ISSN: 2321-2152

IJMECE

International Journal of modern electronics and communication engineering

E-Mail
editor.ijmece@gmail.com
editor@ijmece.com

www.ijmece.com

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 1, 2025

Development and Analysis of Surveillance Robot For Military Applications

Sudhakar

Electronics & Communication Engineering, St.Martin's Engineering college, India, E-mail: ksudhakarece@smec.ac.in

Abstract

1. The Project primary goal is to construct a robot for industrial and military surveillance. The goal of the project is to use Android smartphone to construct a robotic vehicle. Wireless camera installed for remote operation and monitoring. Real-time video is wirelessly sent by the robot and camera, which can see in low light. This type of robot will be useful in conflict zones for espionage purposes. Compared to other technologies, Wi-Fi technology is relatively new, but it has a lot of room to grow and find practical uses. Following these commands, the security system responds to the user. For remote surveillance, the security system is connected to the ESP 32 camera. An individual will be in charge of this type of robot.

2. The importance of surveillance in hospital settings has increased due to the aging population. In this study, we demonstrate a hospital robot that uses audio and visual data to identify anomalous events. In our method, a passive acoustic localization device allows the robot to detect moving targets. The robot then uses a particle filter technique to track the objects. The update system updates the target model on a regular basis to accommodate variations in lighting. In order to guarantee reliable monitoring, the robot tracks an individual's upper body to identify unusual human behavior. Mel frequency costrel coefficients (MFCC) are used to extract features from audio data for audio surveillance. A support vector machine classifier receives these features as input for processing. According to experimental findings, the robot is capable of identifying unusual actions like "running" and "falling down." Additionally, an accuracy rate of 88.17% is attained in the identification of anomalous audio data, such as "crying," "groaning," and "gun shooting." By using a passive acoustic locating device to guide the robot to the area of abnormal occurrences, the abnormal sound detection system reduces false alarms. The robot can then use its camera to confirm the events' occurrence. Finally, the robot will send the image it took to the master's cell phone.

3. INTRODUCTION

4. A surveillance robot is a robotic system designed and equipped for monitoring and collecting information in various environments. These robots are often employed in situations where human presence may be challenging, risky, or impractical. Surveillance robots use sensors, cameras, and other technologies to gather data and transmit it to a control center or

operator. Which do not require sleep, are not hungry, do not experience emotions, and simply carry out their duties and obey orders. Nothing can be more important than human life; the use of such robots can help save countless lives in border areas, freeing up human labor for other tasks. In this scenario, the robot is controlled by an Android device.

5. Small intelligent robotic vehicles have advanced significantly in the last several years for a variety of uses. These vehicles are now essential in the fields including wildlife research, disaster relief, defense and data collection in dangerous environments. The goal of the project is to create a robotic vehicle intended only for data collection.

6. The real-time gathering, processing and timely distribution of data to the operator constitute surveillance. In defense applications, surveillance is critical to keeping an eye out and taking the appropriate precautions to defend its inhabitants. The job of keeping an eye on the conditions is called surveillance. In a military context, this usually happens when battle zones and enemy territory are being monitored. Experienced workers in close proximity to sensitive places conduct human monitoring in order to continuously look for changes. However, there is always a greater chance of losing employees when the enemy catches you. Thanks to advancements in technology in recent years, it is now possible to utilize robots to remotely monitor key locations.

7. Wi-Fi is used to establish a connection with the microcontroller. The robot's wheels at its base allow it to move on both uneven and wet surfaces. Thus, all of the opponent's activity records are accessible to the person seated at the output screen. When human access is not feasible, these robots can readily replace industrial workers and warriors by doing their duties with better improvements. Thus, these intelligent robots will handle combat in the future, which reduce the risk to human life during conflicts. In the future, these robots can used in place of security guards.

8. RELATED WORK

9. The reviews and inspections of the industries potentially dangerous regions, this technology offer a new tool to improve safety in huge enterprises. This will be accomplished by integrating the management of motions on a virtual map and outfitting the system with a navigation system that lets the operator maneuver the robot without worrying about impediments.

10. The goal of the mobile robot's stochastic strategy is to increase the likelihood that it will successfully apprehend the burglar. Describe the strategic interactions between the surveillance robot and the intruder as a Stackelberg game. Firstly, obtain a general upper bound on the capture probability, which represents the surveillance agent's performance limit. It demonstrate that this upper limit is tight across the whole graph and additionally offer sub optimality assurances for the random walk. First characterize dominating tactics for the intruder and the surveillance agent for the star and line graphs. Next, firmly establish the best

course of action for the surveillance agent.

11. The robot will be able to recognize people and read meters: thanks to the system's computer vision capabilities. In a busy area, it can identify unusual activity like someone suddenly sprinting or falling to the ground. The computational action models integrated into the trained support vector machines enable the robot to automatically determine whether freshly acquired audio information is normal by teaching it the distinction between normal and abnormal sound information. The robot can use its camera to further investigate the events controlled by the passive acoustic location device if anomalous audio information is found. Many video surveillance systems have been created to detect and track multiple individuals. Moreover, the primary purpose of none of these systems is to identify deviant behavior. Since it can understand the monitoring method and is aware of the mobile agent's current location, it is presumed that the invader is omniscient.

12. The mobile robot's objective is to create a stochastic plan that will increase the likelihood that it will catch the burglar. Their uses include asset, property and personal security in addition to monitoring and intrusion detection. Benefits like lower costs and more staff expertise in security management are also brought about by it. As a result, a lot of businesses have started implementing autonomous robots to enhance their security and surveillance processes.

13. Robotics' advantages for security and surveillance. Companies want to stay safe at all times because it's such a big problem, and robots like Spot are quite helpful at doing these kinds of jobs.

14. BACKGROUND

15. In order to defend against emerging risks and attacks, many organizations have realized they must look beyond conventional security solutions. The most adaptable and economical way to meet these security and surveillance needs is to use security robots.

16. For years, Plain Concepts has assisted businesses in the industrial and construction sectors with solutions similar to this one, as well as with digitization processes utilizing cutting-edge technologies. Spot, the most sophisticated quadruped robot available is one example. Currently integrating sensors and artificial intelligence (AI) onto its initial hardware, which includes 360-degree vision, unique movement on all surfaces, the ability to carry up to 14 kg, the ability to climb stairs or ramps, the ability to avoid obstructions, etc.

17. Spot can gather data for inspection jobs in dangerous situations, such as high-voltage installations or high-pressure steam pipelines, by accessing places that are inaccessible to humans. In addition, it can map and scan seismic activity, radiation levels, and dangerous gas concentrations both before and after an explosion.

18. With a stronger emphasis on the construction industry, the robot may map new project installations, develop digital twins, assess the status of work sites, and compare real-world circumstances with building

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 1, 2025

information modeling.

19. In an environment (often represented by a graph), mobile robots patrolling and travel between areas in an attempt to apprehend possible intruders. The intruder and the mobile robot play a Stackelberg game in which the surveillance agent's best course of action is determined by presuming that the intruder will act in her best interest. When the surveillance agent is up against the most formidable foe, this formulation represents the worst-case situation. When little to nothing is known about the invader, the equivalent Stackelberg solution makes sense.

20. The task is to find a path for each robot that minimizes the worst idleness for all sensing locations so that the network of robots and base station is connected throughout the mission, given a representation of the area, the number of robots, and the positions of the points of interest (which denote as sensing locations) and the base station. The time elapsed from the last robot visit to a sensing location is the definition of idleness at that point in time, and the worst idleness is the total idleness over all sensing locations and the mission period. Strategies that produce answers over an indefinite time horizon are required because the lifetime of a persistent surveillance mission may be infinite. Because of this connectivity limitation, monitoring an environment continually for an extended length of time is known as persistent surveillance. Robots need a movement strategy that ensures each point of interest in the region is visited by the robots on a periodic basis due to their restricted sensor view and possibly larger area. It is also essential in disaster response scenarios that the mission operators are informed of the situation at all times. This suggests that the robots must notify the base station on a regular basis about the mission's status and the data they have collected. Robots using wireless transceivers can exchange data over certain distance only; if a continuous connection to the base station is needed and the region is wider than the communication range, the data must be transmitted across numerous hops. For safety purposes, it is also possible to track the status of the (aerial) robots, thanks to a network of robots that is always connected to the base station.

21. PROPOSED DESIGN

22. The Internet of Things (IoT) is thought of as a very sophisticated platform that connects things by tagging them for identification. It also consists of sensors, actuators and other technologies. Many open wireless technology standards, such as Bluetooth, Wi-Fi, RFID and smart wireless microcontroller boards like Arduino, Raspberry Pi, and ESP8266 enable the connectivity of objects on the Internet. The IoT is revolutionizing a number of commercial industries, including manufacturing, business administration, robotics, smart intelligent transportation systems and even agriculture. Given the benefits and current advancements of IoT, many research institutions and IT organizations are investing large sums of money in automation systems made possible by IoT. By CISCO predicted that about 30 billion devices would be connected to the Internet of

Things [3], while Morgan Stanley predicted that there would be more than 80 billion devices connected by that year. A number of IT including Microsoft, Facebook, Google and robotics companies like Kuka, are investing billions of dollars in IoT research and will soon release a variety of products, including Google's self-driving autonomous cars, Facebook's Oculus Rift, Microsoft HoloLens, and even a number of companies are developing IoT-based robots and drones. The creation of intelligent real-time robots based on the (IoT) is a major area of focus for researchers and IT organizations worldwide. These robots enable real-time monitoring, perform daily tasks autonomously using smart sensors and even incorporate the idea of "cloud computing" into their overall data management.

23. The current state of computer and mobile computing is not at all like the future wave that will rule the computing era. The Internet of Things is the next big thing that will change computers forever. Observing the situation from the last few months, it is clear that virtually every research component leading the computing "Internet of Things" (IoT) always reflects a new phrase. Currently, five million individuals utilize the Internet for a variety of purposes, including social networking, online gaming, email and web browsing. An increasing number of people have access to the Internet, which serves as a robust platform for communication, computation, and networking amongst people in order to share information.

24. There's been a resurgence of interest in security robots due to recent advancements in AI. Based on the Raspberry Pi, an autonomous interior surveillance robot with face recognition and navigation was created. Using an RGB-D camera, a mobile robot for indoor monitoring and surveillance was proposed and was able to recognize objects. Object detection using CNN and the YOLO (you only look once) algorithm was carried out by a surveillance robot. It was suggested that an autonomous mobile robot for surveillance be able to recognize and detect faces. To perform automatic guard patrol and people detection, a mobile robot was created. However, there are other issues that significantly limit conventional surveillance.

```
#include "esp_camera.h"
#include <WiFi.h>
#include "esp_timer.h"
#include "img_converters.h"
#include "Arduino.h"
#include "fb_gfx.h"
#include "soc/soc.h"           // disable brownout problems
#include "soc/rtc_cntl_reg.h"  // disable brownout problems
#include "esp_http_server.h"
// Replace with your network credentials
const char* ssid = "ESP32";
const char* password = "";
```

```
#define PART_BOUNDARY "123456789000000000000987654321"

#define CAMERA_MODEL_AI_THINKER

//#define CAMERA_MODEL_M5STACK_PSRAM

//#define CAMERA_MODEL_M5STACK_WITHOUT_PSRAM

//#define CAMERA_MODEL_M5STACK_PSRAM_B

//#define CAMERA_MODEL_WROVER_KIT

#if defined(CAMERA_MODEL_WROVER_KIT)

#define PWDN_GPIO_NUM    -1

#define RESET_GPIO_NUM   -1

#define XCLK_GPIO_NUM    21

#define SIOD_GPIO_NUM    26

#define SIOC_GPIO_NUM    27

#define Y9_GPIO_NUM      35

#define Y8_GPIO_NUM      34

#define Y7_GPIO_NUM      39

#define Y6_GPIO_NUM      36

#define Y5_GPIO_NUM      19

#define Y4_GPIO_NUM      18

#define Y3_GPIO_NUM       5

#define Y2_GPIO_NUM       4

#define VSYNC_GPIO_NUM   25

#define HREF_GPIO_NUM    23

#define PCLK_GPIO_NUM    22

#elif defined(CAMERA_MODEL_M5STACK_PSRAM)

#define PWDN_GPIO_NUM     -1

#define RESET_GPIO_NUM    15

#define XCLK_GPIO_NUM     27

#define SIOD_GPIO_NUM     25

#define SIOC_GPIO_NUM     23

#define Y9_GPIO_NUM       19

#define Y8_GPIO_NUM       36

#define Y7_GPIO_NUM       18

#define Y6_GPIO_NUM       39

#define Y5_GPIO_NUM        5

#define Y4_GPIO_NUM       34

#define Y3_GPIO_NUM       35

#define Y2_GPIO_NUM       32

#define VSYNC_GPIO_NUM    22

#define HREF_GPIO_NUM     26

#define PCLK_GPIO_NUM     21

#elif defined(CAMERA_MODEL_M5STACK_WITHO
```

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 1, 2025

```
UT_PSRAM)
#define PWDN_GPIO_NUM     -1
#define RESET_GPIO_NUM    15
#define XCLK_GPIO_NUM     27
#define SIOD_GPIO_NUM     25
#define SIOC_GPIO_NUM     23
#define Y9_GPIO_NUM       19
#define Y8_GPIO_NUM       36
#define Y7_GPIO_NUM       18
#define Y6_GPIO_NUM       39
#define Y5_GPIO_NUM        5
#define Y4_GPIO_NUM       34
#define Y3_GPIO_NUM       35
#define Y2_GPIO_NUM       17
#define VSYNC_GPIO_NUM    22
#define HREF_GPIO_NUM     26
#define PCLK_GPIO_NUM     21
#elif defined(CAMERA_MODEL_AI_THINKER)
#define PWDN_GPIO_NUM     32
#define RESET_GPIO_NUM    -1
#define XCLK_GPIO_NUM      0
#define SIOD_GPIO_NUM     26
#define SIOC_GPIO_NUM     27
#define Y9_GPIO_NUM       35
#define Y8_GPIO_NUM       34
#define Y7_GPIO_NUM       39
#define Y6_GPIO_NUM       36
#define Y5_GPIO_NUM       21
#define Y4_GPIO_NUM       19
#define Y3_GPIO_NUM       18
#define Y2_GPIO_NUM        5
#define VSYNC_GPIO_NUM    25
#define HREF_GPIO_NUM     23
#define PCLK_GPIO_NUM     22
#elif
defined(CAMERA_MODEL_M5STACK_PSRAM
_B)
#define PWDN_GPIO_NUM     -1
#define RESET_GPIO_NUM    15
#define XCLK_GPIO_NUM     27
#define SIOD_GPIO_NUM     22
#define SIOC_GPIO_NUM     23
#define Y9_GPIO_NUM       19
#define Y8_GPIO_NUM       36
#define Y7_GPIO_NUM       18
#define Y6_GPIO_NUM       39
```

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 1, 2025

```
#define Y5_GPIO_NUM       5
#define Y4_GPIO_NUM      34
#define Y3_GPIO_NUM      35
#define Y2_GPIO_NUM      32
#define VSYNC_GPIO_NUM   25
#define HREF_GPIO_NUM    26
#define PCLK_GPIO_NUM    21
#else
#error "Camera model not selected"
#endif
#define MOTOR_1_PIN_1    15
#define MOTOR_1_PIN_2    14
#define MOTOR_2_PIN_1    13
#define MOTOR_2_PIN_2    12
static const char* _STREAM_CONTENT_TYPE = "multipart/x-mixed-replace;boundary=" PART_BOUNDARY;
static const char* _STREAM_BOUNDARY = "\r\n--" PART_BOUNDARY "\r\n";
static const char* _STREAM_PART = "Content-Type: image/jpeg\r\nContent-Length: %u\r\n\r\n";
httpd_handle_t camera_httpd = NULL;
httpd_handle_t stream_httpd = NULL;
static const char PROGMEM INDEX_HTML[] = R"rawliteral(
<html>
<head>
  <title>ESP32-CAM Robot</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <style>
    body { font-family: Arial; text-align: center; margin:0px auto; padding-top: 30px;}
    table { margin-left: auto; margin-right: auto; }
    td { padding: 8 px; }
    .button {
      background-color: #2f4468;
      border: none;
      color: white;
      padding: 10px 20px;
      text-align: center;
      text-decoration: none;
      display: inline-block;
      font-size: 18px;
      margin: 6px 3px;
      cursor: pointer;
      -webkit-touch-callout: none;
      -webkit-user-select: none;
```

```
      -khtml-user-select: none;
      -moz-user-select: none;
      -ms-user-select: none;
      user-select: none;
      -webkit-tap-highlight-color: rgba(0,0,0,0);
    }
    img {  width: auto ;
      max-width: 100% ;
      height: auto ;
    }
  </style>
</head>
<body>
  <h1>ESP32-CAM Robot</h1>
  <img src="" id="photo" >
  <table>
    <tr><td colspan="3" align="center"><button class="button" onmousedown="toggleCheckbox('forward');" ontouchstart="toggleCheckbox('forward');" onmouseup="toggleCheckbox('stop');" ontouchend="toggleCheckbox('stop');">Forward</button></td></tr>
    <tr><td align="center"><button class="button" onmousedown="toggleCheckbox('left');" ontouchstart="toggleCheckbox('left');" onmouseup="toggleCheckbox('stop');" ontouchend="toggleCheckbox('stop');">Left</button></td><td align="center"><button class="button" onmousedown="toggleCheckbox('stop');" ontouchstart="toggleCheckbox('stop');">Stop</button></td><td align="center"><button class="button" onmousedown="toggleCheckbox('right');" ontouchstart="toggleCheckbox('right');" onmouseup="toggleCheckbox('stop');" ontouchend="toggleCheckbox('stop');">Right</button></td></tr>
    <tr><td colspan="3" align="center"><button class="button" onmousedown="toggleCheckbox('backward');" ontouchstart="toggleCheckbox('backward');" onmouseup="toggleCheckbox('stop');" ontouchend="toggleCheckbox('stop');">Backward</button></td></tr>
  </table>
  <script>
  function toggleCheckbox(x) {
    var xhr = new XMLHttpRequest();
    xhr.open("GET", "/action?go=" + x, true);
    xhr.send();
  }
  window.onload = document.getElementById("photo").src = window.location.href.slice(0, -1) + ":81/stream";
</script>
```

```
</body>

</html>

)rawliteral";

static esp_err_t index_handler(httpd_req_t *req){

httpd_resp_set_type(req, "text/html");

return httpd_resp_send(req, (const char *)INDEX_HTML, strlen(INDEX_HTML));

}

static esp_err_t stream_handler(httpd_req_t *req){

camera_fb_t * fb = NULL;

esp_err_t res = ESP_OK;

size_t _jpg_buf_len = 0;

uint8_t * _jpg_buf = NULL;

char * part_buf[64];

res = httpd_resp_set_type(req, _STREAM_CONTENT_TYPE);

if(res != ESP_OK){

  return res;

}

while(true){

  fb = esp_camera_fb_get();

  if (!fb) {

    Serial.println("Camera capture failed");

    res = ESP_FAIL;

  } else {

    if(fb->width > 400){

      if(fb->format != PIXFORMAT_JPEG){

        bool jpeg_converted = frame2jpg(fb, 80, &_jpg_buf, &_jpg_buf_len);

        esp_camera_fb_return(fb);

        fb = NULL;

        if(!jpeg_converted){

          Serial.println("JPEG compression failed");

          res = ESP_FAIL;

        }

      } else {

        _jpg_buf_len = fb->len;

        _jpg_buf = fb->buf;

      }

    }

  }

  if(res == ESP_OK){

    size_t hlen = snprintf((char *)part_buf, 64, _STREAM_PART, _jpg_buf_len);

    res = httpd_resp_send_chunk(req, (const char *)part_buf, hlen);

  }
```

```
  if(res == ESP_OK){
    res = httpd_resp_send_chunk(req, (const char *)_jpg_buf, _jpg_buf_len);
  }
  if(res == ESP_OK){
    res = httpd_resp_send_chunk(req, _STREAM_BOUNDARY, strlen(_STREAM_BOUNDARY));
  }
  if(fb){
    esp_camera_fb_return(fb);
    fb = NULL;
    _jpg_buf = NULL;
  } else if(_jpg_buf){
    free(_jpg_buf);
    _jpg_buf = NULL;
  }
  if(res != ESP_OK){
    break;
  }
  //Serial.printf("MJPG: %uB\n",(uint32_t)(_jpg_buf_len));
}
return res;
}
static esp_err_t cmd_handler(httpd_req_t *req){
char*  buf;
size_t buf_len;
char variable[32] = {0,};

buf_len = httpd_req_get_url_query_len(req) + 1;
if (buf_len > 1) {
  buf = (char*)malloc(buf_len);
  if(!buf){
    httpd_resp_send_500(req);
    return ESP_FAIL;
  }
  if (httpd_req_get_url_query_str(req, buf, buf_len) == ESP_OK) {
    if (httpd_query_key_value(buf, "go", variable, sizeof(variable)) == ESP_OK) {
    } else {
      free(buf);
      httpd_resp_send_404(req);
      return ESP_FAIL;
    }
  } else {
```

```
    free(buf);
    httpd_resp_send_404(req);
    return ESP_FAIL;
  }
  free(buf);
} else {
  httpd_resp_send_404(req);
  return ESP_FAIL;
}
sensor_t * s = esp_camera_sensor_get();
int res = 0;

if(!strcmp(variable, "forward")) {
  Serial.println("Forward");
  digitalWrite(MOTOR_1_PIN_1, 1);
  digitalWrite(MOTOR_1_PIN_2, 0);
  digitalWrite(MOTOR_2_PIN_1, 1);
  digitalWrite(MOTOR_2_PIN_2, 0);
  delay(1000);
  digitalWrite(MOTOR_1_PIN_1, 0);
  digitalWrite(MOTOR_1_PIN_2, 0);
  digitalWrite(MOTOR_2_PIN_1, 0);
  digitalWrite(MOTOR_2_PIN_2, 0);
}
else if(!strcmp(variable, "left")) {
  Serial.println("Left");
  digitalWrite(MOTOR_1_PIN_1, 0);
  digitalWrite(MOTOR_1_PIN_2, 1);
  digitalWrite(MOTOR_2_PIN_1, 1);
  digitalWrite(MOTOR_2_PIN_2, 0);
  delay(150);
  digitalWrite(MOTOR_1_PIN_1, 0);
  digitalWrite(MOTOR_1_PIN_2, 0);
  digitalWrite(MOTOR_2_PIN_1, 0);
  digitalWrite(MOTOR_2_PIN_2, 0);
}
else if(!strcmp(variable, "right")) {
  Serial.println("Right");
  digitalWrite(MOTOR_1_PIN_1, 1);
  digitalWrite(MOTOR_1_PIN_2, 0);
  digitalWrite(MOTOR_2_PIN_1, 0);
  digitalWrite(MOTOR_2_PIN_2, 1);
  delay(150);
  digitalWrite(MOTOR_1_PIN_1, 0);
  digitalWrite(MOTOR_1_PIN_2, 0);
  digitalWrite(MOTOR_2_PIN_1, 0);
```

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 1, 2025

```
  digitalWrite(MOTOR_2_PIN_2, 0);
}
else if(!strcmp(variable, "backward")) {
  Serial.println("Backward");
  digitalWrite(MOTOR_1_PIN_1, 0);
  digitalWrite(MOTOR_1_PIN_2, 1);
  digitalWrite(MOTOR_2_PIN_1, 0);
  digitalWrite(MOTOR_2_PIN_2, 1);
  delay(1000);
  digitalWrite(MOTOR_1_PIN_1, 0);
  digitalWrite(MOTOR_1_PIN_2, 0);
  digitalWrite(MOTOR_2_PIN_1, 0);
  digitalWrite(MOTOR_2_PIN_2, 0);
}
else if(!strcmp(variable, "stop")) {
  Serial.println("Stop");
  digitalWrite(MOTOR_1_PIN_1, 0);
  digitalWrite(MOTOR_1_PIN_2, 0);
  digitalWrite(MOTOR_2_PIN_1, 0);
  digitalWrite(MOTOR_2_PIN_2, 0);
}
else {
  res = -1;
}
if(res){
  return httpd_resp_send_500(req);
}
httpd_resp_set_hdr(req, "Access-Control-Allow-Origin", "*");
return httpd_resp_send(req, NULL, 0);
}
void startCameraServer(){
httpd_config_t config = HTTPD_DEFAULT_CONFIG();
config.server_port = 80;
httpd_uri_t index_uri = {
  .uri       = "/",
  .method    = HTTP_GET,
  .handler   = index_handler,
  .user_ctx  = NULL
};
httpd_uri_t cmd_uri = {
  .uri       = "/action",
  .method    = HTTP_GET,
  .handler   = cmd_handler,
  .user_ctx  = NULL
```

IJMECE International Journal of modern electronics and communication engineering
ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 1, 2025

```
};
httpd_uri_t stream_uri = {
  .uri       = "/stream",
  .method    = HTTP_GET,
  .handler   = stream_handler,
  .user_ctx  = NULL
};
if (httpd_start(&camera_httpd, &config) == ESP_OK) {
      httpd_register_uri_handler(camera_httpd, &index_uri);
      httpd_register_uri_handler(camera_httpd, &cmd_uri);
}
config.server_port += 1;
config.ctrl_port += 1;
if (httpd_start(&stream_httpd, &config) == ESP_OK) {
      httpd_register_uri_handler(stream_httpd, &stream_uri);
}
}
void setup() {
WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0); //disable brownout detector
pinMode(MOTOR_1_PIN_1, OUTPUT);
pinMode(MOTOR_1_PIN_2, OUTPUT);
pinMode(MOTOR_2_PIN_1, OUTPUT);
pinMode(MOTOR_2_PIN_2, OUTPUT);

Serial.begin(115200);
Serial.setDebugOutput(false);
camera_config_t config;
config.ledc_channel = LEDC_CHANNEL_0;
config.ledc_timer = LEDC_TIMER_0;
config.pin_d0 = Y2_GPIO_NUM;
config.pin_d1 = Y3_GPIO_NUM;
config.pin_d2 = Y4_GPIO_NUM;
config.pin_d3 = Y5_GPIO_NUM;
config.pin_d4 = Y6_GPIO_NUM;
config.pin_d5 = Y7_GPIO_NUM;
config.pin_d6 = Y8_GPIO_NUM;
config.pin_d7 = Y9_GPIO_NUM;
config.pin_xclk = XCLK_GPIO_NUM;
config.pin_pclk = PCLK_GPIO_NUM;
config.pin_vsync = VSYNC_GPIO_NUM;
config.pin_href = HREF_GPIO_NUM;
```

```
config.pin_sscb_sda = SIOD_GPIO_NUM;
config.pin_sscb_scl = SIOC_GPIO_NUM;
config.pin_pwdn = PWDN_GPIO_NUM;
config.pin_reset = RESET_GPIO_NUM;
config.xclk_freq_hz = 20000000;
config.pixel_format = PIXFORMAT_JPEG;
if(psramFound()){
  config.frame_size = FRAMESIZE_VGA;
  config.jpeg_quality = 10;
  config.fb_count = 2;
} else {
  config.frame_size = FRAMESIZE_SVGA;
  config.jpeg_quality = 12;
  config.fb_count = 1;
}
// Camera init
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {
   Serial.printf("Camera init failed with error 0x%x", err);
  return;
}
// Wi-Fi connection
/* WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
  delay(500);
  Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");*/
WiFi.softAP(ssid, password);
IPAddress myIP = WiFi.softAPIP();
Serial.print("AP IP address: ");
Serial.println(myIP);
Serial.print("Camera Stream Ready! Go to: http://");
// Serial.println(WiFi.localIP());
// Start streaming web server
startCameraServer();
}
void loop() {
}
```

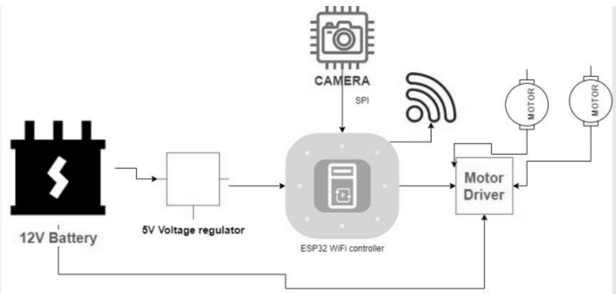

Fig. 1. Diagram of surveillance robot

The camera is connecting to the ESP32 Wi-Fi controller through the serial Peripheral interface (SPI) network media. By using the Wi-Fi module which can operates the surveillance robot. The ESP32 Wi-Fi controller will monitor the object.

The DC motors rotates according the five options they are LEFT, RIGHT, STOP, BACKWORD and FOREWARD.

Based on ESP32, the ESP32-CAM is a compact camera module with minimal power consumption. A micro USB to serial port adapter (ESP32-CAM-MB) and an OV2640 camera are included. Intelligent Internet of Things applications like Wi-Fi picture uploading, QR identification, wireless video monitoring and more can make extensive use of the ESP32-CAM.

Motor drivers are used to give high power to the motor by using a small voltage signal from a microcontroller or a control system. If the microprocessor transmits a HIGH input to the motor driver, the driver will rotate the motor in one direction keeping the one pin as HIGH and one pin as LOW.

A surveillance robot designed to enhance situational awareness and reconnaissance capabilities in various environments would likely incorporate several key features and technologies:

Mobility: The robot should be capable of navigating different terrains and environments efficiently, whether indoors or outdoors. This might involve wheels, tracks, or even legged locomotion for traversing rough terrain.

Communication: The ability to transmit data in real-time or store it for later analysis is crucial. The robot might be equipped with wireless communication capabilities such as Wi-Fi, Bluetooth, or cellular connectivity.

Security: Given the sensitive nature of surveillance missions, the robot's communication and data storage systems would need to be secure against unauthorized access or tampering.

To create a surveillance robot capable of navigating challenging terrains, gathering critical information, and providing real-time data to military personnel, several key features and technologies would be necessary:

Real-Time Data Transmission: The robot should be equipped with robust communication systems capable of transmitting data in real-time to military personnel. This could involve high-bandwidth wireless communication technologies such as secure

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 1, 2025

military-grade radios, satellite communication, or encrypted data links.

Autonomous Navigation and Obstacle Avoidance: Autonomous navigation capabilities would enable the robot to plan its own routes, avoid obstacles, and adapt to changing environments without constant human intervention. This might involve advanced path planning algorithms, obstacle detection sensors, and decision-making systems based on the collected data.

Secure Data Handling: Given the sensitive nature of military operations, the robot's communication and data storage systems must be highly secure to prevent interception or tampering by hostile forces.

User-Friendly Interface: The robot's control interface should be intuitive and user-friendly, allowing military personnel to easily operate the robot, interpret the data it collects, and make informed decisions in real-time.

To enhance operational efficiency and gather valuable intelligence without exposing human personnel to unnecessary risks, a surveillance robot should be designed with the following features:

Persistent Surveillance: Equip the robot with long-endurance capabilities to conduct prolonged surveillance missions without requiring frequent human oversight or battery replacement. This ensures continuous monitoring of the area of interest.

Remote Operation: Provide remote control capabilities to allow human operators to monitor the robot's activities and intervene when necessary. This enables real-time situational awareness and the ability to adjust mission parameters on the fly.

To effectively monitor and gather real-time information from remote or difficult-to-reach areas, a surveillance robot should be equipped with the following capabilities:

Versatile Mobility: The robot should be capable of navigating diverse terrains, including rugged landscapes, confined spaces, and areas with obstacles. This might involve wheels, tracks, or even aerial capabilities such as drones for accessing hard-to-reach areas.

Remote Monitoring and Control: Provide remote operators with intuitive interfaces for monitoring the robot's activities, controlling its movements, and analyzing real-time data streams from remote locations.

25. EXPERIMENTAL RESULTS

A 12-volt battery provides electricity to the motor driver and 5V voltage regulator. Because the voltage regulator maintains a constant voltage of 5V and the motor driver receives 12V to run the DC motors, the ESP32 controller is able to receive 5V of voltage. Via the SPI network medium, the camera is connected to the ESP32 Wi-Fi controller. The Wi-Fi module allows us to control the surveillance robot. The object will be monitored by the ESP32 Wi-Fi controller.

3.7 Existing and Proposed technologies

Existing technologies:

Video cameras are used in closed-circuit television (CCTV), sometimes referred to as video observation, to send a signal to a designated location on a constrained number of displays. Despite the possibility of using mesh wired or wireless networks, point-to-point (P2P), point-to-multipoint (P2MP), or other techniques, the signal is not sent freely like in broadcast television.

Video cameras, which are typically used for surveillance in places like banks, retail establishments, and other places where security is required. One area where video telephony is not often referred to as "CCTV" is in distance education, where it is a valuable tool.

The Node MCU ESP8266, Motor driver L298n, Geared Motor, Power Supply, Battery, and IP Camera are the components of the block diagram. The robot's movement is managed by the Blynk App, and the Blynk App Platform is even used ever to display the output parameters.

The IR sensor will identify any barriers in its immediate environment, and the message it generates will appear in the Blynk app. The geared motor is powered by a motor driver, which allows the rover to move in any direction. IP cameras are utilized for ongoing monitoring and can record video for usage on a mobile device or online.

The tele-sensor programming approach served as the foundation for the unified telerobotic control in ROTEX. This includes sensor-based off-line programming on the ground that is then executed on board, as well as on-board teleoperation conducted both on board and on ground. This approach essentially adopts a "learning by showing" philosophy. Essentially, this strategy involves two key components is a shared control idea (see, for example, /3/, /5/) that was developed at the robot's location (on-board or in a predictive ground simulation). It was based on local sensory feedback, allowing the robot to autonomously refine coarse commands and gain a minimal level of sensory intelligence.

However, force torque and range finder signals were the only on-board sensory feedback available in ROTEX due to CPU restrictions (see below). In the sensor-controlled subspace, adaptive compliance and hybrid (pseudo-)force control based on nominal sensory patterns

The concept of C-frame /2/ was implemented locally. In this framework, gross commands can come from an automated path planner or from a human operator using the sensor ball, a six-degree-of-freedom non-reflecting hand controller. Several earlier studies have covered the methods for projecting bulk commands into the position and sensor-controlled subspaces.

In the event of online teleoperation, the human operator received feedback solely through visual means. For example, the astronaut received feedback through stereo TV images, while the ground operator primarily received feedback using predictive stereo graphics (with the stereo TV images serving as an additional means of verification). This made it possible for us to create a single control structure in spite of the relatively long round-trip signal delays—up to seven seconds—and it will also enable us to gradually

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 1, 2025

give the robot more autonomy in the future without having to modify the fundamental structures.

Predictive 3D stereo modeling was, in fact, a major factor in the space robot experiment's success and, in our judgment, the only effective method of handling significant signal delays. Naturally, the same control structures and path planners needed to be implemented in the predictive graphics ground station as well as on-board for these kinds of concepts to function. This implied that the robot's sensory perception, feedback behavior, and free motion all needed to be represented in the "virtual environment" that was on the ground.

Elemental move concept: any complex task, such as dismounting and remounting the bayonet closure, is assumed to be composed of elemental moves, for which a specific constraint-frame and sensor-type-configuration hold (to be chosen by the operator, for example, using a set of predefined alternatives). This allows for the clear definition of automatic sensor-based path refinement during these motion primitives.

Robot_ect.apk is an Android application that comes with App.inventor. It works with all of the current Android versions that are on the market. This application has three main features: (i) it can control a robot's movement remotely; (ii) it can monitor temperature and humidity in real time; and (iii) it has Bluetooth connectivity.

Figure displays a screen grab of an application on a mobile device.It is composed of two frames: one for robot movement control and the other for monitoring purposes.A list view and a label are present in the first frame.It is possible to choose the specific Bluetooth module using the list view.The applications' labels are used to show the relative humidity as a percentage and the temperature in degrees Celsius.The buttons in the next frame are labeled (a) FOREWARD, (b) BACKWARD, (c) LEFT, (d) RIGHT, and (e) STOP. The user must press a designated button to maneuver the robotic car in a certain direction using these buttons.

REFERENCES

1. Duan, Xiaoming, Dario Paccagnan, and Francesco Bullo. "Stochastic strategies for robotic surveillance as stackelberg games." IEEE Transactions on Control of Network Systems 8.2 (2021): 769-780.
2. Scherer, Jürgen, and Bernhard Rinner. "Multi-robot persistent surveillance with connectivity constraints." IEEE Access 8 (2020): 15093-15109.
3. Asghar, Ahmad Bilal, Stephen L. Smith, and Shreyas Sundaram. "Multi-robot routing for persistent monitoring with latency constraints." 2019 American Control Conference (ACC). IEEE, 2019.
4. George, Mishel, Saber Jafarpour, and Francesco Bullo. "Markov chains with maximum entropy for robotic surveillance." IEEE Transactions on Automatic Control 64.4 (2018): 1566-1580.
5. Nayyar, Anand, et al. "Smart surveillance robot for real-time monitoring and control system in environment and industrial applications." Information Systems Design and Intelligent Applications: Proceedings of Fourth International Conference INDIA 2017. Springer Singapore, 2018.
6. Banfi, Jacopo, et al. "Strategies for coordinated multirobot exploration with recurrent connectivity constraints." Autonomous Robots 42 (2018): 875-894.

7. Asghar, Ahmad Bilal, and Stephen L. Smith. "A patrolling game for adversaries with limited observation time." 2018 IEEE Conference on Decision and Control (CDC). IEEE, 2018.
8. Masehian, Ellips, Marjan Jannati, and Taher Hekmatfar. "Cooperative mapping of unknown environments by multiple heterogeneous mobile robots with limited sensing." Robotics and Autonomous Systems 87 (2017): 188-218.
9. Santos, João Machado, Tomáš Krajník, and Tom Duckett. "Spatio-temporal exploration strategies for long-term autonomy of mobile robots." Robotics and Autonomous Systems 88 (2017): 116- 126.
10. Portugal, David, and Rui P. Rocha. "Performance estimation and dimensioning of team size for multi obot patrol." IEEE Intelligent Systems 32.6 (2017): 30-38.
11. Juhana, Tutun, and Vivi Gusti Anggraini. "Design and implementation of Smart Home Surveillance system." 2016 10th International conference on telecommunication systems services and applications (TSSA). IEEE, 2016. 42
12. Menezes, Virginia, Vamsikrishna Patchava, and M. Surya Deekshith Gupta. "Surveillance and monitoring system using Raspberry Pi and SimpleCV." 2015 international conference on green computing and internet of things (ICGCIoT). IEEE, 2015.
13. Ghamry, Khaled A., and Youmin Zhang. "Cooperative control of multiple UAVs for forest fire monitoring and detection." 2016 12th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA). IEEE, 2016.
14. Rossi, Maurizio, and Davide Brunelli. "Autonomous gas detection and mapping with unmanned aerial vehicles." IEEE Transactions on Instrumentation and Measurement 65.4 (2015): 765-775.
15. Asghar, Ahmad Bilal, and Stephen L. Smith. "Stochastic patrolling in adversarial settings." 2016 American Control Conference (ACC). IEEE, 2016.
16. Menezes, Virginia, Vamsikrishna Patchava, and M. Surya Deekshith Gupta. "Surveillance and monitoring system using Raspberry Pi and SimpleCV." 2015 international conference on green computing and internet of things (ICGCIoT). IEEE, 2015.
17. Ha Manh, Weihua Sheng, and Meiqin Liu. "An open platform of auditory perception for home service robots." 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2015.
18. Salh, Thair A., and Mustafa Z. Nayef. "Face recognition system based on wavelet, pca-lda and svm." Comput. Eng. Intell. Syst. J 4.3 (2013).
19. Güleçi, Mehmetcan, and Murat Orhun. "Android based WI-FI controlled robot using Raspberry Pi." 2017 International Conference on Computer Science and Engineering (UBMK). IEEE, 2017.
20. Shaikh, Shoeb Maroof, et al. "Wireless Video Surveillance Robot Controlled Using Android Mobile Device." International Journal of Advance Foundation and Research in Science & Engineering (IJAFRSE) 1 (2015).
21. Trevelyan, James, William R. Hamel, and Sung-Chul Kang. "Robotics in hazardous applications." Springer handbook of robotics (2016): 1521-1548.
22. Lee, Min-Fan Ricky, and Zhih-Shun Shih. "Autonomous Surveillance for an Indoor Security Robot." Processes 10.11 (2022): 2175.

23. Lee, Hou-Tsan, et al. "Wireless indoor surveillance robot." SICE Annual Conference 2011. IEEE, 2011. 43
24. Salh, Thair Ali, and Mustafa Zuhaer Nayef. "Intelligent surveillance robot." 2013 International Conference on Electrical Communication, Computer, Power, and Control Engineering (ICECCPCE). IEEE, 2013.
25. Rawat, Romil, et al. "Surveillance robot in cyber intelligence for vulnerability detection." Machine Learning for Robotics Applications (2021): 107-123.
26. Wu, Xinyu, et al. "Intelligent household surveillance robot." 2008 IEEE International Conference on Robotics and Biomimetics. IEEE, 2009.
27. Dixit, Dr Shantanu K., and Sharad B. Dhayagonde. "Design and implementation of e-surveillance robot for video monitoring and living body detection." International Journal of Scientific and Research Publications 4.4 (2014): 1-3.
28. Sathyamoorthy, Adarsh Jagan, et al. "COVID surveillance robot: Monitoring social distancing constraints in indoor scenarios." Plos one 16.12 (2021): e0259713.
29. Budiharto, Widodo. "Intelligent surveillance robot with obstacle avoidance capabilities using neural network." Computational intelligence and neuroscience 2015 (2015): 52-52.
30. Telkar, Aishwarya K., and Baswaraj Gadgay. "Iot based smart multi application surveillance robot." 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA). IEEE, 2020016. 2016;327-357.