# MCEN 5228: Project 4 - Multi-View Generation from Single Image Using 3D Reconstruction and Inpainting

Rama Chaganti
rama.chaganti@colorado.edu
Team Roronoa

Srikanth Popuri
srikanth.popuri@colorado.edu
Team Roronoa

***Abstract*—In this project, we explore the generation of multiple viewing angles from a single image by leveraging advanced techniques in depth estimation, 3D reconstruction, and image inpainting. Using RAFT-Stereo for stereo depth estimation, we generated accurate disparity maps to create a 3D point cloud representation of the scene. By manipulating camera poses through transformations, novel viewpoints were rendered to simulate multi-view perspectives. To address occlusions in the newly synthesized views, LaMa inpainting was employed, ensuring visually consistent and realistic outputs. This approach demonstrates the potential for realistic synthesis of unseen perspectives and has significant applications in virtual reality (VR), augmented reality (AR), and immersive content generation. The outcomes highlight the strengths of integrating state-of-the-art computer vision techniques, overcoming challenges such as occlusion handling and maintaining texture realism.**

**LINK TO CODE NOTEBOOK**

## I. Introduction

The ability to generate multiple viewing angles from a single image has transformative potential in domains such as virtual reality (VR), augmented reality (AR), and immersive content creation. Traditional methods for multi-view generation often rely on extensive datasets of multi-view images, which are both costly and time-intensive to capture. This project aims to address this limitation by synthesizing novel viewpoints from a single input image through a combination of depth estimation, 3D reconstruction, and inpainting.

Depth estimation plays a pivotal role in this pipeline, as it provides a foundational understanding of the scene's geometry. Leveraging RAFT-Stereo, a state-of-the-art stereo depth estimation model, we generated accurate disparity maps and converted them into 3D point clouds. These point clouds enable the creation of new camera perspectives through pose transformations, allowing for novel viewpoints to be rendered.

One of the significant challenges in multi-view generation is handling occlusions—regions that become visible in the new views but were not present in the original image. To address this, we utilized LaMa inpainting, an advanced deep learning-based model, to fill occluded areas with visually plausible textures.

The integration of these techniques demonstrates a robust pipeline for multi-view synthesis. This report details the methodology, challenges encountered, and results achieved, highlighting the potential of this approach to revolutionize content generation and visualization technologies.

## II. Methodology

The goal of this project was to synthesize multiple viewing angles from a single stereo image using advanced techniques in computer vision. The process unfolded through a structured pipeline, where each step played a crucial role in ensuring high-quality outputs. The pipeline integrated depth estimation, 3D reconstruction, and inpainting to achieve its objectives. Below, we describe each phase in detail.

### II-A Preparing the Environment and Data

The first step was setting up a robust environment capable of running RAFT-Stereo for depth estimation and LaMa for inpainting. A Conda environment was created to manage dependencies seamlessly, ensuring compatibility with Python and deep learning frameworks.

Next, the necessary resources, including pre-trained weights and datasets, were downloaded. These assets provided the foundation for the subsequent steps, allowing us to test and evaluate our pipeline with reliable data. This setup phase ensured that the tools were ready to process the input stereo images efficiently.

**Key Setup Steps**:

- Cloned the RAFT-Stereo repository.
- Installed dependencies using `environment.yaml`.
- Downloaded RAFT-Stereo weights and example datasets.

### II-B Depth Estimation: Unlocking 3D Information

Depth estimation was the cornerstone of this project, as it provided a 3D understanding of the scene. We used RAFT-Stereo, a state-of-the-art model that estimates disparity maps from stereo image pairs. RAFT-Stereo employs a recurrent architecture to iteratively refine its disparity predictions, ensuring robust and accurate results.

A stereo image pair, consisting of left and right images, was fed into the RAFT-Stereo model. The output was a *disparity map*, which encodes the pixel difference between corresponding points in the two images. Using camera parameters, this disparity map was transformed into a depth map—a 2D representation of the scene's depth.

**Key Equation**:

$$Z = \frac{f \cdot B}{d}$$

Where:

- $Z$ is the depth.
- $f$ is the focal length of the camera.
- $B$ is the baseline distance between stereo cameras.
- $d$ is the disparity.

This step laid the groundwork for 3D reconstruction by providing depth information for every pixel in the image.

### II-C From Pixels to Points: 3D Reconstruction

The depth map was then converted into a 3D point cloud, which represented the spatial structure of the scene. Each pixel from the depth map was mapped to a 3D coordinate using the intrinsic parameters of the camera. This mapping was performed using the equation:

**Key Equation**:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = Z \cdot K^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

Where:

- $X, Y, Z$ are the 3D coordinates.
- $(u, v)$ are the pixel coordinates.
- $K^{-1}$ is the inverse of the camera intrinsic matrix.

The resulting point cloud was a dense, spatial representation of the scene, enabling further transformations to simulate new viewpoints.

### II-D Changing Perspectives: Camera Pose Transformation

To render novel viewpoints, the camera pose was manipulated by applying transformations to the 3D point cloud. Rotation matrices were used to simulate changes in the camera's orientation. For instance, a rotation about the $z$-axis was applied using:

**Rotation Matrix**:

$$R_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

By rotating the 3D point cloud and projecting it back onto the 2D image plane, we created a set of novel perspectives, simulating how the scene would look from different angles.

### II-E Filling the Gaps: Inpainting with LaMa

One of the major challenges of generating new views is handling occlusions—regions that were not visible in the original image but become exposed in the new views. To address this, LaMa inpainting was employed.

LaMa excels at filling large missing regions in images by maintaining texture consistency. Using a combination of fast Fourier convolutions and deep learning, LaMa generated visually plausible content to fill these occluded areas, ensuring seamless integration with the rest of the scene.

This step was critical for enhancing the realism of the rendered viewpoints, as it tackled artifacts and missing textures effectively.

### II-F Automation and Iteration: Scaling the Pipeline

To create a comprehensive set of viewpoints, the pipeline was automated. The process was repeated iteratively, where:

1) A new camera pose was generated using transformations.
2) The scene was rendered from this pose.
3) Inpainting was applied to the occluded regions.

This iterative approach allowed the generation of multiple frames, which were later combined into a dynamic GIF showcasing the transitions between perspectives.

### II-G Key Highlights of the Methodology

- **Tools Used**: RAFT-Stereo for depth estimation, LaMa for inpainting.
- **Equations**: Depth estimation and 3D coordinate mapping were grounded in camera geometry.
- **Automation**: The pipeline was optimized for scalability, enabling the synthesis of multiple views efficiently.

## III. Results

The results from this project illustrate the outcomes of each step in the pipeline, from depth estimation to 3D reconstruction and inpainting. Below, we present key visualizations along with brief explanations.

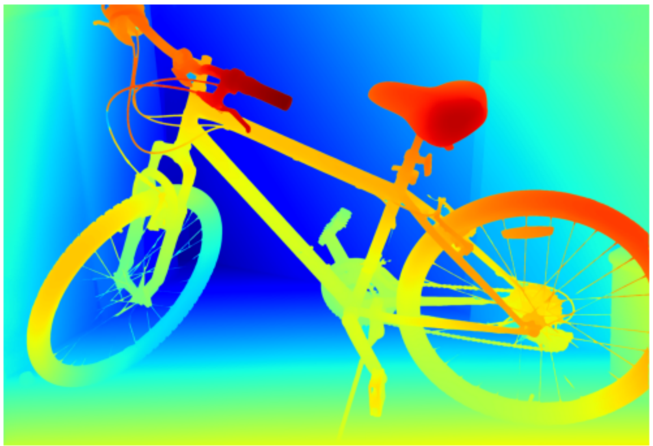

Fig. 1: Disparity map generated by RAFT-Stereo. Warmer colors indicate closer objects, while cooler colors represent greater depth. The bicycle's structure is clearly discernible.

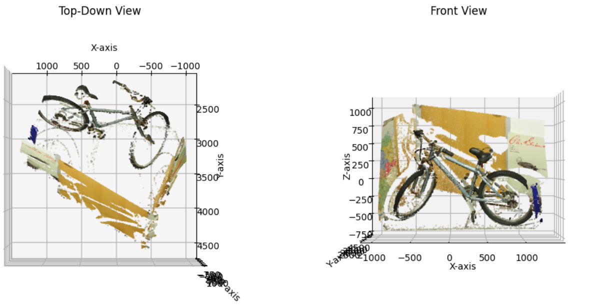

Fig. 2: 3D point cloud generated after running RAFT-Stereo. The spatial structure of the scene, including the bicycle and background, is clearly represented.

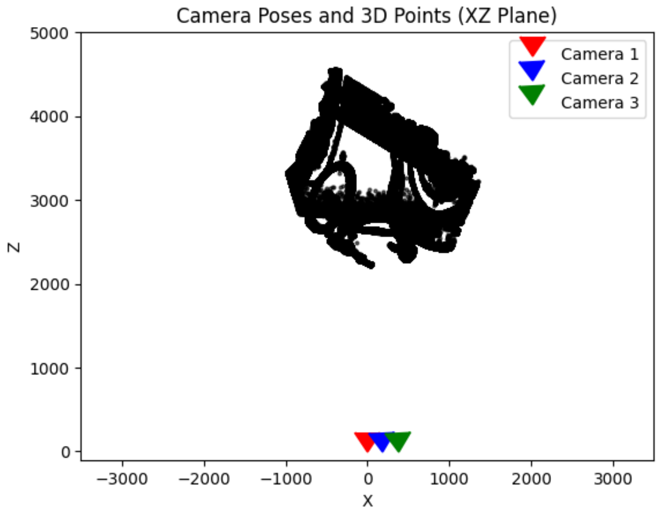

Fig. 3: Camera poses and 3D points visualized in the XZ plane. This plot illustrates the reconstructed point cloud alongside camera positions for incremental 3-degree changes in orientation.

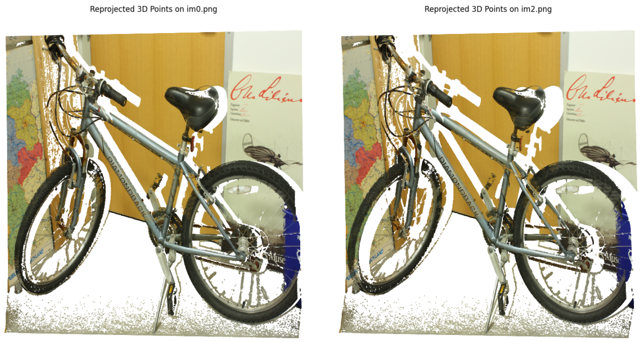
Reprojected 3D Points on im0.png
Reprojected 3D Points on im2.png

Fig. 4: Reprojected 3D points onto two camera views (im0 and im2). These visualizations confirm the accuracy of the point cloud alignment and reprojection.

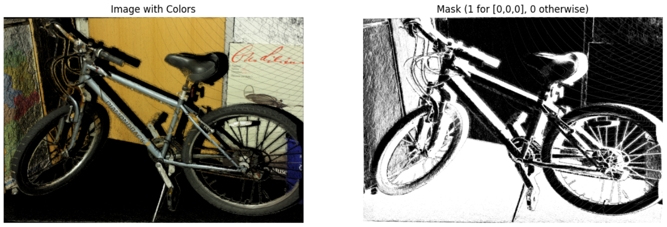

Fig. 5: Masking step before feeding the image to LaMa inpainting. The left image shows the original view, while the right shows the mask highlighting occluded regions for filling.

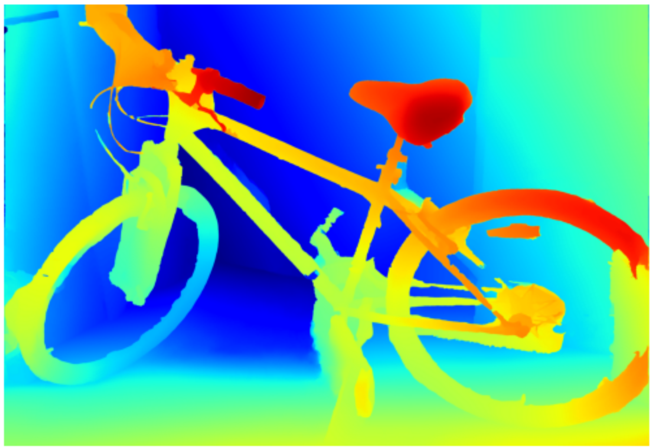

Fig. 6: Updated disparity map generated after transforming the camera view. The reconstructed depth for the new perspective remains consistent with the original geometry.

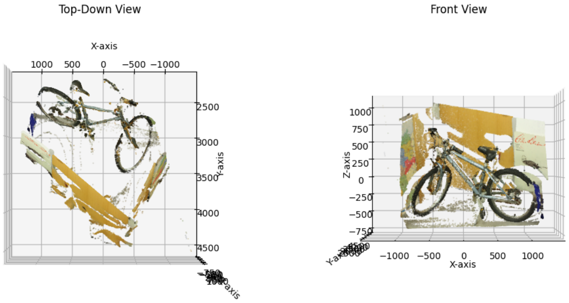

Fig. 7: 3D point cloud for the new output after camera pose transformations. The adjusted viewpoint accurately captures the spatial relationships within the scene.

## IV. Challenges and Observations

### 1. Depth Ordering and Object Separation

Depth ordering is critical to ensure that foreground objects do not mix with the background when creating new image perspectives. Errors in depth estimation and lack of 3d points in foreground object when seen from new perspective can cause these layers to blend, resulting in visual artifacts. Additionally, depth ordering helps address occlusion problems by accurately delineating which areas should be visible and which should remain hidden.

- **Impact:** Without proper depth ordering, foreground and background elements can overlap incorrectly, leading to unrealistic perspectives and unresolved occlusions.

- **Proposed Solution:** while inpainting the missing areas inpaint the fore ground objects first and then fill back ground objects

### 2. Occlusion Mask Generation

Filling occluded areas is highly dependent on generating accurate masks that identify regions requiring inpainting.

- **Impact:** Without precise masks, occluded areas may not be reconstructed correctly, resulting in inconsistencies in the scene.
- **Proposed Solution:** Use multi-view consistency checks or disparity comparisons to automatically generate occlusion masks. These masks can guide inpainting models like LaMa for better context-aware filling.

### 3. Artifacts in Inpainting

Despite using LaMa for occlusion filling, discrepancies occur where foreground objects merge with the background. This indicates a lack of contextual understanding during inpainting.

- **Impact:** The generated views may lose realism, particularly in scenes with complex depth variations or overlapping objects.
- **Proposed Solution:** Integrate depth maps into the inpainting process to enforce depth-aware constraints. Edge-aware masks can also help maintain object boundaries and sharpness.

### 4. Point Cloud Reconstruction and Stitching

Converting disparity maps into 3D point clouds and stitching them across views presents challenges due to alignment inaccuracies.

- **Impact:** Misaligned point clouds lead to errors in the global scene reconstruction, affecting occlusion handling and overall fidelity.
- **Proposed Solution:** Perform point cloud stitching using robust alignment methods like Iterative Closest Point (ICP) or feature-based matching. Refine the alignment further with bundle adjustment to ensure global consistency.

### 5. Bundle Adjustment for 3D Refinement

While RAFT-Stereo provides a dense depth map, integrating these into a cohesive 3D structure can be challenging without optimizing camera poses and reconstructed points.

- **Impact:** Inaccurate 3D points and camera poses can distort the final scene and compromise occlusion handling.
- **Proposed Solution:** Incorporate bundle adjustment to optimize 3D point positions and camera parameters, leveraging the RAFT-Stereo depth as priors to guide the adjustment.

## V. Conclusion

In this project, we successfully implemented a pipeline for multi-view generation from a single stereo image using advanced computer vision techniques. The integration of RAFT-Stereo for depth estimation, 3D reconstruction for spatial representation, and LaMa inpainting for handling occluded regions enabled us to generate realistic novel viewpoints.

The results demonstrate the effectiveness of RAFT-Stereo in producing high-quality disparity maps, which formed the foundation for accurate 3D point cloud reconstruction. Furthermore, the application of LaMa ensured that occluded areas in new views were seamlessly filled, enhancing the realism of the generated perspectives.

Despite these successes, certain challenges were encountered, such as handling reflective surfaces and low-texture regions in disparity estimation. Future work could explore the use of more advanced inpainting techniques and refinement of depth maps to address these limitations.

Overall, the project highlights the potential of combining state-of-the-art methods in computer vision for applications in virtual reality, augmented reality, and immersive content creation. The proposed pipeline provides a robust framework for synthesizing multi-view perspectives, contributing to advancements in visual representation and scene understanding.

## References

[1] @inproceedingslipson2021raft, title=RAFT-Stereo: Multilevel Recurrent Field Transforms for Stereo Matching, author=Lipson, Lahav and Teed, Zachary and Deng, Jia, booktitle=International Conference on 3D Vision (3DV), year=2021

[2] https://github.com/princeton-vl/RAFT-Stereo

[3] Suvorov, R., Logacheva, E., Mashikhin, A., Remizova, A., Ashukha, A., Silvestrov, A., Kong, N., Goka, H., Park, K., & Lempitsky, V. (2021). Resolution-robust Large Mask Inpainting with Fourier Convolutions. arXiv preprint arXiv:2109.07161

[4] https://github.com/saic-mdal/lama

[5] https://github.com/enesmsahin/simple-lama-inpainting

[6] https://doi.org/10.48550/arXiv.2312.04560

[7] https://doi.org/10.48550/arXiv.2404.07199