**Research Training**

**Report**

**On**

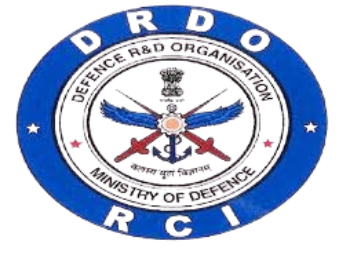
**“ Image Processing ”**

**At**

**Defence Research and Development Organisation**

**Jodhpur - 342001**

**Durations - 5 June 2025 to 31 September 2025**



**Submitted By :- Submitted To :-**

**Ms. Sneha Dadhich Mr. Rajendra Kumar Khatri**

**Computer Science and —----------------------------------**

**Engineering (Data Science)**

**Jodhpur Institute of —-----------------------------------**

**Engineering and Technology**

**Jodhpur - 342001 —-----------------------------------**

**DECLARATION**

I, **Sneha Dadhich**, hereby declare that the training report entitled **Neural Radiance Field (NeRF)** submitted to the **Defence Research and Development Organisation (DRDO), Jodhpur**, is a record of my original work carried out during my training period.

The report has been prepared under the guidance and supervision of my mentor, **Dr. Rajendra Kumar Khatri,** and has not been submitted to any other institute or organization for the award of any degree, diploma, or certification.

I further declare that all sources of information used in this report have been duly acknowledged.

**Submitted By :-**

Ms Sneha Dadhich

Date - 31st September 2025

**ACKNOWLEDGEMENT**

I would like to express my sincere gratitude to my DRDO mentor, **Dr. Rajendra Kumar Khatri**, for his valuable guidance, active support, and timely responses throughout my training. His mentorship not only enriched my learning experience but also helped me gain practical insights into the subject.

I am also deeply thankful to my college mentor, **Prof. Hemant Jain** (JIET, Jodhpur), for his constant encouragement and flexibility during the course of this internship, which allowed me to complete the training and this report smoothly.

Finally, I extend my appreciation to the **Defence Research and Development Organisation (DRDO), Jodhpur**, for providing me with the opportunity to undertake this training and for the resources and environment that contributed significantly to my professional development.

**Sneha Dadhich**

**TABLE OF CONTENTS**

1. **Institute Information……………………………………………………………… 5**
   1. **About DRDO**
   2. **Mission and Vision of DRDO**
   3. **DRDO Contributions**
2. **Image Processing (via OpenCV and MATLAB)........................................... 9**
   1. **Introduction**
   2. **Objectives of Image Processing**
   3. **Applications of Image Processing**
   4. **Image Processing with OpenCV**
   5. **Image Processing with MATLAB**
   6. **Relevance to Training**
3. **FFmpeg.…………………………………………………………………………… 11**
   1. **Introduction**
   2. **Key Features**
   3. **Applications**
   4. **Working Principle**
   5. **Role in the Project**
4. **Neural Radiance Field (NeRF)..................................................................... 13** 
   1. **Introduction**
   2. **Mathematical Background of NeRF**
   3. **Applications of NeRF**
   4. **Volume Rendering Equation**
   5. **Working of NeRF**
   6. **Key Terms in NeRF**
   7. **Steps in NeRF Pipeline**
5. **COLMAP…………………………………………………………………………... 16**
   1. **Introduction**
   2. **Applications of COLMAP**
   3. **Workflow: COLMAP + NeRF Integration**
   4. **Why COLMAP is Important**
6. **3D Model Reconstruction from 360o Product Videos Using NeRF and COLMAP………………………………………………………………………….. 18**
   1. **Introduction**
   2. **Objectives**
   3. **Tools and Technologies Used**
   4. **Methodology**
   5. **Mathematical Simplification**
   6. **Results**
   7. **Example Outcomes**
   8. **Applications**
   9. **Challenges Faced**
   10. **Project Links**
7. **Conclusion………………………………………………………………………. 20**
8. **Reference………………………………………………………………………… 21**

**About DRDO**

The **Defence Laboratory, Jodhpur (DLJ)** was established in **May 1959** to address challenges related to the harsh desert environment and its impact on desert warfare.

The initial charter of the laboratory included:

* Undertaking field trials on weapons and equipment either newly designed or indigenously developed with imported know-how.
* Conducting basic research relevant to the arid zone.
* Carrying out physiological studies, radio-wave propagation studies, and solar energy research.

DLJ is the **westernmost and strategically significant laboratory** of the **Defence Research and Development Organisation (DRDO)**. Earlier, it was located in **Ratanada Palace, Jodhpur** (now functioning as a DRTC Training Centre), and later shifted to the **New Technical Complex (NTC)**.

The laboratory plays a vital role in the **development of electronics and materials for modern warfare and weapon systems**. Its primary research areas are **Materials Science** and **Electronics**, with notable contributions in the following domains:

* Development of advanced radio communication systems.
* Secure data links and networking systems.
* Millimeter wave communication technologies.

### **Research Divisions at DLJ**

DLJ consists of three major divisions:

1. **Camouflage and Low Observation Technology Division**
2. **Nuclear Radiation Management and Applications (NRMA) Division**
3. **Desert Environment Science and Technology Division**

The name suggests the primary functions of these groups, further in these divisions there are various groups working on different technologies. DLJ (Defence Laboratory, Jodhpur) has developed several nuclear radiation monitoring systems and camouflage techniques for the Armed Forces of India . Few scientist are also engazed in field of microbiology and biotechnology.

**Mission and Vision of DRDO**

## **Mission**

The mission of the Defence Research and Development Organisation (DRDO) reflects its responsibility to design and develop cutting-edge defence technologies that strengthen the capabilities of the Indian Armed Forces. Its core objectives can be understood through the following pillars:

* Design, Development, and Production of Advanced Systems:  
   DRDO is committed to creating state-of-the-art sensors, weapon systems, platforms, and allied equipment that meet the dynamic needs of the Army, Navy, and Air Force. By developing technologies that are on par with global standards, DRDO ensures that India’s defence preparedness remains strong in an era of rapid technological change.
* Providing Technological Solutions to Armed Forces:  
   In modern warfare, technological superiority is as important as manpower. DRDO aims to deliver innovative solutions to enhance combat effectiveness, reduce vulnerabilities, and improve operational readiness. Whether in the form of missile defence systems, electronic warfare solutions, or unmanned aerial vehicles, the organisation ensures that the Indian Armed Forces are always equipped with reliable and advanced technologies.
* Promoting Self-Reliance:  
   A central part of DRDO’s mission is to reduce dependency on foreign defence imports. By focusing on indigenous research, design, and production, DRDO contributes to the vision of *“Atmanirbhar Bharat” (Self-Reliant India)*. This not only strengthens national security but also saves valuable foreign exchange and fosters domestic industrial growth.
* Nurturing a World-Class Work Culture:  
   DRDO seeks to create an environment of scientific curiosity, innovation, and professionalism. By encouraging collaboration, knowledge-sharing, and continuous skill development, it attracts bright minds to contribute to national defence research and development. The organisation actively involves young engineers and scientists, fostering a new generation of innovators in defence technology.

## 

## 

## **Vision**

The vision of DRDO defines its long-term aspirations and the role it seeks to play in strengthening India’s global standing:

* **Self-Reliance in Defence Technologies:**  
   DRDO envisions an India where critical defence technologies are designed, developed, and manufactured indigenously. This vision aligns with the country’s strategic goal of reducing dependency on foreign suppliers and ensuring that the nation can independently sustain its defence requirements in any situation.
* **Equipping the Armed Forces with Cutting-Edge Technology:**  
   To safeguard national security in the modern era, DRDO strives to deliver advanced systems that enhance operational effectiveness. From hypersonic missiles and stealth aircraft to cyber defence solutions and AI-based surveillance systems, the vision is to ensure that the Indian Armed Forces remain future-ready.
* **Becoming a Global Leader in Defence Science and Innovation:**  
   Beyond serving national interests, DRDO aspires to establish India as a hub of defence innovation on the global stage. By achieving excellence in research, fostering international collaborations, and continuously pushing the boundaries of science, DRDO contributes not only to India’s security but also to the world’s scientific and technological advancement.

Together, the Mission and Vision of DRDO highlight its dual responsibility: securing India’s defence preparedness today and shaping the future of defence technology for tomorrow.

# **DRDO Contributions and Major Projects**

DRDO has been the backbone of India’s defence modernization, successfully delivering a wide range of indigenous technologies and systems. Some of the landmark contributions include:

1. **Missile Development**
   * Integrated Guided Missile Development Programme (IGMDP).
   * Development of missiles like Agni, Prithvi, Akash, Nag, and BrahMos (jointly with Russia).
2. **Aeronautics & Combat Vehicles**
   * Light Combat Aircraft (Tejas).
   * Arjun Main Battle Tank (MBT) and Bhishma (T-90 modernization).
3. **Naval Systems**
   * Advanced sonar systems like PANCHENDRIYA.
   * Indigenous torpedoes and underwater surveillance systems.
4. **Electronics & Radar**
   * Long-range surveillance radars such as Rohini and Swordfish.
   * Electronic warfare systems for Army, Navy, and Air Force.
5. **Life Sciences & Support Systems**
   * Development of protective gear, combat rations, and high-altitude survival kits for soldiers deployed in extreme conditions (e.g., Siachen Glacier).
6. **Recent Contributions**
   * Development of hypersonic technologies, anti-satellite weapon (ASAT) capabilities, and AI-based defence solutions.
   * Support in indigenization of drones, UAVs, and loitering munitions.

Through these projects, DRDO has strengthened India’s defence preparedness and reduced dependence on foreign technologies, contributing significantly to the concept of “Atmanirbhar Bharat” (Self-Reliant India).

# 

# **Image Processing (via OpenCV and MATLAB)**

## **Introduction**

Image processing is a method of performing operations on images to enhance their quality, extract meaningful information, and prepare them for further analysis. It is widely used in computer vision, artificial intelligence, robotics, and scientific research.

With the advancement of computational power, image processing has become a crucial tool for analyzing digital images in both 2D and 3D forms. Two of the most popular tools for implementing image processing algorithms are:

* OpenCV (Open Source Computer Vision Library): A powerful, open-source computer vision library written in C++ and Python that supports real-time image processing and machine learning.
* MATLAB: A high-level programming platform that provides an extensive set of built-in functions and toolboxes for image processing, mathematical modeling, and visualization.

During my training at DRDO, Jodhpur, I worked extensively with both OpenCV and MATLAB to explore the fundamentals of image processing, and later utilized them in developing my project based on Neural Radiance Fields (NeRF) for converting video into a 3D model.

## **Objectives of Image Processing**

The main goals of image processing include:

1. Visualization – Enhancing image quality for better human interpretation.
2. Image Restoration – Removing noise and correcting distortions.
3. Image Compression – Reducing storage and transmission requirements.
4. Pattern Recognition – Detecting objects, faces, or structures in an image.
5. Machine Perception – Enabling systems to understand and act upon visual data.

## **Applications of Image Processing**

Image processing has numerous real-world applications across diverse fields:

* Defence and Security: Object detection, target tracking, surveillance systems, and night vision enhancement.
* Medical Imaging: MRI, CT scans, X-rays, and automated diagnosis.
* Satellite Imaging: Terrain mapping, environmental monitoring, and disaster management.
* Biometrics: Face recognition, fingerprint recognition, iris scanning.
* Autonomous Vehicles: Road lane detection, obstacle avoidance, traffic sign recognition.
* Augmented and Virtual Reality (AR/VR): Converting 2D data into immersive 3D models.

## **Image Processing with OpenCV**

OpenCV provides efficient tools for:

* Basic Image Operations – Reading, writing, resizing, and rotating images.
* Filtering and Enhancement – Applying Gaussian blur, sharpening, and edge detection.
* Object Detection – Haar cascades, HOG, and deep learning-based detectors.
* Feature Extraction – Identifying corners, edges, and contours.
* Video Processing – Frame-by-frame analysis, motion tracking, and video stabilization.

OpenCV’s strength lies in its real-time performance, making it suitable for applications like surveillance, robotics, and real-time monitoring.

## **Image Processing with MATLAB**

## MATLAB offers a rich Image Processing Toolbox that includes:

## Pre-processing – Noise removal, contrast adjustment, and image segmentation.

## Morphological Operations – Dilation, erosion, opening, and closing.

## Transformations – Fourier transform, wavelet transform, and geometric transformations.

## 3D Visualization – Reconstruction of images and volumetric data.

## GUI Development – Building custom interactive tools for image analysis.

## MATLAB’s strength lies in its ease of visualization, prototyping, and mathematical modeling, which complements OpenCV’s speed and flexibility.

## **Relevance to Training**

## In my project at DRDO, Jodhpur, I utilized OpenCV for handling image and video data, applying pre-processing techniques such as frame extraction, noise removal, and edge detection. Parallelly, MATLAB was used for performing advanced transformations and visualization tasks.

**FFmpeg**

### **Introduction**

FFmpeg (Fast Forward MPEG) is a powerful open-source multimedia framework widely used for handling audio, video, and other multimedia files. It provides a set of libraries and command-line tools for recording, converting, and streaming multimedia content. FFmpeg is highly versatile, supporting a vast range of codecs and file formats, which makes it a standard tool in the field of video processing, computer vision, and research-based projects.

In this project, FFmpeg was primarily used for pre-processing and post-processing tasks, such as converting raw video sequences into suitable input formats, extracting frames, and generating final rendered outputs.

### **Key Features**

* Supports a wide range of video/audio codecs (MPEG, H.264, VP9, AV1, etc.).
* Ability to decode, encode, transcode, mux, demux, stream, filter, and play multimedia.
* High performance with efficient compression techniques.
* Compatible across operating systems (Linux, Windows, macOS).
* Scriptable through the command line, making it easy to integrate into automated workflows.

### **Applications**

1. **Video Conversion:** Convert raw video files into required formats (e.g., .mp4, .avi, .mkv) for model training and visualization.
2. **Frame Extraction:**  Extract individual frames from video sequences, which were used as input for COLMAP and NeRF training.
3. **Image Sequence to Video:** Combine a series of rendered frames generated by NeRF into a single video output.
4. **Compression and Optimization:** Compress large multimedia files without significant loss of quality, helping reduce storage and processing requirements.
5. **Streaming and Visualization:** Used to prepare rendered models for presentation or streaming purposes.

### **Working Principle**

FFmpeg is structured around multiple libraries:

* **libavcodec** – for encoding/decoding audio and video.
* **libavformat** – for demuxing/muxing multimedia container formats.
* **libavfilter** – for applying filters and effects.
* **libavdevice** – for capturing and input/output handling.
* **libswscale** – for scaling and color space conversion.
* **libswresample** – for audio resampling.

Through its command-line tool ffmpeg, these libraries can be accessed using simple commands. For example, extracting frames can be done using:

* ffmpeg -i input.mp4 -vf fps=30 frames/output\_%04d.png

### **Role in Project**

* **Before Training (Pre-processing):** FFmpeg was used to extract image frames from 360° product videos, which were later processed using COLMAP and NeRF.
* **After Training (Post-processing):** The final rendered frames generated by NeRF were combined into smooth videos using FFmpeg. This made it possible to visualize the reconstructed 3D models in an interactive way.

**Neural Radiance Fields (NeRF)**

## **Introduction**

In recent years, Neural Radiance Fields (NeRF) have emerged as a revolutionary approach in computer vision and 3D reconstruction. NeRF uses deep learning techniques to generate novel 3D views from 2D images or video frames by modeling how light radiates from different points in a scene. Unlike traditional 3D reconstruction methods that rely on explicit geometry, NeRF employs neural networks to represent both the geometry and appearance of a scene implicitly.

This makes NeRF particularly useful for applications requiring photo-realistic rendering, immersive visualization, and scene reconstruction, making it a strong research direction for fields such as defence, robotics, gaming, and AR/VR technologies.

During my training at DRDO, Jodhpur, I worked on implementing a NeRF pipeline, where I processed video data into frames (via OpenCV and MATLAB), pre-processed the inputs, and then used NeRF techniques along with COLMAP for camera pose estimation and scene reconstruction.

## **Mathematical Background of NeRF**

## At its core, NeRF represents a 3D scene as a continuous function that maps:

* Fθ ​: (x,y,z,d) → (c,σ)

where:

* (x, y, z) → 3D coordinates of a point in the scene
* d → viewing direction (the direction of the camera ray)
* C = (r,g,b) → predicted color at that point
* σ → density (opacity) of that point
* θ → parameters of the neural network

This function is implemented as a neural network (usually an MLP) which learns to approximate how light interacts with the scene.

## **Applications of NeRF**

NeRF has shown promising results in a wide range of applications:

1. 3D Scene Reconstruction: Converting 2D images or video into detailed 3D models for simulation and visualization.
2. Augmented Reality (AR) and Virtual Reality (VR): Generating immersive environments with high realism for training, gaming, and visualization.
3. Cultural Heritage Preservation: Creating 3D digital twins of monuments and historical artifacts from photographic data.
4. Robotics and Autonomous Systems: Helping robots understand their surroundings by reconstructing scenes in 3D.
5. Medical Imaging: Extending traditional scans (CT/MRI) into volumetric visualizations for diagnosis and surgical planning.
6. Defence and Security: NeRF can enhance surveillance, simulation-based training, and mission planning by providing accurate 3D models of terrains or urban environments.

### **Volume Rendering Equation**

When we look at a pixel in an image, it corresponds to a ray of light passing through the 3D scene. NeRF estimates the pixel color by integrating contributions of many sampled points along that ray.

For a ray r(t)= o + td

​t

where:

* T(t)= exp⁡(−∫σ(r(s))ds ) is the transmittance (probability that light travels without being absorbed).
* σ(r(t)) is the density at point r(t) .
* c(r(t),d) is the color emitted from that point in direction d.

This equation essentially says: the final pixel color is the sum of all light along the ray, weighted by how transparent or opaque each point is.

## **Working of NeRF**

1. **Input Data**
   * NeRF requires multiple 2D images of a scene from different angles.
   * Camera poses (extrinsics and intrinsics) are provided by tools like COLMAP.
2. **Neural Network Architecture**
   * A multi-layer perceptron (MLP) is trained.
   * Input: (x,y,z,d)(x, y, z, d)(x,y,z,d)
   * Output: Color (R,G,B) and Density (σ\sigmaσ).
3. **Ray Sampling**
   * For each pixel, a ray is cast.
   * Points are sampled along the ray (stratified/random sampling).
4. **Volume Rendering**
   * The predicted color and density values are combined using the rendering equation.
   * This produces an estimate of the pixel’s color.
5. **Loss Function:** The predicted pixel colors are compared to ground-truth image pixels using Mean Squared Error (MSE).

L = 1/N + (Cpredicted - Ctrue)2

1. **Optimization:** The neural network parameters (θ\thetaθ) are optimized using gradient descent until the network can accurately reproduce the training images.
2. **Novel View Synthesis:** Once trained, the model can render the scene from new, unseen viewpoints, producing photo-realistic 3D reconstructions.

**Key Terms in NeRF**

To understand NeRF, some important concepts must be clarified:

* Radiance: The amount of light emitted from a point in a particular direction.
* Volume Rendering: A method used to simulate how light passes through a medium and contributes to an image.
* Implicit Representation: Instead of explicitly storing 3D geometry (meshes/voxels), NeRF uses a neural network to encode it.
* Camera Pose: The position and orientation of a camera relative to the scene. This is essential for reconstructing a consistent 3D model.
* Ray Sampling: NeRF models how rays of light travel through a 3D scene, sampling points along each ray to predict color and density.

## **Steps in NeRF Pipeline**

The general workflow of a NeRF project includes the following steps:

* Data Collection: Input is a set of 2D images or frames extracted from a video, captured from different viewpoints.
* Camera Pose Estimation (via COLMAP): The relative positions and orientations of the cameras are computed using a Structure-from-Motion (SfM) algorithm. COLMAP is widely used here.
* Preprocessing: Frames are resized, noise is reduced, and unnecessary distortions are corrected.
* NeRF Model Training: A deep neural network is trained to learn the mapping between 3D coordinates and viewing direction → to output color and density.
* Volume Rendering: Novel views are synthesized by casting rays through the scene and integrating predicted colors and densities.
* 3D Model Generation: The trained NeRF can render highly realistic 3D views from unseen angles, effectively reconstructing the scene.

# 

# **COLMAP**

## **Introduction**

COLMAP is a widely used Structure-from-Motion (SfM) and Multi-View Stereo (MVS) software that reconstructs 3D structures from a set of 2D images. It is particularly powerful in estimating camera intrinsics and extrinsics (pose estimation) and generating a sparse or dense point cloud of the scene.

In my project, COLMAP played a crucial role in preparing the data for NeRF by providing accurate camera pose estimations, which were then used as inputs to train the NeRF model.

## **Applications of COLMAP**

* Camera Pose Estimation: Determining how each image relates spatially to the others.
* Sparse and Dense Reconstruction: Building a point cloud of the scene from multiple images.
* Dataset Preprocessing for NeRF: Supplying essential metadata (camera intrinsics & extrinsics).
* 3D Model Generation: When combined with NeRF, it improves the accuracy of reconstruction and ensures realistic renderings.

## **Workflow: COLMAP + NeRF Integration**

1. Input Frames: Extracted from video or taken as separate images.
2. Feature Extraction (COLMAP): Detecting and matching key points across frames.
3. Camera Pose Estimation (COLMAP): Generating intrinsic and extrinsic parameters.
4. Passing Data to NeRF: Using COLMAP outputs as inputs for NeRF training.
5. 3D Scene Rendering (NeRF): Producing high-quality 3D visualizations of the scene.

## **Why COLMAP is Important**

NeRF’s accuracy heavily depends on knowing where the cameras were located when the images were captured. COLMAP provides:

* Camera Intrinsics: Focal length, principal point, distortion parameters.
* Camera Extrinsics: Rotation and translation of each camera (pose).

This ensures that NeRF understands how images are spatially related, which is critical for consistent 3D reconstruction.

**3D Model Reconstruction from 360° Product Videos using NeRF and COLMAP**

## **Introduction**

With the increasing demand for immersive experiences in e-commerce, virtual reality (VR), augmented reality (AR), and gaming, the ability to generate realistic 3D models from ordinary videos has become an important area of research and application. Traditionally, 3D modeling required manual design using software like Blender, Maya, or CAD tools, which is time-consuming and requires expertise.

Recent advancements in Neural Radiance Fields (NeRF) and Structure-from-Motion (SfM) pipelines such as COLMAP have revolutionized the way 3D objects and scenes can be reconstructed from 2D images or videos. NeRF uses deep learning techniques to synthesize realistic 3D views of a scene by training on multiple 2D images, while COLMAP provides robust camera pose estimation and sparse reconstruction.

In this project, I implemented a system that takes a 360-degree rotating video of a product as input and generates a realistic 3D model. The goal was to automate product digitization for applications like online shopping previews, digital marketing, and AR-based product visualization.

## **Objectives**

The main objectives of this project were:

1. To preprocess 360° product videos and extract high-quality frames for model training.
2. To use COLMAP for feature extraction and camera pose estimation.
3. To apply NeRF for reconstructing the 3D radiance field from the video frames.
4. To render novel views of the product from arbitrary angles.
5. To explore practical applications of 3D model reconstruction in real-world use cases.

**Tools and Technologies Used**

* Programming Language: Python
* Libraries/Frameworks: PyTorch, NumPy, OpenCV
* Neural Rendering Framework: NeRF (Nerfstudio / Original NeRF Implementation)
* 3D Reconstruction Tool: COLMAP
* Visualization: Blender / Matplotlib for rendering and visualization
* Hardware: GPU-enabled system (for faster NeRF training)

**Methodology**

The project followed the below workflow:

### **Data Collection**

* A rotating 360° product video was recorded.
* The video was then converted into frames using OpenCV, generating ~200–300 high-quality images.

### **Preprocessing**

* Images were resized and background noise was reduced to ensure better model convergence.
* COLMAP was applied to extract camera intrinsic and extrinsic parameters, and to generate sparse point clouds.

### **COLMAP for Camera Pose Estimation**

* COLMAP performed feature matching between frames (using SIFT features).
* A Structure-from-Motion (SfM) pipeline was used to recover the relative poses of the camera.
* Output: Camera parameters and sparse 3D points necessary for NeRF training.

### **NeRF Training**

* NeRF was used to model the scene as a continuous 5D function *(spatial location (x,y,z) and viewing direction (θ,ϕ))* that outputs:
  + 1. Density (σ): probability of light being emitted/absorbed
    2. Color (RGB): appearance of the object at that location
* Training involved volumetric rendering: NeRF optimizes a neural network to minimize the error between rendered views and the actual 2D frames.

1. **Rendering**

* Once trained, the NeRF model was used to render novel views of the product from arbitrary angles.
* This enables rotation, zoom, and AR integration without additional video captures.

**Mathematical Simplification:**

* Each camera ray is sampled at intervals.
* The network predicts density + color at each point.
* The final pixel color is computed as:

C(r)=∑i=1N​Ti​(1−e−σi​δi​)ci​

* where TiT\_iTi​ is transmittance, σi\sigma\_iσi​ is density, δi\delta\_iδi​ is distance, and cic\_ici​ is RGB color.

## **Results**

* A realistic 3D reconstruction of the product was obtained from the 360° video.
* The model successfully allowed viewing from unseen perspectives (not present in the original video).
* Rendering quality improved with more training iterations and higher frame sampling.

**Example Outcomes:**

* Smooth 360° rotations of the object.
* Possibility of exporting mesh representations (using marching cubes algorithm) for CAD/AR usage.

## **Applications**

1. E-Commerce – Virtual product try-ons, 360° previews, and AR shopping.
2. AR/VR Experiences – Creating immersive product demonstrations.
3. Digital Marketing – Interactive 3D advertisements.
4. Cultural Preservation – 3D scanning of artifacts using simple videos.
5. Gaming and Simulation – Quick asset creation from real-world objects.

## **Challenges Faced**

* High GPU resource requirement for training NeRF.
* Difficulty in handling reflective or transparent product surfaces.
* Longer training times depending on dataset size and resolution.
* COLMAP failure in low-texture regions (feature extraction issues).

**Project Links**

* Practice Code Github - <https://github.com/Sneha-Dadhich/ImageProcessing>
* Main Project Link -

**CONCLUSION**

During my training internship, I worked on the project **“3D Model Reconstruction from 360° Videos using NeRF and COLMAP”**, which focused on generating realistic 3D models from simple product videos. This project gave me practical exposure to modern technologies like **Neural Radiance Fields (NeRF)** for neural rendering and **COLMAP** for structure-from-motion and camera pose estimation.

Through this work, I learned how raw data in the form of video frames can be processed and transformed into interactive 3D models. The project not only improved my technical skills in **Python, deep learning frameworks, and 3D vision tools**, but also helped me understand the importance of preprocessing, data quality, and computational resources in achieving accurate results.

The final outcome successfully demonstrated the ability to **convert 360° product videos into detailed 3D models**, which can have real-world applications in **e-commerce, AR/VR, digital marketing, and gaming industries**. While challenges such as high GPU requirements and handling reflective surfaces were encountered, the project highlighted possible future improvements through optimization techniques and advanced NeRF variants.

Overall, this internship provided me with valuable **technical knowledge, problem-solving skills, and hands-on experience** in the field of computer vision and AI-driven 3D reconstruction. It has strengthened my foundation for pursuing future projects and research in artificial intelligence, graphics, and immersive technologies.

**REFERENCE**

* **YouTube** – *Introduction to Neural Radiance Fields (NeRF)*. Available at: [https://youtu.be/Z95tFjIPqc0](https://youtu.be/Z95tFjIPqc0?utm_source=chatgpt.com) (Accessed on: September 2025).
* **GeeksforGeeks** – *Neural Radiance Fields (NeRF) Introduction*. Available at: [https://www.geeksforgeeks.org/neural-radiance-fields/](https://www.geeksforgeeks.org/neural-radiance-fields/?utm_source=chatgpt.com) (Accessed on: September 2025).
* **FFmpeg Official Documentation** – *Video and Audio Processing Reference*. Available at: [https://ffmpeg.org/documentation.html](https://ffmpeg.org/documentation.html?utm_source=chatgpt.com) (Accessed on: September 2025).
* **Clipcat Blog** – *A Beginner’s Guide to Using FFmpeg in Python for Video Processing*. Available at: [https://www.clipcat.com/blog/a-beginners-guide-to-using-ffmpeg-in-python-for-video-processing/](https://www.clipcat.com/blog/a-beginners-guide-to-using-ffmpeg-in-python-for-video-processing/?utm_source=chatgpt.com) (Accessed on: September 2025).
* **FFmpeg Command Cheat Sheet** – PDF Resource for Quick Reference.
* **YouTube** – *COLMAP Installation Tutorial*. Available at: [https://youtu.be/kFJiUMELnH8](https://youtu.be/kFJiUMELnH8?utm_source=chatgpt.com) (Accessed on: September 2025).
* **COLMAP Official Documentation** – *Practical Tutorial on Structure-from-Motion*. Available at: [https://colmap.github.io/tutorial.html](https://colmap.github.io/tutorial.html?utm_source=chatgpt.com) (Accessed on: September 2025).
* **COLMAP Python API** – *PyCOLMAP Documentation*. Available at: [https://colmap.github.io/pycolmap/index.htm](https://colmap.github.io/pycolmap/index.htm?utm_source=chatgpt.com) (Accessed on: September 2025).
* Documentation and Correction - <https://chatgpt.com>