

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Landasan Teori**

##### **2.1.1 Definisi Kebakaran**

Kebakaran adalah peristiwa terbakarnya suatu bahan yang dapat terbakar, sehingga menimbulkan nyala api yang tidak terkendali dan tidak terkontrol. Kebakaran dapat merusak dan menghancurkan objek-objek yang ada di sekitarnya (Kelvin, 2019). Kebakaran dapat disebabkan karena beberapa hal yaitu kecelakaan, kelalaian atau kesengajaan. Proses terjadinya kebakaran terjadi Ketika tiga aspek yang disebut dengan segitiga api (*fire triangle*) bertemu, menurut teori terdapat 3 faktor yang menjadi aspek terjadinya kebakaran yaitu:

1. Bahan Bakar: Bahan yang dapat terbakar, yang menjadi sumber nyala api. Bahan bakar bisa berupa benda padat (seperti kayu atau kertas), cairan (seperti bensin atau alkohol), atau gas (seperti propana atau metana).
2. Panas: Sumber energi yang diperlukan untuk memulai pembakaran. Panas meningkatkan suhu bahan bakar hingga mencapai titik nyala, di mana bahan bakar mulai mengeluarkan gas yang dapat terbakar ketika bercampur dengan oksigen.
3. Oksigen: Gas yang diperlukan untuk mempertahankan reaksi pembakaran. Di atmosfer bumi, oksigen hadir dalam jumlah yang cukup untuk mendukung pembakaran, dengan konsentrasi sekitar 21%.



Gambar 2.1 Teori Segitiga Api

Sumber : Kelvin,P.E (2019)

### 2.1.2 Sensor Suhu DHT 22

Sensor DHT 22 adalah sensor yang memiliki *output* digital yang dapat digunakan untuk mengukur suhu dan kelembapan udara. Sensor DHT 22 memiliki stabilitas dan durabilitas yang tinggi apabila digunakan dalam kurun waktu yang lama. Sensor DHT 22 mampu mengukur suhu dengan rentang  $-40^{\circ}\text{C}$  hingga  $80^{\circ}\text{C}$  dengan akurasi  $\pm 0.5^{\circ}\text{C}$ . DHT 22 menggunakan prinsip resistif dan berbasis NTC (*Negative Temperature Coefficient*) yaitu resistansinya akan berubah apabila terdapat perubahan suhu. DHT 22 sangat mudah ketika diintegrasikan dengan berbagai mikrokontroler seperti Arduino, ESP, Raspberry Pi.



Gambar 2.1 Sensor suhu dan kelembapan DHT 22

Sumber : Bhoi, S. K. (2018)

Tabel 2.1 Sensor Suhu DHT 22

Model	DHT22
Tegangan	3,3 – 6 VDC
Output Signal	Digital signal via single-bus
Sensing Element	Polymer Capacitor
Measuring Range	Humidity 0-100%RH; temperature -40-80 Celsius
Accuracy	humidity +-2%RH temperature +-0.5 Celsius
Resolution or Sensivity	Humidity +-1%RH; temp 0.1 Celcius
Repeatability	Humidity+ _1%RH; temp +- 1 Celcius
Humidity hysteresis	+ -1&RH
Long-term Stability	+ _0.5%RH/year
Sensing Period	Average : 2s
Interchangability	Fully interchangeable
Dimension	Size 12*15.5*5.5mm

### 2.1.3 Sensor Gas CO MQ-2

Sensor gas MQ-2 adalah sensor yang digunakan untuk mendeteksi keberadaan gas yang mudah terbakar dan asap dalam konsentrasi yang berbeda-beda. Sensor gas MQ-2 memiliki sensitivitas konsentrasi diantara 200 ppm – 10.000 ppm. Sensor ini sangat sensitif terhadap gas LPG, *carbon monoxide* (CO), *alcohol*, hydrogen (H<sub>2</sub>), metana (CH<sub>4</sub>), dan propane. Cara kerja sensor MQ-2 berdasarkan prinsip konduktivitas gas, yang berarti bahwa konduktivitas sensor meningkat seiring dengan peningkatan konsentrasi gas dalam udara. Sensor ini memiliki sebuah material sensitif gas yang terbuat dari SnO<sub>2</sub> (*Tin Dioxide*), yang konduktivitasnya berubah dengan perubahan konsentrasi gas di udara.

Sensor Gas MQ-2 merupakan sensor gas yang memiliki jenis MOS (*Metal Oxide Semiconductor*). Prinsip pendeteksian sensor gas MQ-2 menggunakan prinsip *chemiresistor* karena pendeteksian gas berprinsip pada selisih nilai perubahan nilai resistansi lapisan sensor yang terbuat dari bahan oksida logam semikonduktor yaitu timah oksida (SnO<sub>2</sub>). Dengan

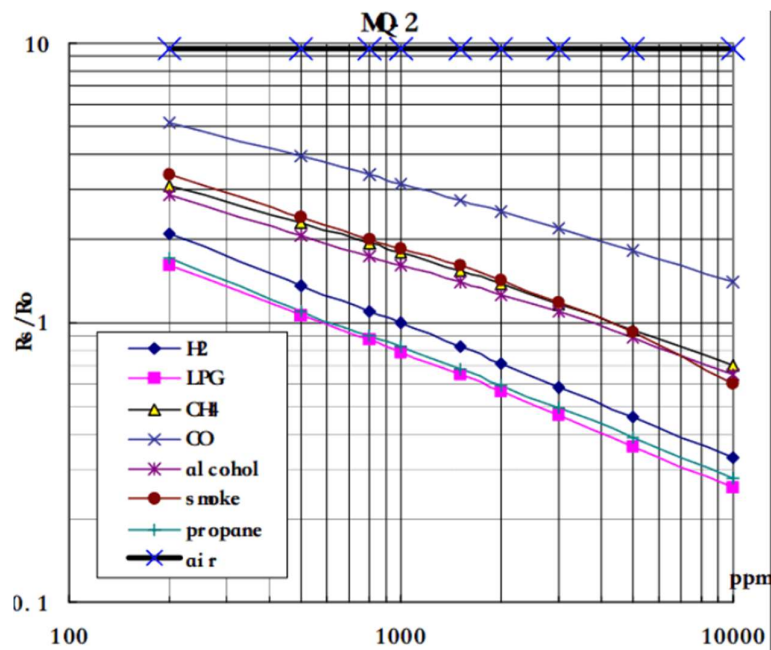
menggunakan rangkaian skematik pembagi tegangan maka nilai resistansi yang diukur akan berbanding lurus dengan konsentrasi gas yang terdeteksi. Secara garis besar berikut adalah rincian prinsip kerja sensor gas MQ-2 :

1. **Pemanasan (*Warming Up*)** : Pada sensor MQ-2 bagian dalam terdapat pemanas internal yang akan memanaskan lapisan oksida logam hingga suhu operasional yaitu sekitar  $300^{\circ}\text{C}$  -  $400^{\circ}\text{C}$ . Pemanasan ini menjadi langkah awal untuk pendeteksian gas melalui reaksi kimia antara gas yang akan dideteksi dan material sensor.
2. **Penyerapan dan Reaksi Kimia** : Pada Lapisan  $\text{SnO}_2$  yang sudah pada suhu panas operasional maka molekul gas eksternal akan diserap dan akan menghasilkan elektron bebas di dalam material semikonduktor yang akan diukur dengan perubahan resistivitas.
3. **Perubahan Resistivitas** : dengan meningkatnya elektron bebas pada material semikonduktor akan menurunkan resistivitas oksida logam, maka perubahan tersebut akan dideteksi oleh sensor.
4. **Pendeteksian dan Output Sinyal** : perubahan resistivitas material sensor MQ-2 akan diukur dan dikonversi menjadi sinyal listrik yang akan dikirimkan ke mikrokontroler sebagai pendeteksian suatu gas.



Gambar 2.2 Sensor Gas MQ-2

Sumber : Bhoi, S. K. (2019)



Gambar 2.3 Datasheet Gas MQ-2

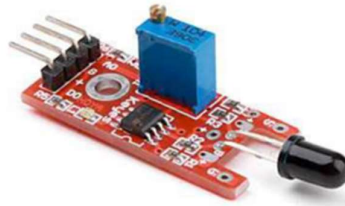
Pada gambar 2.3 menunjukkan *datasheet* yang digunakan untuk mengkalkulasikan hasil output nilai analog output yang dihasilkan oleh sensor gas MQ-2 yang memiliki rentang output analog 0-4095. Sehingga dari nilai sensor tersebut dapat mengidentifikasi jenis gas dan konsentrasi yang ditunjukkan dengan indikator ppm (*part per million*)

#### 2.1.4 Sensor *Flame KY-026*

Sensor api atau *flame* adalah alat *optic* untuk mendeteksi cahaya nyala api *menggunakan sensor optic*. Sensor *flame KY-026* dapat mendeteksi sumber api dan cahaya dengan rentang panjang gelombang atau lamda sebesar 760nm-1100nm dengan jarak kemampuan sensor 100 cm. Dalam penggunaan pada penelitian ini sensor api digunakan untuk mendeteksi keberadaan api, bukan mendeteksi panas. Kebakaran api dapat dideteksi dengan adanya spektrum inframerah dan ultraviolet, dari mana beberapa jenis mikroprosesor beroperasi di dalam sensor api untuk membedakan spektrum cahaya yang terkandung dalam api yang terdeteksi (Tripathi, 2022).

*Sensor flame* atau sensor api adalah alat pendeteksi kebakaran yang bekerja dengan mendeteksi nyala api dengan panjang gelombang 760 nm hingga 1.100 nm menggunakan *transduser infrared*. Sensor ini dapat digunakan di dalam ruangan maupun di luar ruangan yang bersifat terbuka. Terdapat beberapa jenis sensor flame, antara lain visual flame imaging detector, *ultraviolet/infrared flame detector*, dan *multi-spectrum IR flame detector*. Sensor flame dapat membantu mencegah kebakaran dengan mendeteksi nyala api secara dini dan memberikan peringatan kepada pengguna (Rayming, 2023).

*Sensor flame* bekerja dengan cara mendeteksi nyala api melalui spektrum cahaya yang dihasilkan oleh api. Sensor ini dapat mendeteksi nyala api dengan menggunakan beberapa metode, seperti deteksi sinar ultraviolet, deteksi sinar inframerah, deteksi panas, atau deteksi gas ionisasi. *Sensor flame* memiliki beberapa faktor yang perlu diperhatikan dalam pemilihan, antara lain sensitivitas, waktu respons, jangkauan deteksi, ketahanan terhadap alarm palsu, suhu operasi, persetujuan sertifikasi, dan keluaran output.

Gambar 2.4 *Flame Sensor*

Sumber : Bhoi, S. K. (2018)

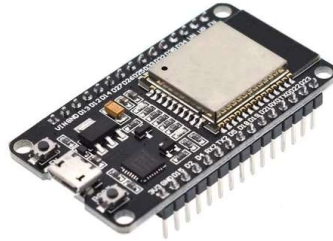
Tabel 2.2 Spesifikasi Flame Sensor

No	Spesifikasi	Keterangan
1	Tegangan Kerja	3,3 V ~ 5.3V
2	Sudut Deteksi	0 – 60 derajat
3	Range Spektrum	760nm ~ 1100nm
4	Operating temperature:	-25 C ~ 85 C
5	Jumlah pin	4 pin ( AOUT,DOUT, VCC, GND)

### 2.1.5 ESP 32 DEV-KIT V1

ESP32 DevKit V1 adalah papan pengembangan untuk IoT (*Internet of Things*) yang dilengkapi dengan modul ESP32-WROOM-32. Modul ESP32-WROOM-32 adalah mikrokontroler dengan mikroprosesor LX6 32-bit dual-core, yang beroperasi pada 160 atau 240 MHz, SRAM 520 kB, dan memori flash 4 MB. Modul ini juga memiliki transceiver WiFi 11b/g/n dan Bluetooth 4.2/BLE. Papan ini memiliki 30 pin, termasuk 25 pin GPIO, 15 pin saluran ADC, 25 pin PWM, dan 2 pin DAC. ESP 32 DevKIT v1 dapat diprogram menggunakan bahasa pemrograman, seperti Arduino IDE. Mikrokontoller ini dapat digunakan untuk berbagai aplikasi IoT seperti sistem keamanan dan sistem monitoring. Mikrokontroler ESP 32 DEC-KIT V1 juga memiliki port USB untuk daya dan pemrograman, dan dapat diberi daya melalui konektor USB Micro B on-board atau secara langsung melalui pin "VIN". Sumber daya dipilih secara otomatis, dan perangkat dapat

beroperasi dengan suplai eksternal 6 hingga 20 volt. Jika menggunakan lebih dari 12V, regulator tegangan dapat menjadi terlalu panas dan merusak perangkat. Papan ini juga memiliki berbagai antarmuka input/output termasuk GPIO, input analog, UART, SPI, dan I2C.



Gambar 2.5 ESP 32 DEV-KIT V1

*Sumber : Bhoi, S. K. (2018)*

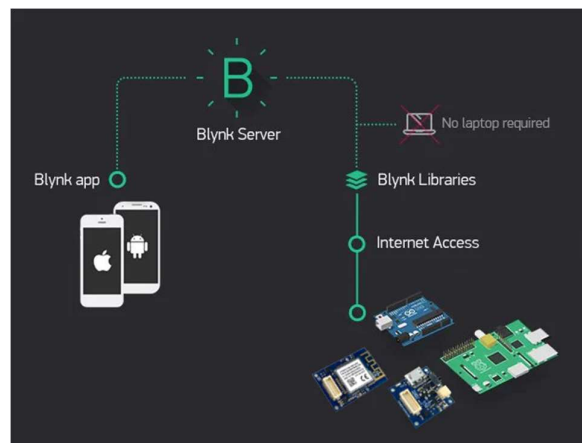
Tabel 2.3 Spesifikasi ESP-32 Dev-kit v1

Jumlah Core	2 (Dual Core)
Wi-Fi	2.4 GHz up to 150 Mbit/s
Bluetooth	BLE (Bluetooth Low Energy)
Architecture	32 bits
Clock Frequency	Up to 240 Mhz
RAM ( <i>Random-Access Memory</i> )	512 KB
Pins	30
Peripherals	<i>Capacitive touch, ADCs (analog-to-digital converter), DACs (digital-to-analog-converter), I2C (Inter-Integrated Circuit), UART ( universal asynchronous receiver/transmitter), CAN 2.0 (Controller Area Network), SPI (Serial Peripheral Interface), I2S (Integrated Inter IC Sound), RMI (Reduce</i>

	<i>Media-Independent Interface</i> ), <i>PWM (Pulse Width Modulation)</i> ,
--	---

### 2.1.5 Blynk IoT

Blynk adalah sebuah *platform* IoT (*Internet of Things*) yang dapat digunakan untuk mengintegrasikan perangkat lunak dan *hardware* secara *realtime*. Blynk IoT dapat menghubungkan hasil pembacaan sensor dan dikirimkan melalui wifi oleh mikrokontroler, dan juga Blynk IoT dapat menjadi pengatur *actuator* secara jarak jauh/*remote*.



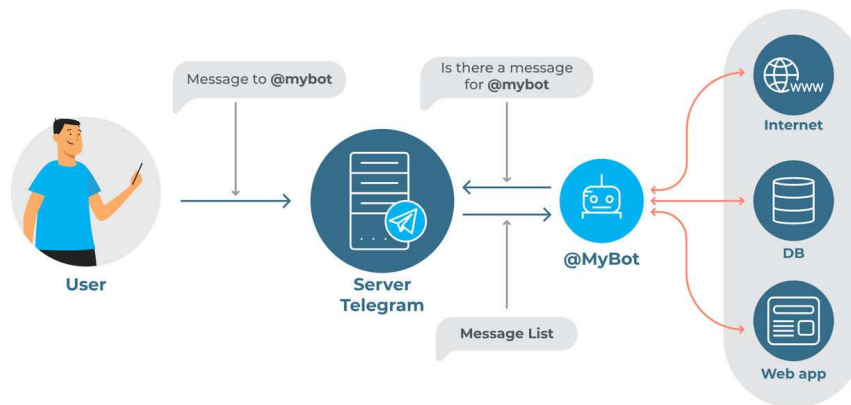
Gambar 2.6 Integrasi Blynk *app* dan mikrokontroler

Sumber Garcia. (2023)

Blynk *app* dapat terhubung dengan Raspberry Pi, ESP 8266, dan WEMOS D1 melalui akses internet dan *blynk server*. Konfigurasi untuk mengintegrasikan antara *sensor* dan *actuator* dapat dilakukan di website *blynk IoT dashboard*. Pada *website* tersebut *users* dapat mengatur secara mandiri sensor mana yang akan digunakan melalui virtual pin. Dan selanjutnya akan didapatkan *blynk id* dan *blynk authentication code* yang dapat *decompile* melalui Arduino IDE.

### 2.1.6 Telegram Bot API (*Application Programming Interface*)

Telegram Bot adalah algoritma *computer* pihak ketiga yang dijalankan di dalam *platform* pesan telegram. Bot telegram bekerja secara otomatis seperti robot digital dan dapat dijalankan oleh pengembang melalui program. Telegram chat bot dapat melakukan tugas secara otomatis seperti memberikan notifikasi secara *autonomous*. Telegram bot memungkinkan para pengembang untuk membangun suatu algoritma bot yang bisa berinteraksi dengan pengguna melalui pesan/perintah khusus.



Gambar 2.7 Gambaran sistem Telegram Bot

Sumber: Devian (2024)

Pada pengaplikasiannya, Telegram bot menggunakan telegram url dan telegram bot token untuk dapat terhubung dengan internet, *database*, dan *web application*. Untuk mengkonfigurasi dan menambahkan fungsi pada Telegram *bot* dapat menggunakan berbagai bahasa pemrograman yang mendukung HTTP *request*. Python adalah salah satu bahasa yang populer digunakan untuk ini karena memiliki *library* python-telegram-bot yang memudahkan interaksi dengan telegram Bot API.

### 2.1.7 *Internet of Things (IoT)*

*Internet of Things (IoT)* adalah konsep yang melibatkan berbagai perangkat atau benda yang ditanamkan dengan teknologi seperti *sensor* dan *software*, yang kemudian terhubung dan berkomunikasi satu sama lain

melalui jaringan komunikasi atau *internet*. IoT ini memungkinkan perangkat untuk mengumpulkan, mengolah, dan mengirimkan informasi secara otomatis, sehingga memudahkan berbagai proses dan kegiatan dalam kehidupan sehari-hari (Bin Suparman, 2019).

Ada tiga komponen utama dalam IoT, yaitu:

1. **Sensor:** Digunakan untuk mengumpulkan data dari objek-objek fisik, seperti suhu, kelembapan, dan gerakan.
2. **Gateway:** Berfungsi untuk mentransmisikan data dari sensor ke *cloud* atau internet yang terhubung.
3. **Cloud:** Memproses dan menyimpan data yang dikumpulkan oleh sensor dan *gateway*.

IoT memiliki berbagai manfaat dalam berbagai bidang, seperti pertanian, lingkungan, dan keamanan. Misalnya, dalam sektor pertanian, IoT dapat membantu mengumpulkan data tentang suhu, kelembapan, dan curah hujan, yang dapat digunakan untuk mengoptimalkan produksi dan keberlanjutan tanaman. Dalam sektor lingkungan, IoT dapat digunakan untuk mengelola sumber daya alam, mengurangi pemborosan, dan meningkatkan efisiensi (Binar, 2023).

IoT juga memiliki potensi untuk mengubah cara kita dalam menjaga kesehatan, dengan mengintegrasikan perangkat medis, seperti sensor melihat dan alat bantu dalam prosedur medis, ke dalam sistem kesehatan. Selain itu, IoT dapat membantu mengoptimalkan operasional bisnis di remote area dengan menggunakan sensor dan perangkat yang terhubung ke internet.

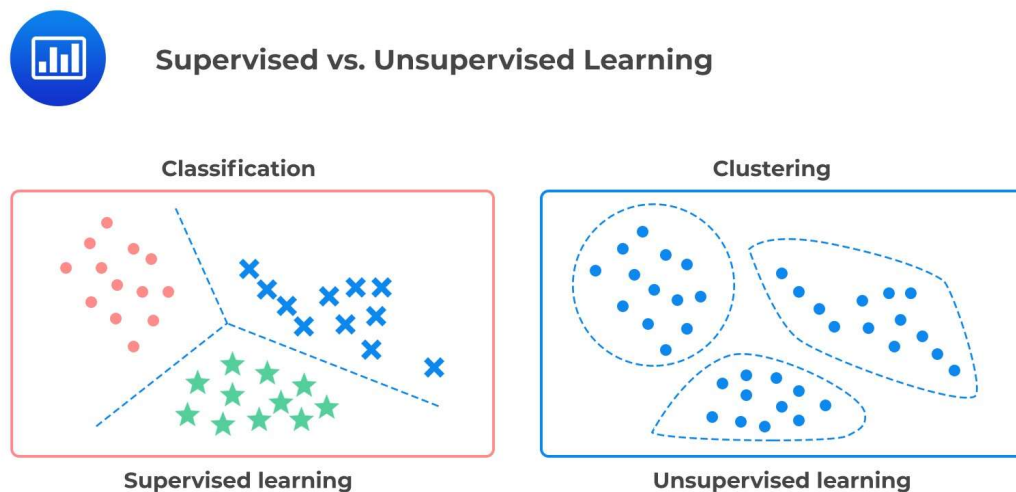
Dalam konteks keamanan, IoT dapat memainkan peran penting dalam mengatasi kebakaran, dengan menggunakan sensor dan perangkat yang terhubung ke internet untuk mengumpulkan informasi dan mengidentifikasi ancaman. Dengan adopsi IoT, kita dapat membangun

sistem deteksi kebakaran yang lebih cepat, efisien, dan efektif, yang dapat membantu melindungi nyawa dan aset manusia (Bin Suparman, 2019).

### 2.1.8 *Machine Learning*

*Machine Learning* atau dalam bahasa Indonesia pembelajaran mesin adalah salah satu cabang dari teknologi AI (*Artificial Intelligence*) yang mengembangkan sebuah program untuk dapat belajar dengan sendirinya tanpa arahan dari penggunanya. Proses *deep learning* berjalan menggunakan algoritma tertentu sehingga perintah yang diberikan kepada computer dapat dilakukan secara otomatis (Hairani, 2020).

Fase pada *machine learning* secara garis besar dibagi menjadi 2 tahap, yaitu fase *training* dan fase *application*. Pertama yaitu fase pelatihan/*training* yaitu pemrosesan dataset input menggunakan model *deep learning* dan fase selanjutnya yaitu *application* yaitu penggunaan model yang telah dilatih untuk menghasilkan sebuah Keputusan tertentu menggunakan *data testing*.



Gambar 2.8 *Algoritma Machine Learning*

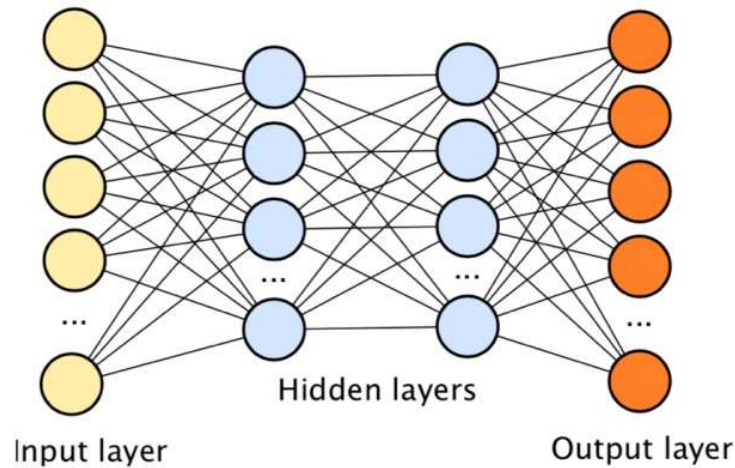
Sumber : Kelvin,P.E (2015)

Pada gambar 2.8 menunjukkan dua algoritma digunakan *machine learning* yaitu *supervised learning* dan *unsupervised learning*. berikut adalah penjelasan mengenai dua algoritma tersebut :

1. ***Supervised Learning*** adalah pelatihan model *machine learning* yang dilatih untuk mengenali pola antara masukan data dan label output. *Supervised learning* juga dapat mengidentifikasi korelasi antara *input* data dan *output* data. Perbedaan yang utama dari *unsupervised learning* adalah algoritma *supervised learning* menggunakan data yang telah diberi label khusus. Contoh dari algoritma *supervised learning* adalah regresi linear, K-NN (K-Nearest Neighbor), *Neural Networks*, *SVM (Support Vector Machine)*.
2. ***Unsupervised Learning*** adalah pelatihan model *deep learning* untuk mengenali pola inputan yang diberikan. Model *deep learning* ini dirancang untuk mempelajari secara mandiri bagaimana pola outputan yang akan didapatkan, karena itu maka algoritma ini memiliki tingkat yang kompleks dan rumit. Contoh dari *unsupervised learning* adalah *clustering* dan *K-Means*.

### **2.1.9 Deep Learning**

*Deep Learning* adalah bagian dari *deep learning* yang berkaitan dengan algoritma yang mempunyai prinsip dan struktur sistem seperti otak atau yang dapat disebut dengan (*neural networks*). Algoritma *deep learning* terdiri dari beberapa lapisan (*layer*), dimana setiap lapisan tersebut mempunyai tujuan untuk mengekstraksi dan memproses informasi dengan tingkat kompleksitas dan kesulitan yang berbeda. Semakin dalam lapisan dalam jaringan *neural networks* mampu mengidentifikasi informasi yang abstrak dan kompleks dibandingkan informasi pada jaringan sebelumnya. *Deep learning* dapat mengolah berbagai macam data seperti data suara, citra, bahasa, video dan lainnya.



Gambar 2.9 Struktur Algoritama *Deep Learning*

Sumber : Kelvin,P.E (2015)

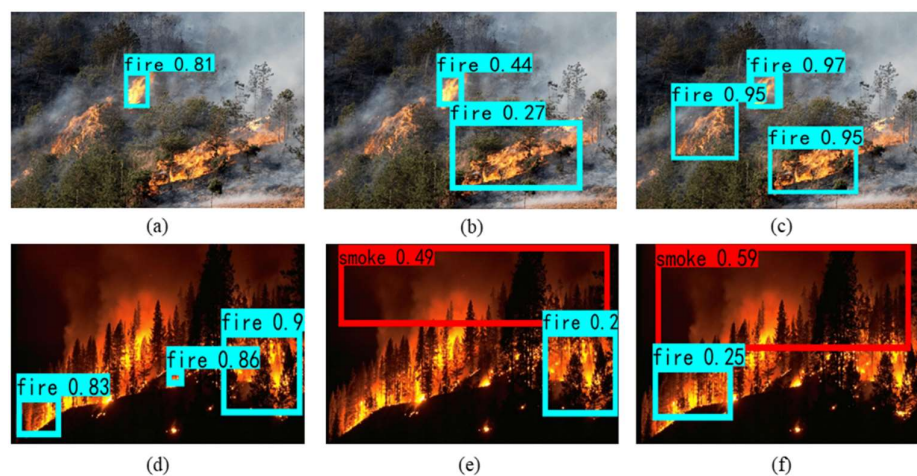
Walaupun *deep learning* adalah cabang dari *machine learning* tetapi terdapat beberapa perbedaan antar keduanya, yaitu :

1. **Peforma komputasi dan Jumlah data:** *Deep learning* memiliki performa yang lebih baik ketika data yang digunakan meningkat, sedangkan *machine learning* membutuhkan data pada saat awal *training* sebagai bahan pembelajaran.
2. **Sistem Algoritma** : *Machine learning* menggunakan sistem algoritma yang membutuhkan rekaya fitur manusia, sedangkan sistem *deep learning* secara otomatis memperlajari data dalam jumlah tinggi tanpa rekayasa fitur manusia.
3. **Waktu Komputasi** : *Deep learning* membutuhkan proses eksekusi algoritma yang lebih lama dibandingkan *machine learning* karena kompleksitas algoritma dan pengolahan data yang tinggi.

#### 2.1.10 Object Detection

*Object detection* adalah salah satu implementasi dari teknologi *deep learning* dan *computer vision* yang memungkinkan mesin untuk mengidentifikasi dan melacak berbagai objek dalam gambar atau video

(Crystal, 2023). *Object detection* dapat mendeteksi objek dalam suatu citra digital yang berasal dari inputan data citra. Dengan menggunakan algoritma dan kecerdasan buatan, *object detection* memungkinkan komputer untuk mengenali objek-objek tertentu dan menentukan posisi serta batas-batasnya dalam citra visual. Ini membantu dalam otomatisasi proses analisis visual yang sebelumnya hanya dapat dilakukan oleh manusia.

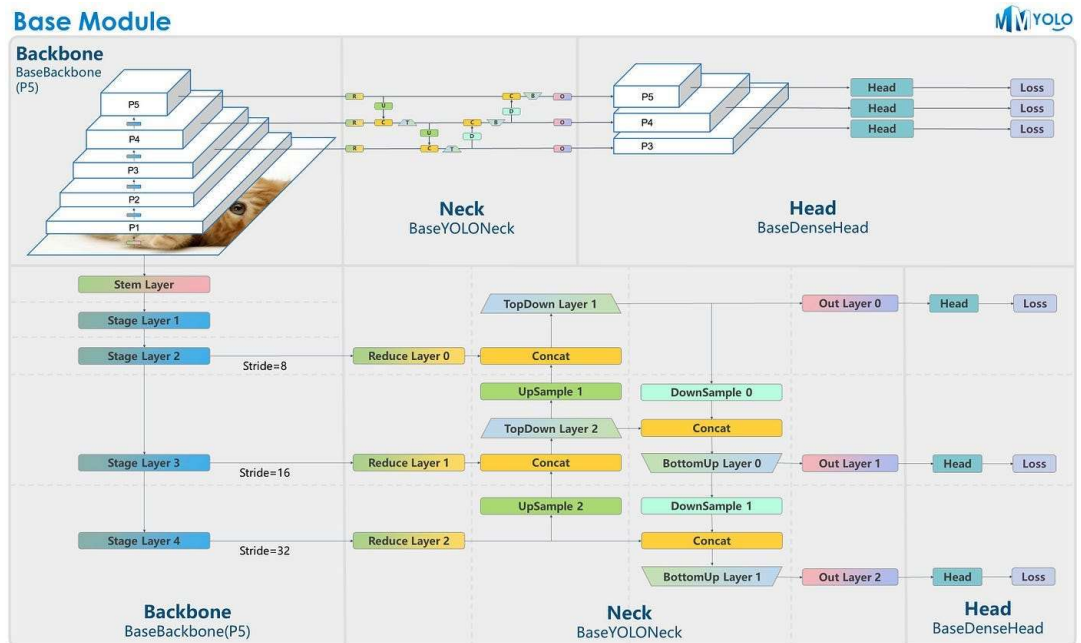


Gambar 2.10 *fire object detection*

Sumber : Crystal. (2023)

### 2.1.11 YOLO (*You Look Only Once*) v8

Yolo (*You Look Only Once*) adalah sebuah algoritma teknologi *computer vision* yang tujuan utamanya untuk mendeteksi objek dan klaasifikasi objek. YOLO pertama kali pada tahun 2015 dengan keluaran YOLOv1 dan selanjutnya YOLOv2, YOLOv3, YOLOv4 dan YOLOv8. Versi YOLOv8 adalah keluaran terbaru yang dikembangkan oleh Ultralytics, YOLOv8 menawarkan pengembangan terbaru pada sistem deteksi. YOLOv8 dapat diakses menggunakan *python package*. YOLOv8 memiliki *focus* pada peningkatan performa, fleksibilitas, dan peningkatan efisiensi untuk berbagai pendeteksian objek menggunakan *deep learning* (Tripathi, 2023).



Gambar 2.11 Arsitektur sistem YOLOv8

Sumber : Mujtaba. (2023)

Pada Gambar 2.11 dapat dilihat bagaimana arsitektur sistem YOLOv8 dimana YOLOv8 membanggakan API baru yang akan membuat pelatihan dan inferensi lebih mudah pada perangkat CPU dan GPU dan kerangka kerjanya akan mendukung versi YOLO sebelumnya (Rohit, 2023).

## 2.2 Hasil Penelitian Terdahulu

Berikut adalah penelitian yang telah dilakukan oleh peneliti terdahulu mengenai teknologi *fire detection* berbasis IoT serta YOLOv8:

Tabel 2.4 Hasil Penelitian Terdahulu

Penulis	Judul	Hasil	Kelebihan	Kekurangan
Jadon, et al. (2019).	A Specialized Lightweight Fire & Smoke Detection Model for Real-Time IoT Applications	Penelitian ini mendeteksi kebakaran yang berbasis <i>Internet of Things</i> (IoT) yang efektif dapat menggantikan sensor fisik serta mampu mengurangi masalah terkait penundaan pendeteksian sensor konvensional. Dalam penelitian ini digunakan <i>Smoke Sensor</i> , <i>Fire Alarm</i> , dan <i>Smoke Alarm</i> dengan menggunakan Arduino UNO sebagai mikrokontroler, yang diintegrasikan dengan Raspberry Pi 3B yang dilengkapi dengan sensor camera. Sistem pendeteksi ini juga dilengkapi dengan teknologi <i>deep learning</i> Algoritama <i>Convolutional Neural Networks</i> ('FireNet') yang dilatih menggunakan dataset <i>image</i> api dalam berbagai macam keadaan pada	Mengintegrasikan mikrokontroler Arduino dan Raspberry Pi 3B untuk melakukan deteksi kebakaran dengan menggunakan <i>smoke</i> sensor dan web cam di mana dilakukan <i>image processing</i> menggunakan CNN.	Penelitian ini hanya menggunakan satu sensor saja yaitu <i>smoke sensor</i> di mana sensor ini lebih <i>sensitive</i> pada berbagai asap, sehingga proses deteksi ini bisa tidak akurat.

		raspberry Pi 3B sehingga menghasilkan <i>sccuracy</i> sebesar 93.91 %.		
Sharma, et al. (2018).	An Integrated Fire Detection System Using IoT and Image Processing Technique for Smart Cities	Pada penelitian ini dirancang sistem untuk mendeteksi dini peristiwa kebakaran dengan menggunakan konsep <i>wireless sensor networks</i> (WSN) dan teknologi UAV. Penelitian ini menggunakan konsep <i>Internet of Things</i> (IoT). <i>Wireless sensor networks</i> (WSN) yang digunakan adalah sensor nirkabel, Unmanned Aerial Vehicle UAV dan cloud computing. <i>Sensor Nodes</i> yang digunakan yaitu HTS-200, MQ2 Sensor, TL Sensor, 2G/3G <i>Gateway Module</i> . Dan IoT menghasilkan waktu standar deviasi tercepat dengan menghasilkan 8.12 ms dibandingkan dengan GSMA 10.92 ms	Penelitian ini menggunakan <i>photodiode</i> untuk mendeteksi kebakaran dan melakukan pengolahan data dengan mengekstraksi gambar kebakaran tersebut dan segmentasi gambar sehingga dapat dilakukan pengenalan kondisi kebakaran tersebut	Jika melihat dari sensor yang digunakan, penelitian ini hanya menggunakan <i>photodiode</i> yaitu mendeteksi Cahaya. Photodiode ini memiliki kekurangan dalam mendeteksi kebakaran sehingga diperlukan sensor yang lebih kompleks lagi.
Bhoi, et al. (2018).	FireDS-IoT: A Fire Detection	Pada Penelitian ini merancang sistem Pendeteksi Kebakaran berbasis <i>Internet of Things</i> atau <i>Internet of Things based</i>	Proses penelitian ini memnggunakan beberapa sensor yang lebih kompleks seperti MQ-135, MQ-7, MQ-2, dan	Penelitian hanya mengambil data-data berupa angka sesuai parameter (tidak

	System for Smart Home Based on IoT Data Analytics	<i>Fire</i> (FireDS-IOT). Sistem ini menggunakan sensor MQ-135 (CO <sub>2</sub> ), MQ-2(smoke), MQ-7 (CO) dengan menggunakan Arduino UNO sebagai mikrokontroller. Data yang telah di dapatkan diklassifikasikan memnggunakan <i>Deep learning Model</i> yaitu <i>K-Nearest Neighbors</i> (KNN) dan <i>Decision Tree Deep learning alghoritms</i> dan menghasilkan akurasi sebesar 93.15% untuk K-NN Model dan 89.25% untuk Decision Tree Model.	DHT-11 kemudian dari data sensor akan dilakukan prediksi menggunakan KNN.	berupa gamabr) untuk diujikan pendeteksi kebakarannya menggunakan KNN.
Khalaf, et al. (2019).	An Improved Fire Detection Approach Based On YOLOv8 for Smart Cities	Pada Penelitian ini membandingkan performa akurasi yang dihasilkan dari berbagai algoritma <i>deep learning</i> dari mulai YOLOv2, YOLOv3, YOLOv4 ,YOLOv5, YOLOv6, YOLOv8. Penelitian ini menggunakan Dataset Api dalam berbagai kondisi, Sehingga didapatkan	Penelitian ini melakukan pendeteksian kebakaran menggunakan Node MCU dan sensor DHT-11 yang data sensor tersebut akan dikirimkan ke handphone sebagai alarm atau peringatannya.	Penelitian ini hanya menggunakan sensor DHT-11 dan tanpa menggunakan proses pengenalan gambar dengan menggunakan kecerdasan buatan atau deep learning. Kemudian, alat hanya bisa mendeteksi

		hasil sebesar 97 %, 98,1 %,98,2 %,94,99 %,93.48 %,97.5 %.		kebakaran melalui kondisi suhu.
Talaat, ZainEldin. (2023).	IoT Fire Detection System Using Sensor With Arduino	Pada penelitian ini membahas mengenai integrasi sensor MQ-2, MQ-135, MQ-5 untuk mendeteksi adanya api menggunakan mikrokontroller Arduino.	Deteksi kebakaran pada penelitian ini menggunakan kecerdasan buatan yaitu algoritma YOLOv8 yang mengolah dataset kebakaran.	Penelitian ini hanya menggunakan dataset gambar yang akan diolah dengan algoritma dan tanpa menggabungkannya dengan <i>internet of things</i> yang dapat mendeteksi kebakaran

Berdasarkan penelitian terdahulu yang telah peneliti analisis, maka terdapat masalah yang belum teratasi, yaitu belum adanya integrasi antara IoT (*Internet of Things*) dan teknologi *deep learning*. Berdasarkan tabel di atas terdapat 5 penelitian terdahulu yang telah peneliti analisis, berikut adalah penjelasan dari penelitian terdahulu tersebut :

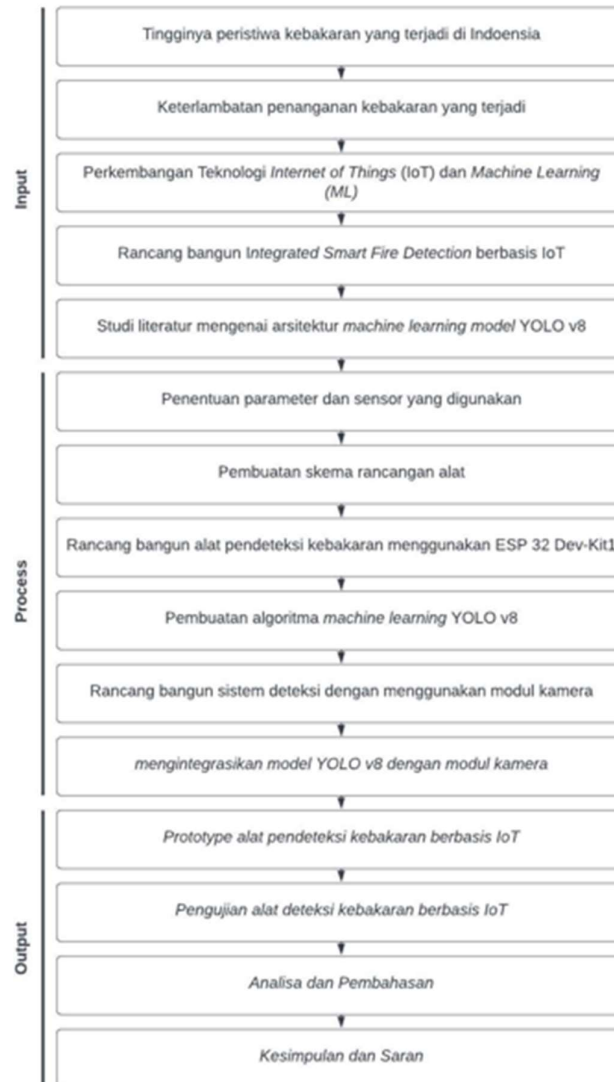
1. Penelitian pertama oleh Jadon dengan judul *A Specialized Lightweight Fire & Smoke Detection Model for Real-Time IoT Applications* menggunakan Arduino dan Raspberry Pi 3B sebagai mikrokontroler dan *smoke sensor* sebagai inputan sensor. Pada penelitian ini telah menggunakan CNN (*Convolutional Neural Network*). Sistem yang telah dirancang belum menggunakan banyak parameter sensor sehingga pengukuran sensor yang dilakukan masih terbatas.
2. Penelitian kedua oleh Sharma dengan judul *An Integrated Fire Detection System Using IoT and Image Processing Technique for Smart Cities* menggunakan teknologi UAV (*Unmanned Aerial Vehicle*) dan IoT (*Internet of Things*) dan menggunakan satu sensor yaitu *photodiode*. Pada penelitian ini parameter sensor masih terbatas karena hanya menggunakan 1 parameter pengukuran.
3. Penelitian ketiga oleh Bhoi dengan judul *FireDS-IoT: A Fire Detection System for Smart Home Based on IoT Data Analytics* menggunakan multisensor gas yaitu MQ-135, MQ-7, MQ-2, dan DHT-11 kemudian dari data sensor akan dilakukan prediksi menggunakan KNN. Tetapi pada penelitian ini tidak menggunakan teknologi *video processing object detection* dan hanya berfokus pada pengolahan data sensor yang didapatkan.
4. Pada penelitian keempat oleh Khalaf dengan judul *An Improved Fire Detection Approach Based On YOLO-v8 for Smart Cities* membandingkan hasil *training* algoritma model YOLO yaitu YOLO v2, YOLOv3, YOLOv4, YOLOv5, YOLOv6, YOLOv8. Serta integrasi dengan mikrokontroler Node MCU dan sensor DHT 11. Tetapi pada penelitian ini tidak terdapat *platform monitoring* secara langsung secara realtime terhadap *users*.

5. Pada penelitian kelima oleh Taalat dengan judul *IoT Fire Detection System Using Sensor With Arduino*. Penelitian ini menggunakan Arduino sebagai mikrokontroler dan YOLOv8 sebagai *algoritma object detection*. Pada penelitian ini menggunakan multi sensor yaitu MQ-2, MQ-135, MQ-5. Salah satu kekurangan dari penelitian ini adalah belum ada integrasi antara hasil *object detection* dan pengukuran sensor secara integratif pada satu *platform*.

Berdasarkan 5 penelitian terdahulu diatas masih terdapat kurangnya integrasi antara teknologi IoT dan *deep learning*. maka peneliti berinisiasi untuk merancang sebuah *prototype* integratif antara hasil pengukuran multi sensor dari mikrokontroler dan hasil *object detection* dari *deep learning*. Peneliti juga merancang *prototype* yang memiliki kemampuan untuk memberikan notifikasi apabila terdapat kebakaran dengan dua *platform* yaitu Blynk IoT dan telegram guna memberikan informasi pendeteksian secara *autonomus* dan *realtime*.

### 2.3 Kerangka Berpikir

Kerangka Berpikir terkait dengan penelitian Implementasi *Artificial Intelligence* pada *integrated smart fire detection berbasis IoT (Internet of Things)* terdapat tiga bagian besar yaitu input, proses dan output seperti bagan 2.1 dibawah :



Bagan 2.1 Diagram Alir Kerangka Berpikir

Dari bagan 2.1 Proses Implementasi *Artificial Intelligence* Pada *integrated smart fire detection* Berbasis IoT (*Internet of Things*) memiliki beberapa langkah dari mulai proses akuisisi data oleh sensor hingga menghasilkan *output notification*. berikut adalah penjelasan mengenai *input, proses dan output* penelitian ini :

### **2.3.1 Input**

Pada Bagian Input, penulis menganalisis data bahwa ternyata tingkat kebakaran di Indonesia masih tinggi dikarenakan Keterlambatan penanganan kebakaran yang terjadi. Sedangkan disisi lain terdapat perkembangan teknologi *Internet of Things* (IoT) dan *deep learning* (DL) yang dapat membantu manusia memonitor secara *autonomus* dan *realtime*. Sehingga penulis merancang suatu sistem *prototype integrated smart fire detection* berbasis IoT serta ditambah dengan studi literatur mengenai arsitektur *deep learning* model YOLOv8, maka sistem tersebut dapat ditambah dengan teknologi *deep learning* yang dapat mendeteksi adanya api menggunakan teknologi *object detection*.

### **2.3.2 Proses**

Pada bagian proses ditentukan parameter dan sensor yang akan digunakan serta pembuatan skema rancangan alat. Selanjutnya *prototype* akan dirancang menggunakan ESP-32 Dev-Kit1 serta pembuatan algoritma *deep learning* YOLOv8. Langkah berikutnya yaitu merancang sistem *object detection* menggunakan modul kamera yang telah diintegrasikan dengan *deep learning model* YOLOV8.

### **2.3.3 Output**

Pada bagian output di kerangka penelitian ini yaitu akan dibuat sebuah Prototype alat pendeteksi kebakaran berbasis IoT dan alat tersebut akan dilakukan pengujian sehingga dapat dianalisa bagaimana kinerja ketepatan dan keakuratan alat tersebut. Selanjutnya hasil analisa akan dibahas dan disimpulkan bagaimana hasil dari penelitian alat pendeteksi kebakaran berbasis IoT.