SMART MIRROR USING RASPBERRY PI

[1]Varun Girdhar, [2]Shraddha Mangelkar, [3]Chandranil Mazumdar, [4] Prof. Pramod Rodge
[1]Student, [2]Student, [3]Student, [4] Professor
[1]Department of Computer Science
[1] Shivajirao S. Jondhale College of Engineering, Dombivli, India

Abstract: *This project has been developed within the context of a time where every day we see more and more connected devices. The Internet transformed our lives by connecting us more easily to information and other people in the virtual world. Mirrors, for example, provide a large surface ideal for displaying information and interacting with. Most people have mirrors at home so the concept of a smart mirror that you can interact with is attractive and has been fantasized in many futuristic movies. Smart mirrors, such as Magic Mirror and Home-Mirror have recently started to be developed by people in the Maker community, with varying degrees of interactivity. This final year project describes how a smart mirror was built from scratch using a Raspberry Pi for the hardware and custom software built on top of Raspbian, a Linux distribution. The goal of the project was to create a Smart Mirror device that people could interact with but also to further develop the technology so that it would let you install and develop your own applications for it. The Smart Mirror was developed in four months, starting with the software and finally integrating it with the hardware.*

Key Words: IoT, Smart Device, Raspberry Pi.

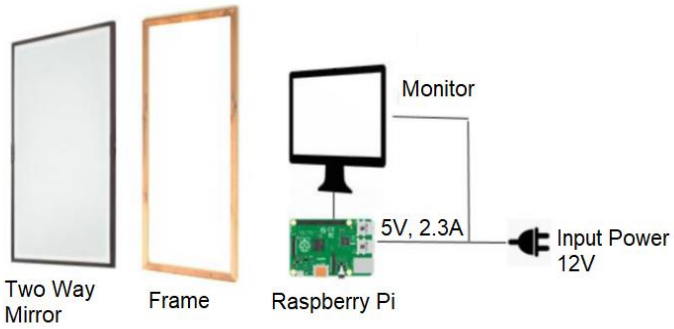
INTRODUCTION

With the introduction of the system on chip (SOC), such as the Raspberry Pi, the notion of creating "smart devices" is a relatively new craze that has taken over hobbyist communities. Many new IoT (Internet of Things) applications are implemented with a single system on a chip (SoC) to provide the highest level of integration and conservation of the area. One currently popular project using a SOC is the development of smart mirrors

A Smart Mirror is a mirror with "smart" capabilities much like how cell phones have become smart. That is, it is a display that looks and acts like a mirror, but has the capability of displaying multimedia data through the mirror glass as if the mirror was a screen on its own accord.

I. PROPOSED SYSTEM

Figure 1: Proposed System



The objective of designing the mirror is to provide a natural interface in the ambient home environment for accessing various services such as location-based weather, time, calendar, etc. and hence reducing user efforts and conveniently saving time. The mirror is eventually a technologically augmented interaction device. The selected processing unit for the system is Raspberry Pi 3(Model B). The operating system used is Raspbian (Stretch) which is based on Debian package and a micro SD card is used for storage and running the Operating System. We are also on including Artificial Intelligence in the Smart Mirror wherein a Voice-enabled assistant will cater to the needs of the user.

II. REQUIREMENT ANALYSIS

[A] Software Requirement:

1. Programming Languages:

- Node JS

The Smart mirror will be fetching requested data in real time. So, each request can be treated as an event. Node.js has extraordinary features when it comes to developing real-time applications. It is also a good fit for the programs that need event-based server, as well as non-blocking driven servers.

- Electron.js (Wrapper)

We'll be using multiple API's to fetch data. But using the API directly will not be efficient. Instead we use an API wrapper. API wrapper lets you call it instead of the API directly. So replacing the API becomes easy.

- Python

Python is a general-purpose scripting language. The AI backend client has been developed in python.

2. API:

- Darksky

The Dark Sky API allows you to look up the weather anywhere on the globe. It provides:

- Current weather conditions.
- Minute-by-minute forecasts out to one hour.
- Hour-by-hour and day-by-day forecasts out to seven days.
- Google maps

In order to fetch geographical data and display it to the user, we use the google maps API

- Wit.ai

Wit.ai makes it easy for us to build applications and devices that you can talk or text to.
Wit learns from what your users say and extract useful information. We've generated our own (Entity: value) pair list and used Wit.ai to extract useful relations between the keywords and their meanings.

- Google Assistant

The Google Assistant Service is the best option for flexibility and broad platform support.
It exposes a low-level API which directly manipulates the audio bytes of an Assistant request and response. The audio response generated by the mirror and also the user can be converted to an audio response using this API.

[B] Hardware Requirements:

1. Raspberry Pi 3 Model B:

The Raspberry Pi is a series of small single-board computers with wireless LAN and Bluetooth connectivity. Here we use Raspberry Pi 3 model B which has a quad-core processor and can easily handle heavy application such as the Smart Mirror. It also allows easy integration with input/output devices like microphone, speakers, etc.

2. Monitor:

Monitor is placed behind the mirror which is used to display the desired information to the user.

3. Mirror:

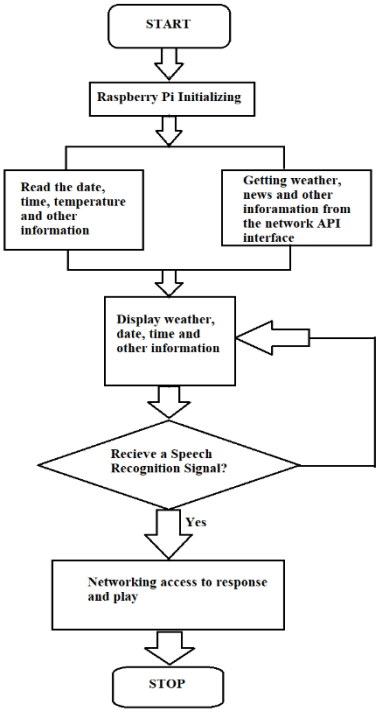
Acrylic two-way see-through mirror which is a special mirror, is used in this project. Unlike the normal household mirror, the two-way mirror is not painted with a color on the back, instead its left untouched. Thus, it was made reflective on one side and transparent on the other side.

4. Microphone:

Interaction with the Smart Mirror is done through a USB microphone.

III. PROCEDURAL DESIGN

Figure 2: Procedural Design



- The Raspberry Pi is initialized when the user inputs a predefined voice command such as “Hello Mirror”. After that, the user issues a Query requesting real-time data like date, time, weather.
- The Wit.ai client consists of an intent : action mapping which understands the intent of the user. The intent is mapped to some action.
- When there is recognition of an user defined command, the Wit.ai maps and acquires the values for the requested entities accordingly to the keywords in the input, which then are used by the python and the Google Assistant to generate the Visual output and Audio Output respectively.
- The required data is fetched from the network API interface which is displayed on the Mirror.

IV. RESULT AND CONCLUSION

The final product is a mirror with an interactive ability. The Mirror displays location-based weather, time, calendar and news headlines. The designed mirror is able to recognise the voice input properly and display us the desired output. It should be able to display Google maps on request.

Smart mirrors have great potential to enhance user experience of accessing and interacting within formation. We have developed a functional prototype to demonstrate our work. Overall, the prototype provides an easily extendable framework that can be utilized to provide even more functionality to the user. This project can be used to investigate how the surrounding context of the user and the environment can be utilized in order to provide optimal service experiences in the home environment. The system can be made much more useful to the users by integrating it into a larger system, such as a home automation system.

REFERENCES

1. Divyashree K J, Dr. P.A. Vijaya, Nitin Awasthi ,Design and implementation of Smart Mirror as a Personal Assistant using Raspberry Pi, International Research Journal of Engineering and Technology, 2018, 438-439.
2. Michael Teeuw, github.com/MichMich/MagicMirror, 2016
3. Sun yong, Geng Liqing, Dan ke, Design of Smart Mirror Based on Raspberry Pi, International Conference on Intelligent Transportation, Big data & Smart City, 2018, 77-79.