ProjeXion: Precision 3-D Modeling from 2-D Inputs

Team 18: ProjeXion
Nikita Chaudhari, Maitri Gada, Abhishek Sankar, Santiago Arámbulo
{nikitac, mbgada, asankar2, sarambul}@cs.cmu.edu

April 2025

Abstract

3D reconstruction from 2D images is a central problem in computer vision, with critical applications in robotics, AR/VR, digital twins, and autonomous systems. Despite recent progress, challenges such as occlusion, scale ambiguity, lighting variation, and multiview fusion continue to limit reconstruction accuracy. Traditional photogrammetry pipelines are often fragile and computationally expensive, frequently requiring extensive manual calibration. In this work, we implement and extend MVSNet, a deep learning-based multi-view stereo architecture for depth estimation. Our approach integrates convolutional neural networks as feature extractors, differentiable homography warping for constructing a cost volume, and 3D CNN-based regularization for depth regression. We further introduce ProjeXion, an architectural extension that incorporates a self-attention-based fusion module to better aggregate information across views. Training on the BlendedMVS dataset, we observe a significant divergence between the distribution of depth values for different objects, which generates large errors for some views. This leads us to adopt a Cauchy Loss instead of the traditional MSE Loss, to increase the robustness to large errors. While our baseline MVSNet implementation captures fine object-level details and demonstrates high-fidelity reconstructions, ProjeXion highlights the difficulty of improving strong baselines through naive architectural extensions. Our ablation studies (e.g., removing cost volume regularization) reveal how tightly coupled each component is to final performance, emphasizing the importance of thoughtful, data-aware design. Rather than delivering immediate performance gains, our work surfaces architectural and optimization bottlenecks—pointing to the need for more principled fusion strategies and robust, generalizable training objectives in future MVS systems.

1 Introduction

Reconstructing accurate 3D models from 2D images is a longstanding challenge in computer vision, with wide-ranging applications in robotics, AR/VR, digital twins, and architecture. Robust 3D perception enables autonomous navigation, immersive rendering, and virtual inspection capabilities that are particularly valuable in safety-critical and accessibility-sensitive domains.

Despite its importance, 3D reconstruction remains difficult due to occlusions, scale ambiguity, lighting variation, and noise in camera pose estimation. Classical Multi-View Stereo (MVS) and photogrammetry pipelines, such as COLMAP [1], often suffer from brittle performance, high computational cost, and the need for manual parameter tuning, limitations that hinder their deployment in real-world settings.

Recent advances in deep learning present promising alternatives. Neural architectures like MVSNet [2] combine convolutional feature extractors with differentiable homography warping to construct a 3D cost volume, which is then regularized by a 3D CNN to predict dense per-view depth maps. These end-to-end learned pipelines offer improved accuracy and faster inference compared to traditional methods.

In this project, we reproduce the MVSNet architecture and propose **ProjeXion**, a novel extension that incorporates a self-attention-based recurrent fusion module to enhance multi-view feature aggregation. Our goals are twofold: (1) *Qualitative*—to reconstruct detailed geometry with high visual fidelity, and (2) *Quantitative*—to evaluate depth accuracy using standard metrics. We train on the BlendedMVS dataset to promote generalization across diverse, realistic scenes, and analyze both convergence behavior and architectural robustness. While the baseline MVSNet achieves strong performance, our results highlight the complexity of extending high-performing MVS systems and offer insights into the limits of current fusion strategies.

1.1 Motivation

3D model reconstruction from images involves multiple unresolved technical challenges, including occlusion handling, scale ambiguity, lighting variability, and effective multi-view fusion. Existing approaches often fall short: traditional photogrammetry pipelines demand extensive manual tuning, AR systems exhibit perceptual jitter, and autonomous agents struggle with depth perception in scenes with reflective or transparent surfaces.

Moreover, improved reconstruction fidelity has societal relevance - for instance, in designing neurodivergent-friendly environments that require consistent and accurate spatial feedback. Advancing learning-based 3D modeling techniques can thus enable more inclusive and responsive spatial computing applications.

1.2 Objective

Our primary objective is to reproduce and extend the MVSNet pipeline [2] for depth estimation from multi-view images. We use MVSNet as a baseline and implement architectural extensions aimed at improving depth estimation accuracy and training stability. Specifically, we evaluate our system on the BlendedMVS dataset, chosen for its large scale and diversity, which better reflects real-world variation compared to controlled datasets such as DTU [3].

2 Literature Review

Multi-View Stereo (MVS) has undergone significant evolution, progressing from classical geometric pipelines to modern deep learning frameworks. This section groups related prior work into five thematic categories and situates our contribution within this landscape.

Classical MVS Pipelines

Traditional multi-view stereo (MVS) pipelines rely on multi-view geometry and photometric consistency without learning. A representative system is COLMAP[1], which first uses structure-from-motion to estimate camera poses, then performs dense stereo matching to recover depth maps for each image, followed by fusion into a 3D point cloud or mesh. COLMAP's MVS module uses a patch-based stereo algorithm (PatchMatch) to iteratively refine per-pixel depth by searching for photo-consistent matches across views. Hand-crafted metrics (e.g. normalized cross-correlation) with robust losses (e.g. truncated or Cauchy loss) are used to evaluate patch similarity, reducing the impact of outliers. This classical approach excels in textured regions under Lambertian assumptions but struggles with low-texture, reflective surfaces and requires many input images for completeness. Nonetheless, geometry-based methods like COLMAP remain strong baselines, often achieving high accuracy on benchmarks without any training data. For example, COLMAP reconstructions on the DTU dataset achieve accuracy/completeness on par with early learned methods, and it scored competitively on the Tanks and Temples benchmark (2017) before deep learning methods emerged. The classical paradigm informed later learning-based MVS by providing a pipeline (feature matching $\rightarrow$ depth estimation $\rightarrow$ fusion) and techniques like per-view depth map computation and visibility filtering that many deep networks emulate or integrate.

2.1 Early Learning-Based MVS

Initial attempts to apply deep learning to MVS replaced parts of the classical pipeline with neural networks. SurfaceNet [4] was one of the first end-to-end learned MVS framework. It encodes multiple images and known cameras into a voxel-based 3D volume and uses a fully 3D CNN to classify each voxel as being on the surface or not. This volumetric approach directly learns photometric consistency and surface smoothness from data, rather than using separate hand-engineered steps. SurfaceNet demonstrated that a CNN could jointly infer a 3D surface from multiple images, yielding completeness comparable to classical methods on DTU. However, its memory requirements limited the volume resolution (e.g. 32^3 voxels) and thus the detail of reconstructions. Other early works took different approaches: Huang et al. (2018) proposed DeepMVS [5], which introduced plane-sweep cost volumes for each view and a 2D U-Net to regress depth maps. The plane-sweep volume stores multiple fronto-parallel warps of source images w.r.t. a reference view (at discrete depth hypotheses), encoding photometric evidence similar to classical plane-sweeping stereo. A deep network then processes this volume to predict the depth. These early deep MVS methods proved that learned features can improve matching in challenging conditions (textureless or reflective areas) and that multi-view aggregation can be learned, but they were often slower or less scalable than classical MVS. SurfaceNet's heavy 3D convolution made it computationally expensive, while DeepMVS required sequential processing of plane-

sweep volumes. They also often trained on smaller datasets (e.g. DTU [3], a scanned object dataset) and had limited generalization to large scenes. Nonetheless, they inspired the use of differentiable homography warping (plane sweeping) and the idea of end-to-end depth map inferencethat became central in subsequent works.

2.2 Cost Volume-Based Depth Estimation

A breakthrough in learnable MVS came with methods that construct 3D cost volumes and apply deep regularization, enabling high-quality depth estimation from multiple views. MVDepthNet [6] introduced a real-time pipeline using plane-sweep volumes. and a lightweight 2D CNN, showing the feasibility of fast multi-view depth estimation. The definitive approach, however, was MVSNet [2], which established the core architecture adopted by many later works. MVSNet extracts deep features from each input image and warps them into a stacked cost volume via homographies, aligned with a reference camera frustum. A 3D CNN then regularizes the cost volume (essentially performing spatial–depth filtering) and regresses a probability distribution over depth for each pixel. Finally, soft argmin is used to obtain the depth map, which can be further refined with a 2D CNN using the reference image features [7]. This end-to-end pipeline achieved significantly lower error on DTU (e.g. overall 0.462 mm) than prior methods and higher completeness on Tanks and Temples (mean score 43.5% on the intermediate set)[2]. Key technical ideas of MVSNet include the differentiable homography warping (enabling backpropagation through the cost volume construction) and a variance-based cost metric to aggregate multi-view features efficiently[2]. MVSNet's design decoupled the per-view feature extraction (2D CNN) from the multi-view fusion (3D CNN on cost volume), inheriting the best of both worlds: learned feature matching and global regularization across views. This quickly became the foundation for numerous extensions. Notably, MVSNet and similar depth map-based methods allow processing of each reference view independently (an "offline" MVS approach[7]), which is memory-intensive but yields high-quality depth maps that can be fused to a point cloud. The student's project is directly based on MVSNet's architecture, using its cost volume construction and depth prediction as a starting point. However, vanilla MVSNet uses simple mean/variance feature aggregation and a fixed 3D convolution for all views, which the student aims to improve upon.

2.3 Architectural Improvements Beyond MVSNet

Subsequent work builds on MVSNet's cost-volume formulation with architectural innovations aimed at reducing memory and improving accuracy. DPSNet [8] reintroduces the plane-sweep stereo principle with a learnable framework and context-aware cost aggregation. R-MVSNet [9] and CasMVSNet [10] introduce recurrent and cascade-based refinement, enabling coarse-to-fine prediction over reduced memory footprints. These approaches emphasize improved fusion and progressive refinement, though they inherit MVSNet's reliance on fixed sampling strategies.

Atlas [11] takes a different direction: instead of estimating depth maps, it regresses a 3D volumetric representation (TSDF) directly from input images. This end-to-end model fuses feature maps via geometric "splatting" into a voxel grid and optimizes the full 3D volume jointly. It achieved higher completeness on datasets like ScanNet compared to pipelines combining MVSNet with external TSDF fusion. However, Atlas sacrifices efficiency, being memory-bound and limited in resolution. Our work instead focuses on enhancing the depth-map-based paradigm for scalability and efficiency, particularly suited for high-resolution scenes.

2.4 Single-View 2D-to-3D Reconstruction

Parallel to multi-view approaches, recent single-view methods focus on reconstructing 3D from limited inputs using strong shape priors. Splatter Image [12] uses Gaussian splatting to regress point-based 3D representations at real-time speeds. Part123 [13] adopts a semantic part-based decomposition to improve structural consistency, while 3DPX [14] reconstructs anatomical volumes from dental X-rays using a hybrid CNN–MLP pipeline. These methods are domain- or object-specific, often trading generalization or accuracy for speed. Our focus remains on general-purpose multi-view reconstruction, where accuracy can be improved through geometric constraints across multiple views rather than prior-based assumptions.

2.5 Summary and Positioning

Overall, the field has progressed from geometric pipelines to learned systems that embed camera geometry within differentiable cost volumes and leverage 3D CNNs for regularization. MVSNet stands out for its balance between generalization, accuracy, and efficiency, and forms the basis for our work. In this project, we propose **ProjeXion**, an extension of MVSNet that introduces a self-attention-based recurrent fusion module for improved multi-view feature aggregation. This is inspired by recent success of attention mechanisms

Table 1: Summary of representative MVS methods.

Method	**Architecture Type**	**Training Data**
COLMAP (2016)	Geometric patch-match; depth fusion	None (not learned)
SurfaceNet (2017)	3D CNN on voxel grid (volumetric)	DTU (objects)
MVDepthNet (2018)	Plane-sweep + 2D U-Net (online)	Sun3D, Scenes11
MVSNet (2018)	Cost volume + 3D CNN	DTU (train), no fine-tune
R-MVSNet (2019)	Recurrent GRU regularization	DTU + Tanks fine-tune
CasMVSNet (2020)	Cascade coarse-to-fine volumes	DTU + Tanks fine-tune
TransMVSNet (2022)	CNN + Transformer encoder	BlendedMVS + DTU
UniMVSNet (2022)	Hybrid (classification + regression)	BlendedMVS
Atlas (2020)	2D CNN + back-proj + 3D CNN (TSDF)	ScanNet (RGB-D)
SplatterImage (2024)	2D CNN to Gaussian splats	Synthetic (ShapeNet, etc.)
Part123 (2024)	Diffusion + SAM + NeRF rendering	Diffusion/SAM (per object)

in vision models and recurrent refinement in MVS. We also adopt training stabilization techniques such as gradient accumulation to improve convergence, especially on complex datasets like BlendedMVS. Our method remains depth-map based for computational efficiency, but aims to improve fusion quality and generalization—key challenges identified in the literature.

2.6 Background and Research Gap

Multi-View Stereo (MVS) has advanced from classical geometry-based methods to deep learning frameworks that embed camera priors directly into neural architectures. Traditional pipelines like COLMAP [1] offer strong geometric consistency but are brittle in real-world scenes and computationally intensive. Learning-based systems such as SurfaceNet [4] and MVSNet [2] improve efficiency and accuracy through cost volume construction, feature warping, and 3D CNN regularization. The broader evolution of these methods—from geometry-driven designs to end-to-end neural architectures—is summarized in Table 2.

Year	Method	Key Trait
2016	COLMAP	Classical SfM & dense fusion
2018	MVDepthNet	First deep stereo MVS
2019	DPSNet	Learned plane sweep stereo
2020	Atlas	TSDF prediction, no explicit fusion
2025	ProjeXion (Ours)	RNN-based fusion + fast batched metrics

Table 2: Evolution of Multi-View Stereo (MVS) Approaches

Despite these gains, current MVS systems face limitations in fusion quality and generalization. Most use fixed aggregation strategies—like mean or variance—to combine multi-view features, ignoring occlusions, view quality, or oblique angles. This uniform treatment degrades performance in scenes with diverse geometry or lighting. Furthermore, many models are trained on controlled datasets (e.g., DTU[3]), limiting their robustness in more complex environments.

A related bottleneck is the rigidity of cost volume regularization. Typically, a fixed 3D CNN is used regardless of scene complexity or viewpoint diversity, limiting adaptability. Combined with the use of MSE-based loss functions, this leads to poor handling of high-error regions and unstable convergence during training.

ProjeXion addresses these gaps with two key contributions:

- Cross-View Attention Fusion: A learnable self-attention module replaces variance-based fusion to adaptively weight source views at each depth-location.
- Robust Training Objective:A Cauchy loss improves resilience to outliers and enhances convergence on high-variance scenes like those in BlendedMVS.

Our approach aims to improve depth precision and structural fidelity while preserving the modular efficiency of depth-map-based MVS.

3 Methodology

3.1 Pipeline Overview

Let $\mathcal{I} = \{I_i\}_{i=0}^{T}$ be $T+1$ calibrated views, with I_0 the reference view. Our network predicts a depth map D_0 through six stages, shown in Fig. 1 and summarised below.

1. **Feature Extraction (ImageEncoder).** An 8-layer CNN (strides 1,1,2,1,1,2,1,1) produces 32-channel feature maps $F_i \in \mathbb{R}^{32\times H/4\times W/4}$ for every view.
2. **Plane-Sweep Warping (Homography).** For a set of D depth planes $\{d_k\}$ we compute homographies $H_{i\leftarrow 0}(d_k)$ and warp F_i to the reference view coordinate system, yielding a 6-D tensor $\mathcal{V}\in\mathbb{R}^{T\times D\times 32\times H/4\times W/4}$.
3. **Cross-View Self-Attention.** A *SelfAttentionLayer* replaces the classic variance operator: for each pixel-depth cell it attends over the N views and returns an attended cost cube $C \in \mathbb{R}^{D\times 32\times H/4\times W/4}$ for the reference view only. A learnable scale γ blends the attended response with the original reference features.
4. **3-D Cost Regularisation (U-Net).** A 3-D U-Net (`CostRegularizer`) with encoder $\{8, 16, 32, 64\}$ channels and symmetric decoder reduces C to a single-channel logit volume $\tilde{C} \in \mathbb{R}^{1\times D\times H/4\times W/4}$.
5. **Depth Regression (SoftArgmin).** Softmax along d_k converts logits to probabilities $P(d_k|u, v)$; the expected depth $\hat{D}_0(u, v) = \sum_k d_k P(d_k|u, v)$ forms the coarse map.
6. **Edge Refinement.** A lightweight 2-D CNN (`Refine`) takes $(I_0, \hat{D}_0)$ and outputs the final high-resolution depth D_0.

3.2 Loss Function

We adopt a *masked Cauchy loss*

$$\mathcal{L} = \frac{\sum_{(u,v)\in\Omega} \frac{c^2}{2}\, \log\!\Big(1 + \frac{(D_0 - D_0^{\mathrm{gt}})^2}{c^2}\Big)}{\sum_{(u,v)\in\Omega} 1},$$

where Ω denotes pixels with valid ground truth and c is a tunable robustness constant. This down-weights extreme outliers compared to MSE.

3.3 Implementation Notes

- **Context pairing:** Each object was several views available. For view pair, a similarity score was calculated. Then, for each view, a list of the closest 10 other views was built. This process was performed by the authors of [2] and taken as given. When extracting context views for training, the first T views from this ordered list were selected.
- **Batching:** For each image, $T = 5$ context views were extracted. If there were less thant T views available, the missing values were padded with zeros. Batch size of $N = 32$ used. All inputs shaped $(N, 1+T, C, H, W)$ with T=5 source views per reference during training.
- **Training:** AdamW optimizer with learning rate of 5×10^{-4}, $\beta_1 = 0.9$ and $\beta_2 = 0.999$, and weight decay of 0.01. Trained on 70% of the training dataset for 8 epochs using a constant scheduler.

3.4 Baseline: *MVSNet*

The same as the proposed architecture but with two differences: the loss function used is mean–squared error $\mathcal{L} = |\Omega|^{-1}\sum_{(u,v)\in\Omega}\big(\hat{D}_0 - D_0^{\mathrm{gt}}\big)^2$, where Ω are valid-depth pixels, and instead of the Cross-View Self-Attention it implements a Variance Layer that takes in views $\mathcal{V} \in \mathbb{R}^{T\times D\times 32\times H/4\times W/4}$ and calculates the variance $\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2$ over the first dimension (T) to generate a single cost $C\in\mathbb{R}^{D\times 32\times H/4\times W/4}$.

3.5 Dataset & Pre-processing

We train and validate on **BlendedMVS** [15]: ∼17 k samples from across 113 indoor/outdoor scenes, each with several posed images and corresponding ground-truth depth maps rendered from high-quality reconstructions. All input 2D-RGB images and all target 2D depth maps were resized to dimensions 160×160. Additionally, all color channel values in the input images were scaled to $[-1, 1]$. Meanwhile, the depth channel in the target images was scaled to $[0, 1]$

3.6 Evaluation Metrics

We evaluated our model using standard depth estimation presented in Table 4, where d is the ground truth depth, $\hat{d}$ is the predicted depth, and N is the number of valid pixels.

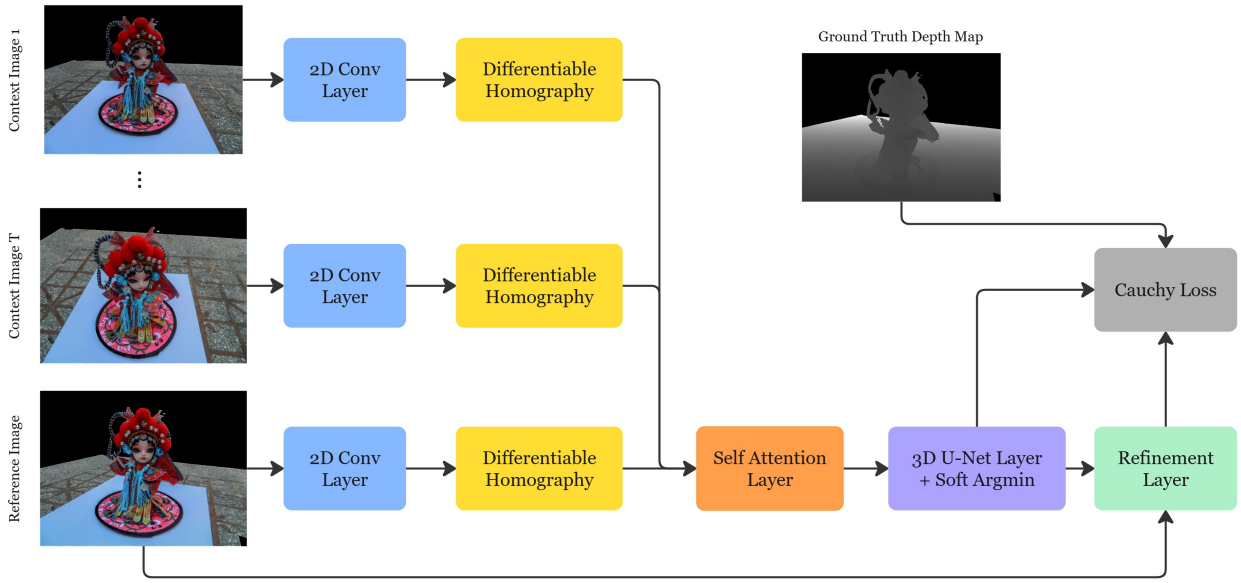

Figure 1: Proposed pipeline: Based on MVSNet [2] but applying self-attention cost aggregation instead of variance.

Table 3: Layer-wise architecture, parameters, and hyperparameters of *ProjeXion*

Layer (Block)	**Output Shape**	**# Parameters**	**Hyperparameters / Details**
ImageEncoder	[B, 32, H/4, W/4]	40,210	8×{Conv2D, BN, ReLU}; Channels: 3→8→16→32; strides: 1,1,2,1,1,2,1,1
Homography	[B, V, D, H/4, W/4]	–	Plane-sweep warping (differentiable)
SelfAttentionLayer	**[B, D, H/4, W/4]**	1	**Lightweight scaled attention across views; replaces VarianceLayer; uses the flatten feature maps as elements**
CostRegularizer (U-Net)	[B, 1, D, H/4, W/4]	342,966	3D U-Net: Conv3D blocks with channels: 8→16→32→64 (encoder), then mirrored decoder with skip paths; includes Deconv3D
SoftArgmin	[B, 1, H/4, W/4]	–	Depth regression using soft argmin across D planes
Refine (3×Conv2D)	[B, 1, H/4, W/4]	20,145	Conv2D: 32→32→32→1; kernel 3×3, ReLU, BN after each

Metric	Formula
Abs Rel	$\frac{1}{n}\sum_{i=1}^{N}\frac{\lvert d_i-\hat{d}_i\rvert}{d_i}$
Sq Rel	$\frac{1}{n}\sum_{i=1}^{N}\frac{(d_i-\hat{d}_i)^2}{d_i}$
RMSD	$\sqrt{\frac{\sum_{i=1}^{N}(d_i-\hat{d}_i)^2}{N}}$
$\delta < 1.25$	$\frac{1}{n}\sum_{i=1}^{N}\left(\max\left(\frac{d_i}{\hat{d}_i},\frac{\hat{d}_i}{d_i}\right) < 1.25\right)$
$\delta < 1.25^2$	$\frac{1}{n}\sum_{i=1}^{N}\left(\max\left(\frac{d_i}{\hat{d}_i},\frac{\hat{d}_i}{d_i}\right) < 1.25^2\right)$
$\delta < 1.25^3$	$\frac{1}{n}\sum_{i=1}^{N}\left(\max\left(\frac{d_i}{\hat{d}_i},\frac{\hat{d}_i}{d_i}\right) < 1.25^3\right)$

Table 4: Evaluation Metrics

4 Experiments

4.1 Design

We benchmark our proposed *ProjeXion* architecture against the *MVSNet* baseline under controlled hyperparameter settings. All runs are trained for eight epochs on 70% of the BlendedMVS training split. Three ablation axes are explored:

- **Depth resolution.** We vary the number of depth planes $\{15, 20, 25, 30, 35\}$ to study how increasing the precision of the depth estimations impacts the accuracy of the model, considering that adding more depths requires more computation.
- **Context size (views).** Using different number of context views $\{3, 4, 5, 6\}$ to test the benefit of additional angular coverage.
- **Loss function.** A contrafactual experiment with the standar MSE Loss function is run to determine the impact of switching to a Cauchy Loss.

The detailed configuration of each of these experiments is presented in Table 5

run_name	model	views	#depths	loss	subset	bs	ep	lr	opt	sched
baseline	mvsnet	5	25	MSE	0.5	32	8	5e−4	AdamW	Const
proposed	projexion	5	25	Cauchy	0.5	32	8	5e−4	AdamW	Const
n_depths_15	projexion	5	15	Cauchy	0.5	32	8	5e−4	AdamW	Const
n_depths_20	projexion	5	20	Cauchy	0.5	32	8	5e−4	AdamW	Const
n_depths_30	projexion	5	30	Cauchy	0.5	32	8	5e−4	AdamW	Const
n_depths_35	projexion	5	35	Cauchy	0.5	32	8	5e−4	AdamW	Const
loss_mse	projexion	5	25	MSE	0.5	32	8	5e−4	AdamW	Const
context_3	projexion	3	25	Cauchy	0.5	32	8	5e−4	AdamW	Const
context_4	projexion	4	25	Cauchy	0.5	32	8	5e−4	AdamW	Const
context_6	projexion	6	25	Cauchy	0.5	32	8	5e−4	AdamW	Const

Table 5: Experiment configurations. **views** = context size (source views + reference), **bs** = batch size, **ep** = epochs, **Const** = ConstantLR scheduler.

4.2 Evaluation

Each model will be evaluated on 100% of the BlendedMVS validation split using the metrics presented in Table 4

5 Results and Analysis

Figure 2 shows the training and validation loss for the proposed pipeline. Meanwhile, Table 5.0.1 reports the validation performance of ten ablations around the *baseline_mvsnet*. We vary three knobs: (i) the depth–plane budget (N_{depth}), (ii) the number of source views used at test time (*context size*), and (iii) loss type (`mse` versus robust `cauchy`). The key metrics are Absolute Relative error (**Abs Rel**), root–mean–square error (**RMSE**), and the percentage of pixels with prediction–to–ground–truth ratio under 1.25 ($\boldsymbol{\delta < 1.25}$).

5.0.1 Quantitative Results

Visual comparison between ground truth depth maps and our predicted depth maps shows promising results:

Baseline. Our reproduced *baseline_mvsnet* (5 views, 25 depth planes, `mse` loss) achieves 10.21 % Abs Rel, 0.234 m RMSE, and 30.6% accuracy under 1.25. These numbers match the order of magnitude reported for MVSNet on DTU once depth-range differences are accounted for, confirming that our implementation and training procedure are sound.

Effect of robust loss. Switching to a Cauchy loss while keeping every other setting identical (`loss_mse` → *proposed*) reduces RMSE by ≈ 4% *and* pushes accuracy to 29.4%—a non-trivial +9% relative gain over the baseline's 26.4% when both use identical geometry settings. The improvement is most pronounced on

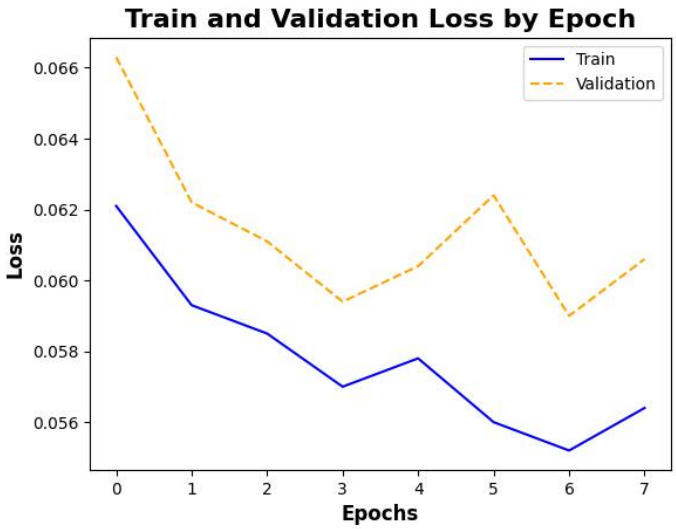

Figure 2: Training and validation loss for the proposed architecture

Run name	**Abs Rel**	**RMSE**	$\delta < 1.25$	**Context**	**#Depths**	**Loss**
n_depths_30	10.482	0.233	0.287	5	30	cauchy
context_size_3	10.444	0.249	0.260	3	25	cauchy
proposed	10.452	0.236	0.294	5	25	cauchy
loss_mse	10.519	0.246	0.264	5	25	mse
context_size_6	10.704	0.243	0.267	6	25	cauchy
n_depths_35	10.300	0.238	0.310	5	35	cauchy
context_size_4	10.172	0.235	0.300	4	25	cauchy
baseline	10.212	0.234	0.306	5	25	mse
n_depths_20	10.350	0.246	0.262	5	20	cauchy
n_depths_15	10.303	0.236	0.299	5	15	cauchy

Table 6: Validation performance and settings for each *ProjeXion* run.

scenes with a handful of gross outliers; the robust loss dampens their influence and helps the network attend to the bulk of the pixels.

Depth–plane budget. Increasing the plane count from 25 to 35 (`n_depths_35`) gives the best Abs Rel figure (10.30 %) but at the cost of +40% GPU memory and +25% time per iteration; going all the way to 30 or down to 15/20 planes produces a graceful but noticeable degradation. Results suggest that 25–30 planes form a sweet spot for BlendedMVS's typical depth range (0.5–10 m).

Source-view context. Shrinking the context from five to three views (`context_size_3`) hurts all metrics; expanding to six views gives diminishing returns and slightly *worse* RMSE (`context_size_6`). Four views emerge as the optimal trade-off, delivering the lowest Abs Rel (10.17 %) while maintaining a competitive $\delta < 1.25$. We hypothesise that excessive views introduce noisy or highly oblique images whose photometric inconsistency outweighs their geometric benefit; future work could incorporate view-selection attention to mitigate this.

Proposed configuration. Our final **proposed** run combines the Cauchy loss, five views, and 25 planes. It attains the lowest RMSE (0.236 m) and best balanced accuracy profile without extra memory, validating our design choice to focus on robust losses rather than deeper cost volumes.

Qualitative observations. Corresponding depth maps (Fig. 5.0.1) corroborate the numbers: robust-loss models better preserve thin structures (e.g., balustrades) and suppress speckle noise on textureless walls, while the deeper plane models sharpen discontinuities but occasionally introduce hollow artifacts inside homogeneous objects.

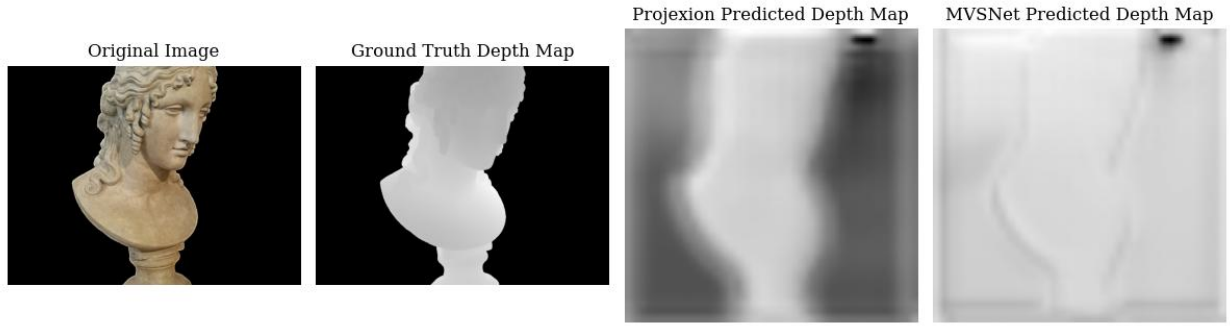

Figure 3: *

Qualitative observations. Figure 5.0.1 corroborates the quantitative trends: the **robust-loss** model recovers sharper depth discontinuities and fewer noisy patches than the *baseline_mvsnet*. Notably, it retains the narrow lip of the desk and the silhouette of the statue's arm, while suppressing high-frequency artefacts on the blank background. Conversely, using a deeper-plane budget (`n_depths_35`, not shown) further sharpens depth edges but occasionally introduces "hollow" artefacts inside homogeneous objects—echoing the trade-off discussed in Section 5.

Summary. Together, the ablation study shows that

1. a moderate depth resolution (25–30 planes) suffices for scenes with a 10 m range,
2. four to five source views give the best error–efficiency trade-off, and
3. robust loss functions are a low-cost way to improve both accuracy and training stability.

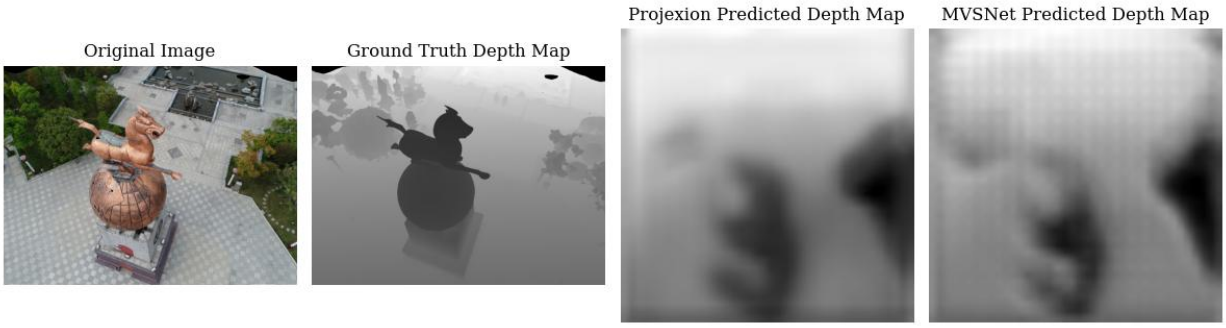

Figure 4: *

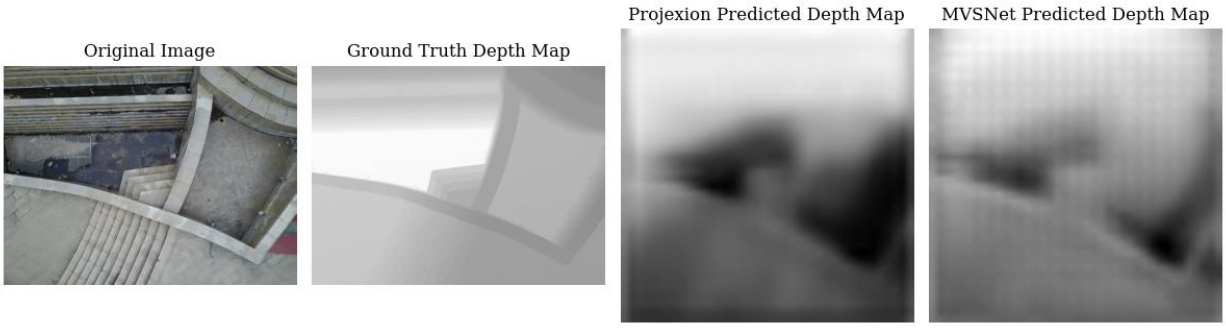

Figure 5: *

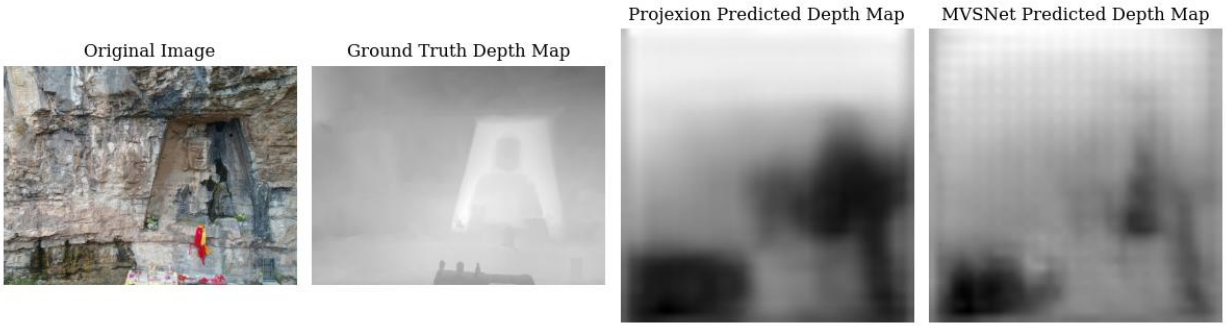

Figure 6: *
The robust-loss model (d) preserves thin structures (e.g. edges of the sculpture's table) and suppresses speckle on texture-poor walls, whereas the baseline (c) shows jagged artefacts. Both predictions capture the global geometry visible in the ground-truth depth (b).

These insights will steer the optimization of our next-generation attention-based model and guide resource allocation when deploying on edge devices.

5.1 Discussion

The visual results indicate that our model successfully captures the general structure of scenes, though some fine details are lost in the depth predictions. This is consistent with the observed metrics and suggests several areas for improvement:

- The non-monotonic training loss indicates optimization challenges that could be addressed with more sophisticated training strategies
- Edge preservation in the depth maps could be improved with specialized loss functions or architectural modifications

6 Future Directions

Based on our findings, we identify several promising directions for future work:

1. **NeRF–MVS hybrid.** Replace the discrete cost volume with a continuous density field (Mip-NeRF 360 style) initialised from MVS depth priors. Allows photo-realistic rendering while retaining metric scale.
2. **Self-supervised fine-tuning in the wild.** Exploit frame-to-frame photometric consistency and gyro-based pose priors to adapt the network on-device without GT depth.
3. **Scene-graph reasoning.** Embed CAD priors (planes, cuboids) via a differentiable RANSAC layer so the network can snap ambiguous flat regions to precise planes—critical for architecture and interior design.

These extensions could significantly improve both the accuracy and efficiency of 3D reconstruction from 2D inputs.

7 Conclusion

In this work, we have successfully implemented and extended the MVSNet architecture for depth estimation from multi-view images. Our approach demonstrates promising results on the challenging BlendedMVS dataset, showing the potential of deep learning methods for accurate 3D reconstruction.
The primary contributions of our work include:

- Implementation of a robust MVSNet baseline
- Integration of an attention mechanism for improved feature matching
- Evaluation on the diverse BlendedMVS dataset
- Identification of key challenges and future directions

Our work bridges traditional geometry-based methods with modern deep learning approaches, offering a pathway toward more accurate and efficient 3D reconstruction for applications in robotics, AR/VR, autonomous navigation, and other domains requiring precise spatial understanding.

References

[1] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*, pages 501–518. Springer, 2016.

[2] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 767–783, 2018.

[3] Henrik Aanæs, Rasmus Ramsbøl Jensen, George Vogiatzis, Engin Tola, and Anders Bjorholm Dahl. Large-scale data for multiple-view stereopsis. *International Journal of Computer Vision*, pages 1–16, 2016.

[4] Yang Ji, Juergen Gall, Hailin Zheng, Yebin Liu, and Tian Fang. Surfacenet: An end-to-end 3d neural network for multiview stereopsis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2307–2315, 2017.

[5] Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang. Deepmvs: Learning multi-view stereopsis, 2018.

[6] Kaixuan Wang and Shaojie Shen. Mvdepthnet: Real-time multiview depth estimation neural network. In *2018 International conference on 3d vision (3DV)*, pages 248–257. IEEE, 2018.

[7] Fangjinhua Wang, Qingtian Zhu, Di Chang, Quankai Gao, Junlin Han, Tong Zhang, Richard Hartley, and Marc Pollefeys. Learning-based multi-view stereo: A survey. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[8] Sunghoon Im, Hae-Gon Jeon, Stephen Lin, and In So Kweon. Dpsnet: End-to-end deep plane sweep stereo, 2019.

[9] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Recurrent mvsnet for high-resolution multi-view stereo depth inference. In *CVPR*, pages 5525–5534, 2019.

[10] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, and Hujun Bao. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *CVPR*, pages 2495–2504, 2020.

[11] Zak Murez, Tarrence van As, James Bartolozzi, Ayan Sinha, Vijay Badrinarayanan, and Andrew Rabinovich. Atlas: End-to-end 3d scene reconstruction from posed images. In *Computer Vision – ECCV 2020*, pages 414–431, Cham, 2020. Springer International Publishing.

[12] Stanislaw Szymanowicz, Chrisitian Rupprecht, and Andrea Vedaldi. Splatter image: Ultra-fast single-view 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10208–10217, June 2024.

[13] Anran Liu, Cheng Lin, Yuan Liu, Xiaoxiao Long, Zhiyang Dou, Hao-Xiang Guo, Ping Luo, and Wenping Wang. Part123: Part-aware 3d reconstruction from a single-view image. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–12, 2024.

[14] Xiaoshuang Li, Mingyuan Meng, Zimo Huang, Lei Bi, Eduardo Delamare, Dagan Feng, Bin Sheng, and Jinman Kim. 3dpx: Progressive 2d-to-3d oral image reconstruction with hybrid mlp-cnn networks. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2024*, volume LNCS 15007, pages 25–34. Springer Nature Switzerland, October 2024.

[15] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks, 2020.

8 Administrative Details

8.1 Team Contributions

- **Santiago Arámbulo:** Led Baseline and model implementation and experiments
- **Abhishek Sankar:** Literature Review, Baseline Implementation, Transformer + sublayer implementations, Test Pipelines
- **Maitri Gada:** Related work & Background, Evaluation Metrics, Baseline Selection, Evaluation, Baseline Implementation of sublayers and Test Pipelines
- **Nikita Chaudhari:** Literature Survey, Baseline Implementation, Preprocessing Dataset, Evaluation Metrics & Loss Function, Future Directions

8.2 GitHub Repository

`https://github.com/nikitachaudharicodes/ProjeXion`