Mirror Mirror on the Wall

Corin Cho, Isaiah Weekes, Siena Lee
Department of Electrical and Computer Engineering, Carnegie Mellon University

Abstract— Mirror mirror on the wall is a smart mirror system designed to provide affordable, accurate, and private skin condition detection and skincare recommendations. Comprising a single-board computer, camera, and display beneath the mirror, the system aims to analyze four common skin conditions (acne, oiliness, sun burn, wrinkles) with high accuracy. Combining accessible hardware, locally processed machine learning models, and user-centered design, the smart mirror seamlessly delivers skin health insights as part of the user's daily routine.

Index Terms—Smart mirror, skin analysis, machine learning, computer vision, user privacy, skincare recommendation, embedded systems

I. Introduction

Skin conditions change regularly due to factors such as weather, diet, or stress, and it is extremely difficult for people to consistently track and manage their skin. Common issues like acne, oiliness, sun burn, and wrinkles tend to vary day by day, and most people do not have the expertise to determine which skincare products are optimal for their specific skin condition. The difficulty in analyzing their skin and choosing appropriate products can leave individuals frustrated about taking care of their skin, discouraging effective long term skincare habits.

Mirror Mirror on the Wall is an accessible, personalized smart mirror that seamlessly integrates into a user's daily routine and guides users to better manage and understand their skin. While a user is in front of our mirror to check their face, the user can press on the touchscreen to start a skin analysis session. Within seconds, a camera captures an image of the user's skin, a single-board computer processes the image using machine learning, and personalized skincare recommendations are generated based on classification results and confidence levels. Our recommendation system is based on general skincare ingredients, developed through discussions with a dermatologist. By mapping skin conditions and confidence levels to commonly accepted active ingredients, the system provides consistent and appropriate skincare guidance during each session. Once the analysis is complete, the results and product suggestions are displayed on a touchscreen beneath the mirror, allowing users to view both their reflection and the feedback simultaneously.

Existing alternatives to our product include expensive dermatologist consultations, generic skincare apps[15], or cloud-based analysis tools[7] that may raise privacy concerns. These solutions are difficult for individuals to rely on consistently because they are costly, inefficient, generic, or potentially unsafe. Our system provides a lower-cost, private, and user-friendly alternative that makes skin analysis more accessible, efficient, and personalized. The primary goal of the product is to make skincare analysis a simple part of daily life while providing consistent, trustworthy feedback that help users select the most suitable product.

II. Use-Case Requirements

Our use-case requirements were designed with a goal of creating a system that makes skincare analysis accurate, accessible, and practical for everyday use. The requirements we defined ensure that our system not only provides reliable technical performance but also integrates into daily routines, encouraging users to engage with it consistently. The requirements are divided into four main categories: reliable image capture, accurate classification of skin conditions, low-latency skin analysis, and user interface.

A. Image Capture

The mirror must capture reliable images that provide sufficient detail for detecting small blemishes, faint wrinkles, and subtle textural differences. To achieve this, the camera must have resolution high enough to preserve fine-grained details when subsectioned into patches. Consistency in lighting is also essential, so the product requires integrated illumination that minimizes shadows and ensures uniform image quality. Overall, reliable image capture will be achieved through built-in LED lighting that maintains consistent conditions under different environments and a carefully selected camera that provides accurate input data.

B. Accurate Classification of Skin Conditions

The system must accurately classify four primary skin conditions: acne, oiliness, sunburn, and wrinkles. The product's primary goal is to provide personalized skincare recommendations using a local database which groups information by skin conditions and their corresponding confidence levels. To ensure reliable analysis, skin classification results must achieve an accuracy greater than 85% consistently. 85% accuracy represents a reasonable average/upper bound for most lightweight commercial vision models on common classification tasks. With high classification accuracy and confidence levels, the system ensures that the feedback provided to users is trustworthy and supports informed skincare decisions. The goal is for individuals to have confidence in the personalized feedback delivered by the mirror.

C. Low-latency Machine Learning Inference

To fit naturally into a user's daily routine, the system must deliver results quickly. From the moment the user initiates the analysis, the complete process–image capture, processing, classification, generating skin care recommendations, and displaying results–should finish in less than 5 seconds. This latency threshold was chosen because it's short enough to feel seamless, especially since the system doesn't require the user to remain still for the duration of the analysis.

D. User Experience

The mirror must present results in a clear, intuitive, and non-intrusive way. The user app should feature an intuitive layout that allows users to quickly understand and navigate the interface. A local database of skincare, organized by condition and model confidence, must provide tailored recommendations for each session. To improve user engagement, the system must also retain past analysis results, and display trends for at least the past 5 sessions, providing users insights into their skin health over time.

III. Architecture and/or Principle of Operation

As the MVP, Mirror Mirror on the Wall will 1) perform skin analysis on image captured 2) bring up the history of their skin condition 3) display recommendations that have been made in the past. The skin evaluation will be done through customized machine learning models. Predetermined treatment options will be provided based on the analysis result. Users can navigate through app features by the LCD that provides the user-friendly touchscreen interface. A touchscreen allows users to interact with the system quickly and silently, making it better suited for shared spaces such as bathrooms or bedrooms. Meanwhile, the embedded control block will conduct real-time interaction and updates between the system app, "Magic Mirror," and the hardware peripherals.

A. Principles of Engineering

Our project makes use of multiple individual sub-domains and subsystems, which come together via core engineering principles. The design of the mirror employs modularity. This allowed for the parallel development and implementation of multiple core components, such as the skin inference models and the core electronics layout. An additional engineering principle reflected in the system design, and one that guided its development, is the integration of stakeholder needs throughout all aspects of the system. The design requirements directly address core stakeholder concerns, being affordability, user convenience and accuracy, and ease of use.

B. Principles of Science

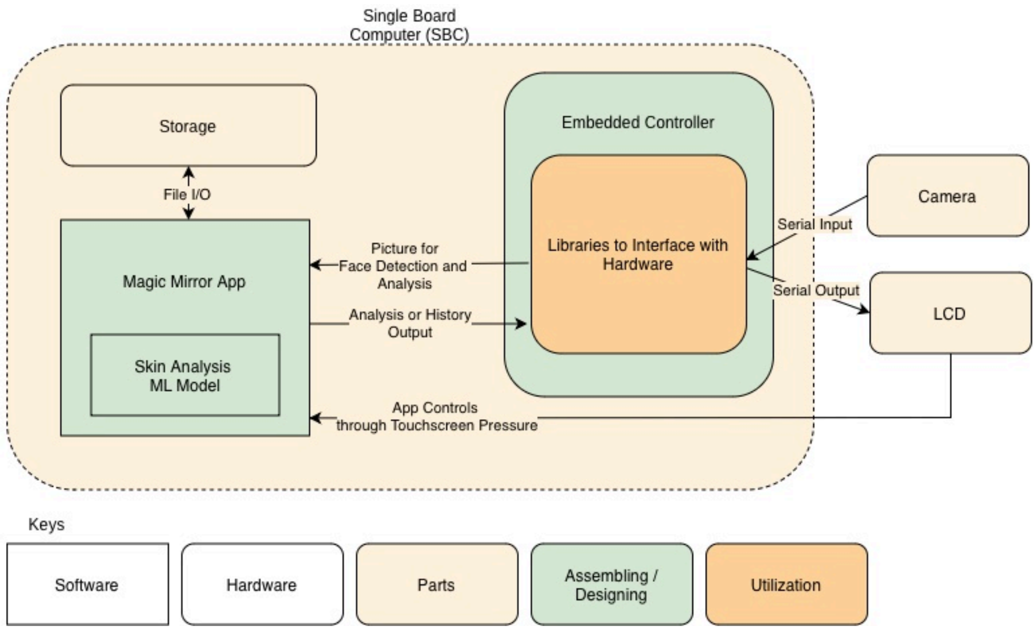
One of the core principles of science is iteration and evaluation. Referencing and expanding upon previous research is a core component of the scientific method. Our project made intentional use of previously existing knowledge and products to develop our own novel implementation. An example of this would be the skin oiliness classification model. The design and architecture of the model was inspired by existing research, however the referenced research paper only proposed the architecture as a solution to classifying highly zoomed-in images of skin. Our system used the same training, augmentation and architectural machine learning techniques, and applied them to the scenario where data is lower-resolution and higher variance. This allowed us to iterate on the existing body of knowledge and expand pre-existing methods into new domains.

C. Principles of Mathematics

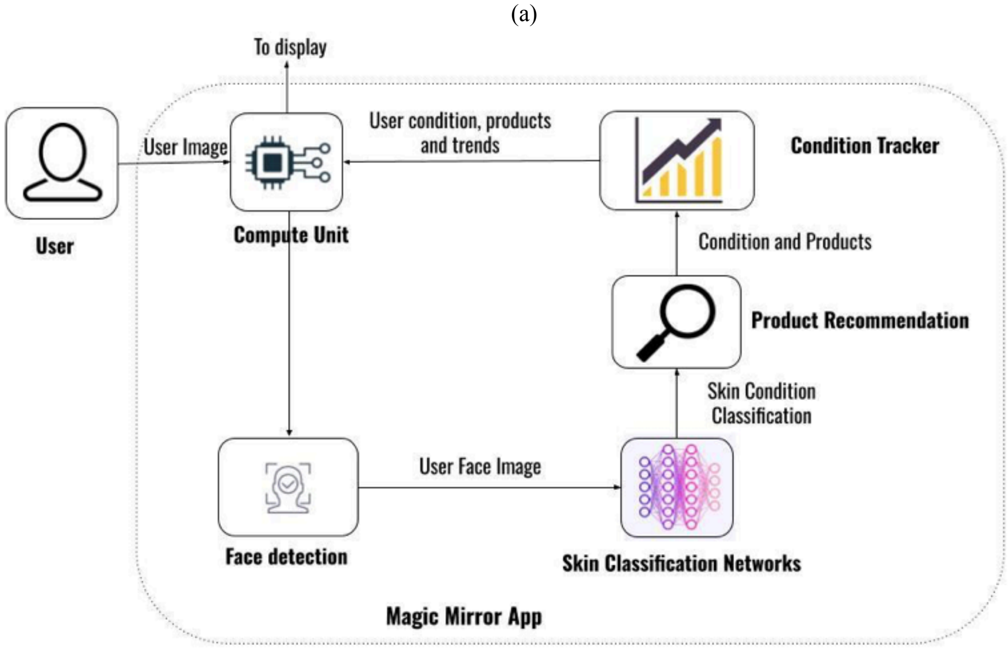
The core methods of skin inference and analysis for our system make heavy use of machine learning methods. In particular, deep learning systems utilize statistical inference methods to generate results. Training the skin classification systems involved creating and optimizing loss functions unique to each specific task. At a mathematical level, each model represents a function with learnable parameters:

$$f: R^{HxWxC} -> Y \qquad (1)$$

For the classification task, the output of the function is optimized with respect to a cross-entropy loss function, however this changes to a more complex joint-optimization problem for object detection. In particular, YOLO models use a composite loss function including bounding box regression loss, objectness loss and classification loss. Each model is optimized via mini-batch stochastic gradient descent. Ultimately, the principles of mathematics are critical in developing the core skin inference system, utilizing statistics, calculus, linear algebra, and optimization.



(a)



(b)

Fig. 1. System drawing or rending (a) overall system (b) software system of the Magic Mirror App

IV. Design Requirements

The design requirements are also split into four main categories: reliable image capture, accurate classification of skin conditions, low-latency machine learning inference, and user interface. Each requirement is mapped to its corresponding use-case requirement and includes specific technical considerations.

A. Image Capture

The camera must capture images at a resolution of at least 16 megapixels to ensure sufficient detail for detecting small blemishes, faint wrinkles, and subtle textural differences. The captured images will be downsampled and subsectioned into 224 × 224 pixel inputs for the model, while retaining sufficient detail to preserve key visual features. DeepFace, a lightweight, real-time face detection model will be used to locate the face for image capture when the user initiates a session by pressing the button. We want to ensure that DeepFace accurately detects the location of the face with a 95% or higher accuracy, so that the face is captured for input data. Also, we want the image captures to be under optimal lighting conditions, which will be controlled by a LED system. Together, these requirements ensure reliable, high quality input data.

B. Accurate Classification of Skin Conditions

To achieve at least 85% accuracy (a reasonable average/upper bound for most lightweight commercial vision models on common classification tasks) per class on the test datasets, our model will be trained with a dataset large enough to represent variations in skin tone, lighting, and condition severity. To maximize efficiency on edge devices, our design requirements include the use of lightweight architectures, balancing speed and accuracy through depth wise separable convolution. The model architecture will provide sufficient capacity to meet accuracy targets while remaining small enough to be able to function on a single-board computer. By leveraging this type of architecture, the system can perform inference with required latency constraints while preserving classification accuracy.

C. Low-latency Machine Learning Inference

The total latency of the entire system should be less than 5 seconds. The ML inference stage is anticipated to be the primary bottleneck and is required to complete in under 4 seconds while the communication between subsystems – including camera to processor and processor to display – must take less than 1 second.

Breaking it down into its constituent stages, we expect the following latency contributions: camera capture and transfer at less than 120mS, preprocessing operations at less than 100mS, face detection at approximately 100mS, local data base lookup at less than 100mS, user interface rendering with result composition at approximately 150mS, and display latency of less than 50mS. The total communication latency sums up to approximately 620mS, well within the 1 second budget for our subsystem interaction.

With this breakdown, the majority of the latency budget remains allocated to machine learning inference. We were able to achieve an inference time of less than 4 seconds.

D. User Interface

The system must present skin analysis results and corresponding skincare recommendations in a manner that requires minimum effort and is intuitive. A 7-inch display will be in the lower half of the mirror, ensuring that the output is clearly visible without being obstructive. The overall size of the mirror should be large enough for easy usability (16.25× 14.5 inches).

The application will also provide a history of past results, retaining and displaying data from at least the five most recent sessions. Upon completion of the analysis, the display will present classification results for each skin condition, along with tailored skincare recommendations derived from the local database. This design ensures that users receive feedback in an intuitive and unobtrusive manner.

V. Design Trade Studies

The design trade study for our system focused on these main areas: the selection of the single-board computer (SBC), the camera module and screen display, the face detection system, the ML architecture, and the user-interaction and physical interface design.

A. Single-Board Computer

The SBC serves as a core processing unit, responsible for handling image capture, ML inference, and user interface. The primary tradeoff lies between computational performance and cost efficiency.

Candidate options included Raspberry Pi 4, Raspberry Pi 5, Nvidia Jetson Orin Nano, and Coral Dev Board. While the Jetson Orin Nano[16] and Coral[17] offer dedicated GPU/TPU acceleration for ML, they were more costly than the RaspberryPi boards, which was detrimental to our objective of making our product affordable.

The main focus when selecting our SBC board was to ensure sufficient computing power to satisfy our latency design requirements. Our specification requires an end-to-end latency of less than 5 seconds, with ML inference completing in under 4 seconds. Although Raspberry Pi boards lack the raw compute throughput of devices such as the Jetson Nano or Coral Dev Board, they offer significantly lower cost, lower power consumption, and an extensive developer ecosystem. Although there is no strict formula for the inference time, for the inference time depends on the content of each matrix, we’re confident Raspberry Pi 5 will achieve inference in less than 4 seconds based on the size of light weight ML architectures. The negligible cost difference for Raspberry Pi 5 and Raspberry Pi 4, coupled with a substantially higher on-CPU inference throughput on the Pi 5 versus the Pi 4 led us to select the Raspberry Pi 5 as our SBC.

	Raspberry Pi 4	Raspberry Pi 5	Nvidia Jetson Orin Nano	Coral Dev Board
GPU/Accelerator	Broadcom VideoCore VI	Broadcom VideoCore VII	NVIDIA Ampere Architecture (1024 CUDA cores / 32 tensor cores)	Google Edge TPU coprocessor
RAM	4GB	8GB	8GB	4GB
Camera Support	2× MIPI CSI ports	2× MIPI CSI ports	2× MIPI CSI-2 connectors	1× MIPI CSI, USB cameras supported
Cost	$81	$85	$250	$130
Ease of Development	Excellent, python + TFLite	Excellent, python + TFLite	Strong Jetson ecosystem	Limited to Edge TPU model compatibility
Inference Throughput	poor	moderate	good	best

B. Camera Module and Screen Display

For the camera module and screen display, our objective was to capture fine-grained skin detail and present results unobtrusively on a two-way mirror.

We evaluated CSI-2 camera options and a USB webcam with the goals of low latency and low cost. CSI-2 modules paired with libcamera offer tighter integration with the Raspberry Pi image pipeline than the USB webcams. Therefore, we gravitated towards CSI-2 cameras to enhance image accuracy and latency requirements. We selected the Arducam IMX519 (16MP) which provides higher spatial resolution for micro-textures. For the display, we compared 5-inch and 7-inch panels. Initially we intended to use the 5 inch DSI panel for its compact footprint and lower cost. However, during prototyping, we observed that the 5-inch display made the user interface difficult to view and navigate. Therefore, for our final design, we settled on the 7 inch display.

	RPi AI Camera (Sony IMX500)	RPi Camera Module 3	Arducam IMX519	Logitech C270 HD Webcam
Price	$77	$30	$25	$22
MP	12	12	16	720p (video)
RPi Compatibility	Compatible through CSI	Compatible through CSI	Compatible through CSI	Compatible through USB
Interface	CSI-2 serial data output I2C for camera control/config	CSI-2 serial data output I2C for camera control/config	CSI-2 serial data output I2C for camera control/config	USB 2.0
Autofocus	No	Yes	Yes	No

C. Face Detection Model

Face detection is required to be fast and accurate. The overall end-to-end skin inference time cannot exceed 5 seconds, and likely a majority of that time will be used by the skin classification models. The face detection model also needs to work 95% of the time in standard lighting and user distance conditions. The initial implementation used the BlazeFace model from the Google MediaPipe framework.[1] It was selected due to its < 95% average precision and small model size. However, MediaPipe was failing during integration testing, as the framework only has minimal support for the ARM architecture used on the Raspberry Pi 5. This led to the adoption of the DeepFace model for face detection. DeepFace reports a 97.53% accuracy on facial recognition tasks, with similar optimizations made for increasing inference speed on edge devices.[9]

D. Machine Learning Architecture

The ML architecture used for the skin classifiers must allow for an inference time of less than 4 seconds, while still having the model performance to achieve 85% accuracy on the four ML tasks. Originally, each of the four conditions (oiliness/dryness, acne, sunburn, wrinkles) were structured as classification tasks. However, this approach led to sub-optimal results that failed to meet the design requirements of 85% accuracy by a significant margin. Acne, sunburn, and wrinkle conditions were all shifted to be object detection tasks instead of image classification. This problem realignment aligns closer with the goal of the product in informing the user about the specifics of their skin, and allows for increased performance and a more effective training procedure.

For the oiliness model, out of the class of performant image classification architectures, lightweight Convolutional Neural Networks provide sufficient performance while being small enough to perform inference on an edge device within the allotted time limit. As such, four potential options were considered: MobileNetV2, MobileNetV3, EfficientNet, and ShuffleNet.

We decided to use MobileNetV3. MobileNetV3-small has a lower parameter count than MobileNetV2 and EfficientNet, resulting in faster inference. Furthermore, MobileNetV3 is a popular and commonly-used architecture, meaning that more resources would be available for guidance in training and development. Skin type analysis was previously performed with MobileNetV2, and MobileNetV3 consistently outperforms its predecessor, indicating that the model selection was appropriate for this task. [2], [3]

To solve the object detection objective, YOLO models emerged as the best clear alternative among popular models. Other models such as Masked R-CNNs, Cascade R-CNNs and Single-Shot MultiBox detectors report slower speeds than the nano variants of YOLO models. YOLO models achieve efficient computation by only performing one pass on the image with a single network, reducing compute time. YOLOv8-nano was selected as the specific model due to the large community and developer support and small model size.

E. User Interaction and Physical Interface Design

An additional design trade study focused on user experience, specifically on the user interaction method and physical interface of the system.

Our initial design employed a two-way mirror configuration, with an LCD mounted behind the mirror film and physical buttons used for application navigation. This approach was intended to create a visually cohesive “smart mirror” experience in which the mirror and display appeared closely integrated.

However, during prototyping and user testing, we found that the two way mirror film significantly reduced display visibility. In addition, navigating the application using physical buttons was unintuitive and inefficient, particularly for multi-step interactions such as browsing past results or viewing trend graphs. These limitations negatively impacted the overall usability and user engagement.

To address these challenges, the design was revised to use a one-way mirror paired with a touch screen display mounted beneath the mirror surface. This configuration allows the display to present the analysis with improved clarity and brightness. Replacing the button-based control with a touchscreen interface significantly improved ease of navigation.

The physical dimensions of the system also increased from 12 × 18 inch frame to a larger 16.25 × 14.5 inch frame. This change enhanced visibility and provided additional space to integrate the larger display. Overall, the migration from a two-way mirror with button controls to a one-way mirror with touch screen input represents the tradeoff favoring usability over tightly coupled visual integration.

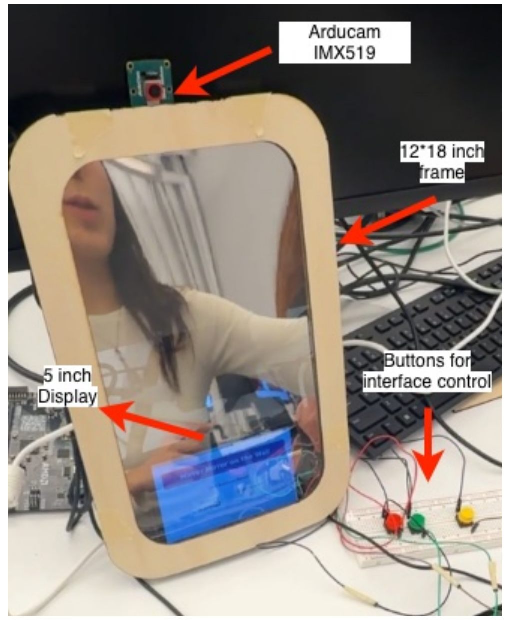


Fig. 2. Prototype of Mirror Mirror on the Wall

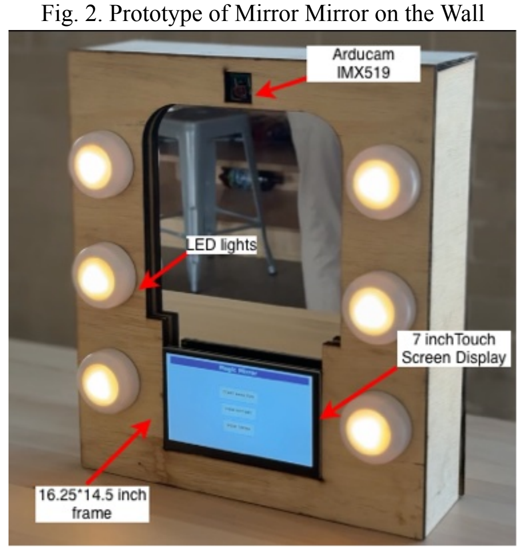


Fig. 3. Mirror Mirror on the Wall final product

VI. System Implementation

A. Physical Design

Mirror Mirror on the Wall was intentionally designed in order to build an intuitive system that combines usability with aesthetics. The primary interface of this product will be through a one-way mirror with an LCD that sits in the front, allowing users to use the product as a regular vanity mirror with a seamless integration of our skin analysis and tracking mechanism. Mirror Mirror on the Wall has a sizable dimension, measuring 14.5” × 16.25” × 3.0”. The 7.0” LCD in the front provides a comfortable user experience, as it is easy to view and interact with through touch. The full physical system is shown in Fig 3.

The LCD is hot-glued to the mirror for secure attachment. The camera module is mounted at the top center of the exterior. To prevent potential obstructions to the user view, the camera is mounted at a slight angle using wood adhesive to optimize its alignment with the user's face. LED lights are bonded along the sides of the exterior using a wood glue. This will provide uniform lighting for image capturing.

Finally, the RPi 5 sits hidden in the rear compartment. A 64 GB microSD card is used to provide sufficient storage for the large number of images and analysis results required to track and display the user history. The connections among the single board computer, camera, and the LCD are routed in the rear compartment, minimizing interference with the user experience. There is a small hole made in the back to allow the power cable to run through.

B. Embedded Controls

RPi 5 hosts the embedded controller. The controller acts as the bridge layer between software and hardware and handles user inputs and system outputs in real-time to communicate smoothly via touchscreen and serial interfaces. To achieve this, it utilizes existing libraries to 1) capture and send the image to our face detection and ML models and 2) display and manage interaction with our system app Magic Mirror.

B. 1. Camera

The embedded controller will manage communication with the Arducam IMX519 camera module through the Raspberry Pi's MIPI CSI-2 interface. In addition to the camera's high resolution and auto-focusing capabilities, its interface supports high-speed image transfer with minimal latency directly into the Raspberry Pi's Image Signal Processor (ISP). A custom version of the libcamera library provided by Arducam to enable focus control is used to manage all userspace camera operations using a cleanly defined API for preview, focus, capture, and image processing. When the user initiates a skin analysis, the controller will utilize libcamera to initiate a preview stream. Once the Magic Mirror app decides capture is ready, it will send a capture signal shown in Fig 4 to the embedded controller. The captured frame will be passed on to the ML block for post-processing. This is a functionality that has to be closely coordinated in terms of timing so that the image data is transmitted smoothly and without corruption. All imagery is written to the microSD card for processing and also archived for history tracking. To note, all pictures are stored locally on the RPi to ensure privacy, and the users have full control to delete their photos if they wish.

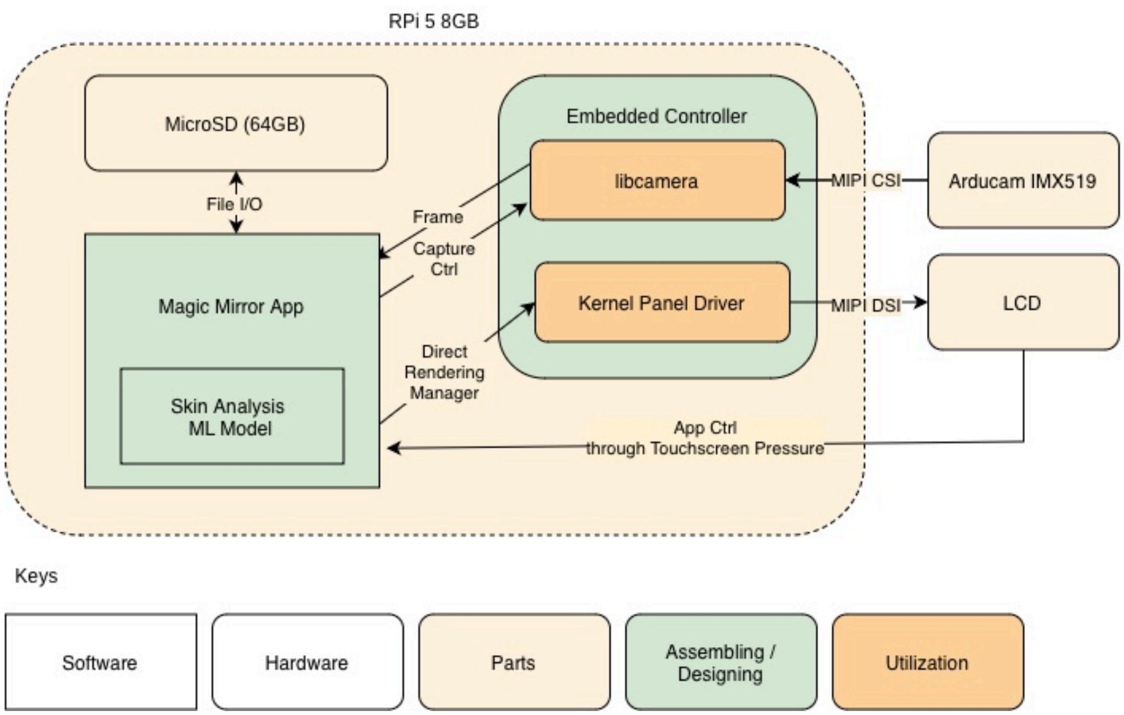


Fig. 4. Updated overall system diagram that reflects our Design Trade Studies

B. 2. LCD

The LCD is connected to the Raspberry Pi via the MIPI DSI interface, which provides a high-speed serial link to transmit image data from the graphics subsystem to the display panel. The display is driven by the Linux kernel's panel driver in the Direct Rendering Manager (DRM), which allows the system app to directly write frames to the framebuffer for low latency updates.

The LCD provides a touchscreen interface that allows users to kickstart and manage the Magic Mirror application. It operates on a capacitive touch mechanism, meaning that it works by measuring capacitance differences when a finger touches the surface. Generally, this supports a smoother interaction as it is more responsive than pressure-based touchscreens.

C. Face Detection

The face detection system receives image data from the camera. Upon receiving button control input, if a face is located, the bounding box for the face is checked to determine whether the face is an appropriate size for proper processing. Given a sufficiently sized bounding box, a crop is taken within the bounding box, and that cropped patch is sent to the skin condition classification models. If any of the above conditions are not met (no face detected, or bounding box is insufficiently large), the detection system does not send data to the skin classification system. Feedback on ensuring proper face detection conditions is sent to the display. This takes the form of informing the user to move closer/farther depending on current bounding box size.

Detection and bounding box generation is handled using the DeepFace model. Given an image, DeepFace generates a bounding box with an associated confidence score. Bounding boxes with < 0.85 confidence scores are rejected. This confidence value is selected to minimize false positives, while still maintaining reasonable leniency. From there, the bounding box with the highest confidence score is selected, and tested for area. The area of the bounding box is required to be within the range of 10-50% of the area of the image. From there, if the bounding box is sufficiently sized, the image is cropped for patch extraction. The image is then split into 4 separate equal-sized patches. This allows for a finer-grain analysis of the user's skin. Each of the four patches are sent through all four models, and the results of each are stitched together. The size of the patches before downsampling can be estimated as shown in equation (2):

$$\sqrt{16,000,000px \ * \ 25\% \ * \ 10\%} \ \approx 632px \quad (2)$$

This represents about $\approx 2.49\%$ of the area of the original image, allowing for fine grain detail of the user's skin before downsampling. A flowchart of the face detection system can be found in Fig 7.

D. Skin Condition Evaluation

The Skin Condition models receives a patch extracted from the image of the user's face, and returns a set of four values representing the classifiers' output in the four respective categories:

1) Bounding boxes/confidences for sunburns
2) Bounding boxes/confidences for wrinkles
3) Bounding boxes/confidences for acne
4) Class confidences for the oiliness of the skin (oily, normal, dry).

The classification system consists of four deep learning models. The oily model is a MobileNetv3-small architecture with an attached fully connected network and a classification head. The acne, wrinkle, and sunburn models are all YOLOv8-nano architectures. Each model was pre-trained with self-supervised learning on ImageNet and the FFHQ dataset. The acne, wrinkles, and sunburn models all output bounding boxes with associated confidences for each detection. If a box has a confidence of greater than 0.5, then it is accepted as a valid detection. The bounding boxes and the number of bounding boxes are returned to the recommender app.

The oiliness model outputs class confidences for its three classes: oily, dry and normal. The confidences are then normalized, scaled, and interpolated to generate a value from 0-5 representing the user's skin moisture. A numeric value is generated to reflect the nature of skin moisture as a gradient of values rather than discrete classes.

The system feeds the patch into the four models, and the results from the current patch are returned to the rest of the mirror software.

E. Datasets

Multiple datasets were collected for use in training each individual model. The FFHQ-wrinkle dataset was used for training the wrinkle model. This dataset consists of over 70,000 unlabeled faces with wrinkle masks, and around 1000 directly labeled images[5]. Burn and acne datasets used a combination of kaggle datasets and scraped data. The acne Kaggle dataset[14] consisted of 940 images, and we scraped and labeled 60 additional images. The burn Kaggle dataset[13] consisted of labeled 1289 images, and we scraped and labeled around 80 additional images. Lastly, the oiliness dataset was a combination of two datasets. One with 3152 scraped images[11], and a second dataset[12] with 112 curated images. We did not manually scrape any additional images for the oiliness dataset due to the difficulty of manually classifying oiliness.

F. Product Recommendation System

The product recommendation system uses the classification results from the skin condition classifiers, and returns appropriate product recommendations to the users. Each skin condition and its severity has different assigned products stored locally on the board, and the final result provides up to two combinations of treatment. For example, a person with 2 acnes detected and 10 acnes detected will have a different recommendation, even if they both have acne. The following are the thresholds for having different treatments:

i) Oily (scale of 0~5): {0-1.5}, {1.6-3.2}, {3.3-5}
ii) Acne (quantitative counts): {0},{1-4}, {5+}
iii) Wrinkles (quantitative counts):{0}, {1-4}, {5+}
iv) Sunburn (quantitative counts): {0},{1-2}, {5+}

Each recommendation will include the generic product name, and for which condition the product is needed for. Products were chosen based on thorough research and a consultation with a dermatologist. In cases where multiple conditions fall within the same severity range, recommendations are prioritized in the following order: sunburn, acne, wrinkles, and oiliness. A special case is when a user falls within the mid-range for oiliness (1.6-3.2) and has no detected acne, wrinkles, or sunburn. Then, the system provides a default maintenance skincare recommendation.

G. Magic Mirror App

The Magic Mirror App is the central software that the user interacts with, and that interacts with the other software subsystems. The app is developed in Python to ensure easy compatibility with the face detection subsystem, the skin condition classifiers, and frontend design.

Upon powering the product and launching the app, the home screen is displayed. From the home screen, three options are available: Start Analysis, View History, and View Trend. Each of these subpages have a “Back” button so the user can return to the home screen at any time. The touchscreen is used to navigate the app.

G. 1. Start Analysis

This functionality kickstarts the skin analysis process. Upon choosing this option, the camera will start a preview and stream frames to the face detection system. At this time, the face detection system will also provide positional feedback i.e. move forward or backward to ensure the face is visible with great details. After running the analysis on the ML models, the skin analysis results will be displayed as shown in Fig 5.

Fig. 5. Skin analysis results page

G. 2. View History

The Magic Mirror App stores the user’s past skin condition analyses and product recommendations. It shows the 5 most recent results (each result is the same format as what is shown in Fig 5). This page is also where users can delete pictures from specific sessions if they want to.

G. 3. View Trend

Compared to the View History page, this page gives you access to a numerical history. It has a graph for each of the four categories, representing condition against time. Selecting a specific point on a particular day will display the picture taken of the user for that particular analysis.

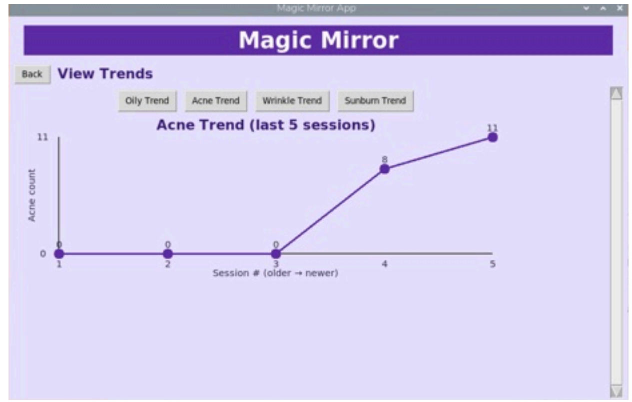


Fig. 6. View Trend page

VII. Test, Verification and Validation

The Magic Mirror will be rigorously tested to ensure the system meets the use-case and design requirements. Testing takes two forms: unit testing for each of the individual components, and a user study for overall user experience testing.

A. Tests for Latency

Latency for analysis should be kept to a minimum. The latency of the skin condition and recommendation analysis will be tested to ensure no more than 5 seconds between capture and analysis and recommendation. Within the 5 seconds, we wanted the ML inference model to take at most 4 seconds. To test if these goals were met, the analysis pipeline will be timed for 40 trials. In the end, we were able to achieve an average end-to-end latency of 2.16 seconds and an average inference latency of 1.5 seconds, both well below our threshold.

B. Tests for Model Accuracy

Model performance was firstly evaluated on the corresponding models’ test sets first. For the oiliness model, classification accuracy on the test set was used as the metric. For acne, wrinkles, and sunburn, recall on the test set was used instead, since they are object detection tasks, not classification tasks. The oiliness model achieved an 88% accuracy on the test set. Wrinkles achieved an 87% recall on the test set. Burn achieved an 80% recall on the test set, and lastly, the acne model achieved a 65% recall on the test set. Oiliness and wrinkles achieved the 85% accuracy/recall requirement, whereas acne and sunburn did not.

To evaluate the performance of the models on live data, a mixture of live faces and printed out images of faces was tested on the three object detection models. The live dataset was small, consisting of 12 live user sessions and 8 printed out images. The models consistently failed to find every instance of wrinkles, sunburns and acne on an image or face, but found at least one of each category (when present) in 80%, 85% and 70% respectively. These values are still below our requirements, but show the models’ ability to make detections under live conditions.

C. *Tests for Face Detection Accuracy*

In order to ensure the image capture system is reliable, as defined by the corresponding use-case, it is necessary to test the robustness and accuracy of the face detection system. Face detection reliability was tested across 40 user sessions. Across all 40 user sessions, the system provided correct feedback on which direction the user should move to be in the optimal position for analysis. Additionally, the system was able to detect a face at the optimal distance 100% of the time. The results of these tests conclude that the system is able to meet the design requirements of at least 95% accuracy on face detection and positional feedback.

D. *Test for Captured Image Quality*

To validate the design requirement of the camera capturing a clear, analyzable image <95% of the time, image quality was analyzed across 40 user sessions. An image was considered unsuitable for analysis if the image was blurry or out of focus at the point it was passed on to the skin condition models. The camera was able to capture a clear, in-focus image 87.5% of the time, representing 35/40 sessions. The testing results show that the system did not meet the design requirement listed. The likely reason behind this failure is the time it takes for the camera to autofocus. Analysis is programmed to start directly when a valid-sized face is detected in the image. This can cause the system to capture a photo of the user's face before the camera autofocus has time to complete the focusing procedure. The tradeoff was made to ensure a lower latency in the skin analysis pipeline.

E. *Tests for Hardware Robustness*

In order to ensure a smooth, consistent user experience, the system hardware is required to be robust. In particular, the LCD display and the LEDs should all maintain optimal performance across any length of use. As such, each component will be tested for consistency.

For the LCD display, there should be no jitters or instances of lost signal/connection. The display should never turn off unless the mirror is off. To test this, the display should be on and free from graphical glitches for 25 minutes straight. This represents a reasonable upper bound on use time for one user session.

The LEDs should be able to stay on for as long as a typical user session lasts, without any noticeable flickering. As such, to ensure LED robustness, the LEDs should stay consistently on for 25 minutes.

Across 20 longer user sessions (we intentionally tested longer user sessions for hardware robustness), all 20 sessions had reliable connection to the LCD display and the LED lights for 25 minutes.

F. *Tests for Software Robustness*

To ensure a consistent, frictionless user experience as outlined in the design requirements, the Magic Mirror App needs to work as intended each time the user interacts with it. This requires each feature of the app to be consistently navigable by the user, without unexpected state changes occurring. To ensure this, each app state transition should behave the same each time they occur. To test this, each transition was tested 10 times using the touch screen navigation. A trial is considered to be successful if no unexpected transitions occur, and the system responds to touchscreen inputs correctly each time. If any of these trials fail, then the robustness of the app needs to be adjusted and reexamined. The software was proved to be robust as we achieved 100% accuracy for all of our transitions that were tested 10 times each.

G. *Test for User Experience - User Study*

The mirror must not have a poor user experience, as outlined in the use-case requirements. In order to ensure that the system is simple, intuitive, and easy to use, a user study was conducted. The user study was conducted over 3 days with 5 additional participants other than our group members (total of 40 sessions consisting of 8 different users). Because accuracy is hard due to the nature of this project, participants who have undergone previous dermatological consolations were chosen to get as close as we could to ground truth.

Over the course of the study, the participants used Mirror Mirror on the Wall to receive an analysis of their skin condition, and observe the trends and product recommendations produced by the mirror. At the end of the 5 days, the participants were given a survey consisting of questions with Likert scale responses that gauged user experience and accuracy.

The Likert scale questions are as follow:

1. The Magic Mirror's user interface was intuitive to navigate through.
2. The analysis of my skin given by the Magic Mirror matches my personal intuitions and/or consultations.
3. It would not be difficult to integrate the Magic Mirror into my regular routine.

The free response questions are as follow:

1. What was your experience like using the Magic Mirror?
2. Are there any features you feel are missing from the Mirror?
3. Are there any other thoughts you have about the Magic Mirror?

G. 1. *Likert Scale Results*

The results shown are the averaged score from a scale from 0 to 5 over 40 sessions, 0 being strongly disagree and 5 being strongly agree.

Question	Average Score
Q1 - UI	4.5
Q2 - Accuracy	4
Q3 - UX	5

The result of the survey shows an overall positive experience using Mirror Mirror on the Wall. An average score of 4.5 indicates that the UI was intuitive and easy to navigate, reaching the objective of creating a simple interface suitable for daily use. The perceived accuracy also achieved an average score of 4.0, which reflects a high degree of confidence among respondents about the accuracy conveyed by the system, more so given that respondents have previous knowledge about their skin from professional consultations. Lastly, the question regarding ease of integration with a daily routine received a perfect average rating of 5. This finding ensures that one of the main goals of this project, which aims at seamlessly integrating skin analysis with daily life, is met.

G. 2. Free Response Results

Free response feedback further reinforced findings from the Likert scale responses. In general, participants described the Mirror Mirror on the Wall as easy to use and informative. However, there were consistent concerns regarding the touchscreen being unresponsive at times. The responses emphasized how interface responsiveness relates to maintaining user trust and satisfaction. This feedback helped us make an informed design decision to resize the on-screen buttons and scroll bars to be bigger for better accuracy of touch and reducing missed inputs. Although this didn't fully solve our problem, we believe future work in further calibration of touch sensitivity or alternative input redundancy could help address this issue.

Overall, the user feedback confirmed that the system met its usability goals while also providing actionable insight into specific areas for refinement, demonstrating the value of the user study in guiding iterative design improvements.

VIII. Project Management

A. Schedule

The schedule shown in Fig. 8. is divided into 4 sections: Research and Design, Prototyping, Implementation, and Testing. Each subsection includes tasks, the person in charge, and their corresponding start and end dates we aimed for.

Compared to the initial plan, we were one week behind schedule. This was mainly due to our parts coming in late. However, everything else went according to plan, so we were able to recover by just reducing user testing from two weeks to one.

B. Team Member Responsibilities

Corin was primarily responsible for bringing up the LCD and ensuring that the display is showing what the Magic Mirror application intends to show the users. This involves the communication from the Magic Mirror app to the embedded controller, as well as the connection of the MIPI DSI.

Siena was primarily responsible for bringing up the camera and its integration with the system app and face detection pipeline. This requires communication between the frame inputs and the signal coming back from the app in regards to when to capture the image for the skin analysis. Corin and Siena both worked on the frontend and parallelization of the ML models.

Isaiah was primarily responsible for the core ML logic within the Magic Mirror app. This involves designing and training the skin analysis models. Isaiah also helped integrate his models to our system app.

Each person performed testing and validation for their corresponding subsystem. All of us worked together to assemble the mirror itself and conduct integration testing.

C. Bill of Materials and Budget

It took a total of $135.56 to build Mirror Mirror on the Wall. The breakdown is shown in Fig 9. The total spent on our entire project is $360.72, as we went through different iterations of our design. The spreadsheet shown in Fig 10 details our entire parts list. It includes the parts that were purchased along with their quantities, vendors, and final prices.

D. TechSpark Usage

We used laser cutters in TechSpark to cut the frame for the mirror, as well as the hind compartment that will house our Raspberry Pi. We also worked in TechSpark to assemble our physical structure.

E. Risk Management

Our first risk was towards the beginning of the semester, where our parts came in around a week late. This pushed us back in our schedule. We tried to mitigate this challenge by working on the software/ML aspects first and conducting research on skin care recommendations. Eventually, we were able to complete our project after cutting down our user study from two weeks to one.

We encountered our biggest risk during the interim demonstration. As detailed in our Design Trade Studies, our initial design was to have a two-way mirror with an LCD behind it. App controls would be done through buttons. Our user feedback, however, expressed that the LCD was hard to see through the two-way mirror film and that the controls were unintuitive. Thus, we went through a second iteration of our physical design to address this. We worked over Thanksgiving to put everything together since we were running short on time.

Lastly, although we were meeting both our inference and end-to-end latency requirement, users expressed that it would be better to make our system faster. To achieve this, we parallelized the four ML models and saw a 1.25x speedup compared to when they were run sequentially. Inference would only take an average of 1.5 seconds, making the use of our product more seamless.

IX. Ethical Issues

The design and implementation of our project raises several important ethical considerations related to user privacy, bias and fairness in ML, and user health and safety. Since our system analyzes the user's face and provides personalized skin care recommendations, we made careful design choices to ensure our product is safe, inclusive, and respectful for all users.

One primary ethical concern in a culture that values privacy is proper handling of sensitive personal data. In the case of our project, we ensured that there were no privacy violations or misuse of our user's facial images by operating the entire system locally on our SBC, without reliance on cloud-based processing or external servers.Image capture, ML inference, and recommendation system are performed on-device, minimizing the risk of data leakage.

From a global perspective, bias and fairness were serious ethical issues we tried to address while building our ML models. CV models are highly dependent on the diversity of the dataset they are trained on, and if certain skin tones or age groups are underrepresented, the system may produce less accurate results for some users. To mitigate the risk, we intentionally used large and diverse datasets. We used the FFHQ-wrinkle dataset for wrinkle training, known for its size and demographic diversity. For burns[13] and acne[14], we combined the Kaggle datasets with approximately 100 hand scraped images to increase diversity. For oiliness[11,12] detection, we used a combination of a small curated dataset (~110 images) designed specifically for identifying oiliness and a larger scraped dataset (3150 images) that is more diverse but lower in image quality and consistency. These dataset choices were intended to reduce demographic disparities and improve model reliability across a wide range of skin tones and age groups.

Lastly, considering the public health, safety, and welfare, we were careful to ensure that the users do not misinterpret the system's output as medical diagnosis. The system is explicitly designed and presented as a non-diagnostic, wellness support tool. Users are informed that the results should not replace professional medical advice, particularly for serious or persistent skin conditions.

X. Related Work

Skin analysis systems are an active consumer market. There are existing products out there from brands with big names and high credibility offering similar services. For example, Neutrogena[7] and Sephora[9] offer web/app based systems. The one by Neutrogena has more thorough checks of the surroundings such as background noise and lighting. However, these requirements are difficult for users to meet consistently, and the analysis process is relatively slow. Our product, on the other hand, sets up consistent lighting for the user and is insensitive to background noise as it crops to the biggest face before running the ML models. In addition, there is a 3.55 speedup compared to Neutrogena.

The one by Sephora is 1.83x faster than our system. This increased speed comes at the cost of reliability, as the system does not verify lighting conditions to ensure sufficient facial visibility, which can lead to inconsistent or inaccurate results. Aside from offering reliability, our system also provides general recommendations without promoting products from specific brands. Additionally, we make sure that all data stays local, giving users a sense of security.

Existing embedded smart mirrors such as the beauty mirror from Mues-Tec[10] are too expensive and are hard to purchase.

Relative to these, our system ensures an affordable, reliable, and safe integration to your personal routine, aiming to bridge market-grade hardware with reliable, repeatable skin-analysis workflows.

XI. Summary

Mirror Mirror on the Wall aims to address a common issue for those interested in skincare: the lack of accessible, personalized, and real-time feedback about their skin condition. Whether for casual users who require quick daily feedback or skincare enthusiasts tracking long-term progress, many are dissatisfied with the inconvenience and inaccuracy of app-based or manual assessments. This product delivers a self-contained, customized solution that performs on-device image capture, analysis, and feedback through an integrated LCD display and camera system. Through the incorporation of embedded controls, ML, and natural physical interaction, it offers a personal, interactive, and user-friendly experience with no reliance on external devices or cloud connectivity.

We encountered many challenges during this journey, some within, and some out of our control. Through working on a project that was student-led, we learned to work around and be flexible in the process of working towards the goal. Although we didn't have time to achieve our MVP, future work can be in increasing accuracy and expanding the different number of skin conditions it can detect.

Glossary of Acronyms

MQTT – Message Queuing Telemetry Transport
OBD – On-Board Diagnostics
RPi – Raspberry Pi
MIPI – Mobile Industry Processor Interface
DSI – Display Serial Interface
ML – Machine Learning
CSI – Camera Serial Interface
SBC – Single Board Computer
UI – User Interface
UX – User Experience
MVP – Minimum Viable Product

References

[1] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, et al., "MediaPipe: A framework for perceiving and processing reality," in Proc. 3rd Workshop on Computer Vision for AR/VR at IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), vol. 2019, 2019.

[2] S. Saiwaeo, S. Arwatchananukul, L. Mungmai, W. Preedalikit, and N. Aunsri, "Human skin type classification using image processing and deep learning approaches," Heliyon, vol. 9, no. 11, p. e21176, 2023, doi: 10.101

[3] A. Howard et al., "Searching for MobileNetV3," 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 1314–1324, Oct. 2019. doi:10.1109/iccv.2019.00140

[4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788, Jun. 2016. doi:10.1109/cvpr.2016.91

[5] J. Moon, H. Chung, and I. Jang, "Facial wrinkle segmentation for cosmetic dermatology: Pretraining with texture map-based weak supervision," Lecture Notes in Computer Science, pp. 319–334, Dec. 2024. doi:10.1007/978-3-031-78104-9_22

[6] "Introduction," MagicMirror2 Docs, https://docs.magicmirror.builders/ (accessed Oct. 10, 2025).
[7] Neutrogena, "Neutrogena Skin360® Web App," skin360.neutrogena.com, 2024. https://skin360.neutrogena.com/
[8] H. Cate, F. Dalvi, and Z. Hussain, "DeepFace: Face Generation using Deep Learning," *arXiv preprint arXiv:1701.01876*, 2017.
[9] "Smart Skin Scan: Virtual Skin Analysis Tool | Sephora," Sephora, 2025. https://www.sephora.com/beauty/skin-analysis-tool
[10] "Beauty Mirror - Mues-Tec," Interactive smart mirrors with personality! - Mues-Tec, Feb. 25, 2025. https://mues-tec.com/products/beauty-mirror/ (accessed Dec. 12, 2025).
[11] Dissanayake, S. (2024, December). Oily, Dry and Normal Skin Types Dataset, Version 1. Retrieved December 11, 2025 from https://www.kaggle.com/datasets/shakyadissanayake/oily-dry-and-normal-skin-types-dataset/data.
[12] Katoch R.. (2025, April). Normal/Dry/Oily Skin type, Version 1. Retrieved December 11, 2025 from https://www.kaggle.com/datasets/ritikasinghkatoch/normaldryoily-skin-type/data.
[13] Shubham. (2021) .Skin Burn Dataset, Version 1. Retrieved December 11, 2025 from https://www.kaggle.com/datasets/shubhambaid/skin-burn-dataset.
[14] Kurnaz O.(2023). Acne Dataset in YOLOv8 Format, Version 1. Retrieved December 11, 2025 from https://www.kaggle.com/datasets/osmankagankurnaz/acne-dataset-in-yolov8-format.
[15] [1] FeelinMySkin, "FeelinMySkin skin analysis application." [Online]. Available: https://www.feelinmyskin.com. Accessed: Dec. 12, 2025.
[16] "Jetson Nano Brings the Power of Modern AI to Edge Devices," NVIDIA, 2025. [Online]. Available: https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-nano/product-development/. Accessed: Dec. 12, 2025.
[17] "Get started with the Dev Board," Coral. [Online]. Available: https://www.coral.ai/docs/dev-board/get-started/#requirements. Accessed: Dec. 12, 2025.

Section VI: Additional Figures

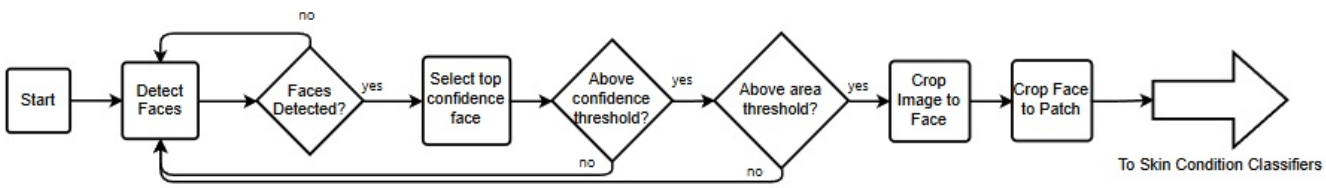


Fig. 7. Face Detection Pipeline Flowchart

Section VIII: Additional Figures

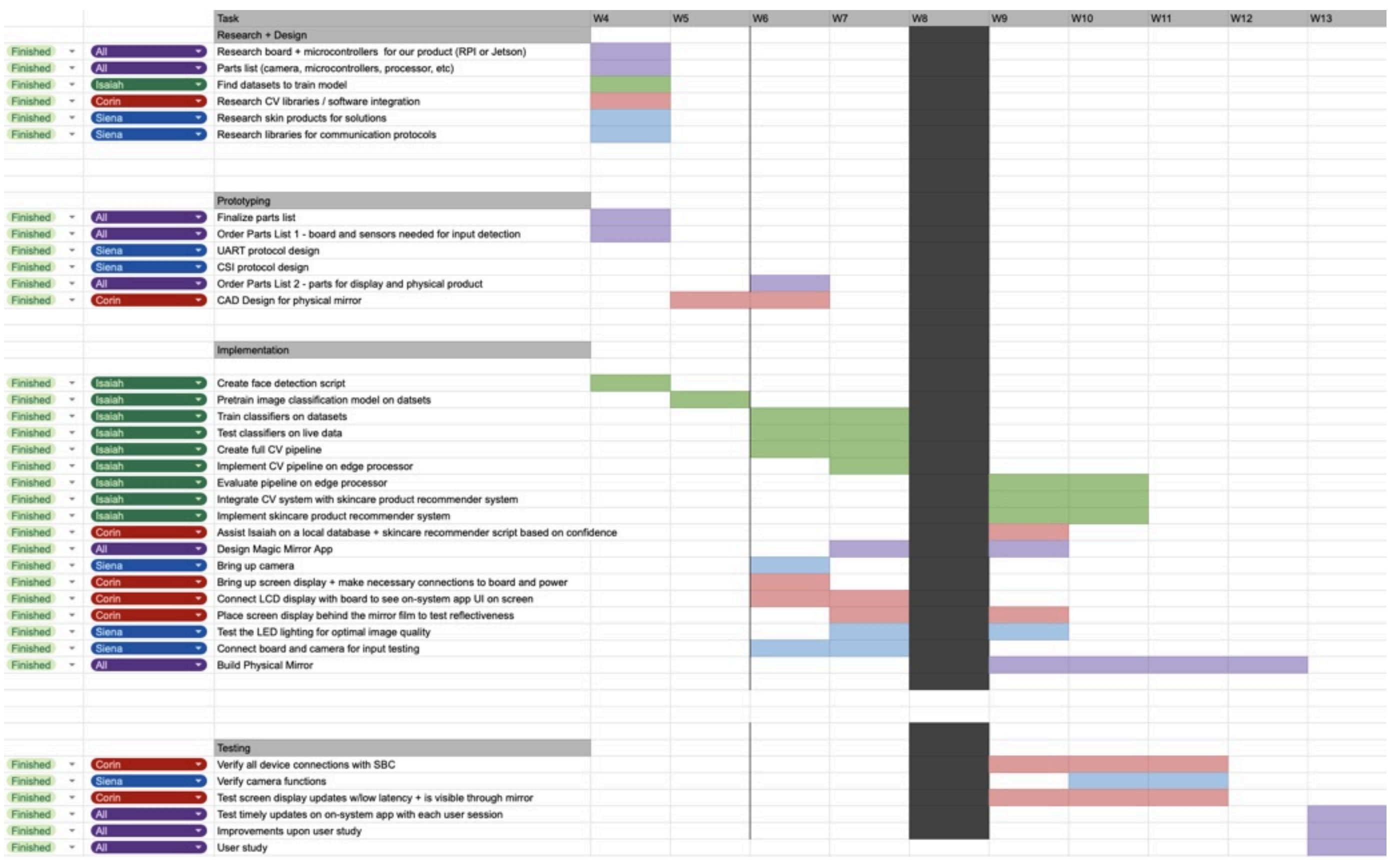


Fig. 8. Gantt Chart

Item	Quantity	1 Unit Cost		Total Cost
RPi5	1	x (borrowed)		$135.56
Arducam IMX519	1	$24.99		
LCD (7")	1	$37.59		
Plywood 2x4	1	$14.94		
Micro SDCard	1	$15.08		
LED Lights	1	$17.99		
Mirror	1	$24.97		

Fig. 9. Total cost of Product

Item	Quantity	1 Unit Cost	Price + Shipping	Vendor	Link	Status		Total Spent
RPi 5	1	x	x	Capstone		Receiv...		360.72
Arducam IMX519	1	$24.99	$24.99	Amazon	https://www.ama	Receiv...		
LCD (5")	1	$34.99	$34.99	Amazon	https://www.ama	Receiv...		
Micro SDCard	1	$15.08	$15.08	Amazon	https://www.ama	Receiv...		
USB-A Keyboard and Mo	1	$19.99	$19.99	Amazon	https://www.ama	Receiv...		
LED Lights	1	$17.99	$17.99	Amazon	https://www.ama	Receiv...		
Plywood Sheets	1	$18.99	$18.99	Amazon	Amazon.com: Ba	Receiv...		
M2.5 x 6mm	1	$5.99	$5.99	Amazon	https://www.ama	Receiv...		
Two-Way Mirror Glass	1	$26.99	$26.99	Amazon	https://www.ama	Receiv...		
Wood Glue	1	$10.97	$10.97	Amazon	https://www.ama	Receiv...		
Super Glue	1	$9.99	$9.99	Amazon	https://www.ama	Receiv...		
Buttons	1	$13.98	$13.98	Amazon	https://www.ama	Receiv...		
Ribbon Cable	1	$9.99	$9.99	Amazon	https://www.ama	Receiv...		
Plywood 2x4	4	$14.94	$62.75	Home Depot	https://www.hom	Receiv...		
Batteries	1	$12.49	$12.49	Amazon	https://www.ama	Receiv...		
Mirror	1	$24.97	$27.96	Home Depot	https://www.hom	Receiv...		
LCD (7")	1	$37.59	$37.59	Amazon	https://www.ama	Receiv...		
CSI Cable	1	$9.99	$9.99	Amazon	https://www.ama	Receiv...		

Fig. 10. Total cost of project