facial recognition technology, smart mirrors become more than simple reflective surfaces; they become dynamic interfaces that deliver customized content to individual users.

The purpose of this project is to explore the possibilities of enhancing personal mirrors with smart technology, creating a more interactive and useful device for everyday life. Traditional mirrors only serve as a reflection tool, but smart mirrors can provide a seamless flow of information without requiring active user input, making them highly suited for smart homes, offices, hotels, and even public spaces.

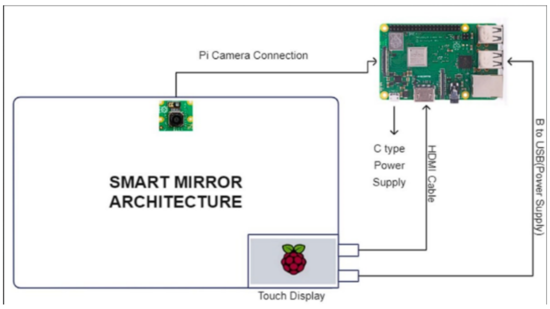

Figure 1. *Project Layout*

1. Problem Statement: As smart home technology becomes increasingly prevalent, there is a growing need for devices that can provide personal and context-aware experiences. A mirror, being an object used by almost everyone daily, offers a unique opportunity for such a transformation. This project aims to provide a practical implementation of a smart mirror that recognizes different users and displays personalized information without requiring manual inputs, adding convenience and efficiency to daily routines.
2. Applications: Smart mirrors, as demonstrated in this project, are applicable in various environments:
    (a) Homes: Providing real-time updates (e.g., weather, news, to-do lists) tailored to each family member.
    (b) Offices: Displaying meeting schedules or reminders for employees.
    (c) Hotels: Offering personalized greetings and suggestions based on guest preferences.
    (d) Public Spaces: Enhancing customer engagement with personalized advertisements or information in malls, airports, and stores.

2. Literature Review

Smart mirrors have been explored in the research community, particularly in the context of smart homes and IoT. Early versions of smart mirrors focused on displaying static information such as the time or weather. However, with advancements in AI and IoT technologies, smart mirrors now have the potential to deliver dynamic, personalized content.

- Existing Smart Mirror Technologies: Various smart mirror systems have been developed in recent years. Most commercial products focus on providing information via touch or voice interaction. For example, products like Google's Home Hub and other similar smart devices can display information when prompted by voice commands. However, they lack advanced personalization features like facial recognition.
- Facial Recognition in IoT Devices: Facial recognition technology has gained popularity across different domains, from security systems to smartphones. It offers a unique and secure way of identifying users, making it an ideal technology for personalizing content in smart home devices. Recent advancements in machine learning and AI, particularly deep learning models, have significantly improved the accuracy and reliability of facial recognition systems.
- Challenges in Smart Mirror Development: Although smart mirrors are gaining attention, there are several challenges to be addressed:
    1. Privacy: Facial recognition systems need to ensure data privacy and security to avoid unauthorized access or misuse of personal information.
    2. User Experience: Creating a smooth, intuitive interface that feels natural for users is crucial.
    3. Scalability: Developing a system that can handle multiple users efficiently without significant performance loss.

3. System Architecture

The smart mirror system consists of both hardware and software components working together to provide a seamless user experience. The architecture was designed to be modular, allowing easy integration of new features in the future, such as mood detection.

1. Hardware Components:
    - Raspberry Pi 4: The Raspberry Pi 4 serves as the core processing unit of the smart mirror. It was selected

due to its affordability, flexibility, and sufficient computational power for facial recognition tasks. The Raspberry Pi operates as the system's central hub, coordinating the flow of data between the display and the facial recognition algorithm.
- Two-Way Mirror: A two-way mirror was chosen to maintain the traditional reflective functionality of a mirror while allowing the display of digital content behind it. This mirror is mounted on a standard LCD display, which shows personalized information to the user.
- LCD Display: Behind the two-way mirror is an LCD display that shows digital content, including weather updates, calendars, and other personalized data. The display remains hidden when not in use, making the device look like a regular mirror.

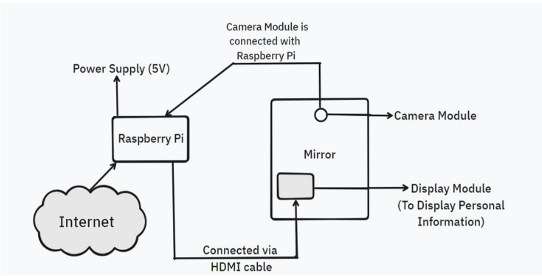

Figure 2. *Project Layout*

2. Software Components:
   - Magic Mirror Software Framework: The open-source Magic Mirror software framework serves as the backbone of the user interface. It allows for easy customization of the display and provides various modules that show information like time, weather, calendar, and news. The framework is highly modular, enabling developers to integrate additional functionality such as facial recognition and mood detection seamlessly.
   - Facial Recognition Module: A facial recognition algorithm was implemented using OpenCV and Python. The system identifies users by comparing facial features to pre-registered profiles. The Raspberry Pi processes the facial recognition locally to ensure data privacy, with all user profiles stored securely on the device.
3. System Flow:
   - When the user stands in front of the mirror, their face is detected by the facial recognition module.
   - The system matches the detected face with stored profiles and identifies the user.
   - Once identified, the smart mirror displays personalized information on the screen, such as the user's calendar, weather forecast, and to-do lists.
   - The mirror can handle multiple user profiles, automatically switching the displayed content based on the recognized user.

4. Methodology

The smart mirror system was developed following a systematic approach to ensure robust performance and a smooth user experience. The methodology includes the processes of facial recognition and data display.

- Facial Recognition Process: The system uses a Python-based facial recognition library integrated with the Magic Mirror framework. The process starts with users registering their facial profiles by taking several pictures from different angles. These images are processed and stored locally on the Raspberry Pi. Once the profiles are set, the system matches users' faces with stored images, identifying them accurately even with small variations in lighting or facial expressions.
  The system avoids live video capture and instead relies on static image matching, where user profiles are created based on a set of images during the initial setup. This ensures privacy and simplicity, eliminating the need for continuous video streaming.
- Data Display Customization: Once a user is identified, personalized content is pulled from various modules within the Magic Mirror software. For instance:
  1. The weather module retrieves location-based weather data.
  2. The calendar module accesses the user's Google or local calendar to display upcoming events.
  3. The news module provides personalized news feeds, based on user preferences.

  The data is displayed in a clean, user-friendly manner, ensuring that the user can gather all necessary information at a glance.
- User Registration and Data Privacy: Users must register their facial data by capturing a set of static images during the system's setup phase. The data is stored securely on the Raspberry Pi with encryption to ensure privacy. No cloud services are used for storing facial data, minimizing security risks and ensuring that user data stays local.

5. Data Structures:

1. For storing the facial data and PCA components, you can use a dictionary data structure in Python. Here's how you can structure it:

```python
# Dictionary to store facial data and PCA components
facial_data = {
    "faces": [],          # List to store preprocessed facial images
    "labels": [],         # List to store labels (identifiers) associated with each face
    "mean_face": None,    # Mean face vector
    "eigenfaces": [],     # List to store top-k eigenfaces
    "projection": []      # List to store projected faces onto the eigenfaces
}
```

2. In this dictionary: "faces" would store preprocessed facial images. "labels" would store unique identifiers associated with each face. "meanface" would store the mean face vector calculated during PCA. "eigenfaces" would store the top-k eigenfaces obtained from PCA. "projection" would store the projected faces onto the eigenfaces.
3. Using a dictionary allows you to easily access and manipulate different components of the facial recognition

system, making it convenient for implementation on the Raspberry Pi using Python.

6. Data structure and algorithm

1. Image Preprocessing:
   - Grayscale Conversion: The RGB image is converted to grayscale without any specific formula.
   - Resizing: The image is resized to a predefined width and height without any specific formula.
   - Normalization: The formula for normalization is: $\text{Normalized\_Pixel\_Value} = \frac{\text{Pixel\_Value}}{255}$, assuming pixel values range from 0 to 255.
2. Feature Extraction:
   - Haar Cascades: Various mathematical calculations based on Haar wavelets are utilized for feature detection.
3. Classification:
   - Template Matching: Pixel values of the detected facial features are compared with templates of known faces without any specific formula.
   - Nearest Neighbor Classification (Euclidean Distance): The formula for Euclidean distance is: $d(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$.
4. Thresholding:

- Distance Thresholding: The match function is defined as: $\text{Match}(d) = \begin{cases} \text{True}, & \text{if } d < \text{Threshold} \\ \text{False}, & \text{otherwise} \end{cases}$.

7. Results and Implementation

The smart mirror was tested under a variety of conditions to ensure robustness, usability, and accuracy.

1. Facial Recognition Performance:
   - The system demonstrated a facial recognition accuracy of approximately 92percent in controlled environments. Factors such as consistent lighting and clear visibility of facial features contributed to this high accuracy.
   - In situations with varying lighting conditions, the system still performed effectively, with only a minor drop in accuracy. These results indicate the potential of the system to operate efficiently in real-world conditions.
2. User Experience Feedback: The mirror was tested with several users, each registering their profiles. Users reported a positive experience, noting that the system quickly recognized them and displayed the correct personalized information. The two-way mirror design was appreciated, as it maintained the appearance of a traditional mirror when the display was inactive.
3. Applications in Smart Homes and Public Spaces: In a smart home setting, the mirror functioned as a central hub for personal information. Users could check their schedules, receive reminders, and view weather updates without having to interact with any other device. The smart mirror demonstrated its usefulness in enhancing day-to-day routines.
   In commercial settings, the mirror's ability to provide personalized content makes it an attractive solution for businesses looking to engage customers. Hotels, for example, could use smart mirrors in guest rooms to offer customized greetings, information, and services tailored to each guest.

Figure 3. *Smart Mirror*

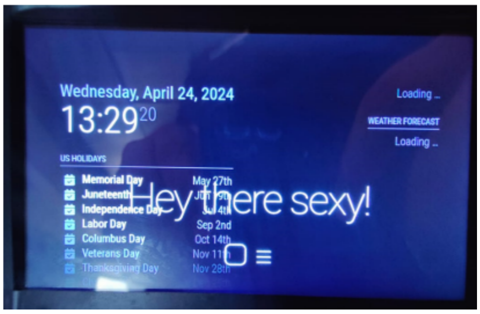

Figure 4. *Display*

8. Working

- Start:This is the initial state where the smart mirror system is powered on and begins its operations. It indicates the beginning of the process flow.
- Camera Activation: The Pi Camera module is activated to start capturing images. This step ensures that the system is ready to detect and recognize faces as users approach the mirror. The camera may be continuously active or triggered by a motion sensor, depending on your setup.
- Capture Image: The camera captures an image of the person standing in front of the mirror. This image will be used for facial recognition to identify the user and provide personalized content.
- Facial Recognition Process: The captured image is processed using facial recognition software (such as OpenCV). The system analyzes the image to detect and identify facial features. This step involves running the image through a pre-trained model to determine if the face matches any stored profiles.
- Face Detected?: This decision point checks if the facial recognition system has successfully detected a face in the captured image.
  Yes: If a face is detected, the system proceeds to the next step to identify the user.

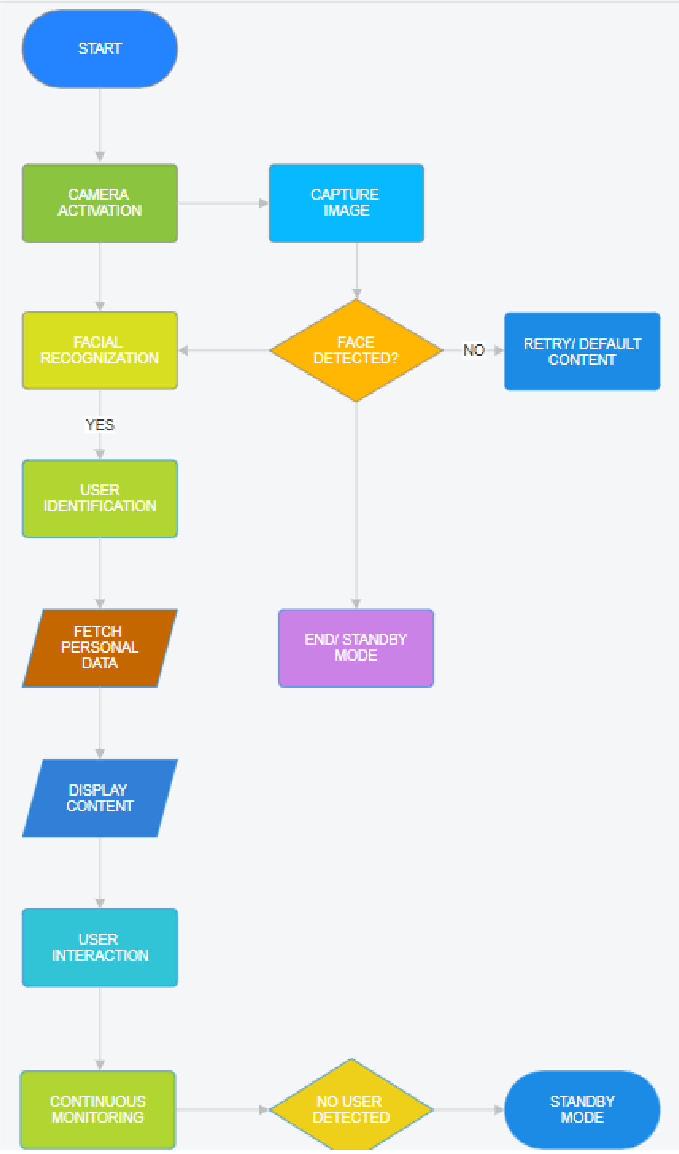

Figure 5. *Flowchart*

No: If no face is detected, the system may retry capturing another image or display default content if no recognized user is present.

- User Identification: If a face is detected, the system attempts to identify the user by matching the detected face with stored profiles. This involves comparing the facial features from the captured image to those in the user database.
- Fetch Personalized Data: Once the user is identified, the system retrieves personalized data based on the user's profile. This can include weather updates, news headlines, calendar events, and other relevant information tailored to the user.
- Display Content on Mirror: The personalized content retrieved is then displayed on the mirror's screen. This is where the user interacts with the smart mirror and sees real-time information.
- User Interaction: If the smart mirror supports user interaction, this step allows the user to interact with the mirror through various means, such as voice commands, gestures, or touch. This interaction can enable the user to request additional information or control certain aspects of the mirror.
- Continuous Monitoring: The system continuously monitors for the presence of the user. This ensures that the mirror remains active and responsive while the user is in front of it. If no user is detected, the system prepares to enter standby mode.
- No User Detected?: This decision point determines if the user is still present in front of the mirror.
  Yes: If the user is still present, the system continues monitoring and displaying content.
  No: If no user is detected, the system proceeds to enter standby mode to save power.
- End / Standby Mode: When the system detects that no user is present, it enters standby mode. In this state, the mirror may turn off the display or enter a low-power mode to conserve energy until a new user approaches and the cycle starts again.

9. Applications and Benefits

The smart mirror project has several potential applications, particularly in the areas of smart homes, smart cities, and customer engagement.

1. In Smart Cities:
   - Public Spaces: In malls or airports, smart mirrors can serve as interactive kiosks. Facial recognition can personalize the displayed content, offering advertisements or information based on the user.
   - Workplaces: Smart mirrors can display personalized schedules or announcements in offices or coworking spaces, enhancing productivity.
2. In Daily Life:
   - Health Monitoring: Future integrations with health-related sensors could allow smart mirrors to display health metrics (e.g., heart rate, sleep patterns)

10. Conclusion

The smart mirror project successfully demonstrates the potential of integrating modern technology into daily life. By combining facial recognition with real-time information display, the smart mirror provides a seamless, hands-free interface for users to access personalized content, such as weather updates, calendars, and news. Its use of a Raspberry Pi 4 and Magic Mirror software shows how affordable and scalable solutions can be implemented for smart home applications.
This project also lays the foundation for further enhancements, such as the mood detection system, which could open up new possibilities for personalized user experiences. In the context of smart cities, the smart mirror offers benefits in areas like home automation, personalized healthcare, and efficient living, contributing to the broader goal of connected and intelligent urban environments.
Overall, the project highlights the viability of smart mirrors as practical, user-friendly devices that can enhance both daily life and the future of smart city infrastructures.

11. References

- Yusri, M. M. I., Kasim, S., Hassan, R., Abdullah, Z., Ruslai, H., Jahidin, K., Arshad, M. S. (2017, May). Smart mirror for smart life. In 2017 6th ICT International Student Project Conference (ICT-ISPC) (pp. 1-5). IEEE.
- Johri, A., Jafri, S., Wahi, R. N., Pandey, D. (2018, December). Smart mirror: A time-saving and affordable assistant. In 2018 4th International Conference on Computing Communication and Automation (ICCCA) (pp. 1-4). IEEE.
- Sun, Y., Geng, L., Dan, K. (2018, January). Design of smart mirror based on Raspberry Pi. In 2018 International Conference on Intelligent Transportation, Big Data Smart City (ICITBS) (pp. 77-80). IEEE.
- Patel, M. N., Shabandri, K. F., Balse, G., Naik, S. (2019). Smart Mirror. International Journal of Engineering Research and Technology (IJERT), 8(05).
- Sun, Y., Geng, L., Dan, K. (2018, January). Design of smart mirror based on Raspberry Pi. In 2018 International Conference on Intelligent Transportation, Big Data Smart City (ICITBS) (pp. 77-80). IEEE.
- Kumar, A., Singh, J. (2018, August). Smart Mirror with Advanced Features and IoT Integration. In 2018 International Conference on Computer Communication and Informatics (ICCCI) (pp. 1-6). IEEE.
- Yusof, Y., Kamarudin, K. (2020, July). Implementation of Smart Mirror System using Raspberry Pi and Facial Recognition.