16.	Fast Smoke Detection for Video Surveillance Using CUDA	Alexander Filonenko, Danilo C´aceres Hern´andez, Kang-Hyun Jo	2018	Smoke detection is a key component of disaster and accident detection. Despite the wide variety of smoke detection methods and sensors that have been proposed, none has been able to maintain a high frame rate while improving detection performance. In this paper, a smoke detection method for surveillance cameras is presented that relies on shape features of smoke regions as well as colour information. The method takes advantage of the use of a stationary camera by using a background subtraction method to detect changes in the scene. Due to the variable density of the smoke, not all pixels of the	This paper presents a smoke detection algorithm for video surveillance. A single CPU is not able to rapidly process HD video sequences. Processing of the most time-consuming parts should be carried out by specialized devices like FPGAs or by GPGPU. This work was based upon the use of GPGPU. The implementation of the proposed method could achieve time performance for HD-resolution video, which is appropriate for video surveillance tasks. After tests were conducted for multiple datasets, it became clear that the method is sensitive to noise in the video. The best environmental condition for the proposed

				actual smoke area appear in the foreground mask. These separate pixels are united by morphological operations and connected-component labelling methods.	algorithm is indoors with artificial light.

Block Diagram:

The circuit contains an Arduino board (ATmega328P), LED display, wifi module, buzzer and a gas sensor. The data set is collected using the gas sensor with the help of Arduino and is transferred on mobile with the help of esp82666 and is displayed on LCD screen to the mask so that the person is able to see the quality of air in the surrounding environment. The mask also navigates the person from source to destination with the path containing minimum smoke. If the smoke level is greater than a certain threshold limit then the buzzer connected to the mask buzzes and a warning is displayed on the LCD screen.

So the block diagram consists of mainly Arduino and gas sensor others are secondary elements. The sensor is first calibrated using standard values and after that the measured values are noted by using the sensor. The data is serially transferred to a website or application for the navigation part. The navigation algorithm works on the simple concept that the path which contains less smoke is chosen in this way the person can avoid the intake of harmful gases and indirectly reduce the risk of health diseases such as asthma, lung diseases etc. So in this way the mask helps the person to avoid the intake of harmful gases like CO2, NO2, SO2 and many more which affect human health.

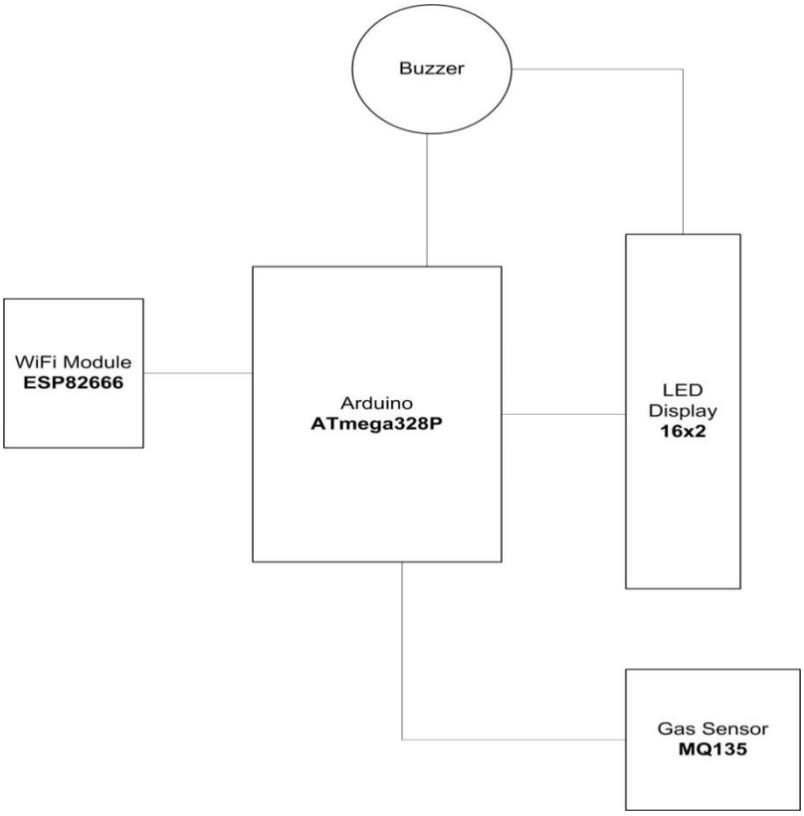


Fig1: Block diagram of the working model

First of all connection is done between the ESP8266 with the Arduino. ESP8266 runs on 3.3V and if you may provides it 5V from the Arduino then it won't work properly and it should get injury. Connect the VCC and also the CH_PD to the 3.3V pin of Arduino. The RX pin of ESP8266 works on 3.3V and it'll not communicate with the Arduino after we will connect it on to the Arduino. So, to compel to create a resistor for it which can convert the 5V into 3.3V. This may be done by connecting 3 resistors asynchronous like we tend to did within the circuit. Connect the TX pin of the ESP8266 to the pin ten of the Arduino and also the RX pin of the esp8266 to the pin nine of Arduino through the resistors.

ESP8266 Wi-Fi module offers your comes access to Wi-Fi or net. Its low-cost device and very powerful. It will communicate with any microcontroller and it's the foremost leading devices within the IOT platform.

Then we'll connect the MQ135 sensing element with the Arduino. Connect the VCC and also the ground pin of the sensing element to the 5V and ground of the Arduino and also the Analog pin of sensing element to the A0 of the Arduino.

Connect a buzzer to the pin eight of the Arduino which can begin to beep once the condition becomes true. In last, connect alphanumeric display with the Arduino.

Arduino:

Arduino is an open-source hardware and software company, project and user community that designs and manufactures single-board microcontrollers and microcontroller kits for building digital devices. Its products are licensed under the GNU Lesser General Public License (LGPL) or the GNU General Public License (GPL), permitting the manufacture of Arduino boards and software distribution by anyone. Arduino boards are available commercially in preassembled form or as do-it-yourself (DIY) kits.

Fig2: Arduino board

Arduino board designs use a variety of microprocessors and controllers. The boards are equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards ('shields') or breadboards (For prototyping) and other circuits. The boards feature serial communications interfaces, including Universal Serial Bus (USB) on some models, which are also used for loading programs from personal computers. The microcontrollers can be programmed using C and C++ programming languages. In addition to using traditional compiler toolchains, the Arduino project provides an integrated development environment (IDE) based on the Processing language project.

WiFi Module

The ESP8266 is a low-cost Wi-Fi microchip, with a full TCP/IP stack and microcontroller capability, produced by Espressif Systems in Shanghai, China.

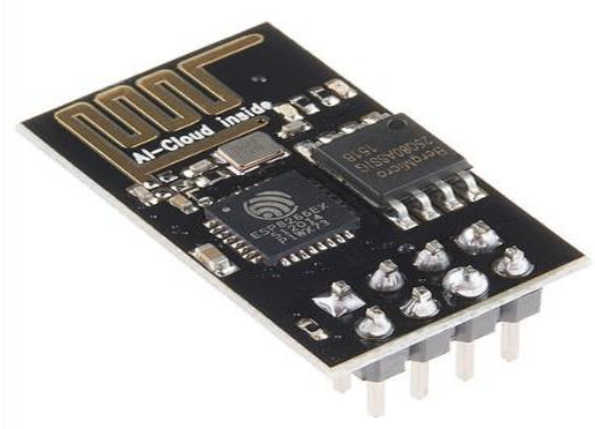


Fig3: ESP8266

The chip first came to the attention of Western makers in August 2014 with the ESP-01 module, made by a third-party manufacturer Ai-Thinker. This small module allows microcontrollers to connect to a Wi-Fi network and make simple TCP/IP connections using Hayes-style commands. However, at first there was almost no English-language documentation on the chip and the commands it accepted. The very low price and the fact that there were very few external components on the module, which suggested that it could eventually be very inexpensive in volume, attracted many hackers to explore the module, the chip, and the software on it, as well as to translate the Chinese documentation.

The ESP8285 is an ESP8266 with 1 MiB of built-in flash, allowing the building of single-chip devices capable of connecting to Wi-Fi.

LED Display

An LCD (Liquid Crystal Display) screen is an electronic display module and has a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. The 16 x 2 intelligent alphanumeric dot matrix display is capable of displaying 224 different characters and symbols. This LCD has two registers, namely, Command and Data.

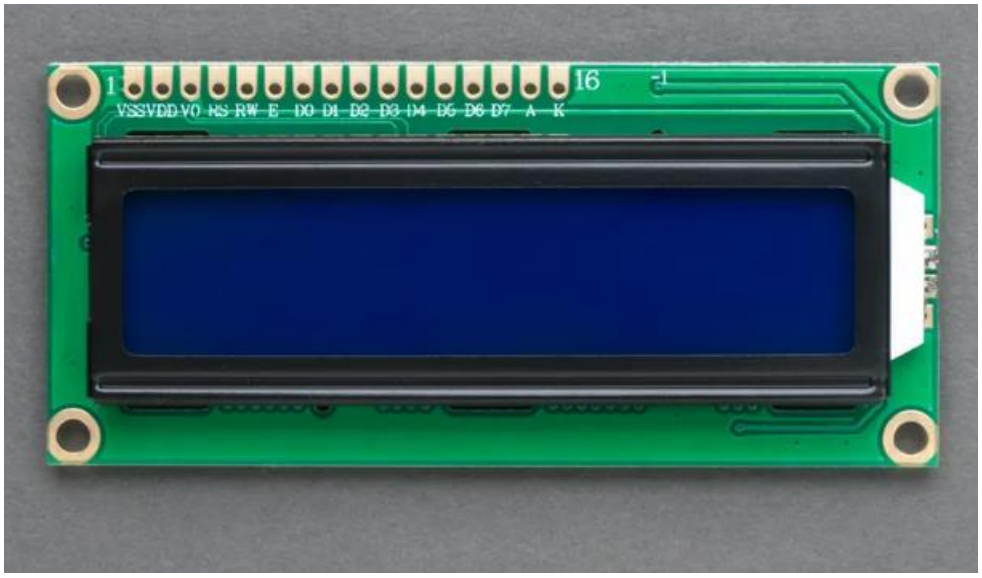


Fig4. LCD display

Command register stores various commands given to the display. Data register stores data to be displayed. The process of controlling the display involves putting the data that form the image of what you want to display into the data registers, then putting instructions in the instruction register. In your arduino project Liquid Crystal Library simplifies this for you so you don't need to know the low-level instructions. Contrast of the display can be adjusted by adjusting the potentiometer to be connected across VEE pin.

Gas Sensor

The MQ-135 Gas sensors are used in air quality control equipments and are suitable for detecting or measuring of NH3, NOx, Alcohol, Benzene, Smoke, CO2. The MQ-135 sensor module comes with a Digital Pin which makes this sensor to operate even without a microcontroller and that comes in handy when you are only trying to detect one particular gas. If you need to measure the gases in PPM the analog pin need to be used. The analog pin is TTL driven and works on 5V and so can be used with most common microcontrollers.

MQ-135 gas sensor applies SnO2 which has a higher resistance in the clear air as a gas-sensing material. When there is an increase in polluting gases, the resistance of the gas sensor decreases along with that. To measure PPM using MQ-135 sensor we need to look into the (Rs/Ro) v/s PPM graph taken from the MQ135 datasheet.

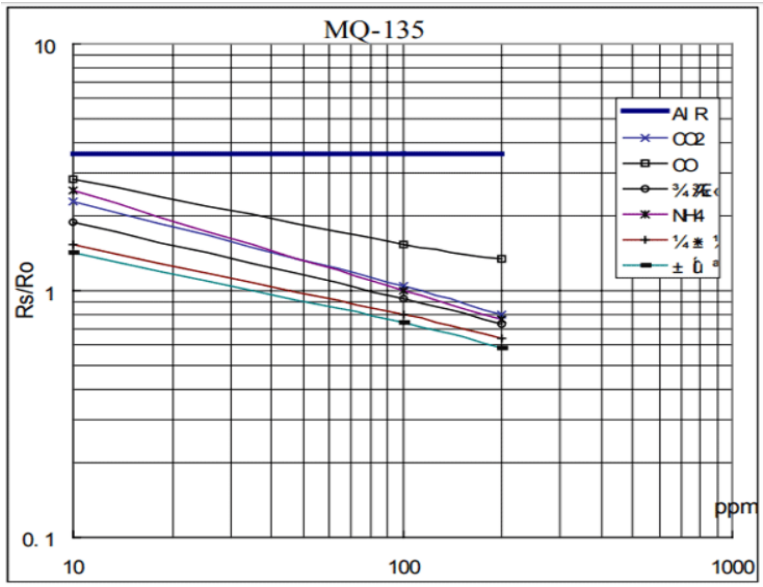


Fig5: Characteristics of MQ135

Buzzer

A buzzer or beeper is an audio signalling device,[1] which may be mechanical, electromechanical, or piezoelectric (piezo for short). Typical uses of buzzers and beepers include alarm devices, timers, and confirmation of user input such as a mouse click or keystroke.

Fig6: Buzzer

Model of SMOKE MASK:

The mask is created using 3D modelling by using pc software AutoCAD. The mask can cover the person's mouth and nose thus reducing the risk of intake of harmful gases. The front part of mask contains two small holes so that the

person can breathe easily and those holes are covered using filters such that the intake of harmful gases should be least.

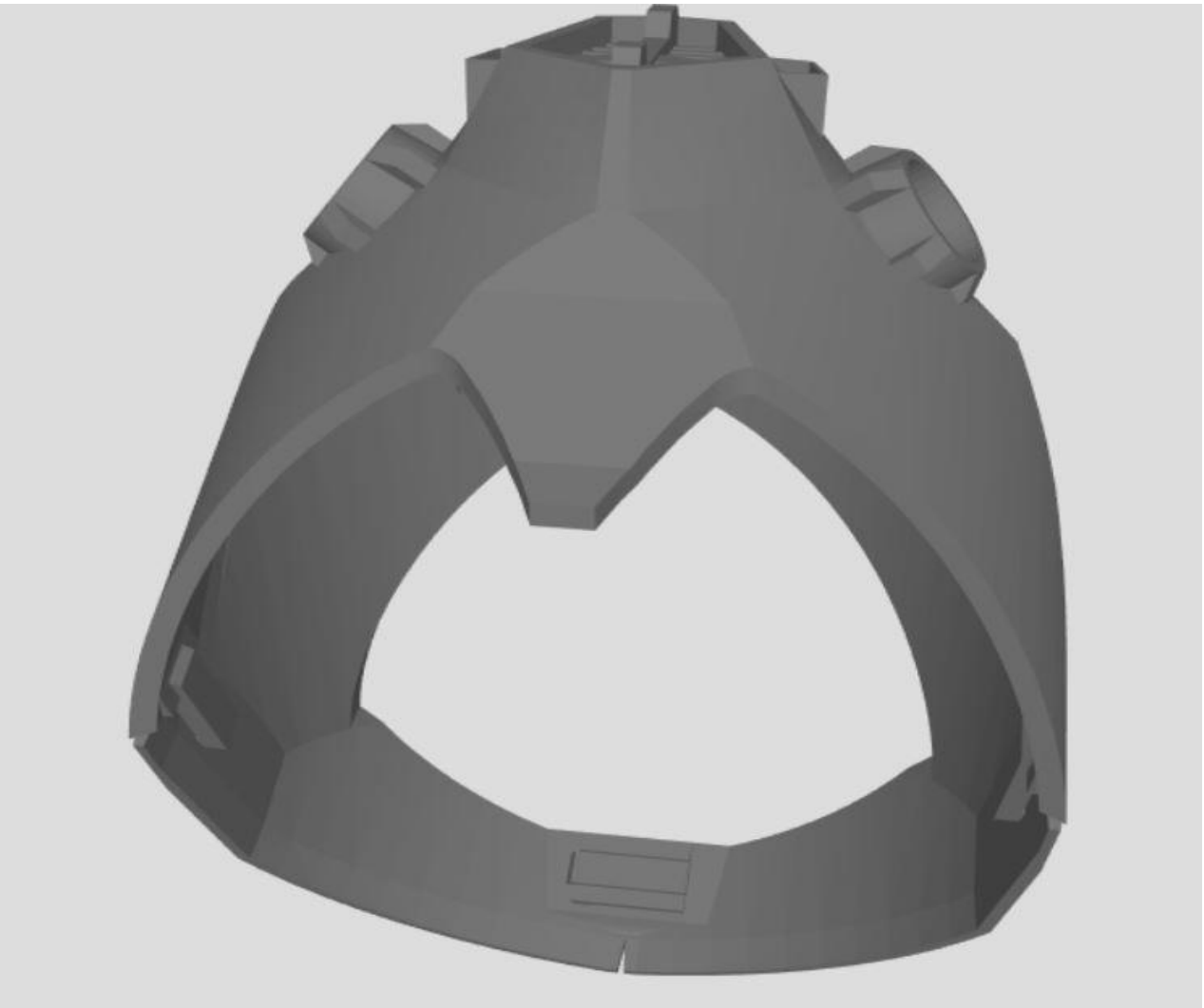


Fig7: Smoke mask

The mask is light weighted and shock proof it is constructed using 3D printing. The mask can be constructed in different sizes depending on the needs of mankind.

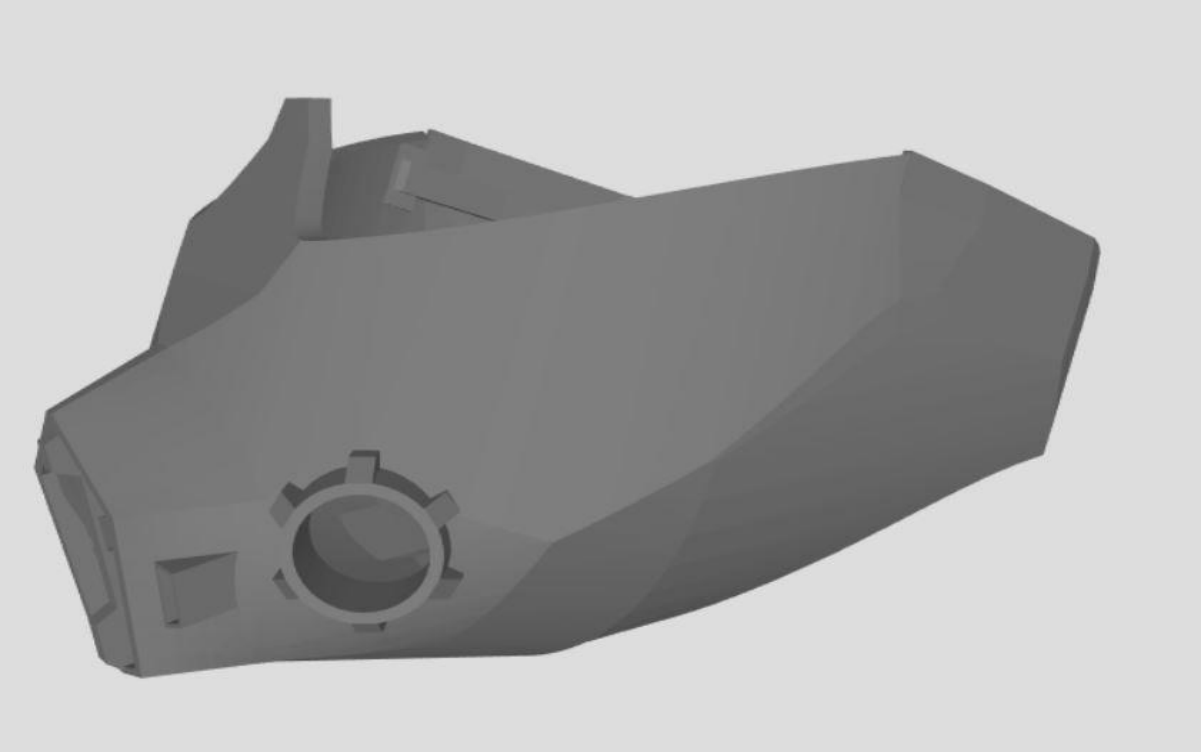


Fig 8: Side View of 3D model

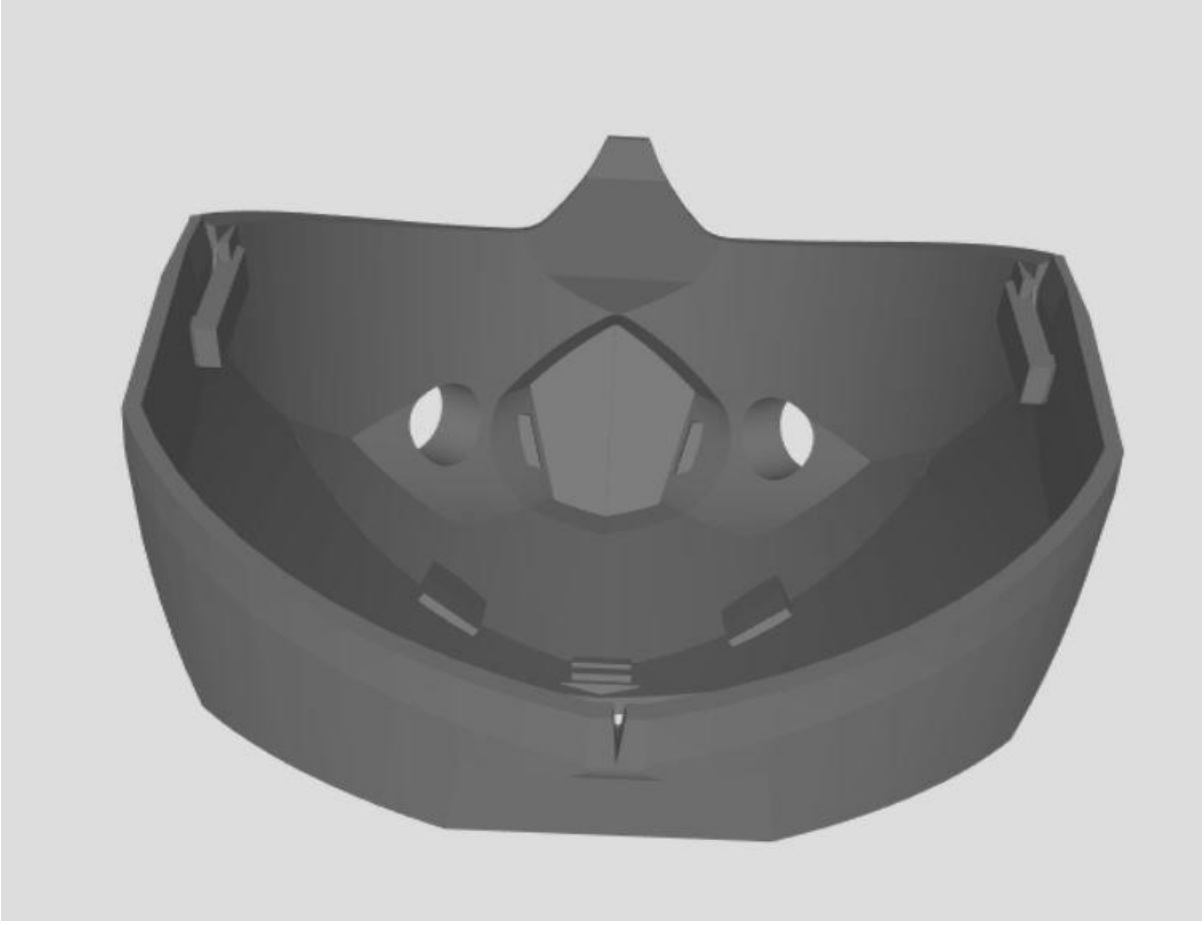


Fig 9: Top View of 3D mask

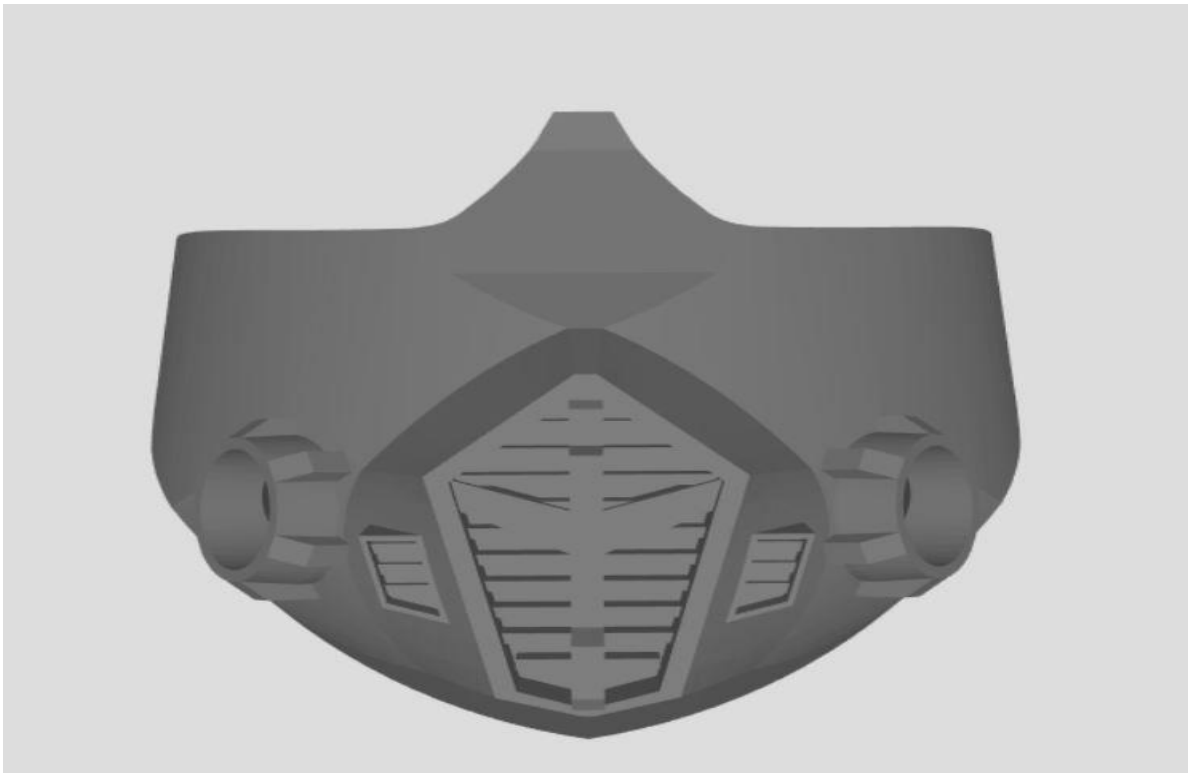


Fig 10: Front View of 3D model

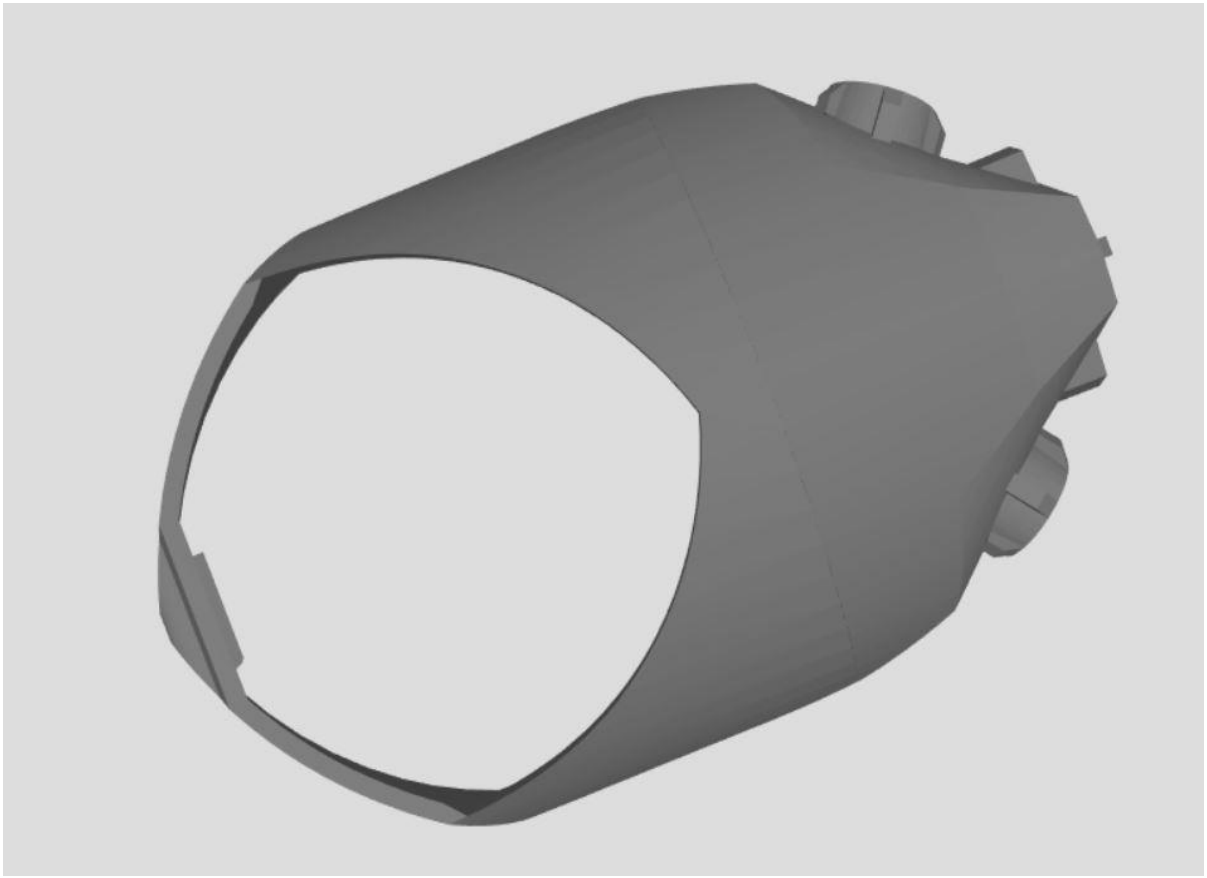


Fig 11: Bottom View of 3D mask

Working

The MQ135 sensing element will sense smoke, NH3, Benzene, carbon dioxide and a few different gases, thus its gas sensing element for watching smoke. Once the smoke sensing element is connected to Arduino it'll sense the gases, and record the Pollution level in PPM (parts per million). MQ135 gas sensing element offers the output in type of voltage levels and that we ought to convert it into PPM. Thus for changing the output in PPM, here we've got used a library for MQ135 sensing element.

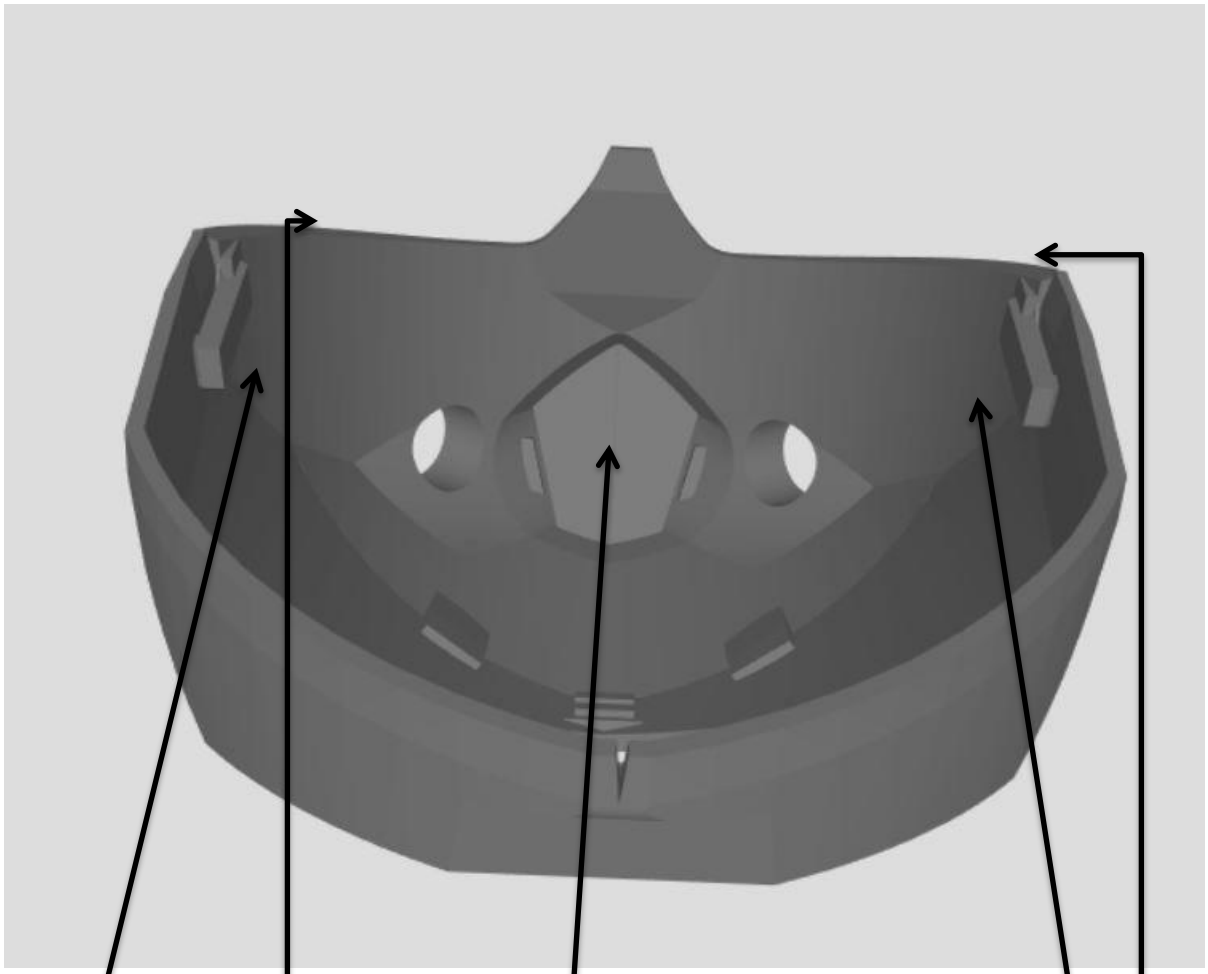
Sensor was giving America price of ninety once there was no gas close to it and therefore the safe level of air quality is 350 PPM and it shouldn't exceed a thousand PPM. once it exceeds the limit of a thousand PPM, then it starts cause Headaches, temporary state and stagnant, stale, stuffy air and if exceeds on the far side 2000 PPM then it will cause enlarged vital sign and plenty of different diseases.

When the value is going to be but a thousand PPM, then the digital display {alphanumeric display} and webpage can display “Fresh Air”. Whenever the worth can increase a thousand PPM, then the buzzer can begin beeping and therefore the show digital display {alphanumeric display} and webpage can display “Poor Air”. If it increases to 2000 then the buzzer can keep beeping and therefore the show digital display {alphanumeric display} and webpage will display “Danger! Move to recent Air”.

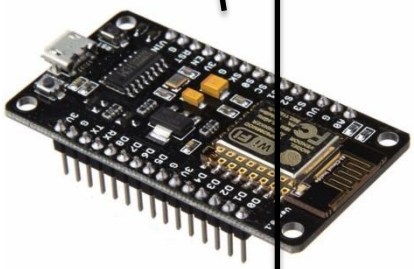
When the LCD is displaying “Fresh Air” the person will take the path allowing him to avoid the intake of harmful gases. The app will navigate the person in the direction of minimum smoke from source point to destination.

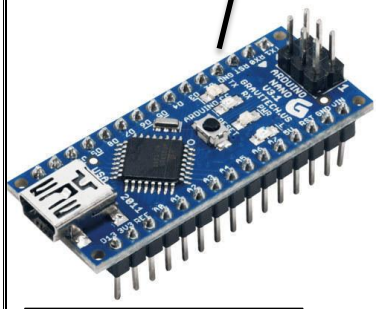
The app is invented using Android Studio. The app navigates the person from source to destination which contains the minimum smoke. So that the person can avoid intake of harmful gases present in the smoke.

The dataset is fed into the app which shows values such as pm2, pm10, NO2 content, Co2 content, SO2 content in the nearby surrounding. So that the person will be aware of the surrounding air quality which will help him to avoid unnecessary health diseases. So in this way this app and our mask will also help mankind to reduce the risk of health diseases.



ESP8266

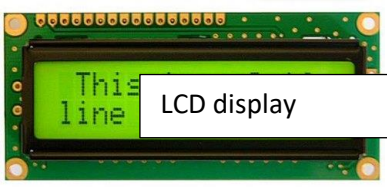




Arduino Nano

Smoke sensor

Buzzer



LCD display

Fig 12: Components in place