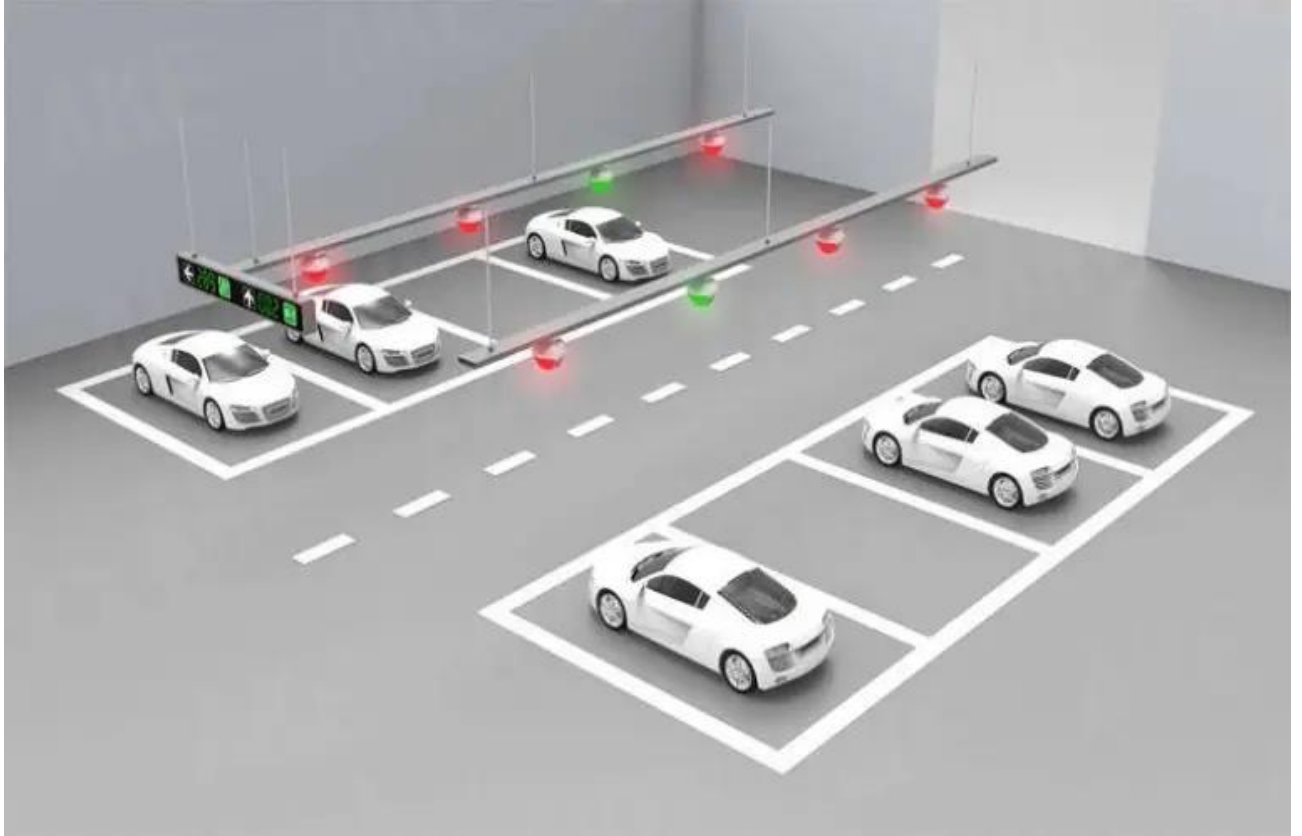


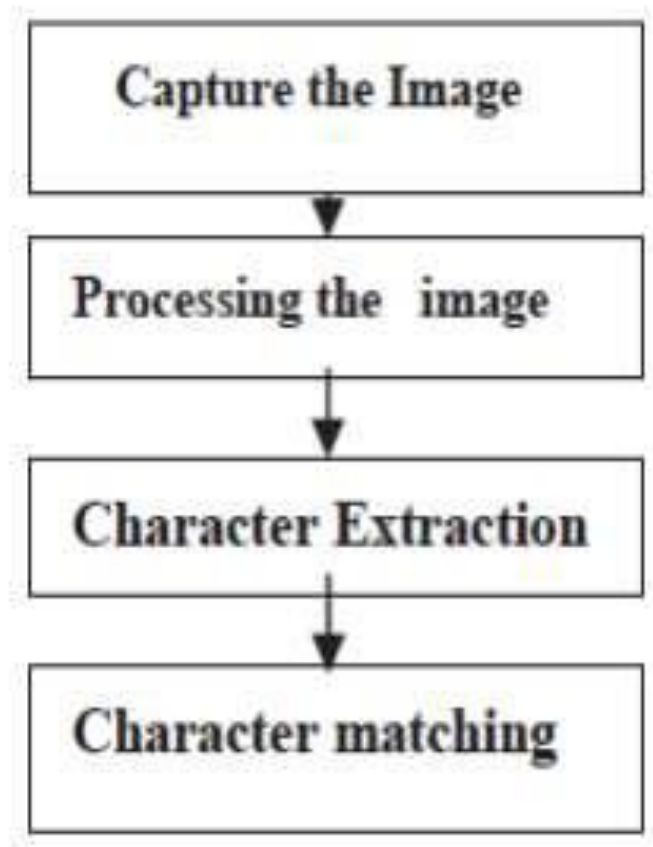
# SMART PARKING



## Technologies Used:

- IOT, Advanced Smart Parking System, Sensors, Automatic License Plate Recognition System, Optical Character Recognition.

## Steps:



### **A. Capturing of Images:**

In this process a high resolution analog /digital camera is used to capture the image.

### **B. Processing the Image:**

In this process firstly gray scale conversion is used to convert the image from RGB to grayscale. After that image will be resized by resize function. Then median filter is used to remove the salt and pepper noise.

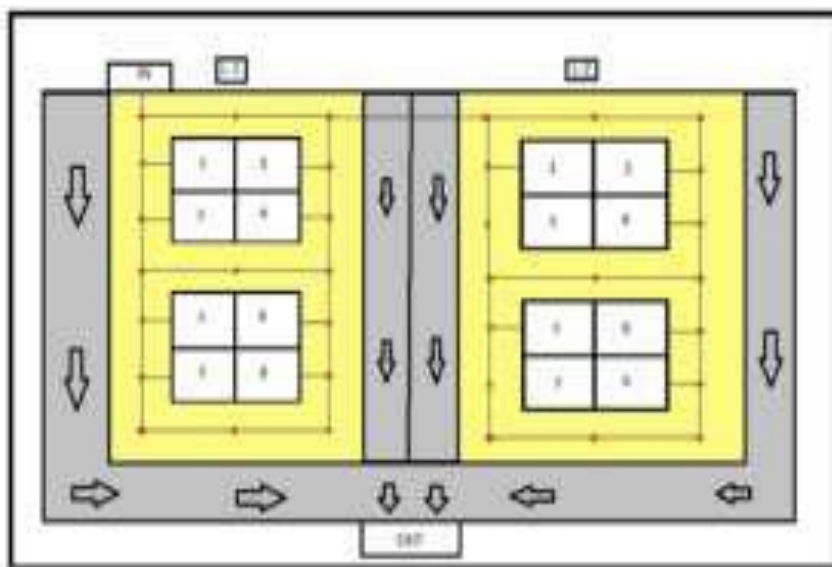
### **C. Character Extraction:**

After median filtering process we used smearing process to find out the text area from the plate and get the erode image Fig.4 and then we applied morphological process to remove the unwanted edges of the plate as shown in Fig.5, it also include dilation process. Dilation means to fill the gap/to separate the character from the image and after that each character is cut separately from done by finding starting and end points of the characters in horizontal and vertical direction .Characters cut from plate areas.

#### **D. Character Matching:**

Before the Character Matching Normalization process is occurring in order to normalize the character means not to include any other white or extra spaces in all the four sides of the character. Then each character is to fit to equal size.

#### **Working Model:**



## Arduino Code for Parking System:

```
#include <SoftwareSerial.h> //Software Serial library

const int ProxSensor1=5;
const int ProxSensor2=6;
const int ProxSensor3=7;


int inputVal = 0;

SoftwareSerial mySerial(12, 13);
SoftwareSerial espSerial(2, 3); //Pin 2 and 3 act
as RX and TX. Connect them to TX and RX of ESP8266

#define DEBUG true

String mySSID = "ram"; // WiFi SSID
String myPWD = "ramnath123$"; // WiFi Password
String myAPI = "I3SMDXSR4EMUK7CR"; // API Key
String myHOST = "api.thingspeak.com";
String myPORT = "80";
String myFIELD1 = "field1";
String myFIELD2 = "field2";
String myFIELD3 = "field3";
int sendVal1; int sendVal2; int sendVal3;
int k=0;


void setup()
{
  Serial.begin(9600);
  mySerial.begin(115200);
  espSerial.begin(115200);
```

```

    pinMode(ProxSensor1, INPUT);    //Pin 2 is
connected to the output of proximity sensor
    pinMode(ProxSensor2, INPUT);
    pinMode(ProxSensor3, INPUT);

    espData("AT+RST", 1000, DEBUG);
//Reset the ESP8266 module

    espData("AT+CWMODE=1", 1000, DEBUG);
//Set the ESP mode as station mode

    espData("AT+CWJAP=\""+ mySSID +"\", \""+ myPWD
+"\"\"", 1000, DEBUG);    //Connect to WiFi network

    /*while(!esp.find("OK"))
    {
        //Wait for connection
    }*/
    delay(1000);

}

void loop()
{

    sendVal1=digitalRead(ProxSensor1);
    sendVal2=digitalRead(ProxSensor2);
    sendVal3=digitalRead(ProxSensor3);

    String sendData1 = "GET /update?api_key="+
myAPI +"&"+ myFIELD1 +"="+String(sendVal1);

    String sendData2 = "GET /update?api_key="+
myAPI +"&"+ myFIELD2 +"="+String(sendVal2);

```

```

    String sendData3 = "GET /update?api_key="+
myAPI +"&" + myFIELD3 +"="+String(sendVal3);

    espData("AT+CIPMUX=1", 1000, DEBUG);
//Allow multiple connections

    espData("AT+CIPSTART=0,\"TCP\", \""+ myHOST
+"\", "+ myPORT, 1000, DEBUG);

    espData("AT+CIPSEND=0, "
+String(sendData1.length()+4), 1000, DEBUG);

    espSerial.find(">");

    espSerial.println(sendData1);

    espData("AT+CIPMUX=1", 1000, DEBUG);
//Allow multiple connections

    espData("AT+CIPSTART=0,\"TCP\", \""+ myHOST
+"\", "+ myPORT, 1000, DEBUG);

    espData("AT+CIPSEND=0, "
+String(sendData2.length()+4), 1000, DEBUG);

    espSerial.find(">");

    espSerial.println(sendData2);

    espData("AT+CIPMUX=1", 1000, DEBUG);
//Allow multiple connections

    espData("AT+CIPSTART=0,\"TCP\", \""+ myHOST
+"\", "+ myPORT, 1000, DEBUG);

    espData("AT+CIPSEND=0, "
+String(sendData3.length()+4), 1000, DEBUG);

    espSerial.find(">");

    espSerial.println(sendData3);

    Serial.print("Value to be sent: ");

    Serial.println(sendVal1);

    Serial.print("Value to be sent: ");

    Serial.println(sendVal2);

    Serial.print("Value to be sent: ");

    Serial.println(sendVal3);

```

```
if(k==0&&sendVal1==1&&sendVal2==1&&sendVal3==1)
{
    SendMessage();
    k=1;
}
```

```
if (mySerial.available()>0)
    Serial.write(mySerial.read());
    espData("AT+CIPCLOSE=0",1000,DEBUG);
    //delay(100);
}
```

```
String espData(String command, const int timeout,
boolean debug)
```

```
{
    Serial.print("AT Command ==> ");
    Serial.print(command);
    Serial.println("      ");

    String response = "";
    espSerial.println(command);
    long int time = millis();
    while ( (time + timeout) > millis())
    {
        while (espSerial.available())
        {
            char c = espSerial.read();
            response += c;
        }
    }
}
```

```

    }
    if (debug)
    {
        //Serial.print(response);
    }
    return response;
}

}

```

In the code, we have just used different digital pins for different IR sensors. They will be giving the input to the controller.

### *Registration:*

**SIGNUP PAGE:** The user will enter the details such as first name, last name, email address and password. These details will be used for login into our website.

```

<?php
require_once('config.hp');
?>

<!DOCTYPE html>

<html>

<head>
    <title>User Registration | PHP</title>
    <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
</head>

<body>

<div>
    <?php

    ?>

```



</div>

<div>

<form action="registration.php" method="post">

<div class="container">

<div class="row">

<div class="col-sm-3">

<h1>Registration</h1>

<p>Fill up the form with correct values.</p>

<hr class="mb-3">

<label for="firstname"><b>First Name</b></label>

<input class="form-control" id="firstname" type="text"  
name="firstname" required

<label for="lastname"><b>Last Name</b></label>

<input class="form-control" id="lastname"  
type="text" name="lastname" required>

<label for="email"><b>Email Address</b></label>

<input class="form-control" id="email" type="email" name="email"

<label for="phonenumber"><b>Phone Number</b></label>

<input class="form-control" id="phonenumber"  
type="text" name="phonenumber" required>

<label for="password"><b>Password</b></label>

<input class="form-control" id="password"  
type="password" name="password" required>

<hr class="mb-3">

<input class="btn btn-primary" type="submit"  
id="register" name="create" value="Sign Up">

```
</div>
```

```
</div>
```

```
</div>
```

```
</form>
```

```
</div>
```

```
<style>
```

```
h2{
```

```
    text-align:center;
```

```
}
```

```
body
```

```
{background-color:lightblue;
```

```
}
```

```
.row
```

```
{
```

```
    position:relative;
```

```
    left:400px;
```

```
    top:60px;
```

```
}
```

```
.col-sm-3
```

```
{
```

```
    height: 700px;
```

```
    width: 1050px;
```

```
    margin-top: auto;
```

```
    margin-bottom: auto;
```

```
    background: #f39c12;
```

```
    position: relative;
```

```
    display: flex;
```

```
    justify-content: center;
```

```
    flex-direction: column;
```

```
    padding: 10px;
```

```
    box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2), 0 6px 20px 0 rgba(0, 0, 0, 0.19);
```

```
-webkit-box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2), 0 6px 20px 0 rgba(0, 0, 0, 0.19);
-moz-box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2), 0 6px 20px 0 rgba(0, 0, 0, 0.19);
border-radius: 5px;
}
</style>
```

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/sweetalert2@8"></script>
<script type="text/javascript">
    $(function(){
        $('#register').click(function(e){

            var valid = this.form.checkValidity();

            if(valid){

                var firstname    = $('#firstname').val();
                var lastname     = $('#lastname').val();
                var email         = $('#email').val();
                var phonenumber  = $('#phonenumber').val();
                var password      = $('#password').val();

                e.preventDefault();

                $.ajax({
                    type: 'POST',
                    url: 'process.php',
                    data: {firstname: firstname,lastname: lastname,email: email,phonenumber: phonenumber,password: password},
                    success: function(data){
```

```
        Swal.fire({
            'title': 'Successful',
            'text': data,
            'type': 'success'
        })

    },
    error: function(data){
        Swal.fire({
            'title': 'Errors',
            'text': 'There were errors while saving the data.',
            'type': 'error'
        })
    }
});

}else{
}

});

});
</script>
</body>
</html>
```

## Conclusion:

The Internet of Things integrates the hardware, software and network connectivity that enable objects to be sensed and remotely controlled across existing network. Such integration allows users to monitor available and unavailable parking spots that lead to improved efficiency, accuracy and economic benefit.