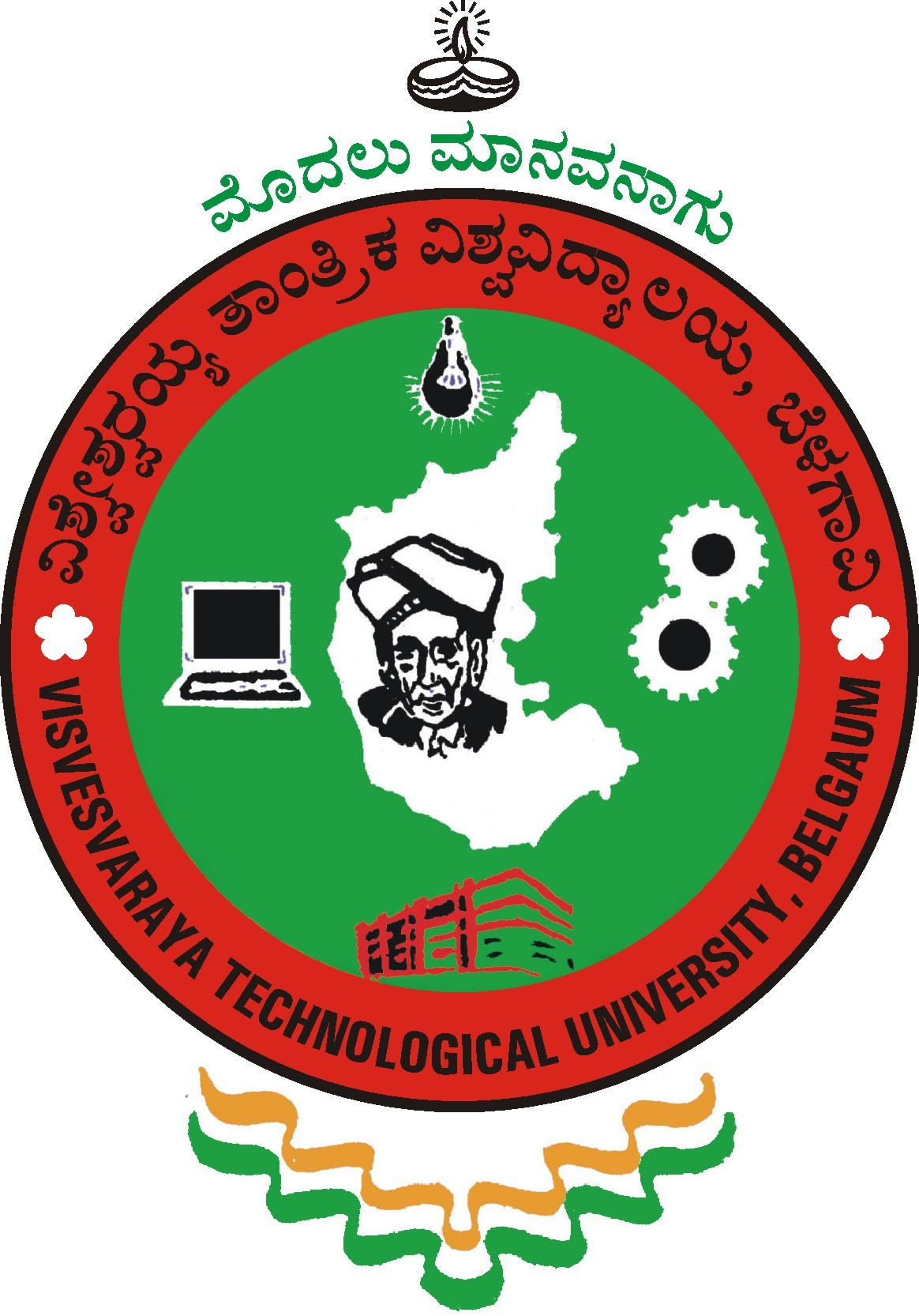
**VISVESVARAYA TECHNOLOGICAL UNIVERSITY**

**Jnanasangama, Macche, Santibastwada Road**

**Belagavi-590018, Karnataka**



**Internship Report**

**"Live Streaming with Raspberry Pi and OpenCV for Object Detection"**

***Submitted in Partial fulfillment for the award of Degree of***

**Bachelor of Engineering**

**in**

# ELECTRONICS AND COMMUNICATION ENGINEERING

**19EC8ICINT**

**Submitted by**

**THOTA SAI GAUTHAM**

**1DS19EC742**

***Internship carried out***

**at**

**DIGITECTURA TECHNOLOGIES PVT LTD**

1016, 4th block, 1st Stage, HBR Layout, Bengaluru, Karnataka 560043

**Internal Supervisor name in the college**

Dr.Dinesha P

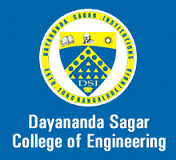
Professor

**External Supervisorin the industry**

Siddanth Jain

Technical lead

Degitectura Technologies Private Limited



**Department of Electronics & Communication Engineering**

(An Autonomous College affiliated to VTU Belgaum, accredited by NBA & NAAC)

ShavigeMalleshwara Hills, Kumaraswamy Layout,

Bengaluru-560078, Karnataka, India

December 2018

|  |  |
| --- | --- |
| **download (6).jpg** | **Dayananda Sagar College of Engineering**  **Shavige Malleshwara Hills, Kumaraswamy Layout, Bangalore-560078, Karnataka**  **Tel : +91 80 26662226 26661104 Extn : 2731 Fax : +90 80 2666 0789**  **Web -** [**http://www.dayanandasagar.edu**](http://www.dayanandasagar.edu)**Email :**[**hod-ece@dayanandasagar.edu**](mailto:hod-ece@dayanandasagar.edu)  **( An Autonomous Institute Affiliated to VTU, Approved by AICTE & ISO 9001:2008 Certified )**  **( Accredited by National Assessment & Accreditation Council (NAAC) with 'A' grade )** |

# CERTIFICATE

Certified that the internship work (19EC8ICINT) entitled, “**Live Streaming with Raspberry Pi and OpenCV for Object Detection**” is carried out by **THOTA SAI GAUTHAM** (**1DS19EC742)** is the bonafide student of ECE Department, Dayananda Sagar College of Engineering, Bangalore, Karnataka, India in partial fulfillment for the award of Bachelor of Engineering in Electronics & Communication Engineering , Visvesvaraya Technological University, Belagavi, Karnataka during the academic year 2022-23. It is certified that all corrections / suggestions indicated for internship work have been incorporated in the internship report deposited to the ECE department. The internship report (19EC8ICINT) has been approved as it satisfies the academic requirement in respect of internship work prescribed for the said undergraduate degree.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Internship Guide/Supervisor Evaluator Head of the Department

Name: Name: Dr. T.C. Manjunath

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Dr. B G Prasad

Principal

**External Internship (to be signed by SEE examiners) Viva-Voce:**

**Name of the internship examiners**

1: Signature & date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2: Signature & date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Certificate:**

**A certificate of completion

Description automatically generated with medium confidence**

|  |  |  |
| --- | --- | --- |
| **download (6).jpg** | Declaration |  |

Certified that the internship work (19EC8ICINT) entitled, “**Live Streaming with Raspberry Pi and OpenCV for Object Detection**” is a bonafide work that is carried out by me in the “**DIGITECTURA TECHNOLOGIES PVT LTD**” in partial fulfillment for the award of degree of Bachelor of Engineering in Electronics & Communication Engg. (Autonomous) of the Visvesvaraya Technological University, Belagavi, Karnataka during the academic year 2022-23 for the 4th year B.E. course. I hereby declare that the entire internship work has been done on my own in the company& is a novel work under the leadership of the industry / company guide. The results embedded in this internship report have not been submitted elsewhere for the award of any type of degree in any other university& is genuine related to the company aspects and the work done in the company/industry.

Student Name: THOTA SAI GAUTHAM USN: 1DS19EC742

Date: 19/05/2023 Place: Bengaluru -78

Sign: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
| **Table of Contents**      Certificate iii  Declaration iv  Table of Contents v  List of Figures vi  Abstract vii      Chapter 1: About the Company 8  Chapter 2: About the Department 10  Chapter 3: Tasks Performed in the Industry 12  3.1 Learnings 13  3.2 Tools Exposed 14  3.3 Objective 17  3.3 Proposed Methodology 18  3.4 Implementation 21  3.5 Results 26  Chapter 4: Reflections  Applications 27  Future scope 28  Reflections 29  References 30 |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| List of Figures Fig. 1: Block diagram 18  Fig. 2: Raspberry Pi 20  Fig. 3: Pi Camera module 20  Fig. 4: SD card and power supply 20  Fig. 5: Monitor 20  Fig. 6: Raspberry pi with camera module 25  Fig. 7: Captured image 1 26  Fig. 8: Captured image 2 26 |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# 

**Summary**

**Title:** Video Streaming Using Raspberry Pi with Server and Client in Headless Mode

This project focused on developing a video streaming solution using Raspberry Pi in a headless mode, where the server and client applications were implemented on separate devices. The aim was to create a system that enables remote video streaming without the need for a dedicated monitor or keyboard. The internship was carried out at Digitectura Technologies Private Limited in the technical department. The need for a cost-effective and compact video streaming solution that can be controlled remotely without the requirement of a dedicated monitor or keyboard. The challenge of implementing video streaming capabilities using Raspberry Pi in a headless mode. Setup and configuration of Raspberry Pi devices: The Raspberry Pi devices were prepared for the project, including the installation of necessary software and libraries. Development of the server application: A server application was created to capture video from a connected camera or video source, encode it, and stream it over a network. Implementation of the client application: The client application was developed to receive the video stream from the server, decode it, and display it on a connected screen or device. Integration and testing: The server and client applications were integrated and tested to ensure seamless video streaming and remote-control capabilities. Successful implementation of a video streaming solution using Raspberry Pi in headless mode. Remote control of the video streaming system without the need for a dedicated monitor or keyboard. Efficient video encoding and decoding processes to ensure smooth and high-quality video streaming. In conclusion, this project successfully developed a video streaming solution using Raspberry Pi with server and client applications in headless mode. It addressed the need for a cost-effective and compact video streaming system that can be controlled remotely. The findings highlight the efficient encoding and decoding processes and the capability to stream high-quality video remotely.

**Keywords:** Video streaming, Raspberry Pi, Headless mode, Server-client setup, Remote control.

**Chapter 1**

**About the Company**

**Company:** DIGITECTURA TECHNOLOGIES PVT LTD

**Location:** 1016, 4th block, 1st Stage, HBR Layout, Bengaluru, Karnataka 560043

Digitectura Technologies Private Limited is a Private incorporated on 28 November 2018. It is classified as Non-govt company and is registered at Registrar of Companies, Bangalore. Its authorized share capital is Rs. 2,000,000 and its paid up capital is Rs. 1,990,000. It is involved in Other computer related activities. Digitectura Technologies is a technology company specializing in software development and IT solutions. The company offers a range of software products and solutions for various industries, including healthcare, finance, e-commerce, and more. Their product portfolio includes custom software applications, mobile apps, enterprise solutions, and cloud-based platforms. Digitectura Technologies is known for its innovative approach to software development and technology solutions. They continually strive to stay ahead of the curve by adopting the latest technologies and trends in the industry.

The company emphasizes research and development to create cutting-edge solutions for their clients. Digitectura Technologies is committed to sustainable practices in its operations and product development. They prioritize environmentally friendly solutions and aim to minimize their carbon footprint. The company actively promotes sustainability initiatives within their organization and collaborates with clients who share the same values. Digitectura Technologies has built a strong reputation in the IT industry for delivering high-quality software products and solutions. They have a track record of successfully completing projects for clients across different sectors.

The company is known for its professionalism, reliability, and customer-centric approach, which has contributed to their positive reputation in the market. The company is engaged in designing, developing, and providing security systems and technology-driven products. They offer benchmarking, training, and consultancy services in various areas related to software and solutions. They specialize in mobile application development, including emerging niche segments such as the Internet of Things (IoT), applications and tools, home automation, and industrial automation. The company focuses on information technology (IT) and IT-enabled applications, including customization, smart devices, and testing. They deal with the use of technology in personal and commercial areas, implementing technology-enabled solutions in different fields.

The company operates in both India and overseas markets, providing their services and solutions globally. They cover all aspects of hardware and software, offering comprehensive solutions to their clients. The company's expertise lies in leveraging technology to address various business needs and challenges across industries. They provide implementation, maintenance, and support services for their software and solutions. The company's offerings encompass a wide range of IT solutions, catering to diverse industries and sectors.

In summary, the company engages in designing, developing, and providing security systems, technology-driven products, and software solutions. They specialize in mobile application development, IoT, home automation, industrial automation, and other emerging niche segments. They operate both in India and overseas, offering comprehensive IT services, implementation, and maintenance support. Overall, Digitectura Technologies Pvt Ltd is a technology company based in Bengaluru, Karnataka, offering innovative software products and solutions across various industries. They prioritize sustainability and have earned a strong reputation for their professionalism and customer satisfaction.

**Chapter 2:**

**About the Department**

The software development department is responsible for designing, developing, and maintaining software solutions within an organization. Here is an overview of the software development department:

**1. Software Design and Architecture:** The department is involved in designing the overall structure, components, and modules of software systems. They define the architecture, frameworks, and patterns that guide the development process.

**2. Coding and Programming:** Software developers in this department write code using programming languages like Java, Python, C++, or JavaScript. They implement the functionality, logic, and algorithms that drive the software application or system.

**3. Software Testing and Quality Assurance:** The department conducts testing activities to identify and fix defects or issues in the software. They perform unit testing, integration testing, and system testing to ensure that the software meets quality standards and functional requirements.

**4. Version Control and Configuration Management:** Software development teams utilize version control systems (e.g., Git) to manage code versions, track changes, and collaborate effectively. They also handle configuration management to maintain consistent software environments across different stages of development.

**5. Software Deployment and Maintenance:** The department ensures that software applications are properly deployed, configured, and maintained. They may work with operations or DevOps teams to manage infrastructure, automate deployment processes, and address any operational issues.

Overall, the software development department plays a crucial role in creating, maintaining, and enhancing software solutions within an organization. They are responsible for the entire software development lifecycle, from initial design to deployment and ongoing maintenance, while utilizing best practices and staying updated with the latest industry trends.

**Chapter 3:**

**Tasks Performed in the company:**

In the software development department, an employee working in image processing would typically be involved in the following tasks:

**1. Algorithm Design and Development:** The employee would be responsible for designing and implementing algorithms that perform various image processing tasks such as image enhancement, image segmentation, object detection, feature extraction, and image recognition. This involves understanding the mathematical and computational principles behind image processing and translating them into code.

**2. Coding and Software Development:** The employee would write code in programming languages such as Python, MATLAB, or C++ to implement the image processing algorithms. They would work with software development tools and frameworks specific to image processing, such as OpenCV or MATLAB's Image Processing Toolbox, to efficiently process and manipulate images.

**3. Testing and Debugging:** The employee would conduct thorough testing of the image processing software to ensure its functionality, performance, and accuracy. They would identify and fix any bugs or issues that arise during testing, ensuring that the software delivers the expected results for a wide range of images and scenarios.

**4. Optimization and Performance Improvement:** The employee would optimize the image processing algorithms and code to enhance performance, speed, and efficiency. This could involve techniques such as parallelization, code optimization, or utilizing hardware acceleration (e.g., GPUs) to accelerate the processing of large-scale images or video streams.

Overall, an employee in the image processing role within the software development department would focus on designing, developing, testing, and optimizing image processing algorithms and software to deliver efficient and accurate image analysis and manipulation capabilities.

**Learnings**

**Technical:**

Digital image fundamentals involve the basic concepts and components related to digital images. Here are some key elements:

Pixels: A digital image is composed of small picture elements called pixels. Each pixel represents a single point in the image and contains information about its color and intensity. Pixels are arranged in a grid-like pattern, and the resolution of an image refers to the number of pixels in that grid.

Resolution: Resolution refers to the number of pixels in an image, typically expressed as width x height (e.g., 1920 x 1080 pixels). Higher resolutions result in sharper and more detailed images, while lower resolutions may appear pixelated or blurry. Resolution is often measured in pixels per inch (PPI) for printed images or dots per inch (DPI) for digital displays.

Color Depth: Color depth, also known as bit depth, determines the number of colors that can be represented in an image. It is expressed in bits per pixel (bpp) and influences the richness and accuracy of color reproduction. Common color depths include 8-bit (256 colors), 24-bit (true color), and 48-bit (high dynamic range).

Image File Formats: Digital images are typically stored in various file formats, each with its own characteristics and purposes. Common formats include JPEG (lossy compression for photographs), PNG (lossless compression for graphics and transparent images), GIF (limited color palette and animation support), and TIFF (high-quality and lossless format for professional use).

Compression: Image compression techniques are used to reduce the file size of images, making them easier to store, transmit, and manipulate. Lossless compression algorithms preserve all image data without quality loss, while lossy compression algorithms discard certain details to achieve higher compression ratios.

Color Models: Color models define how colors are represented and displayed in digital images. Common color models include RGB (Red, Green, Blue), which is used for most digital displays and image capture devices, and CMYK (Cyan, Magenta, Yellow, Black), which is used for print reproduction. Other color models include HSB, Lab, and grayscale. Image Editing: Digital image editing involves manipulating and enhancing images using software tools. It allows adjustments to color, brightness, contrast, sharpness, and other image attributes. Image editing software like Adobe Photoshop or GIMP provides a range of features for editing, retouching, and manipulating images.

Metadata: Digital images often contain metadata, which is descriptive information about the image. This can include details such as the camera settings, date and time of capture, geolocation, copyright information, and keywords. Metadata helps organize and categorize images and provides additional context or information.

**TOOLS EXPOSED:**

**Raspberry pi**

Raspberry Pi 3B is a small, single-board computer designed for educational and hobbyist purposes. It features a 1.2GHz quad-core ARM processor, 1GB RAM, built-in Wi-Fi and Bluetooth capabilities, HDMI output, and multiple USB ports. It runs on various operating systems, including Linux distributions, making it versatile for different projects. With its affordable price and GPIO pins for hardware connections, Raspberry Pi 3B is widely used for coding, robotics, home automation, and DIY projects. Raspberry Pi is a popular single-board computer that can be used for a wide range of projects. To work effectively with a Raspberry Pi, you may need various tools and accessories. Here are some common tools used with Raspberry Pi:MicroSD Card: Raspberry Pi uses a microSD card as the primary storage medium for the operating system and data. You'll need a microSD card with sufficient capacity (recommended 16GB or higher) to install and run the Raspberry Pi OS.

Power Supply: A micro-USB power supply is required to power the Raspberry Pi. The power supply should have the appropriate voltage and current rating for your specific Raspberry Pi model (usually 5V and 2.5A or higher).

HDMI Cable: If you plan to connect your Raspberry Pi to a monitor or TV, you'll need an HDMI cable to transmit audio and video signals.

USB Keyboard and Mouse: A USB keyboard and mouse are necessary for input and interaction with the Raspberry Pi. You can connect them to the USB ports on the Raspberry Pi board.

Display: You can connect a display to the Raspberry Pi for visual output. It can be a computer monitor, TV, or Raspberry Pi-compatible touchscreen display.

Ethernet Cable or Wi-Fi Adapter: Raspberry Pi has built-in Ethernet ports for wired internet connectivity. If you don't have a wired connection available, you can use a USB Wi-Fi adapter to connect to a wireless network.

GPIO Accessories: Raspberry Pi has a General-Purpose Input/Output (GPIO) header that allows you to connect external components and sensors. You may need breadboards, jumper wires, LEDs, resistors, sensors, and other electronic components to interface with the GPIO pins.

Cooling: Depending on your project and the computing load, you may need cooling solutions like heat sinks or fans to prevent the Raspberry Pi from overheating. Programming Tools: To develop software for the Raspberry Pi, you'll need a text editor or integrated development environment (IDE) for writing code. Popular options include Python IDEs like Thonny, IDLE, or using the command-line interface.

**Camera module:**

The camera module for Raspberry Pi 3B is a small accessory that allows users to capture images and videos directly from the Raspberry Pi board. It connects to the CSI (Camera Serial Interface) port on the Pi, providing high-quality imaging capabilities. The module typically features a small sensor and lens that enable the Pi to capture still images or record video footage. It is widely used in various applications such as surveillance systems, robotics, computer vision, and photography projects, adding visual capabilities to the Raspberry Pi 3B board.

**Advanced ip scanner:**

Advanced IP Scanner is a Windows-based network scanner that allows you to scan IP addresses and evaluate the performance of your network. It provides additional features such as remote shutdown, wake-on-LAN, and the ability to connect to shared folders using the SMB protocol.

**Visual studio:**

Visual Studio is an integrated development environment (IDE) developed by Microsoft. It provides a comprehensive set of tools and features for software development across various platforms, including Windows, macOS, Android, iOS, and web applications. Here are some key aspects of Visual Studio:

Code Editing: Visual Studio offers a powerful code editor with features like syntax highlighting, code completion, code refactoring, and intelligent code suggestions. It supports a wide range of programming languages, including C#, C++, Python, JavaScript, TypeScript, and more.

Debugging: Visual Studio provides robust debugging capabilities, allowing developers to identify and fix issues in their code. It offers features like breakpoints, step-through debugging, variable inspection, call stacks, and real-time monitoring of application execution. Integrated Tools: Visual Studio integrates various tools to streamline the development process. It includes tools for version control (e.g., Git), project management, build automation, code analysis, and testing. These tools help developers collaborate, manage their projects, and ensure code quality.

Project Templates: Visual Studio provides a wide range of project templates for different types of applications, such as desktop applications, web applications, mobile apps, game development, and cloud services. These templates offer a starting point and predefined structures for specific platforms and frameworks.

**Project:**

The "Video Streaming Using Raspberry Pi with Server and Client in Headless Mode" project aims to implement a video streaming solution utilizing Raspberry Pi devices. In this project, the Raspberry Pi is utilized both as a server and a client, enabling video streaming without the need for a dedicated display or user interface. By leveraging the capabilities of Raspberry Pi, the project focuses on developing a system that can efficiently stream video content over a local network. This includes setting up the Raspberry Pi devices in headless mode, where they operate without a connected monitor, keyboard, or mouse.

The project involves configuring the Raspberry Pi as a server to host the video content, utilizing its processing power and network connectivity to facilitate video streaming. Additionally, the Raspberry Pi acts as a client that can receive and display the streamed video content. The implementation includes various aspects such as network setup, communication protocols, video encoding/decoding, and streaming protocols. It may utilize technologies like Real-Time Streaming Protocol (RTSP) to enable real-time transmission of video data.

The project offers an opportunity to explore the capabilities of Raspberry Pi devices and gain hands-on experience in setting up, configuring, and programming them for video streaming purposes. It also involves working with networking protocols, video encoding/decoding techniques, and implementing industry-standard streaming protocols. By successfully completing this project, participants can develop a deeper understanding of video streaming principles, enhance their technical skills in Raspberry Pi configuration, networking, and video processing, and gain practical experience in implementing a streaming solution using modern tools and techniques.

**Objectives:**

The objectives of the "video streaming using Raspberry Pi with server and client in headless mode" project include:

**1. Implementing a video streaming solution:** The primary objective is to develop a functional video streaming system using Raspberry Pi as both the server and client, allowing for the transmission of video data over a network connection.

**2. Headless operation:** The project aims to enable the Raspberry Pi to operate without a dedicated display or user interface, making it suitable for headless deployments where remote access and control are utilized.

**3. Efficient video encoding and decoding:** The project seeks to employ efficient video encoding and decoding techniques on the Raspberry Pi to ensure smooth and high-quality video streaming with minimal latency.

**4. Network connectivity and communication:** Establishing a reliable network connection between the server and client on the Raspberry Pi is an essential objective. This includes configuring IP addresses, network protocols, and ensuring seamless communication for video streaming.

**5. Scalability and performance optimization:** The project may involve optimizing the video streaming solution for scalability and performance, allowing it to handle multiple clients and stream videos efficiently without significant degradation in quality.

**6. Robustness and error handling:** The objective is to create a robust video streaming solution that can handle network disruptions, recover from errors, and ensure a seamless streaming experience for users.

**7.Annotation:** Bounding Box Annotation: This involves drawing rectangles around objects of interest within an image. The coordinates of the bounding box (x, y, width, height) are usually recorded to specify the object's location.

**Proposed Methodology:**

**The proposed methodology is mentioned below:**

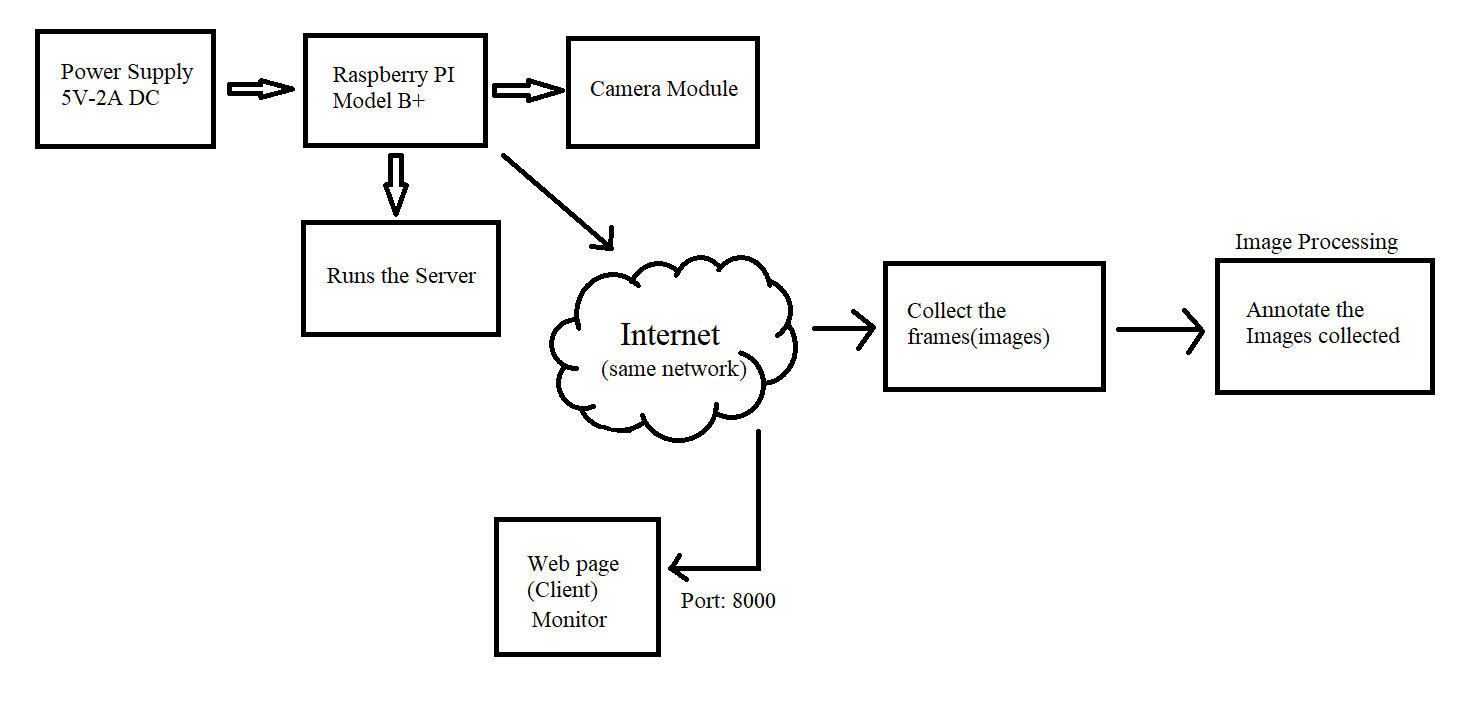


Fig1: Block diagram

1. Raspberry Pi Setup and Configuration:
   1. Set up Raspberry Pi devices for headless operation, including installing the necessary operating system (e.g., Raspbian) and software dependencies.
   2. Configure network settings, such as assigning static IP addresses for the server and client.
2. Server-Side Development:
   1. Develop the server-side software on the Raspberry Pi to handle video streaming functionality.
   2. Implement video encoding and streaming algorithms to compress and transmit the video data.
   3. Set up necessary buffers and data structures to ensure smooth streaming.
3. Client-Side Development:
   1. Develop the client-side software on the Raspberry Pi to receive and decode the video stream.
   2. Implement video decoding algorithms to decompress the received video data.
   3. Render and display the video stream on the client-side display or output device.
4. Testing and Optimization:
   1. Conduct thorough testing of the video streaming system to ensure functionality and performance.
   2. Test different scenarios, network conditions, and streaming configurations to validate the system's robustness.
   3. Optimize the software components and algorithms for improved video streaming quality and reduced latency.
5. Annotation:

Bounding Box Annotation: This involves drawing rectangles around objects of interest within an image. The coordinates of the bounding box (x, y, width, height) are usually recorded to specify the object's location.

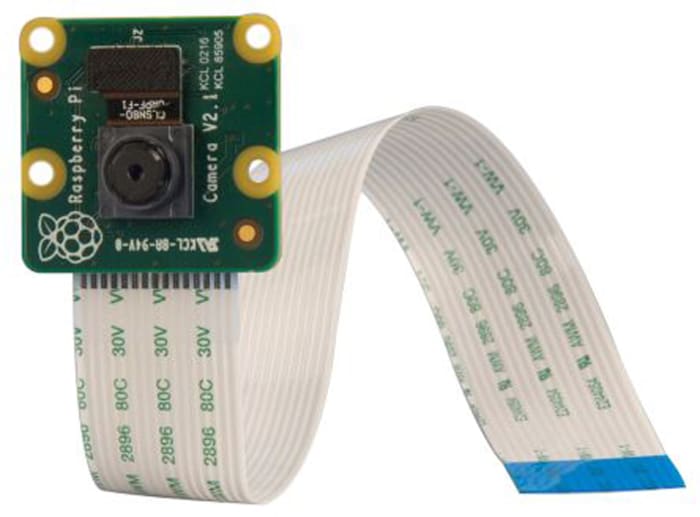


Fig 3: Pi Camera module

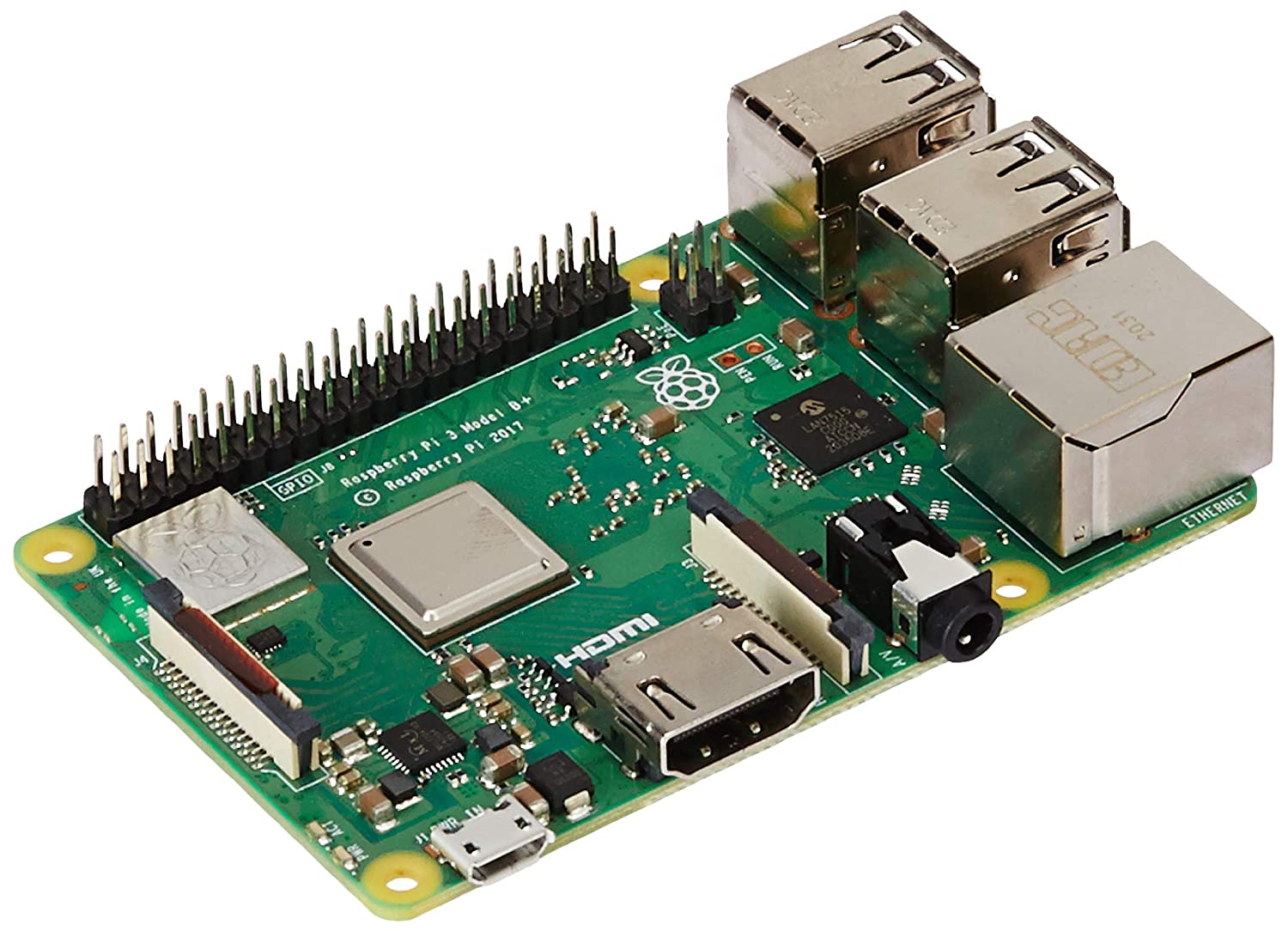


Fig 2: Raspberry pi 3B



Fig 4: SD card and Power supply



Fig 5: Monitor

**Implementation:**

**1) Different image types:**

Image types, also known as image file formats, are specific file formats used to store and encode digital images. Each image type has its own characteristics, advantages, and limitations. Here's an overview of used image types:

1. JPEG (Joint Photographic Experts Group): A widely used image format that supports high compression levels. It is suitable for photographs and complex images, but it uses lossy compression, which may result in some loss of image quality.

2. PNG (Portable Network Graphics): A lossless image format that supports transparency and high-quality images. PNG files are often used for graphics, logos, and images with sharp edges or text. However, they tend to have larger file sizes compared to JPEG.

3. GIF (Graphics Interchange Format): A format commonly used for animated images and graphics with a limited color palette. GIF supports animation and transparency, making it popular for short, animated clips or simple graphics.

4. BMP (Bitmap): A simple and uncompressed image format that stores pixel data directly. BMP files are typically large in size and do not support compression. They are commonly used in Windows-based applications and can store images in different color depths.

6. RAW: A file format that stores unprocessed and uncompressed image data captured directly from a camera's image sensor. RAW files retain maximum image quality and provide greater flexibility for post-processing. However, they require specialized software to convert and edit the images.

**2) Different types of images based on their color representation:**

1. Black and White Image: A black and white image, also known as a monochrome or binary image, only contains two colors - black and white. It is typically represented using a single bit per pixel, where black is represented by 0 and white by 1. Black and white images have no color information and are commonly used for simple line art, text, and binary representations.

2. Grayscale Image: A grayscale image uses shades of gray to represent different levels of brightness. It typically has 8 bits per pixel, allowing for 256 different shades of gray ranging from black (0) to white (255). Grayscale images lack color information but provide varying levels of intensity, making them suitable for representing images with different levels of brightness and contrast.

3. Color Image: A color image contains multiple color channels, typically representing the three primary colors - red, green, and blue (RGB). Each pixel in a color image is assigned a combination of RGB values, ranging from 0 to 255, to specify its color. Color images can represent a wide range of colors and are commonly used for photographs, illustrations, and multimedia applications.

**2. Raspberry Pi Configuration and Setup:** Through the project, participants gain knowledge and expertise in setting up Raspberry Pi devices for headless operation. This includes installing the required operating system, configuring network settings, and optimizing the Raspberry Pi for video streaming.

**3. Video Encoding and Decoding:** The project involves implementing video encoding and decoding algorithms to compress and decompress the video data. Successful execution results in the ability to encode video on the server-side and decode it on the client-side for seamless video playback.

**4. Networking and Communication:** Participants gain insights into networking protocols and communication methods for transmitting video data over a network. They learn how to establish a connection between the server and client in a headless environment and ensure smooth video streaming.

**5. Video Quality and Latency:** The project offers an opportunity to optimize the video streaming system to achieve high-quality video playback with minimum latency. Through testing and optimization, participants can improve video quality, reduce buffering, and minimize delays during streaming.

A green circuit board with a white ribbon

Description automatically generated with low confidence

Fig 6: Raspberry pi with camera module

The overall working of the "Video Streaming using Raspberry Pi with Server and Client in Headless Mode" project involves the following steps:

1. **Enabling the Raspberry pie camera module :** If you’re using the Raspberry Pi Camera Module, you need to enable the camera software in your Raspberry Pi in order to use it. In the Desktop environment, go to the Raspberry Pi Configuration window under the Preferences menu, open the Interfaces tab and enable the Camera as shown in figure below.

A screenshot of a computer

Description automatically generated

2. pi @ raspberry: ~ $ **sudo raspi-config**

A screenshot of a computer

Description automatically generated with medium confidenceRaspberry Pi software configuration tool. Select the **Interfacing Options**:

Enable the camera and reboot your Pi:

A screenshot of a computer

Description automatically generated with medium confidence

2. Server-Side Development:

- Develop the server-side software on the Raspberry Pi server to handle video streaming functionality.

- Implement video encoding algorithms to compress the video data.

- Set up a server program to transmit the encoded video stream to the client.

3. Client-Side Development:

- Develop the client-side software on the Raspberry Pi client to receive and decode the video stream.

- Implement video decoding algorithms to decompress the received video data.

- Render and display the video stream on the client-side display or output device.

4. Networking and Communication:

- Establish a network connection between the Raspberry Pi server and client, ensuring proper IP addressing and connectivity.

- Implement a communication protocol (e.g., RTSP) to facilitate the transmission of video data from the server to the client.

5. Video Encoding and Decoding:

- On the server side, encode the video frames captured by the Raspberry Pi camera or any other video source.

- Transmit the encoded video stream over the network to the client.

6. Video Reception and Decoding:

- On the client side, receive the transmitted video stream from the server.

6. pi @ raspberrypi: ~ $ **python3 rpi\_camera\_surveillance\_system.py**

7. Once the script is running, you can access your video streaming web server at: http://<Your\_Pi\_IP\_Address>:8000. You can access video streaming through any device that has a browser and is connected to the same network as your Pi.

9. Storing the images through pie; taking the continuous images through pi module in order to train the camera module in aspects of identifying the person called as annotation of the images, annotation of 2000 images

**Results:**



Fig 7: Captured image 1



Fig 8: Captured image 2

**CHAPTER 4:**

**Applications:**

Applications of "Video Streaming using Raspberry Pi with Server and Client in Headless Mode" project:

1. Remote Surveillance: The project can be utilized for remote surveillance applications, allowing users to monitor live video feeds from remote locations using Raspberry Pi devices as both server and client. This can be beneficial for home security, monitoring industrial processes, or remote site surveillance.

2. IoT Applications: The project can be integrated into Internet of Things (IoT) systems, where the Raspberry Pi devices can stream video data to other IoT devices or cloud platforms for further analysis, machine learning, or automation purposes.

3. Education and Research: The project can be used as a learning tool for educational institutions or individuals interested in studying video streaming, network protocols, and Raspberry Pi. It can serve as a hands-on project to understand video encoding/decoding, network communication, and software development.

**Future Scope:**

1. Mobile Application Integration: The project can be extended to develop mobile applications that can connect to the Raspberry Pi server and act as clients, allowing users to view live video streams on their smartphones or tablets.

2. Multiple Client Support: Enhancing the system to support multiple clients simultaneously can enable broader access to the video streams, catering to scenarios where multiple viewers or users need to access the same video feed.

3. Enhanced Security Features: Implementing encryption and authentication mechanisms to secure the video streams and prevent unauthorized access or tampering can enhance the project's security aspect.

4. Integration with Cloud Services: Integrating the project with cloud platforms or services like AWS or Azure can provide scalability, storage, and additional processing capabilities for the video streams.

5. Video Analytics and Machine Learning: Introducing video analytics and machine learning algorithms to analyze the video streams in real-time can unlock advanced capabilities such as object detection, motion tracking, or event recognition, opening up opportunities for various intelligent video applications.

The future scope of the project lies in further enhancing its features, expanding its capabilities, and exploring integration possibilities with other technologies to address specific industry requirements and use cases.

**REFLECTIONS :**

The "Video Streaming using Raspberry Pi with Server and Client in Headless Mode" project enables video streaming capabilities using Raspberry Pi devices, providing a cost-effective solution for various applications. It demonstrates the feasibility of utilizing Raspberry Pi as a video streaming server and client, even in headless mode.

By implementing the project, we have successfully achieved video encoding, streaming, and decoding functionalities on the Raspberry Pi platform. The project demonstrates the potential of Raspberry Pi devices in video streaming applications, showcasing their versatility and capability to handle video processing tasks

**References**

Textbooks:

[1] Rafael C. Gonzalez and Richard E. Woods, '' Digital Image processing '', 3rd edition 2008

Websites:

[1] <https://randomnerdtutorials.com/video-streaming-with-raspberry-pi-camera/>

[2] <https://www.javatpoint.com/dip-types-of-images>

[3] <https://en.wikipedia.org/wiki/Image_file_format>