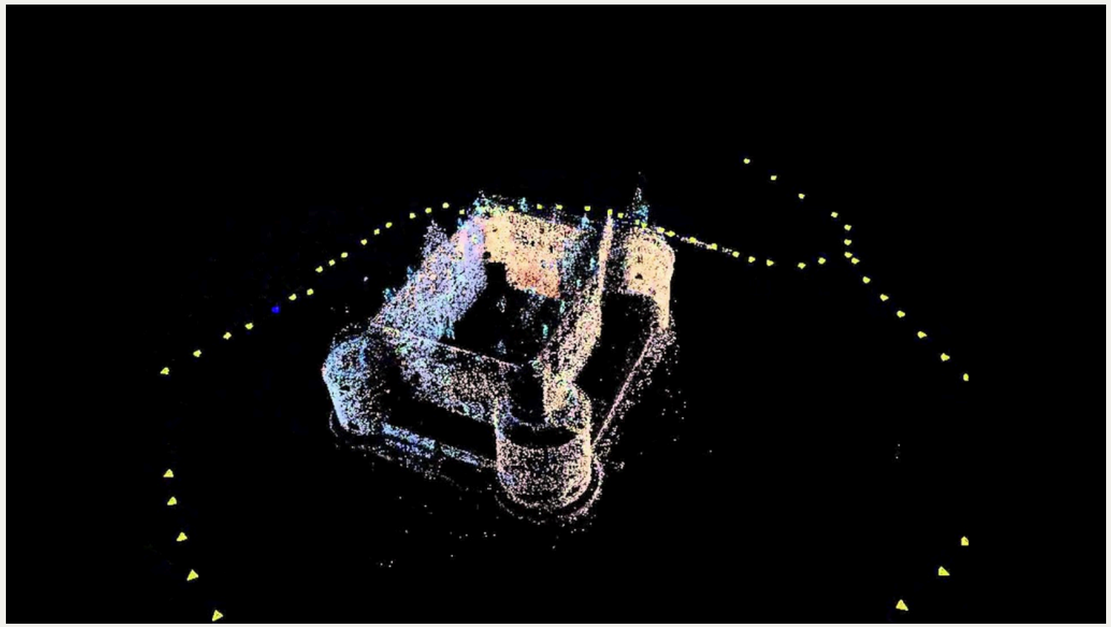
Abubakar Siddiq | 120403422 | absiddiq
Gayatri Davuluri | 120304866 | gayatrid

3D Reconstruction Buildings Built in minutes

Classical and Deep Learning Aprroaches



Key Concepts

SfM

What is Structure from Motion ?

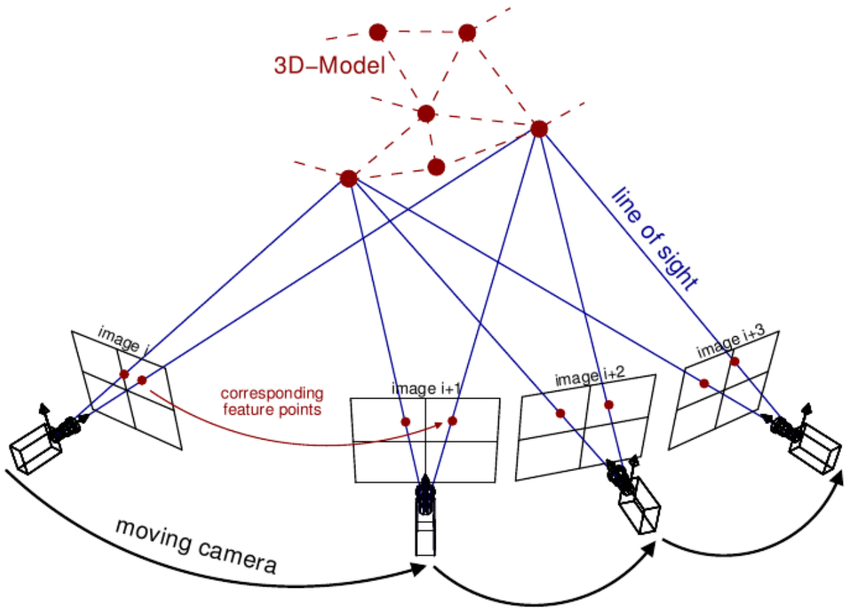
- Photogrammetric Technique
- 2D images --> 3D scene

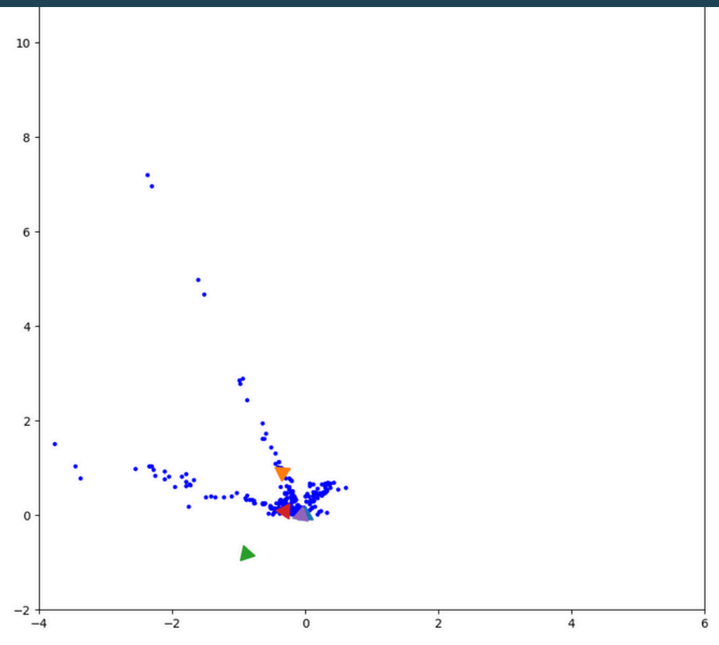
Classical

- Geometric techniques and Algorithms
- Optimization techniques

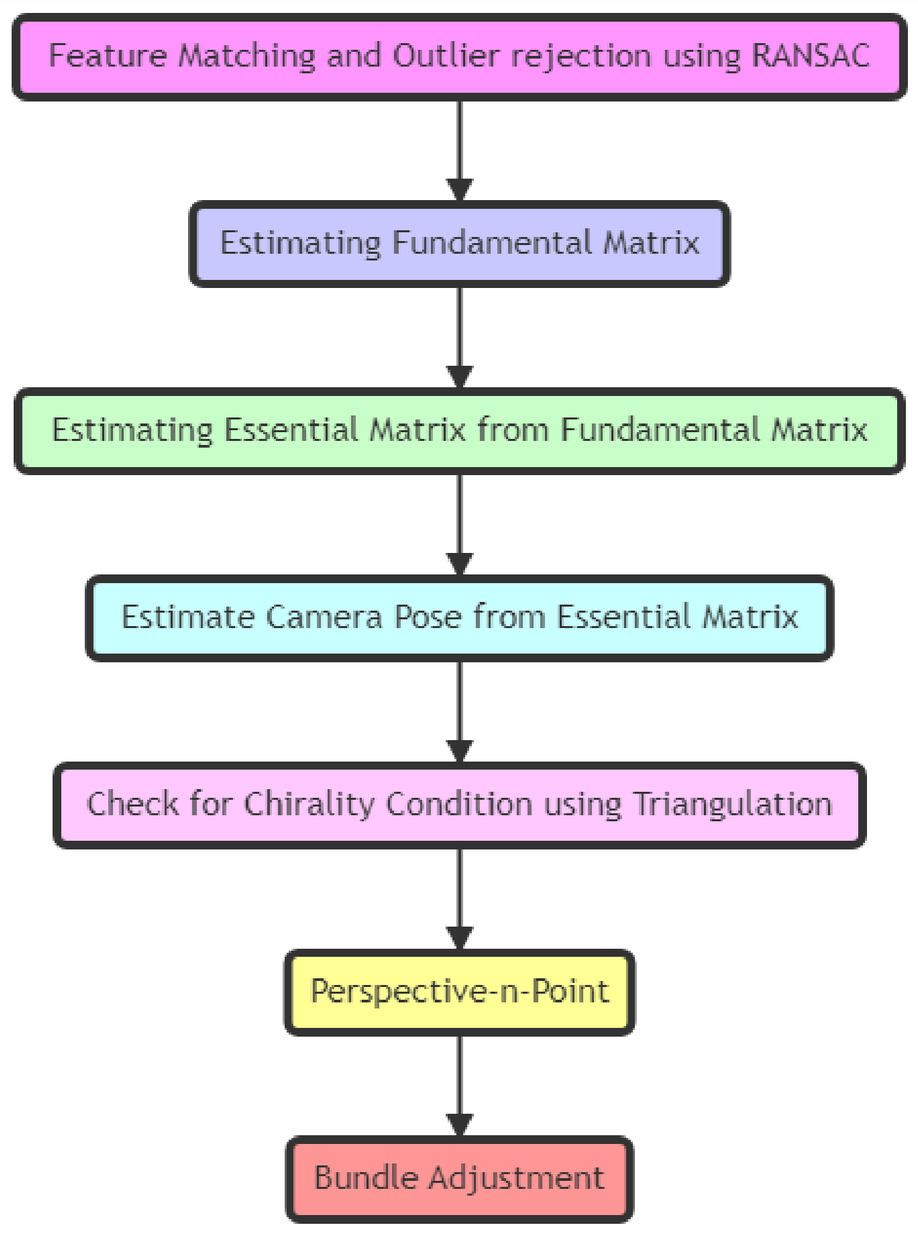
Deep Learning

- Leverage Neural Networks
- Enhanced structure understanding and accuracy.





Classical Approach



1. Feature Matching

- SIFT (Scale-Invariant Feature Transform) to detect features
- FLANN (Fast Library for Approximate Nearest Neighbors)

2. Fundamental Matrix estimation and Outlier rejection using RANSAC

- Fundamental Matrix (F) is an algebraic representation of Epipolar geometry

$$\begin{bmatrix} x_i' & y_i' & 1 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = 0$$

$$x_i x_i' f_{11} + x_i y_i' f_{21} + x_i f_{31} + y_i x_i' f_{12} + y_i y_i' f_{22} + y_i f_{32} + x_i' f_{13} + y_i' f_{23} + f_{33} = 0$$

- RANSAC (Random Sample Consensus) is utilized to further refine the estimation of the fundamental matrix by iteratively selecting a subset of the original matched points to find the best fitting model.
- It helps in eliminating outliers in the matches which could skew the estimation, ensuring that the resulting fundamental matrix is more reliable and robust against noise and incorrect matches.

2. Essential Matrix from Fundamental Matrix

- **Essential Matrix (E)** finds the relative camera poses between the two images using camera intrinsic parameters.

$$E = K^T F K$$

- As in the case of **F** matrix computation, the singular values of **E** are not necessarily (1,1,0) due to the noise in **K**.

- This can be corrected by reconstructing it with (1,1,0) singular values

$$\mathbf{E} = U \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} V^T$$

3. Estimate Camera Pose from Essential Matrix

- The camera pose consists of 6 degrees-of-freedom (DOF) Rotation (Roll, Pitch, Yaw) and Translation (X, Y, Z) of the camera with respect to the world.
- From Essential Matrix decomposition four camera poses can be derived

$$\mathbf{E} = UDV^T \text{ and } W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

1. $C_1 = U(:,3)$ and $R_1 = UWV^T$
2. $C_2 = -U(:,3)$ and $R_2 = UWV^T$
3. $C_3 = U(:,3)$ and $R_3 = UW^TV^T$
4. $C_4 = -U(:,3)$ and $R_4 = UW^TV^T$

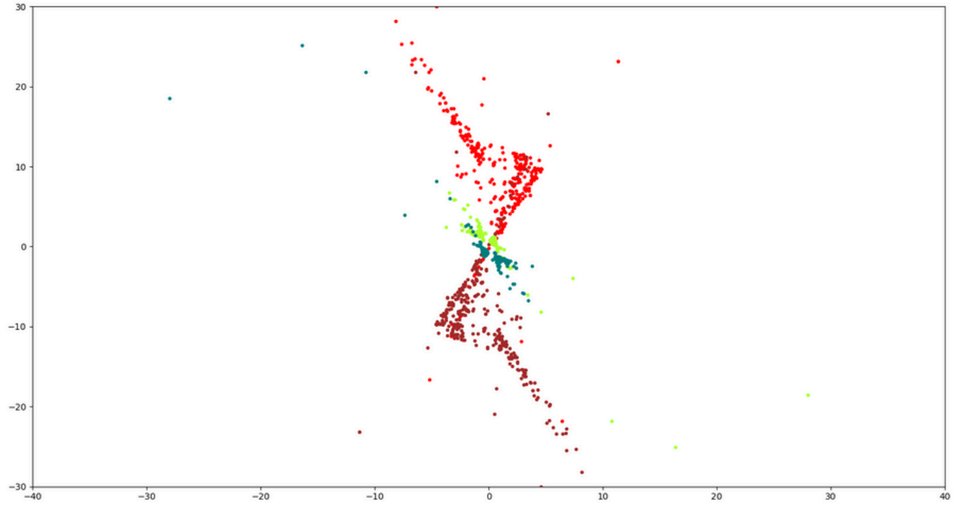
(C1, R1), (C2, R2), (C3, R3), (C4, R4)

Triangulation

- **3D Point Triangulation:** Using the camera poses, corresponding feature points from multiple images we can triangulate to estimate their 3D coordinates. This is achieved by finding the intersection of rays from each camera to the points, optimized to minimize projection errors.

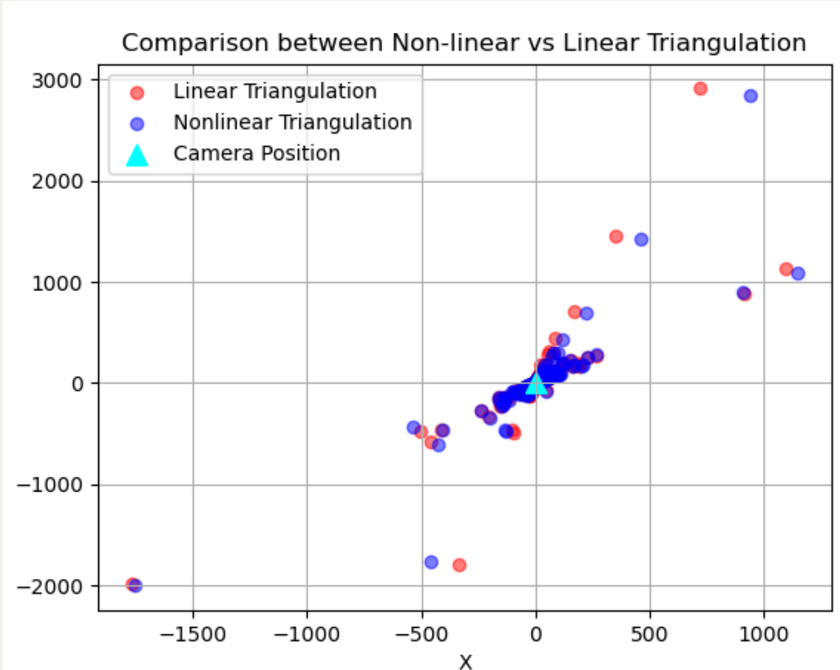
- Given four camera pose configurations and their triangulated points, we need to find the unique camera pose by checking the ***cheirality condition*** (reconstructed points must be in front of the cameras). This check is crucial to confirm that the reconstructed 3D scene is physically and geometrically valid.

- **Non-linear Optimization:** Refine the initial 3D point estimates through non-linear optimization techniques to further reduce reprojection errors. This step adjusts the 3D coordinates to provide a more accurate fit to the observed image data.



[1] Initial triangulation plot with disambiguity, showing all four possible camera poses

[2] Linear Triangulation vs Non Linear Triangulation



Perspective n Point

PnP methods estimate the camera pose from 3D points and their corresponding 2D projections. This is crucial for integrating new views into SfM reconstructions.

1. Linear PnP: Provides initial pose estimates quickly using linear algorithms, essential for preliminary alignment of 3D and 2D points.
2. RANSAC: Enhances pose accuracy by iteratively filtering out outliers, ensuring robustness in the presence of erroneous data.
3. Nonlinear Optimization: Refines camera poses by minimizing reprojection errors, employing advanced techniques like quaternion-based rotation to ensure precise alignment.

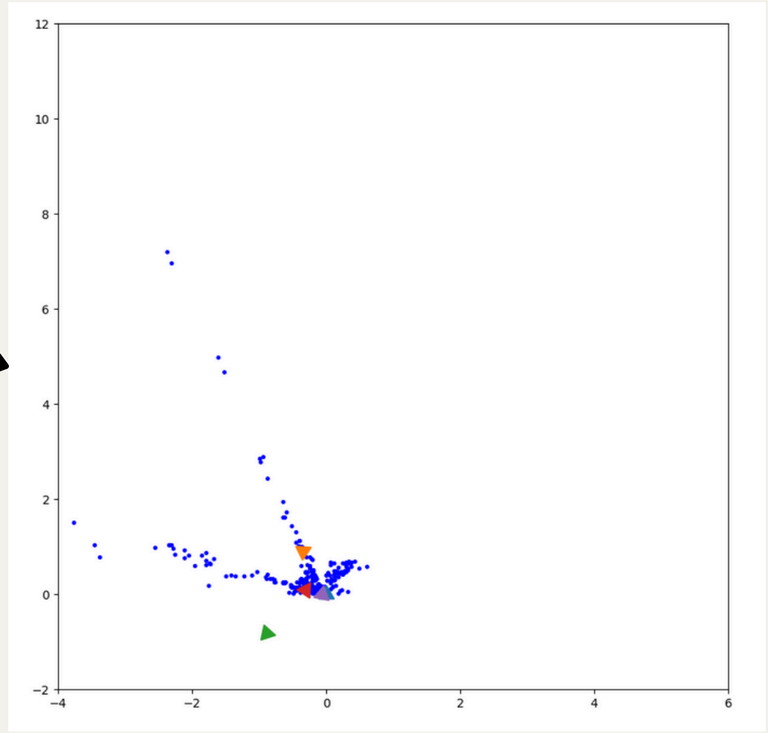
PnP method is used to accurately estimate the camera pose relative to a known 3D scene from its 2D image projections. This process is crucial when integrating new images into an existing 3D reconstruction

Bundle Adjustment and Result

- Once we have computed all the camera poses and 3D points, we need to refine the poses and 3D points together, initialized by previous reconstruction by minimizing reporjection error.
- The reprojection error is the difference between the observed positions of points in the images and the predicted positions based on the 3D model and the camera parameters.
- This is very complex and sensitive to tuning parameters and utilizes Visibility Matrix which indicates whether a point is visible in a specific image or not.

Original Image

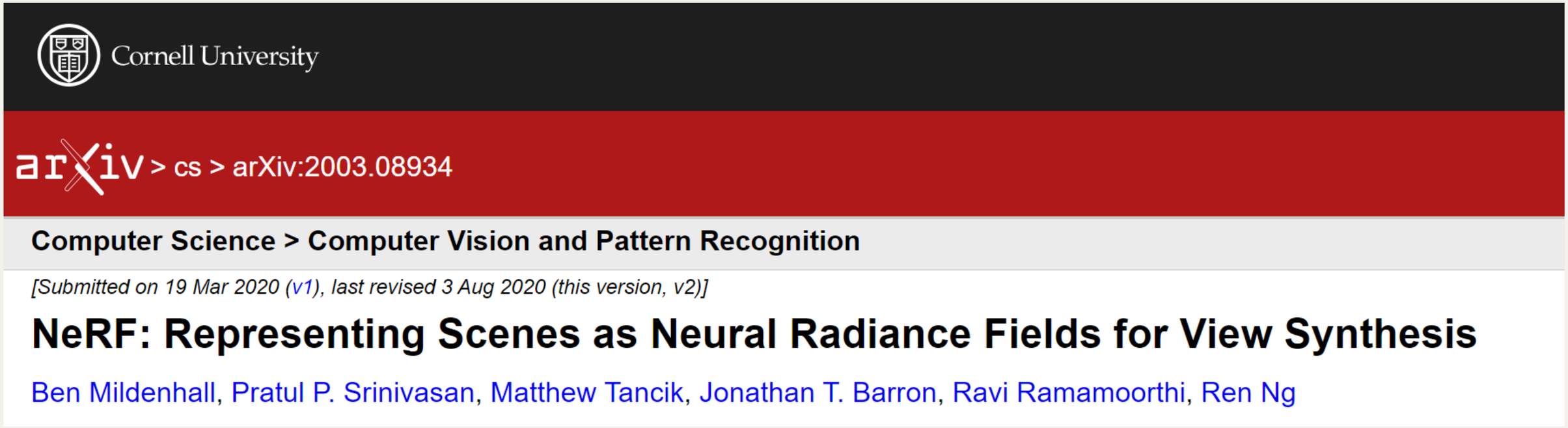
SfM Pipeline



Top view of the reconstructed scene

Deep Learning Approach to SfM

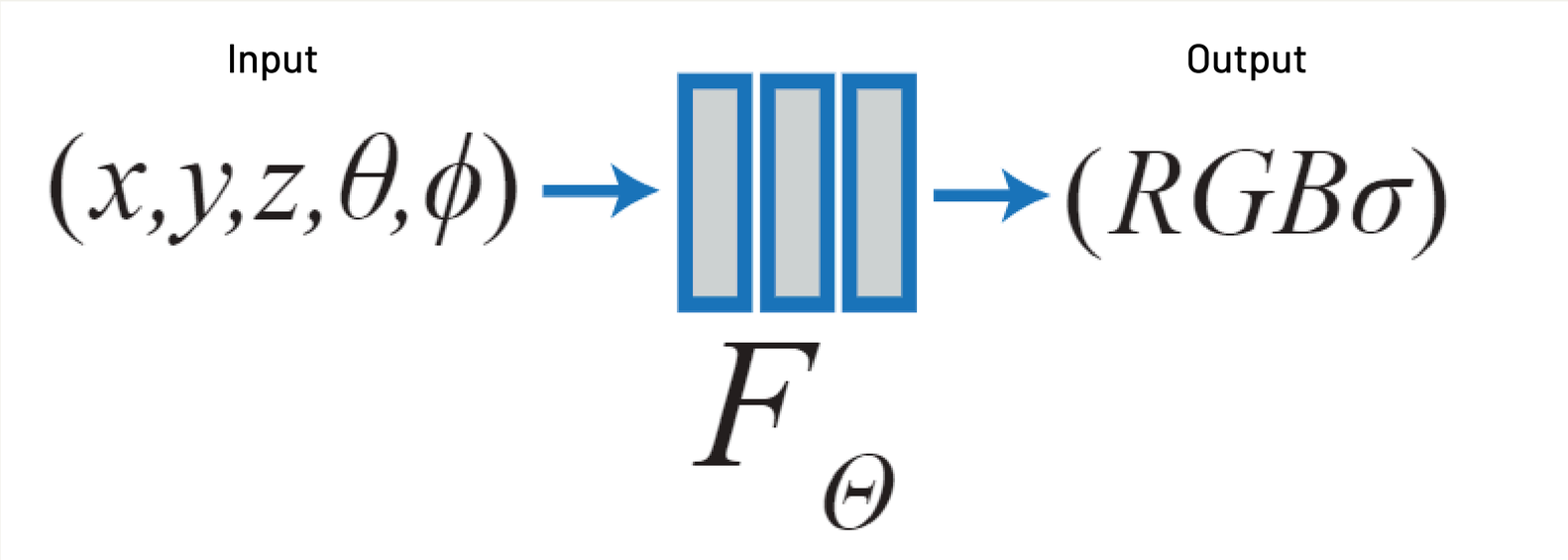
NERF (Neural Radiance Fields)



NERF

- Neural Radiance Fields (NeRF) is a deep learning technique used in Structure from Motion (SfM) to create highly detailed 3D models of scenes from a sparse collection of 2D images. It optimizes scenes with complicated geometry and appearance.

- Unlike traditional methods that directly compute 3D points, NeRF builds a neural network model that predicts the color and density of light at any point in a scene.

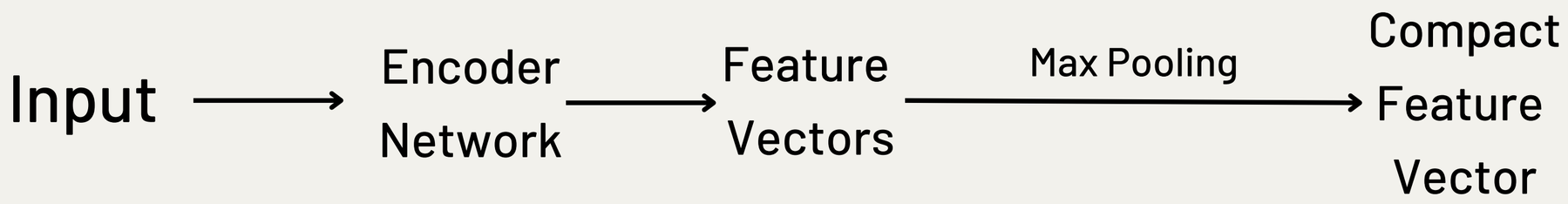


We Implemented Tiny Nerf

- TinyNerf is a lightweight variant of the Neural Radiance Fields (NeRF) architecture, designed to render high-quality 3D images from a sparse set of input views.
- **Input :** set of 2D images, along with their camera poses and intrinsics.

- Each input image is first passed through a small encoder network, which produces a set of feature vectors and the feature vectors from all the input images are then aggregated using a simple max-pooling operation to produce a single, compact feature vector

- Multi-layer perceptron (MLP) network, which consists of several fully connected layers with ReLU activations.
- Radiance Estimation: The output of the MLP network is then passed through a final layer that produces the RGB color and opacity values at each 3D point in the scene



DataSet and Workflow

- We utilized Lego Data set from the original *NERF* paper from *Cornell University*
- Ray generation from each pixel of the image which gives the direction and origin of the ray
- Performed uniform sampling along the ray with added noise and utilized.
- Positional encoding
- Input data is fed into the Tiny Nerf network
- Output: RGB color value and volume density at a specific location are predicted
- These predictions are inserted into the rendering equation which gives the final color and this is known as ***RADIANCE FIELD***

Network Training Parameters

Epochs = 1000

Mini Batch size = 128

Learning rate = 0.005

Results

NERF MODEL

Comparision

Classical approach SfM

- Outputs are sparse or dense point cloud. These point clouds consist of discrete points in 3D space, each with associated XYZ coordinates.
- Positions and orientations (poses) of the cameras used to capture the input images are also estimated as output
- The data from SfM is explicit and geometrically descriptive. The point clouds directly represent physical points in the scene.
- SfM relies heavily on accurately detecting and matching features across multiple images, which are susceptible to errors.
- SfM is used when direct measurements and structural details of the environment are needed

NERF

- NeRF does not produce point clouds or discrete geometric models. Instead, it outputs a continuous volumetric representation.
- The scene is represented implicitly by a neural network that predicts the color and density for any point given its coordinates and viewing direction.
- Unlike SfM, NeRF does not inherently calculate or output camera poses, instead it uses this data for training.
- The focus is more on synthesizing new views rather than determining camera parameters.
- NeRF is preferred for scenarios where visual quality and the realism of renderings are more critical.

Challanges Faced

- The SfM output is extremely proned to errors based on quality of the feature points detected and the optimization techniques used to refine the results.
- NERF is heavy to run in a local system and downgrading the model by reducing the number of layers and parameters in the model resulted in a lot of errors and debugging

THANK

YOU