Novel view synthesis from sparse inputs

Qian Li

▶ **To cite this version:**

Qian Li. Novel view synthesis from sparse inputs. Computer Vision and Pattern Recognition [cs.CV]. Université de Rennes, 2023. English. NNT : 2023URENS036 . tel-04280320

HAL Id: tel-04280320

https://theses.hal.science/tel-04280320v1

Submitted on 10 Nov 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

COLLEGE MATHS, TELECOMS
DOCTORAL INFORMATIQUE, SIGNAL
BRETAGNE SYSTEMES, ELECTRONIQUE

Université de Rennes

THÈSE DE DOCTORAT DE

L'UNIVERSITÉ DE RENNES

ÉCOLE DOCTORALE N° 601
Mathématiques, Télécommunications, Informatique, Signal, Systèmes, Électronique
Spécialité : *Informatique*

Par
Qian Li

Novel View Synthesis from Sparse Inputs

Thèse présentée et soutenue à INRIA Rennes - Bretagne Atlantique, le 17 Octobre 2023
Unité de recherche : INRIA

Rapporteurs avant soutenance :

Hubert SHUM Associate Professor Durham University
Céline LOSCOS Principal Research Engineer, Huawei

Composition du Jury :

Président :	Alexandre Krupa	Directeur de Recherche Inria
Examinateurs :	Alexandre Krupa	Directeur de Recherche Inria
	Hubert SHUM	Associate Professor Durham University
	Céline LOSCOS	Principal Research Engineer, Huawei
Dir. de thèse :	Franck MULTON	Directeur de Recherche Inria

Invité(s) :

Adnane BOUKHAYMA Chargé de Recherche Inria
Stefanie WUHRER Chargée de Recherche Inria

Résumé en français

Au cours des dernières années, il y a eu une volonté croissante parmi les consommateurs de vivre des expériences immersives et réalistes dans diverses applications, y compris les films, les vidéoconférences, la réalité virtuelle et augmentée, et les jeux vidéo. Cette demande croissante signifie un marché commercial substantiel avec des perspectives d'investissement prometteuses, capturant l'attention à la fois de l'industrie et du monde académique.

La synthèse de points de vue novateurs a été l'un des domaines de recherche les plus populaires, montrant un grand potentiel dans diverses applications de graphiques et de vision par ordinateur. Parmi les algorithmes de synthèse de points de vue novateurs, le rendu neuronal exploite la puissance des réseaux neuronaux pour générer de nouvelles perspectives visuellement réalistes, présentant de nouvelles possibilités pour une synthèse d'image contrôlée et photoréaliste tout en évitant la nécessité d'une adhérence stricte aux paramètres physiques. En utilisant des méthodes d'apprentissage profond et en exploitant d'importants ensembles de données d'observations visuelles préexistantes, le rendu neuronal montre la capacité de gérer des scènes complexes et de produire des résultats visuels réalistes. Une illustration marquante de cette approche est les Champs de Radiance Neuronaux (NeRF), qui déploient un perceptron multicouche (MLP) pour approximer les champs de radiance et de densité au sein d'une scène 3D. En acquérant cette représentation volumétrique, NeRF facilite le rendu de scènes à partir de divers points de vue de caméras virtuelles grâce à des techniques de rendu analytiquement différentiables, telles que l'intégration volumétrique.

Néanmoins, le modèle NeRF original implique des réseaux neuronaux complexes, des algorithmes intensifs en calcul, et une demande pour un grand nombre d'images méticuleusement calibrées, entraînant des exigences élevées en mémoire computationnelle et des temps de formation prolongés. Par exemple, le NeRF conventionnel nécessite généralement environ dix heures pour la formation d'une seule scène, limitant ainsi sa viabilité pratique dans les applications réelles. De plus, une baisse notable de la qualité des points de vue novateurs synthétisés devient évidente à mesure que le nombre de vues d'entrée diminue.

Parmi ces défis, la capacité à générer de nouveaux points de vue à partir de données

d'entrée clairsemées est d'une importance primordiale, non seulement pour les champs de radiance neuronaux, mais aussi pour d'autres scénarios et applications, tels que les réseaux de champs lumineux ou le rendu de plusieurs sujets humains. En particulier, la tâche de synthétiser de nouveaux points de vue pour plusieurs sujets humains présente un défi significatif, surtout lorsqu'il s'agit de traiter des problèmes liés aux occlusions, aux détails complexes et aux poses humaines compliquées. La synthèse de nouveaux points de vue pour des scénarios multi-humains à partir d'entrées clairsemées introduit un objectif encore plus redoutable mais très prometteur, capable de fournir des expériences immersives et interactives. Surmonter ces défis permettrait non seulement de faire progresser le domaine de la synthèse de nouveaux points de vue (y compris les champs de radiance neuronaux et les réseaux de champs lumineux) mais aussi d'ouvrir une multitude de possibilités passionnantes dans les applications réelles, couvrant la réalité virtuelle, la réalité augmentée, la robotique, la création de contenu, et divers autres domaines où la capacité de synthèse d'images réalistes est d'une importance primordiale.

Contributions de cette thèse

Notre thèse s'est concentrée sur un sujet important concernant la synthèse de points de vue novateurs à partir d'entrées clairsemées. Plus précisément, nous étudions et proposons des solutions pour trois techniques importantes en matière de synthèse de points de vue novateurs à partir d'entrées clairsemées : les champs de radiance neuronaux avec peu d'exemples et la synthèse de points de vue novateurs, les réseaux de champs lumineux avec peu d'exemples et la synthèse de points de vue novateurs, la reconstruction multi-humaine et la synthèse de points de vue novateurs. Nous introduisons chacune de ces méthodes.

Champ de radiance neuronal à partir d'entrées clairsemées:Nous avons exploré les champs de radiance neuronaux souffrant d'une dégradation de la qualité à partir de vues clairsemées. Nous avons présenté une nouvelle approche pour améliorer le champ de radiance neuronal (NeRF) à partir d'entrées clairsemées pour relever ce défi. Nos méthodes proposées comprennent une stratégie d'échantillonnage global, une régularisation géométrique utilisant des pseudo-vues augmentées, et un schéma d'échantillonnage local par patch avec une régularisation basée sur des patchs. Nous avons introduit l'utilisation d'informations de profondeur pour une régularisation géométrique explicite. L'approche proposée a surpassé plusieurs références sur des benchmarks réels et a obtenu des résul-

tats à la pointe de la technologie. Cependant, l'une des limitations est qu'elle nécessite des informations de profondeur précises à partir de vues clairsemées. Dans cette thèse, nous avons choisi d'utiliser la profondeur du capteur à partir de l'ensemble de données, tandis que des études futures pourraient explorer comment utiliser la profondeur estimée à partir du réseau. De plus, les recherches futures pourraient incorporer l'amélioration du champ de radiance neuronal et la reconstruction de surface implicite à partir d'entrées RGBD clairsemées.

Champ lumineux neuronal à partir d'entrées clairsemées: Nous avons proposé une nouvelle approche basée sur une représentation neuronale de champ lumineux pour une synthèse de points de vue novateurs avec peu d'exemples. Notre méthode proposée utilise un réseau neuronal implicite conditionné sur des caractéristiques locales de rayon générées à partir d'un rendu volumétrique grossier. Nous avons exploré différentes architectures de réseaux neuronaux convolutionnels. Avec l'échantillonnage basé sur la profondeur et le réseau MVS, nos méthodes peuvent généraliser l'apparence réaliste à travers les scènes. La méthode proposée offre des performances compétitives sur différents ensembles de données et offre une vitesse de rendu bien plus rapide. Les méthodes proposées nous permettent de bien généraliser vers de nouveaux points de vue de scènes vues et non vues à partir de quelques entrées. Parallèlement, notre approche réduit considérablement le coût computationnel du rendu tout en maintenant l'apprentissage de relations complexes. Bien que notre méthode offre un rendu efficace, elle éprouve encore des difficultés à reproduire le plus haut niveau de détails dans de grandes images réelles comme le font les méthodes basées sur NeRF. Cela est dû à la résolution réduite de notre volume de caractéristiques et à notre rendu grossier de caractéristiques, ce qui contribue à réduire la mémoire.

Rendu de forme 3D et de radiance de multi-humains à partir d'entrées clairsemées: Nous avons proposé une méthode basée sur l'apprentissage pour générer plusieurs humains à partir d'images clairsemées. Notre approche a abordé les défis de l'occlusion et du désordre dans les scènes multi-humaines en incorporant des contraintes géométriques à l'aide de maillages pré-calculés, une régularisation de rayon basée sur des patchs pour la cohérence de l'apparence, et une régularisation de saturation pour une optimisation robuste. Des expériences approfondies sur des données réelles et synthétiques ont démontré les avantages de notre méthode et ses performances de pointe par rapport aux méthodes existantes de reconstruction neuronale sur des ensembles de données multi-humains réels (CMU Panoptic [1], [2]) et sur des données synthétiques

(MultiHuman-Dataset [3]). Notre approche présente encore plusieurs limites. Premièrement, nous nous appuyons sur des ajustements SMPL, qui ne sont parfois précis que dans certains cas, en particulier pour des scènes avec de nombreux humains. Une solution possible est d'améliorer les reconstructions SMPL tout en formant les réseaux de géométrie et d'apparence. Deuxièmement, notre méthode ne modélise pas les interactions humaines proches, car il s'agit d'un cas bien plus difficile.

Notre thèse a centré son attention sur un sujet pivot, à savoir, la synthèse de points de vue novateurs à partir d'entrées clairsemées. Étant donné les exigences inhérentes en données de la plupart des algorithmes de synthèse de points de vue novateurs pour une formation précise, notre recherche s'efforce de concevoir un algorithme capable d'apprendre habilement à partir de données limitées ou clairsemées. Notre exploration englobe des algorithmes de premier plan, y compris les champs de radiance neuronaux, les réseaux de champs lumineux, et le rendu et la reconstruction multi-humains. Dans chacun de ces domaines, nous avons introduit des solutions innovantes visant à améliorer la performance des algorithmes de synthèse de points de vue novateurs lorsqu'ils sont confrontés à des vues d'entrée clairsemées. Notre aspiration est que ce travail serve à étendre les frontières de la connaissance et offre des orientations précieuses aux chercheurs qui se lancent dans de futures avancées dans ce domaine de recherche dynamique.

Les limites et les défis

Malgré ces contributions, plusieurs limites existent dans notre travail. Cependant, de nombreuses voies prometteuses existent pour relever ces défis et pour faire progresser et améliorer les découvertes et méthodologies articulées dans cette thèse.

Notre approche introduit des techniques de régularisation pour les champs de radiance neuronaux, mettant l'accent sur la régularisation de la profondeur et la déformation basée sur l'image. Pour améliorer davantage les champs de radiance neuronaux à partir d'entrées limitées, une exploration des modalités supplémentaires, comme les normales de surface, l'intégration d'informations temporelles pour la synthèse basée sur la vidéo, ou l'adoption de méthodes d'apprentissage non supervisées ou auto-supervisées pour atténuer la dépendance aux données étiquetées, est justifiée. La dépendance à une information de profondeur précise est une limitation ; donc, les travaux futurs peuvent se pencher sur la substitution de la profondeur dérivée du capteur par une profondeur monculaire estimée via des réseaux neuronaux ou en exploitant l'information de profondeur à travers

des techniques de structure à partir du mouvement. De telles améliorations pourraient notablement élever la qualité des surfaces estimées et des apparences rendues.

De plus, notre méthode dans ce chapitre utilise des coordonnées plus robustes et conditionne les représentations de rayons à l'aide de volumes de caractéristiques extraits. Pour renforcer la qualité du rendu, les efforts futurs devraient explorer l'utilisation de techniques de conditionnement avancées. Par exemple, l'incorporation de modèles d'attention ou de diffusion dans le processus d'extraction du volume de caractéristiques a le potentiel d'améliorer les résultats. Les modèles d'attention facilitent l'apprentissage par le réseau d'informations plus pertinentes, capturant des détails complexes et améliorant finalement la qualité du rendu. Pendant ce temps, les modèles de diffusion permettent le transfert efficace d'informations à travers différentes régions, permettant au réseau de mieux comprendre et modéliser les relations complexes au sein de la scène. De plus, l'utilisation de modèles pré-entraînés avancés sur d'importants ensembles de données peut aider le réseau à capturer des détails de scène complexes, conduisant à des rendus de meilleure qualité.

De plus, notre méthodologie pour le rendu de scénarios multi-humains, impliquant l'utilisation de SMPL à partir d'entrées clairsemées, ouvre des possibilités pour une génération de scène plus complexe. Bien qu'une limitation soit l'accent mis sur les scènes statiques, les travaux futurs devraient étudier des méthodes pour améliorer les résultats de rendu vidéo en considérant l'information temporelle. Il est possible d'étendre l'approche proposée au rendu multi-humain à partir de vidéos monculaires ou de vidéos multi-vues, permettant la synthèse de scènes multi-humaines dynamiques. De plus, relever les défis associés à divers objets, à des arrière-plans variés et à des conditions d'éclairage difficiles dans des scènes complexes présente une direction intrigante pour une exploration plus approfondie.

En conclusion, la montée en puissance de la synthèse de points de vue novateurs dans divers domaines souligne son importance. Cette thèse a apporté d'importantes contributions pour relever les défis de la synthèse de points de vue novateurs à partir d'entrées limitées. Les avancées dans ces domaines faciliteront une adoption plus large et une application pratique des technologies de synthèse de points de vue novateurs à travers diverses industries et disciplines académiques. Les travaux et perspectives futurs énoncés ci-dessus sont prêts à repousser encore plus les limites, favorisant le progrès dans ce domaine de recherche passionnant et bénéficiant à la communauté élargie travaillant dans des domaines connexes, tels que la réalité virtuelle, la réalité augmentée et les graphiques informatiques.

Acknowledgement

I would like to take a moment to express my deepest gratitude to all those who have supported and assisted me throughout the process of writing this thesis.

First and foremost, I would like to extend my heartfelt appreciation to my thesis supervisors for offering me the opportunity to work on this thesis topic and for the discussions we have had in the past time. Thanks for all your patience and effort in guiding me during the research period.

I would also like to thank all my dissertation board members. Your valuable suggestions and insightful comments have significantly contributed to the improvement and ultimate completion of my thesis. I am immensely grateful for your time and effort invested in reviewing my work.

I would like to extend my heartfelt thanks to all the people who love me and whom I love: my parents, my friends, and those who have accompanied me on my life journey.

Finally, thanks for going through difficulties and hope that all the best in the future.

Table of Contents

Chapter 1

Introduction

1.1 Context

In recent years, there has been a growing desire among consumers for immersive and realistic experiences in various applications, including movies, video conferences, virtual and augmented reality, and video games. This increasing demand signifies a substantial commercial market with promising investment prospects, capturing the attention of both industry and academia.

Classic view synthesis is widely used in computer graphics and computer vision to simulate the appearance of a scene or object from different viewpoints or under different conditions. However, it requires the explicit input of all physical parameters associated with the scene, including camera parameters, illumination conditions, and material properties of the objects. When generating controllable imagery of real-world scenes, classic view synthesis methods face significant challenges due to the lack of explicit physical parameters. The absence of complete physical parameter information constrains the development of many practical applications, such as real scene editing, augmented reality (AR), and virtual reality (VR).

In contrast, neural rendering utilizes neural networks to synthesize visually realistic novel views, which offers new possibilities for controllable and photo-realistic image synthesis alleviating the requirements of all physical parameters. By leveraging deep learning techniques and learning from large datasets of existing observations, neural rendering could handle complex scenes and produce realistic appearances. A prominent example of such techniques is Neural Radiance Fields (NeRF) [4]. NeRF employs a multi-layer perceptron (MLP) to approximate the radiance and density fields of a 3D scene. By learning this volumetric representation, NeRF enables the rendering of the scene from any virtual camera viewpoint using analytic differentiable rendering techniques, such as volumetric integration.

However, original NeRF involves complex neural networks, computationally algo-

rithms, and hundreds of calibrated images, which costs high computational memory and a long time for training. For instance, original NeRF usually takes ten hours to learn one scene, constraining the development of real-world applications. Moreover, the quality of synthesized novel view drops dramatically with the decrease of input views' number. Among those challenges, generating novel views from sparse inputs is not only crucial for neural radiance fields but also important in other novel view synthesis scenarios or applications such as light field networks or multiple humans rendering. Specifically, though multiple humans rendering brings many universal and significant applications, novel view synthesis of multiple humans remains a challenging task, especially in handling occlusions, fine details, and complex human poses. Synthesizing novel views of multi-humans from sparse inputs becomes an even more challenging but potential task for providing immersive and interactive experiences. Addressing these challenges would not only advance the field of novel view synthesis (e.g. neural radiance fields, light field network, etc.) but also unlock a range of exciting possibilities in real-world applications, such as virtual reality, augmented reality, robotics, content creation, and other domains where realistic image synthesis is crucial.

Synthesizing novel views from a limited number of input images holds great significance and offers numerous practical applications across various fields. Examining these algorithms and their application contributes to the advancement of novel view synthesis techniques and their practical use in various fields. The potential impact of these approaches extends beyond academic research, as they hold promise for transforming industries and enhancing immersive experiences.

1.2 Motivation and goals

Novel view synthesis from sparse inputs is a crucial area of research in computer vision and computer graphics with various potential practical applications. It involves generating new viewpoints of a scene or object from a limited number of input images, typically fewer than ten [5]. This task presents several challenges due to the limited availability of input data, such as inconsistent geometry representation, degradation of rendering quality, etc. This thesis focuses on addressing these challenges associated with view synthesis algorithms when trained on sparse inputs. Specifically, we investigate and propose solutions for three important techniques regarding synthesizing novel views from sparse inputs: neural radiance fields, light field networks, and novel view synthesis for

multi-humans.

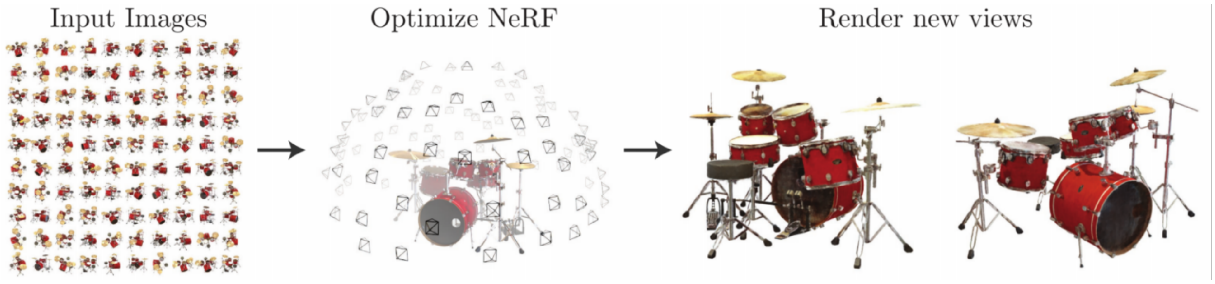


Figure 1.1 – Neural Radiance Fields (NeRFs) optimize a continuous 5D neural radiance field representation (volume density and view-dependent color at any continuous location) of a scene from a set of input images. It achieves impressive photo-realistic view synthesis results when trained on dense input images [4]. Image is taken from NeRF [4].

Recently, Neural Radiance Fields (NeRF) stands up as one of the most popular novel view synthesis algorithms [4]. NeRF optimizes a continuous 5D neural radiance field representation (volume density and view-dependent color at any continuous location) of a scene from a set of input images, e.g. 100 images. It achieves photo-realistic renderings when given dense inputs (see Figure 1.1), while its performance drops dramatically with the decrease of training views. We observe that the original NeRF is prone to over-fitting inputs rapidly at the beginning of training and lacks geometry regularization due to the scarcity of training views, usually resulting in rendering quality degradation. To address those challenges, we have chosen to explore classic image-based rendering and depth-based rendering algorithms, such as the image-warping technique. On the one hand, novel views generated by image warping can serve as pseudo truth and alleviate the over-fitting of NeRF's training. On the other hand, the powerful 3D scene representation capability of NeRF assists to remove artifacts in the warped images and render photo-realistic novel views. In addition, we propose to utilize sensor depth as explicit geometry regularization, so as to improve the synthesized quality of novel views. The combination of the classic rendering algorithm with NeRF together with the utilization of depth information helps to improve the rendered novel views' quality in NeRF when the training views are limited.

Light field network is another ongoing and representative algorithm for novel view synthesis. Recent method [6] leverages an implicit neural network to map each ray directly to its target pixel's color based on a given target camera pose. This light field representation facilitates faster rendering speed compared to volumetric rendering methods (e.g. NeRF), while the light field network still suffers from several challenges: rendering qual-

ity degradation from sparse inputs, limited generation capability across scenes, etc. Our work addresses those challenges and explores how to enhance the light field network's generation ability, including from limited input views and across scenes. We propose to utilize convolutional networks to extract feature volumes and condition the light field network with extracted local ray features. On one side, the convolutional network aids in inferring the implicit scene geometry of both unseen scenes as well as unseen views, enabling the light field network generalizable across scenes. On the other side, the light field network also enhances the quality of rendered novel view, effectively addressing the issue of blurriness commonly associated with convolutional networks.

The challenges of novel view synthesis become more pronounced when dealing with complex scenes, such as those involving multiple human shapes and radiance generations. These challenges mainly come from the complexity and variability of multiple humans' appearances, poses, occlusions, and interactions. Limited training views further exacerbate these difficulties due to insufficient geometry and appearance information available. Our key insight is that human-specific geometric constraints can be leveraged to tackle the challenging sparse-view setting. We propose to utilize the Skinned Multi-Person Linear Model (SMPL), a human body model, as a geometry prior and explicit geometry regularization. To further enhance the rendering quality when training views are sparse, we propose a patch-based regularization technique. This regularization ensures consistency across different rays, allowing for improved synthesis of novel views. By incorporating these strategies, our approach aims to overcome the challenges posed by complex scenes, limited training views, and the need for accurate geometry and appearance information in novel view synthesis tasks involving multiple human subjects.

In summary, our main goal is to enhance various algorithms regarding novel view synthesis from a limited number of input images. We have focused on the current most popular novel view synthesis algorithms. There are some common difficulties in those novel view synthesis techniques when given sparse inputs. For instance, the scarcity of input data will lead to the over-fitting of training and inconsistency in geometry estimation. Our proposed algorithms not only address those general challenges but also could serve as sub-modules within larger systems to tackle broader problems related to different novel view techniques, such as neural radiance fields and light field networks. By utilizing these algorithms as standalone solutions or as part of larger systems, we can effectively address the challenges and improve novel view synthesis from sparse inputs. In addition, our methods strive to develop algorithms that can adapt to different types of data, including

both synthetic and real-world data with varying baseline ranges, including light field data, multi-human data etc. Furthermore, the ability of our methods to learn from sparse inputs brings additional advantages in terms of data transportation, memory usage, and computational complexity, making it well-suited for industrial and real-world applications, where versatility and efficiency are key considerations.

1.3 Thesis structure and contributions

The dissertation is organized in the following manner:

In **Chapter 1**, we first present context, motivations, and goals in this chapter.

Chapter 2 provides a comprehensive overview of novel view synthesis. The initial part presents the state of the art of existing approaches and the associated challenges in generating novel views. It specifically highlights recent representative approaches such as neural radiance fields and light field networks. The subsequent section focuses on state-of-the-art methods for generating novel views of both single-human and multi-human.

Chapter 3 introduces our work of improving neural radiance fields (NeRF) from sparse inputs. We propose a global sampling strategy together with a geometry regularization utilizing warped images as augmented pseudo-views to encourage geometry consistency across multi-views. In addition, a local patch sampling scheme with a patch-based regularization is introduced to guarantee appearance consistency. Furthermore, our method exploits depth information for explicit geometry regularization and faster training. Our approach outperforms existing baselines on real benchmarks from sparse inputs and achieves the state of the art performance.

Chapter 4 proposes a novel approach for few-shot novel view synthesis based on a light field representation. Our method leverages an implicit neural network to map each ray directly to its target pixel's color based on a given target camera pose. We propose to condition the network with local ray features generated by coarse volumetric rendering from an explicit feature volume through convolutional neural networks. Our proposed conditioning scheme enables us to generalize well to novel views of both seen and unseen scenes from sparse inputs. Our approach achieves competitive performance across different datasets and offers a much faster rendering speed than those baselines.

Chapiter 5 presents a learning-based method for reconstructing multiple humans from sparse images including the following contributions: First, we propose to use geometry constraints by exploiting pre-computed meshes using a human body model (SMPL).

Specifically, we regularize the signed distances using the SMPL mesh and leverage bounding boxes for improved rendering. Second, we propose a patch-based ray regularization to minimize rendering inconsistencies and a saturation regularization for robust illumination conditions. Extensive experiments on both real and synthetic datasets demonstrate the benefits of our approach and show state-of-the-art performance against existing neural reconstruction methods.

In the **Chapiter 6**, we conclude our thesis by summarizing our contributions and discussing the future perspectives of our work.

Chapter 2

Background

2.1 Novel view synthesis: definition and development

Novel view synthesis refers to generating images or views of a scene from unseen or unobserved viewpoints, as shown in Figure 2.1. It has gained significant attention from the computer vision and graphics communities, leading to numerous advancements and applications [7].

Previous classic novel view synthesis techniques have explored diverse representations such as multi-plane images [8], depth-layered images [9], light fields [10], and depth-based warping [11], and so on. Those works usually use geometric modeling, optimization, and image warping to generate novel views.

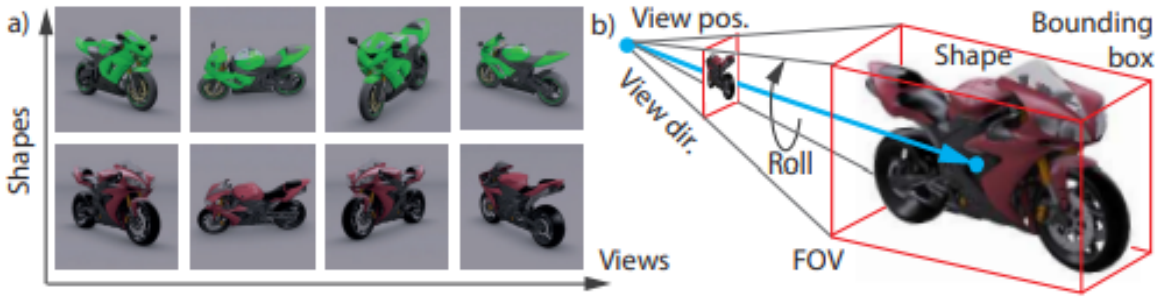


Figure 2.1 – Example of novel view synthesis [5]. It refers to generating new views of a scene or object from a limited set of input views or data. This task has applications in virtual reality, augmented reality, 3D content creation, and computer graphics. Image is taken from [5].

For instance, depth-based methods aim to estimate the depth information of a scene or object to synthesize novel views [12], [13]. Techniques like stereo matching, structure from motion, or depth from focus/defocus can be employed to recover depth maps. These depth maps are then used in the rendering process to generate novel viewpoints. Image-

based rendering approaches [14] leverage the available views to synthesize novel views by warping and blending the existing images. These techniques typically rely on the assumption of image consistency and use algorithms like texture mapping, morphing, and image-based modeling. They can be effective in generating new views by leveraging existing information. Multi-view stereo techniques aim to reconstruct the 3D geometry of the scene or object from multiple views [15]. Once the 3D geometry is obtained, novel views can be generated by rendering the scene from different viewpoints. MVS techniques often involve depth map estimation, surface reconstruction, and rendering algorithms to synthesize new views [16]. These techniques formed the foundation for subsequent research in novel view synthesis and provided valuable insights into the field.

While those traditional approaches have contributed significantly to novel view synthesis, they often had limitations in handling complex scenes, occlusions, and large disparities between views. They heavily relied on accurate depth estimation and geometric modeling, which could be challenging in certain scenarios. However, recent advances in deep learning have led to the development of more sophisticated and data-driven methods.

In recent years, deep learning-based approaches have shown promising results in generating high-quality novel views, which include convolutional neural networks (CNNs [17]), generative adversarial networks (GANs [18]), and variational autoencoders (VAEs [19]). In the early stages of deep learning-based novel view synthesis, 2D convolutional encoder-decoder architectures were commonly employed. These architectures aimed to map the sparse input to the target image while conditioning on the desired view. Some methods directly predicted colors [20], [21], while others predicted 2D flow fields [22]–[24] that were subsequently applied to the input. However, these approaches were outperformed by 3D-aware convolutional methods. These 3D-aware convolutional approaches utilized techniques such as volumetric rendering [25], rasterization [26], or learnable neural rendering [27], [28] to encode and render explicit 3D latent. These latents took the form of intrinsic scene representations [26], [29], [30] or extrinsic volume grids [25], [27], [28]. Later works [31] learn complex mappings between input views and target views, capturing intricate scene details and handling challenging scenarios more effectively. Although many of these methods could learn to generate 360-degree views from very sparse inputs, especially for synthetic central object data, most of them could not scale to high-resolution images, complex scenes, and real data such as multi-view stereo datasets (DTU [32]). Some also require foreground segmentation masks at training (e.g. [27]).

More recently, there has been a recent surge of interest in implicit neural shape and

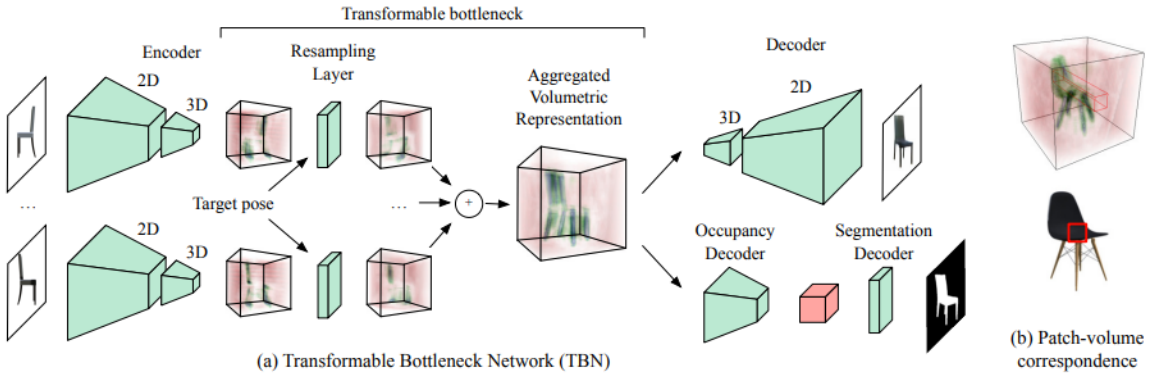


Figure 2.2 – Example of transformable bottleneck networks [27]. It consists of three main parts: an encoder, a resampling layer, and a decoder. The encoder includes 2D convolutional layers, reshaping operations, and 3D convolutional layers. The decoder is a mirror image of the encoder architecture. Image is taken from the transformable bottleneck networks [27].

appearance representations, exemplified by works like Neural Radiance Field (NeRF [4]), Scene Representation Networks [33], Neus [34], Neural Stages [35], SHARP [36], and NerfingMVS [37]. Additionally, neural rendering techniques such as Free View Synthesis [38], Stable View Synthesis [39], Deferred Neural Rendering [40], and Neural Volumes [41] have gained attention. Among those works, implicit neural radiance fields (NeRF [4]) stand up as a powerful representation for novel view synthesis and demonstrate photo-realistic renderings. In addition, light Field Networks (LFNs [6]) are ongoing research topics designed to process and leverage light fields and have shown promising results in exploiting the additional information available in light field data for various computer vision tasks. Our work includes investigating and exploring these two recent representative approaches for novel view synthesis, including neural radiance fields and neural light fields. Except for object rendering, we also explore human rendering and reconstruction methods. The following section highlights these two types of research.

2.2 Novel view synthesis: state of the art

2.2.1 Neural radiance field

The neural radiance fields (NeRF [4]) technique represents a scene as a continuous 5D function that maps spatial coordinates to radiance values. It has demonstrated remarkable

success in generating realistic novel views of scenes, even for intricate and highly detailed scenes. The NeRF network comprises a multi-layer perceptron (MLP) that maps spatial points to volume density and view-dependent colors (Figure 2.3 [4]). Images are rendered using hierarchical volumetric rendering. However, NeRF suffers from several limitations, including high computational and rendering time complexity, the requirement for dense training views, limited generalization capability across scenes, and the need for test-time optimization. In recent research, various approaches have targeted these challenges individually or jointly. The goals in those works include but are not limited to improving rendering speed, enhancing rendering quality from sparse views, and extending the generalization ability across different scenes or datasets. This section introduces recent research efforts in addressing these challenges.

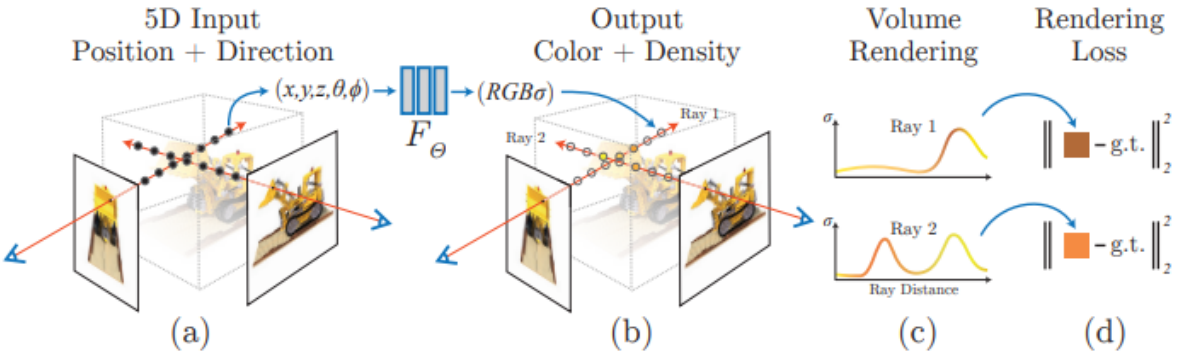


Figure 2.3 – Scene representation and differentiable rendering procedure in neural radiance field [4]. Image is taken from NeRF [4].

Fast neural radiance field The original neural radiance field (NeRF) [4] requires a significant amount of training time (hours or even days) to achieve photo-realistic rendering results. Extensive research has been conducted to explore different approaches for improving the efficiency of training and rendering speed while maintaining high-quality results [42]–[47].

Among those methods, Yu et al. [42] predict radiance spherical harmonic coefficients instead of density and continue to improve the efficiency leveraging plenoxels [48], alleviating NeRFs' rendering complexity by learning view independent radiance features. More recently, instant NGP utilizes hash encoding, and TensoRF adopts tensor decomposition; both achieve significant improvements in real-time training and rendering. Though those related works improve NeRF's training and inference speed more or less, most of them [43],

[48], [49] still focus on the single scene representing and require a dense well-calibrated view for training, ignoring the generalization ability across scenes. Sitzmann et al. [6] recently introduced a new implicit representation for modeling multi-view appearance. A neural light field function maps rays i.e. target pixels directly to their colors without any need for physical rendering. However, the method uses a hypernetwork for conditioning, making it expensive to scale to bigger images in computing and memory. This method was not demonstrated on real-world datasets (e.g. DTU [32], LLFF [10]). It was implemented in the auto-decoding setup, which means it requires test time optimization. We take an insight into light field networks [6] and explore strengthening light field networks' generalization ability across real-world data, especially on sparse setups.

Neural radiance field from sparse inputs The original NeRF demonstrated a photo-realistic rendering capability by representing a scene as a neural radiance field, while at the expense of well-calibrated dense views for training. The rendering quality of vanilla NeRF [4] drops significantly with fewer inputs due to the lack of geometry regularization. Current research works are trying to solve this issue (e.g. [42], [49]–[59]) in different ways, e.g. utilizing image encoder [50], [55], [56], additional depth information [51], [59], different regularization [52], [53], and so on.

Specifically, regularization-based methods adopt additional supervision on unobserved viewpoints to improve the generation capability of NeRF, particularly in sparse setups. These methods introduce extra supervision using various techniques, such as utilizing features extracted from a pre-trained visual encoder like CLIP-ViT [50], incorporating rendered depth or density from sampled patches [52], [53], and so on. These regularization techniques enhance NeRF's ability to generate novel views from sparse inputs, but they often ignore training efficiency and computational complexity.

Depth-based methods [51], [59], [60] highlight the significance of depth information in achieving faster rendering and higher rendering quality. However, these methods primarily focus on per-scene fine-tuning, limiting their generalization ability across different scenes. Several approaches propose augmenting NeRF with 2D [56], [61], [62] and 3D convolutional features [55] extracted from input images. By incorporating encoded features, these methods offer forward-pass prediction models that eliminate the need for test-time optimization and enable generalization across scenes. However, they still rely on evaluating hundreds of 3D query points per ray during inference, similar to NeRF, resulting in slow rendering speeds. In this thesis, we investigate and propose solutions to improve the

neural radiance fields from sparse inputs.

	NeRF	PixelNeRF	MVSNeRF	LFN
Scene prior	✗	✓	✓	✓
Features	✗	2D	3D	HyperNet
Generalization	✗	✓	✓	✓
Speedup	✗	✗	✗	✓
Real-world scene	✓	✓	✓	✗
Coordinate	Point + direction	Point + direction	Point + direction	Ray

Table 2.1 – Neural radiance fields (NeRFs [4]) map a 5D coordinate to its' corresponding radiance and composites the color via volumetric rendering. PixelNeRF [56] and MVS-NeRF [55] are conditional versions of NeRF that aim to improve its generalization ability using an image encoder. LFN [6] learns a function that maps 4D light field samples to color space.

2.2.2 Neural light field

Light field networks have shown great promise in novel view synthesis with high efficiency and rendering quality, enabling multiple applications such as virtual reality, augmented reality, and image and video processing [63]–[67]. There are types of light field networks, including convolutional neural networks (CNNs) [63], [65] and recurrent neural networks (RNNs) [66], [68]–[70], etc. Recent works combine light field networks with neural rendering, which leads to improved rendering quality, as it allows for more accurate modeling of complex scenes and better handling of reflections. However, challenges remain in dealing with large-scale scenes and generating high-quality, photo-realistic images.

To address these challenges, researchers continue to progress in the cross-domain of the neural light field with large-scale scenes, and efficient and photo-realistic rendering. For instance, Wang et al. [64] distill a neural radiance field to a neural light field to produce a compact and lightweight model that could generate high-quality light field data with fewer computational resources. However, the quality of the generated light field data may be lower than that produced by the original NeRF [4], especially in complex lighting conditions and scene geometries. To model complex scenes' appearance and geometry accurately, Suhail1 et al. [70] combines epipolar geometry with a light field network to represent a novel view dependently. However, generating novel views across different scenes remains challenging for the above works [64], [69], [70].

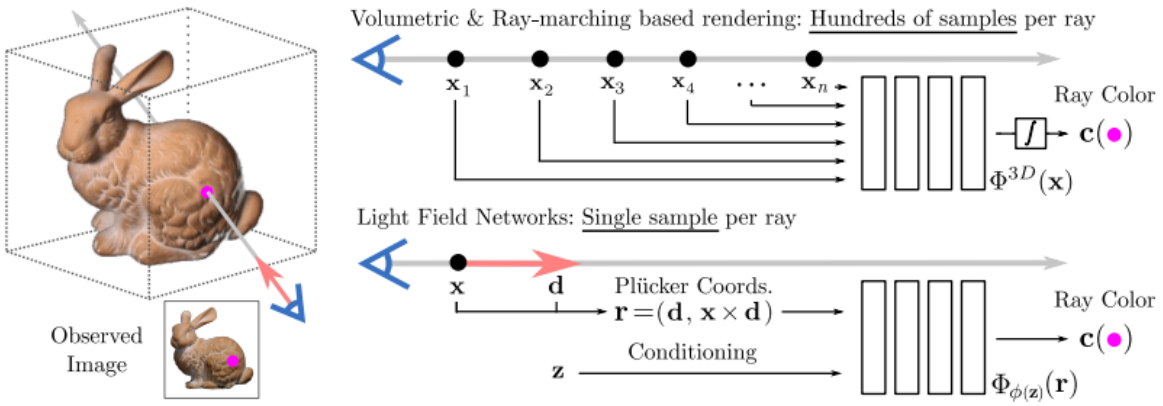


Figure 2.4 – Light field networks (LFNs) encode the full 360-degree light field of a 3D scene [6], which enables subsequent real-time novel view synthesis of simple scenes. Image is taken from [6].

The later work [68] learns a light field representation across different scenes by leveraging an epipolar geometry transformer and directly predicting the color of a target. It enables novel view generation across scenes with considerable good quality at the expense of more training views than existing conditional NeRF methods. Specifically, MVSNeRF [55] demonstrates realistic rendered novel views by using only three reference views as inputs, whereas [68] requires ten input views for achieving similar performance as shown in its paper. To sum up, synthesizing novel views with neural light fields from sparse views is still promising yet under exploring methods. Thus, we explore and address the challenges of novel view synthesis with neural light fields from sparse inputs.

2.2.3 Hybrid surface and volume rendering

Surface and volume rendering of object Neural implicit surface rendering has leveraged neural networks to approximate the implicit surface function and extended to novel view synthesis and rendering of implicit surfaces, [35], [71]–[73]. By describing the 3D geometry as the zero-level set of the function, the neural network takes input parameters (e.g., 3D coordinates [71], occupancy values [72]) and outputs a value representing the signed distance to the surface or a signed distance function (SDF [74]). The rendering process involves sampling points in 3D space and evaluating the neural network to obtain the SDF values. These values provide information about the distance to the surface and can be used to estimate surface normals, shading, and appearance properties (Figure.

2.5). Techniques such as ray marching, sphere tracing, or volume rendering are used to generate high-quality renderings, [71].

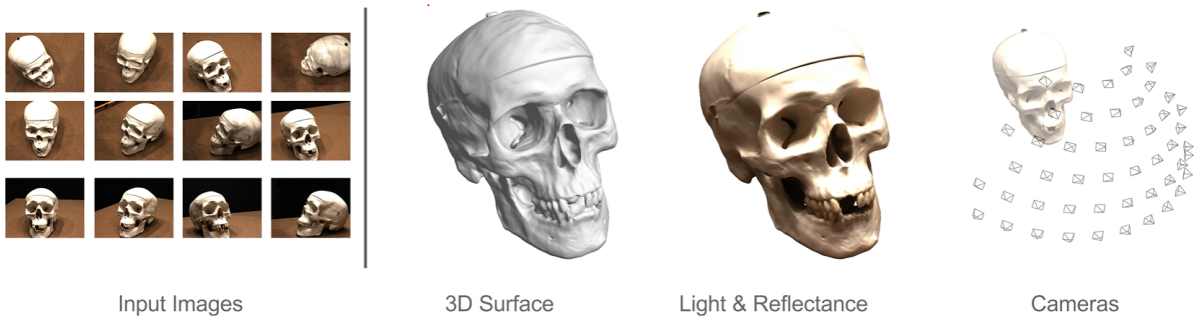


Figure 2.5 – Example of multi-view 3D surface reconstruction [71]. The network aims to learn the geometry surface from given input images and model a wide range of lighting conditions and materials. By incorporating the rendering equation principles, the neural renderer captures the complex interplay of light and surface properties, enabling realistic rendering of various lighting scenarios and material appearances. Image is taken from [71].

Specifically, various neural network architectures have been employed for Neural implicit surface rendering, including multi-Layer perceptrons (MLPs) [35], [71], [72], convolutional neural networks (CNNs) [74], and more advanced architectures like deep implicit functions (DIFs) [73], occupancy networks [75], and neural radiance fields (NeRF [4]). These architectures capture the intricate details and complex topology of implicit surfaces.

Among those works, implicit differentiable renderer (IDR [71]) proposes to learn an implicit representation directly from multi-view images to recover more accurate 3D geometry along with appearance. The geometry is represented as the zero-level set of a neural network, which allows for flexible and adaptable modeling of the scene's shape. Given a set of masked 2D images as inputs, IDR aims to learn three components simultaneously: the unknown geometry of the scene, the camera parameters, and a neural renderer that approximates the light reflected from the surface to the camera. As shown in Figure 2.6, the MLP takes the surface point x and normal n, the viewing direction v, and a global geometry feature vector z as inputs and outputs the RGB values of corresponding camera position. Optimization mainly depends on the pixel color of the input images while enabling simultaneous learning of the geometry, its appearance, and camera parameters. However, this work requires accurate mask images as inputs, which are unavailable in many real-life applications.

Consequent works propose to combine implicit representations with volume rendering

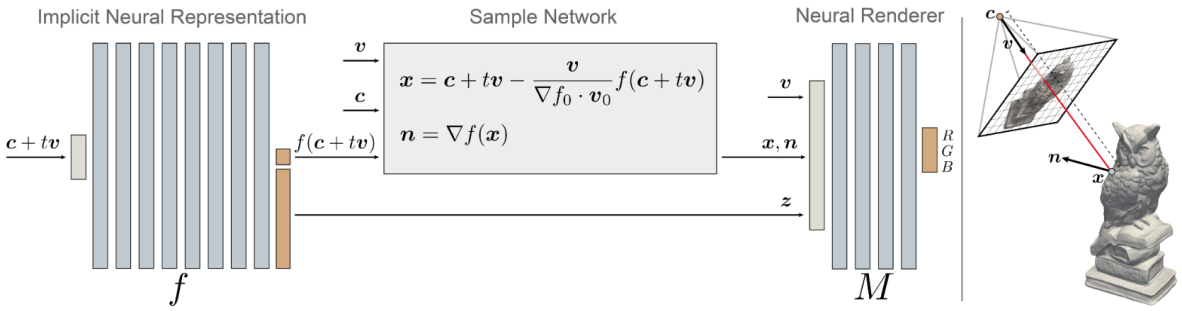


Figure 2.6 – The implicit differentiable renderer (IDR [71]) forward model generates differentiable RGB values of a learnable camera position c and a fixed image pixel p. This is accomplished by defining a viewing direction v based on the camera parameters and pixel. IDR computes the intersection x of the viewing ray, represented as $c + tv$, with the implicit surface. Image is taken from [71].

[34], [76], [77]. Precisely, VolSDF [77] is proposed to combine an implicit SDF representation with volume rendering, transforming SDF values into volume densities by using the cumulative distribution function of the Laplace distribution. NeuS [34] uses an SDF to represent the surface and develops a new volume rendering method to train a neural SDF representation. Though those methods could reconstruct 3D geometry and appearance simultaneously, it remains challenging to reconstruct a geometry-consistent surface, especially for scenes containing complex geometry. Moreover, these methods show remarkable reconstruction results but still suffer from rendering degradation when the number of input views is limited.

Surface representation and volume rendering of human Nowadays, there has been extensive research on reconstructing 3D humans from various types of input data, including from single images [78]–[82], monocular video [83]–[85], RGB-D data [86]–[88] and multi-view data [89]–[92]. Most existing methods are focused on single human reconstruction, especially in multi-view settings.

Some early works proposed by Starck and Hilton et al. [89], employ a combination of visual hull and stereo reconstruction techniques to capture the human surface. However, these high-end multi-view capture systems require complex studio setups that are expensive and not easily accessible. To address this issue, researchers have developed methods that utilize a sparse set of RGB cameras, typically ranging from 2 to 15 cameras. These methods compensate for the limited views and comprehensive baselines by employing various strategies. One approach is to track a pre-scanned template using the available camera

views. Gall et al. [93] and Vlasic et al. [94] proposed methods that rely on template tracking and utilize temporal information to reconstruct the 3D human shape. Carranza et al. [95] and De Aguiar et al. [96] also explored template-based techniques for reconstructing human motion from multi-view data. Another direction of methods involves leveraging parametric body models to aid the reconstruction process. Researchers like Huang et al. [97] and Balan et al. [98] introduced parametric models of the human body, which can be fitted to the observed views and used to estimate the 3D shape and pose.

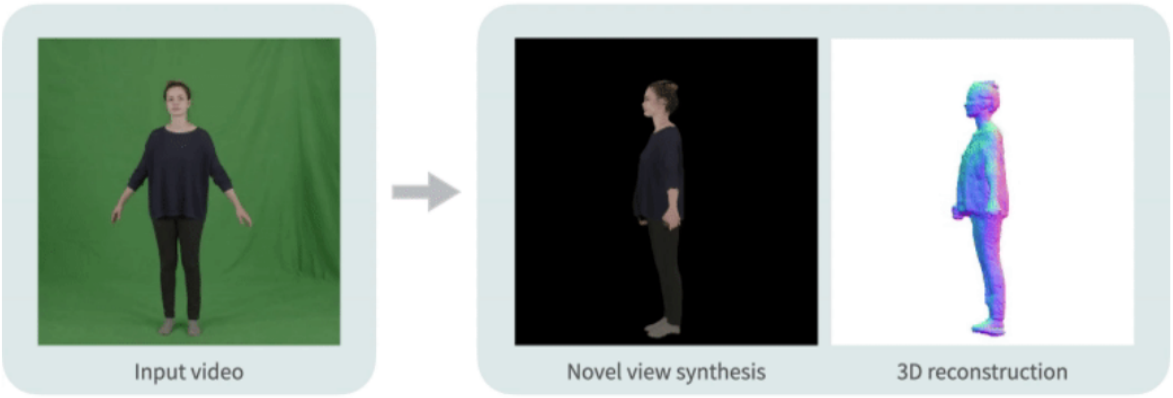


Figure 2.7 – Novel view synthesis and shape reconstruction of single human [99]. Image is taken from [99].

More recently, deep learning techniques have been applied to tackle the problem of multi-view human reconstruction. Huang et al. [92] proposed a deep learning approach that employs a convolutional neural network (CNN) to estimate the 3D human shape from multiple camera views. Liang et al. [100] introduced a method that combines a CNN with a differentiable renderer to reconstruct detailed 3D human shapes from sparse views. Current works have explored the use of neural networks and deep learning for multi-view human reconstruction, such as ARAH [101], Neural Body [102], HumanNeRF [103], and so on [99], [104]. These methods utilize neural networks to learn the mapping between the input views and the 3D human shape, enabling more accurate reconstructions. It's important to note that the field of single-human reconstruction is rapidly evolving, and new techniques and advancements are being proposed to improve the quality and efficiency of the reconstructions.

Despite the great progress of existing research on single human generation, only a limited number of studies have addressed the challenging problem of multiple human generations [105]. This task becomes difficult due to the increased geometric complexity

introduced by the presence of multiple people, resulting in occlusions and amplified ambiguities that hinder the accurate assignment of commonly used features such as color, edges, or key points.

The predominant approach for generating multi-human models from single images and videos involves regressing the parameters of the SMPL [106] body model [105], [107]–[117]. While this approach can robustly handle even a single view, the resulting reconstructions need more fine geometric details and accurately capture characteristics such as hair, clothing, and intricate details. Notably, Mustafa et al. [118] proposed an exception to this approach by performing model-free reconstruction of multiple humans, combining an explicit voxel-based representation with an implicit function refinement. However, this method requires training on a large synthetic dataset specific to multiple people, limiting its generalization to diverse scenes.

Multi-view capture setups can help resolve depth ambiguities and some of the occlusions. Classic methods for estimating multiple humans rely heavily on segmentation masks and template mesh tracking [119]–[121]. We avoid using segmentation masks by adopting volumetric rendering for implicit surfaces [34]. More recently, deep learning-based approaches were proposed, but they either require temporal information [3], [122]–[124], pre-training on a large dataset [3] which cannot work on general scenes, or a coarse body model [122]–[124] which lacks geometric detail.

Our work focuses on novel view synthesis of multi-human on static scenes from sparse inputs and proposes a method that recovers accurate reconstructions and produces renderings of novel viewpoints.

Chapter 3

Few Shot Neural Radiance Field based Novel View Synthesis

3.1 Introduction

NeRF [4] has achieved photo-realistic rendering, while it requires many well-calibrated dense inputs for training. The rendering quality of original NeRF [4] drops significantly with few shot inputs due to the lack of geometry regularization and over-fitting to training views.

Recent approaches solve this issue by utilizing image encoder [55], [56], depth regularization [51], [52], [59], ray density regularization [53], etc. On the one hand, those methods require pre-training on a large-scale dataset [50], [56] or ignore the training efficiency [52]. On the other hand, depth information is one of the key factors for 3D geometry learning in previous works [51], [52], [59], which improves the rendering quality of NeRF efficiently. Thus, we propose to utilize depth information and introduce a novel framework combining geometry and appearance regularization to improve neural radiance fields from sparse views.

In this chapter, instead of sampling rays randomly like vanilla NeRF [4], we propose a local patch-based ray sampling scheme and a global patch-based ray sampling scheme for different purposes. Firstly, it remains challenging for the neural radiance field to infer a reasonable 3D geometry from only a limited number of views. To encourage geometry consistency across multi-views, we propose to sample a patch from images globally, warp those patches from an observed camera to an unobserved camera, and optimize the rendering of unobserved views using those warped patches' feature extracted by a pre-trained visual encoder (CLIP-ViT) [125]. In addition, NeRF's supervision depends on mean squared error (MSE) loss to optimize the pixel's color prediction while not considering each pixel's neighborhood information. By utilizing 2D local neighborhood information of each pixel, we sample the local patch from the training image and supervise the rendering with per-

ceptual loss. Moreover, depth information could provide essential geometry cues for 3D reconstruction and also facilitate the training and rendering. Our method exploits depth information for explicit geometry regularization and faster training and inference speed. Each part of the proposed method is analyzed and evaluated in ablations.

We evaluate our approach on real-world dataset (DTU dataset [32]) and compare it with recent representative NeRF-related works [43], [50], [52], [55], [56], [59], [60], [126]. It outperforms existing methods quantitatively and qualitatively, and achieves state-of-the-art performance.

3.2 Related work

Novel View Synthesis: There has been a vast amount of research focused on novel view synthesis, which is classified into traditional methods (e.g. light field [127], image-based warping [12] etc.) and learning-based methods (e.g. neural rendering [38], implicit neural representations [4], [43], [45], [50], [52], [56], [126], etc.). Among those, NeRF [4] stands out by representing 3D scenes as neural radiance fields and demonstrating the photo-realistic quality of the synthesized novel view. While NeRF requires dense training views, a long time for training, etc.

Current research work is tackling each of these limitations (e.g. [42], [43], [48], [126], [128], etc.). For instance, some research improve NeRF for faster training and rendering speed, including octree-based 3D representation [42], [48], multi-resolution hash encoding [45], tensor decomposition (TensoRF [43]). However, those methods still require dense training views, and the rendering quality decreases when given sparse inputs. We build our work on TensoRF [43] for faster training speed. While our goal is to improve the rendering quality from sparse inputs directly and our proposed regularization applies to most of the existing NeRF and fast NeRF architecture.

Few Shot Radiance Field: Existing methods improve NeRF's generalization ability from few shot inputs by exploiting conditional encoder [55], [56], different regularization(e.g. multi-view Stereo [129], feature [50],color [52], density [53], etc.) or depth-based method [51], [59], [60]. Among those, conditional NeRF leverages convolutional neural network (CNN) to extract 2D features [56], [129] or 3D cost volumes [55]. Although those methods improve rendering quality, they require pre-training on large-scale data

and cost more computation and time complexity. The regularization approaches often introduce additional information for supervision, e.g. a pre-trained visual encoder such as CLIP-ViT [50], [125], rays sampled from unobserved cameras [52], [53] to enhance the NeRF' generation ability, while ignoring training efficiency. Depth-based methods [51], [59], [60] have proven that depth is an important factor for fast and high-quality rendering. We take advantage of both the regularization and depth-based methods and propose a novel framework combining geometry and appearance regularization with an assistant of depth for view synthesis from sparse inputs.

3.3 Methodology

3.3.1 Preliminary

NeRF represents 3D scenes as neural radiance fields using a multi-layer perception(MLP). For a sampled ray r starting at camera origin o with view direction v, the color $C(r)$ is rendered as follows [4]:

$$C(r) = \sum_{i=1}^{N} T(p_i)\alpha(p_i)c(p_i), T(p_i) = \prod_{j}^{i-1}(1 - \alpha(p_j)) \tag{3.1}$$

where $p_i(p_i = o + t_i v, i = 1, ..., N)$ is a sampled point along the ray r, $c(p_i)$ is the predicted color at the point p_i, $T(p_i)$ is the accumulated transmittance, and $\alpha(p_i)$ is the opacity value.

When given dense inputs, NeRF [4] could achieve realistic rendering quality by optimizing the network through photo-metric loss [4]:

$$L_r = ||I_t - I_r||_2^2 \tag{3.2}$$

where I_t is the ground truth of the target image and I_r is the rendered image.

TensoRF [43] models radiance field scenes as 4D tensors and factorizes those tensors into multiple compact low-rank components aiming at speedup and higher rendering quality from dense inputs, see Figure 3.1. When given sparse inputs, the rendering quality of the synthesized novel view of TensoRF [43] decreases significantly as NeRF. Our approach builds on TensoRF [43] for faster training speed, but it could be plugged into most NeRF [4] based methods.

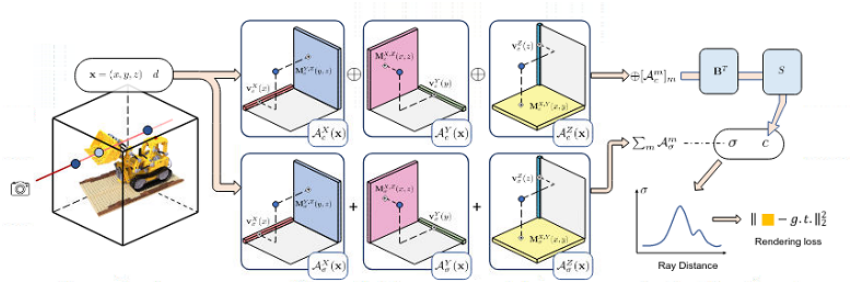


Figure 3.1 – Overview of TensoRF [43]: it models a scene as a tensorial radiance field using a set of vectors (v) and matrices (M). These vectors and matrices are utilized to represent and encode the appearance and geometry of the scene along their corresponding axes. Image is taken from [43].

Our goal is to improve neural radiance fields from sparse inputs. The framework is shown in Figure. 3.2. Instead of randomly sampling rays like vanilla NeRF [4], we propose a combined local and global patch-based ray sampling strategy. Specifically, we sample a patch from an input image globally, warp the sampled patches from seem camera to an unobserved camera and optimize the rendering of unobserved views using those warped patches' feature extracted by a pre-trained visual encoder (CLIP-ViT [125]). While for rays sampled on a local patch from a training image, we supervise its' rendering with features encoded by VGG network [130]. It enables the use of 2D local neighborhood information of each pixel. Moreover, we utilize depth information for explicit geometry regularization and faster training and inference speed. The following section will introduce the proposed geometry regularization, appearance regularization, and joint optimization with depth regularization separately.

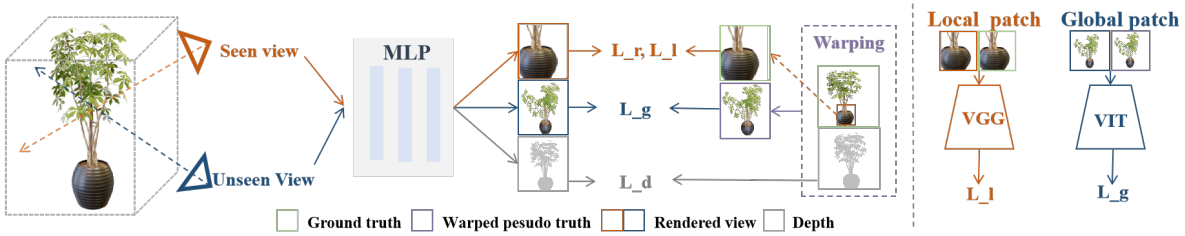


Figure 3.2 – Overview of the proposed framework. Our proposed regularizations include photo-metric loss L_r, global geometry regularization L_g, local patch appearance regularization L_l and depth regularization L_d.

3.3.2 Geometry regularization

Vanilla NeRF [4] is prone to over-fit to sparse inputs when only a few training views are available. Meanwhile, the neural network is hard to estimate a reasonable 3D geometry based on sparse images. To address this challenge, we utilize depth information corresponding to the sparse training views and augment the training data with a forward warping scheme. As suggested by [52], [53], regularizing the unobserved rays with seen rays could improve the overall rendering quality.

Thus, we take insight from that and transfer the pixel x_s from the observed camera to the unobserved camera as follows:

$$x_t = K_t T_t D(x_s) K_t^{-1} x_s \tag{3.3}$$

where x_t is the pixel in the target view from the unobserved camera, K_t, T_t is the camera intrinsic matrix and camera transformation matrix, respectively, and $D(x_s)$ is the depth map corresponding to seen views. The depth map could be obtained from Structure from Motion (SfM), e.g. COLMAP [131].

Due to the forward warping and sometimes inaccurate depth map, the warped image usually contains some holes, which might degrade the rendering quality if the supervision entirely depends on those warped pixels. To optimize the geometry estimation while minimizing the influence of unreliable pixels, we propose to sample a patch globally (e.g. sampling a down-sampled image). We render the hole patch and optimize it using the warped pseudo-views as follows:

$$L_g = ||\phi(I_w) - \phi(I_r)||_2^2 \tag{3.4}$$

I_w is the warped patch of pseudo-views, I_r is the rendered patch, and ϕ denotes an image encoder, where we use a pre-trained visual encoder (CLIP-ViT [125]) for it. Please note that DietNeRF [50] adopts CLIP-ViT to ensure semantic information and uses only training views for supervision. Different from it, we take the warped image as a pseudo-view and utilize it to optimize the unobserved views' rendering. This function encourages geometry consistency across multi-view by utilizing depth information and forward warping.

3.3.3 Appearance regularization

NeRF and its related works [4], [52], [53] usually utilize mean squared error(MSE) loss to optimize the pixel's color prediction. However, depending on MSE loss solely ignores each pixel's information regarding its' neighborhood, which might result in blur images. Different from sampling pixels randomly in vanilla NeRF [4] and global patch sampling above, we propose to sample a complete patch from training images. The appearance regularization is defined based on the sampling strategy:

$$L_l = ||\varphi(I_t) - \varphi(I_r)||_2^2 \tag{3.5}$$

where I_t and I_r is ground truth patch and the rendered patch respectively, φ denotes image encoder. We use a pre-trained VGG model [132] to extract features in our experiments. By regularizing the extracted features, the key point is to take advantage of each pixel's 2D local neighborhood to encourage more realistic appearance rendering.

3.3.4 Joint optimization

During the training, we optimize all parameters of the network jointly by back-propagating a combination of total loss L:

$$L = L_r + \lambda_g L_g + \lambda_l L_l + \lambda_d L_d \tag{3.6}$$

L_r is the $L2$ reconstruction loss between the predicted image and ground truth in equation 3.2, L_g and L_l are the geometry consistency loss and appearance regularization loss in equation 3.4 and equation 3.5 respectively, L_d denotes depth regularization, λ_g, λ_l and λ_d are hyper parameters setting to 0.01, 0.01,1 in experiments. We exploit a depth map for explicit geometry regularization:

$$L_d = ||M \odot (D_r - D_t)||_1^1 \tag{3.7}$$

where $\odot$ denotes hadamard product, M is the mask for removing invalid depth information, D_t is the ground truth depth, and D_r is the rendered depth. D_r is rendered as

follows [4]:

$$\tilde{D}_r = \frac{1}{\sum_{z=1}^{N} T_z \alpha_z} \sum_{z=1}^{N} T_z \alpha_z t_z \tag{3.8}$$

where T and α are detailed in equation 3.1.

3.4 Experiments

3.4.1 Implementation details

We implement our approach on TensoRF [43] codebase with the PyTorch framework on a Quadro RTX 5000 gpu. We optimize the training with the Adam solver using learning rate decay from 10^{-4} to 10^{-5}. Each scene's training takes around one hour. Please note that the original NeRF takes around 8 hours in training, the proposed methods enable much faster training speed.

Method	Setting	PSNR↑			SSIM↑			LPIPS↓		
		3-view	6-view	9-view	3-view	6-view	9-view	3-view	6-view	9-view
SRF [129]	Trained on DTU	15.32	17.54	18.35	0.671	0.730	0.752	0.304	0.250	0.232
PixelNeRF[56]		16.82	19.11	20.40	0.695	0.745	0.768	0.270	0.232	0.220
MVSNeRF[55]		18.63	20.70	22.40	0.769	0.823	0.853	0.197	0.156	0.135
SRF [129]	Trained on DTU & Optimized Per-scene	17.07	16.75	17.39	0.436	0.438	0.465	0.529	0.521	0.503
PixelNeRF[56]		16.17	17.03	18.92	0.438	0.473	0.535	0.512	0.477	0.430
MVSNeRF [55]		17.88	19.99	20.47	0.584	0.660	0.695	0.327	0.264	0.244
FWD [60]	Optimized Per-scene	21.98	-	-	0.791	-	-	0.208	-	-
mip-NeRF [126]		8.68	16.54	23.58	0.571	0.741	0.879	0.353	0.198	0.092
TensoRF [43]		13.77	15.84	17.27	0.545	0.614	0.662	0.382	0.296	0.267
DietNeRF [50]		11.85	20.63	23.83	0.633	0.778	0.823	0.314	0.201	0.173
RegNeRF [52]		18.89	22.20	24.93	0.745	0.841	0.884	0.190	0.117	0.089
DSNeRF [59]		16.9	20.60	22.30	0.57	0.75	0.81	0.45	0.29	0.24
Ours		22.02	24.16	25.74	0.802	0.829	0.858	0.135	0.151	0.076

Table 3.1 – Comparison of the average PSNR, SSIM, and LPIPS of reconstructed images in the DTU [32] dataset, using 3/6/9 views for training. The higher the better for both PSNR and SSIM. The lower the better for LPIPS [133]. The color represents the performance of ranking, the darker the better.

3.4.2 Evaluation on DTU dataset

We demonstrate our method for novel view synthesis from sparse inputs using real-world multi-view datasets DTU benchmark [32]. Following the PixelNeRF [56] and MVS-

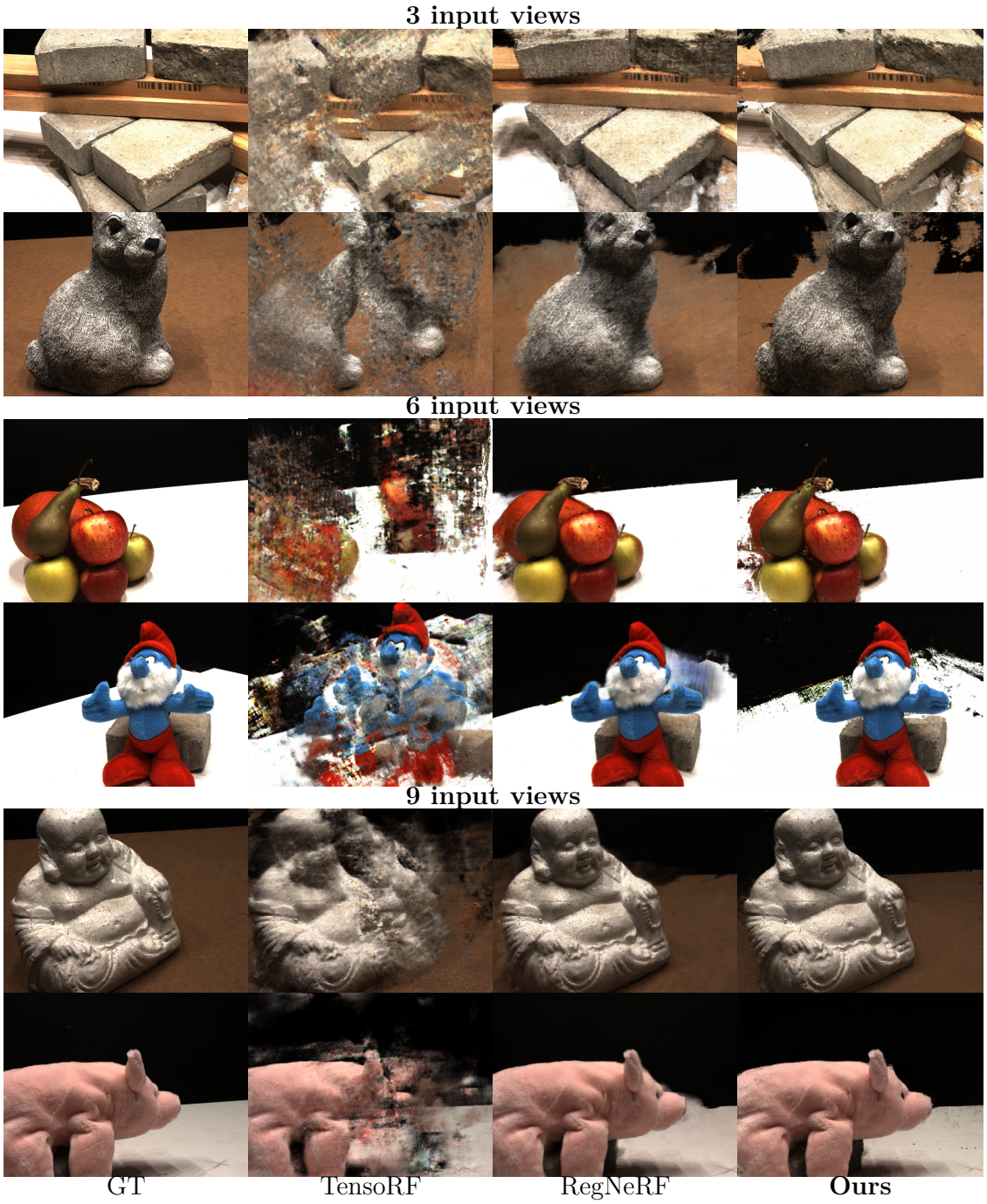


Figure 3.3 – Qualitative comparison with TensoRF [43] and RegNeRF [52] from 3/6/9 input views on the DTU dataset [32].

NeRF [55] experimental settings, the data is split into 88 training scenes and 16 testing scenes, each scene including 49 images with an orginal resolution of 1600×1200. We note

that this is a challenging scenario due to the complex illumination, geometry, and so on. For instance, the lighting and backgrounds are also inconsistent between the scenes. We pick the same training and testing views as RegNeRF [54].

Following the RegNeRF [52] experimental settings, we evaluate all experiments on their reported test set of 15 scenes with down-sampled resolution (400×300). For quantitative comparison, we report the peak signal-to-noise ratio (PSNR), structural similarity (SSIM), and learned perceptual image patch similarity (LPIPS) reconstruction metrics in Table 3.1 for 3/6/9 training views averaged across all testing scenes. We report evaluations of both generalizable approaches (SRF [129], PixelNeRF [56], MVSNeRF [55]) and unconditional baselines (MipNeRF [126], DietNeRF [50] and RegNeRF [52]). We build our work on TensoRF [43] and improve it with sample space annealing [52], so we report the evaluation of improved TensoRF on Table 3.1. Moreover, we compare with depth-based methods, including the method using colmap depth (DSNeRF [59]) and the method using sensor depth (FWD [60]). Table 3.1 demonstrates that our method is robust and achieves comparable performance with generalizable approaches [55], [56], [129] and unconditional baselines [43], [50], [52], [59], [126] in all three metrics. Especially, our method also outperforms methods utilizing depth information [59], [60].

We demonstrated qualitative comparisons in Figure 3.3 from different scenes in 3/6/9 input views. Since RegNeRF [52] is the former state of art method average on 3/6/9 inputs on Table 3.1, we mainly compare our visual results with RegNeRF and our baseline TensoRF [43]. Figure 3.3 shows that our model improves TensoRF's generation capability on 3/6/9 settings and could render images with higher frequency details than RegNeRF. Furthermore, Figure 3.4 shows that compared with baseline TensoRF [43], our methods not only render a realistic appearance but also reconstruct more accurate depth.

3.4.3 Evaluation on LLFF dataset

We also demonstrate our method for novel view synthesis from sparse inputs using real forward-facing datasets (LLFF benchmark) [10]. The LLFF dataset consists of 8 scenes. Each scene includes 20-62 images with a resolution of 1008×756. This dataset has a different camera distribution from the DTU dataset. We follow standards from [4], [52] to choose every 8th image as testing views and sampling training views from the remaining images.

Specifically, similar to baseline [52], we first compare with generalizable approaches, including SRF [129], PixelNeRF [56], and MVSNeRF [55] using its pre-trained model from

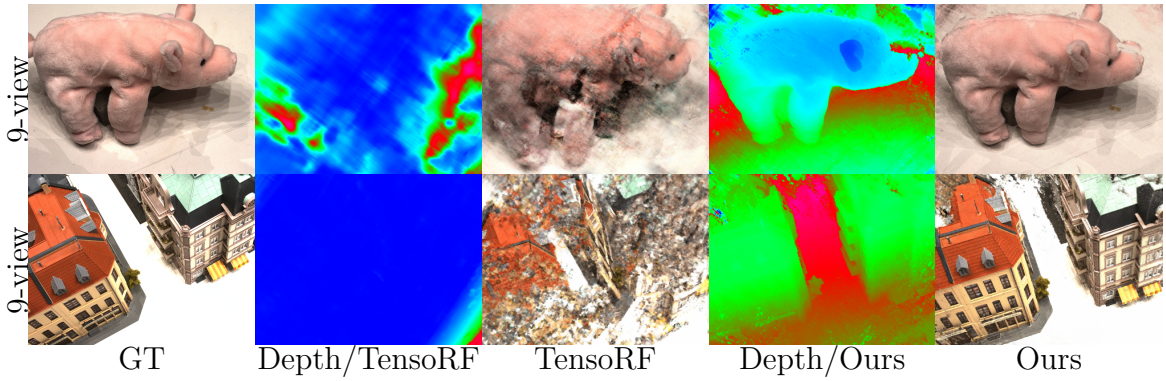


Figure 3.4 – Qualitative comparison of depth and appearance with TensoRF [43] from 9 input views on the DTU dataset [32].

3/6/9 input views. Table 3.2 also demonstrates the per-scene optimization evaluations of generalizable methods [55], [56], [129] from 3/6/9 input views. More importantly, we also evaluate recent representative methods relating to sparse neural radiance fields. As shown in Table 3.2, our method not only outperforms baseline TensoRF [43], generalizable methods [55], [56], [129], but also achieves better results than recent sparse NeRF methods in most cases, including InfoNeRF [53], DietNeRF [50] and RegNeRF [52].

Figure 3.5 demonstrates qualitative comparison with baselines from three input views. Compared with the baseline [43] and sparse NeRF method [52], our method generates a more realistic appearance with fewer geometry errors.

Method	Setting	PSNR↑			SSIM↑			LPIPS↓		
		3-view	6-view	9-view	3-view	6-view	9-view	3-view	6-view	9-view
SRF	Tested on LLFF without Optimization	12.34	13.10	13.00	0.250	0.293	0.297	0.594	0.594	0.605
PixelNeRF		7.93	8.74	8.61	0.272	0.280	0.274	0.682	0.676	0.665
MVSNeRF		17.25	19.79	20.47	0.557	0.656	0.689	0.356	0.269	0.242
SRF	Tested on LLFF with Per-scene Optimazation	17.07	16.75	17.39	0.436	0.438	0.465	0.529	0.521	0.503
PixelNeRF		16.17	17.03	18.92	0.438	0.473	0.535	0.512	0.477	0.430
MVSNeRF		17.88	19.99	20.47	0.584	0.660	0.695	0.327	0.264	0.244
mip-NeRF	Tested on LLFF with Per-scene Optimization	14.62	20.87	24.26	0.351	0.692	0.805	0.495	0.255	0.172
TensoRF		11.85	12.95	13.72	0.224	0.275	0.312	0.563	0.559	0.542
DietNeRF		14.94	21.75	24.28	0.370	0.717	0.801	0.496	0.248	0.183
RegNeRF		19.08	23.10	24.86	0.587	0.760	0.820	0.336	0.206	0.161
Ours		19.21	23.21	24.73	0.590	0.765	0.811	0.329	0.201	0.159

Table 3.2 – Comparison of the average PSNR, SSIM, and LPIPS of reconstructed images in the LLFF [134] dataset, using 3/6/9 views for training. The higher the better for both PSNR and SSIM. The lower the better for LPIPS [133]. The color represents the performance of ranking, the darker the better.

Figure 3.5 – Qualitative comparison on LLFF dataset [134] with 3 training views, respectively. Compared with baselines, our methods generate realistic appearances with more details and fewer geometry errors.

3.4.4 Ablations and analysis

We demonstrate an ablative analysis showing the importance of each part proposed in our method. Specifically, we add each regularization individually, including only with global patch regularization (L_g), only with local patch regularization (L_l), only with depth regularization (L_d), and a combined full model. Table 3.3 shows the numerical improvement of each part of our proposed methods from 3/6 input views on the DTU scene(scan 41). In addition, figure 3.6 demonstrates qualitative comparisons for 3-view and 6-view cases. Compared with the baseline [32], each regularization in our proposed method improves the results quantitatively and qualitatively. More importantly, the full model ($L_g + L_l + L_d$) achieves the state of art performance.

3.5 Conclusion

This chapter introduces a novel approach to improve neural radiance fields(NeRF) from sparse RGBD inputs. Specifically, we propose three different regularizations, including geometry regularization, appearance regularization, and depth regularization. We evaluate our method quantitatively and qualitatively on a real benchmark DTU dataset. Compared

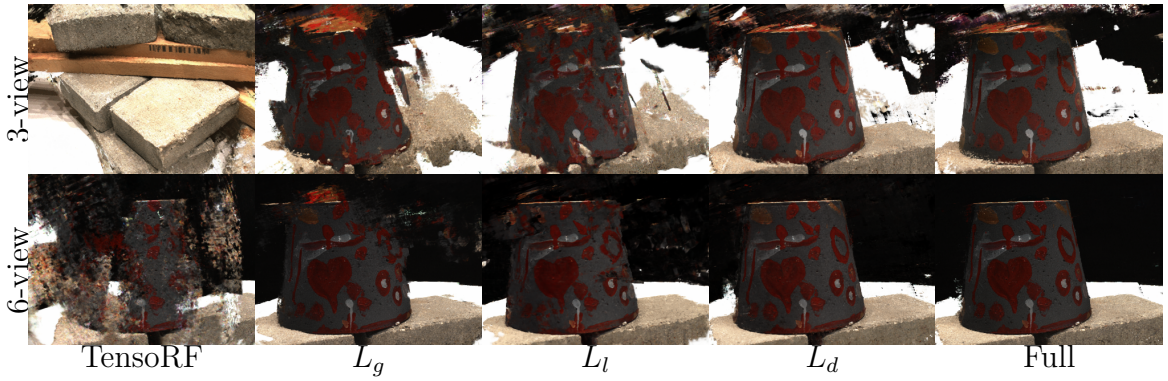


Figure 3.6 – Qualitative ablations of different regularization from 3/6 input views on the DTU dataset [32].

Method				PSNR↑		SSIM↑		LPIPS↓	
Baseline [43]	L_g	L_l	L_d	3	6	3	6	3	6
✓				12.45	13.34	0.472	0.498	0.455	0.453
✓	✓			16.97	20.65	0.614	0.706	0.295	0.235
✓		✓		17.28	19.46	0.635	0.691	0.357	0.241
✓			✓	21.09	23.27	0.695	0.752	0.332	0.127
✓	✓	✓	✓	21.80	24.16	0.760	0.801	0.132	0.097

Table 3.3 – Quantitative ablations of different regularization approaches from 3/6 inputs on the DTU dataset [32].

with existing approaches [43], [50], [52], [55], [56], [59], [60], [126], our work achieves state of art performance across different metrics. However, one of the limitations is that it requires accurate depth information, e.g. sensor depth. Future work could explore how to replace the sensor depth with a monocular depth estimated by networks. More research includes improving the neural radiance field together with implicit surface reconstruction from sparse RGBD inputs.

Chapter 4

Few Shot Neural Light Field based Novel View Synthesis

4.1 Introduction

In this thesis, we aim to build a novel view machine that could generalize novel views outside the training data. Given a few input-calibrated color images at test time, we expect our method to generate novel target images given new query viewpoints. We are also interested in fast rendering novel view synthesis that can generate novel views in a single forward pass without test time optimization.

The recently popularized implicit neural representations offer numerous advantages in modeling 3D shape [74], [135] and appearance [4], [33] in comparison to their traditional alternatives, while being conditionable using e.g. encoders [56], [75], [136] and meta-learning [137]. In particular, neural radiance fields (NeRF) [4] provide impressive novel view synthesis performances from dense input images. When coupled with convolutional encoders (e.g. [55], [56]), they can additionally achieve across-scene generalization and test-time optimization free inference, in addition to reconstructing from fewer inputs. However, the rendering of these methods is expensive. They require sampling hundreds of 3D points along each target pixel ray, evaluating densities and view-dependent colors for all these points through a multi-layer perceptron (MLP), and building the final image through the volumetric rendering of all the samples' colors and densities. Multi-scale sampling is also necessary to achieve satisfactory results.

To reduce this complexity, we propose to use an implicit neural network operating in ray space rather than the 5D Euclidean $\times$ direction space, thus alleviating the need for per-ray multi-sample evaluation and physical rendering. For a given target pixel, an MLP (i.e. light field network) maps its ray coordinate and ray features to the color directly. Key to efficient generalization, and differently from [6], we build the ray features by computing and merging 3D convolution feature volumes from the input images. These

features are then rendered volumetrically into a coarse ray feature image. The method is fully differentiable and trained end-to-end.

We evaluate our method on both synthetic (ShapeNet [138]) and real multi-view stereo data (DTU [32]). In the few-shot novel view optimization-free setting, we outperform comparable convolutional methods, including our 3D convolutional baseline, and extend them to real complex data. Our proposed method achieves competitive results compared to generalizable encoder-decoder NeRF-based models while providing orders of magnitude faster rendering.

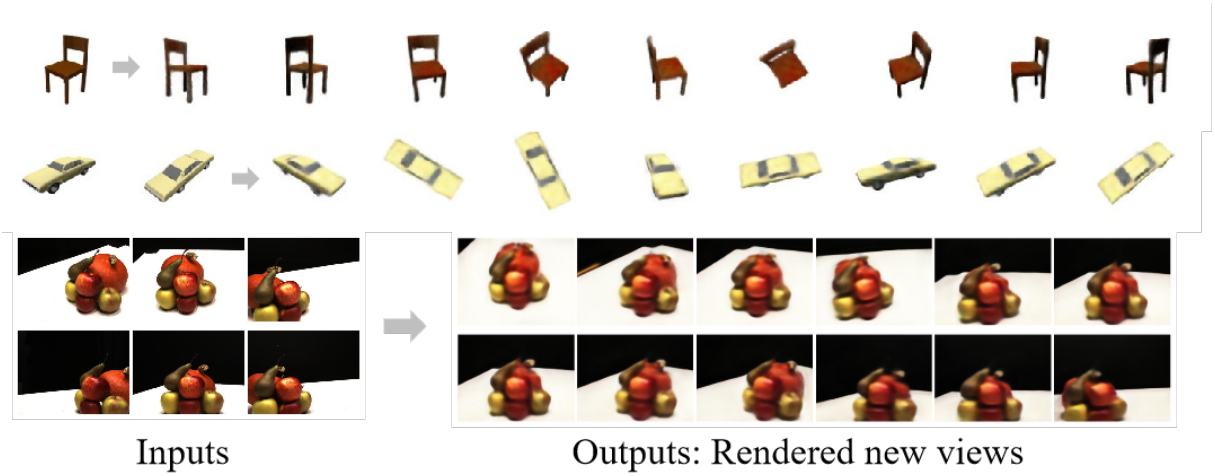


Figure 4.1 – Our method enables the fast generation of novel views from sparse input images without 3D supervision in training. We generate the above novel views for objects (ShapeNet dataset [138]) and a scene (DTU dataset [32]) never seen at training.

4.2 Related work

Few Shot Radiance Fields Valline NeRF demonstrated a photo-realistic rendering ability by representing a scene as a neural radiance field at the expense of well-calibrated dense views for training. The rendering quality of vanilla NeRF [4] drops significantly with fewer inputs due to the lack of geometry regularization. Recent researchers are keening to tackle this issue in various ways (e.g. [42], [49]–[59], [139]), including utilizing image encoders ([50], [55], [56]), exploiting additional depth information [51], [59], or using different regularization techniques [52], [53], etc.

Specifically, one line of work has explored various regularization-based methods for NeRF. These methods introduce extra supervision on unobserved viewpoints, such as

using pre-trained visual encoders like CLIP-ViT [50], or rendered depth or density from sampled patches [52], [53]. These approaches improve the generation ability of NeRF, especially from sparse inputs, at the expense of training efficiency and computation complexity or generalization ability across scenes. Other lines of approaches propose to improve the rendering quality of neural radiance field using depth information ([51], [59], [60], [140], [141]) from sensors, colmap [131] and so on. Those methods prove that accurate depth maps contribute to higher rendering quality and speed. However, obtaining precise depth information can be expensive or computationally intensive. To further enhance NeRF's generation ability and generalization across scenes, another line of approaches augment NeRFs with 2D ([56], [61], [62], [140]) or 3D [55] convolutional features extracted from input images. These methods offer forward pass prediction models, which do not require test-time optimization. However, they still need to evaluate hundreds of 3D query points per ray for inference, making them slow to train and render.

Taking inspiration from conditional-based NeRF method ([56], [61], [62]) and light filed network [6], we explore here a tangent strategy, consisting in bypassing 3D implicit radiance modeling altogether. Unlike PixelNeRF [56] and MVSNeRF [55] conditioning NeRF with local image features, we propose here a more efficient local conditioning mechanism for the light field network [6]. In addition, different from LFN utilizing hyper-network to condition network and lacking demonstration on real-world scenes, our proposed local ray conditioning approach enables real-world scene reconstruction and offers optimization-free inference.

Fast Neural Rendering To achieve photo-realistic renderings, original neural radiance fields (NeRF) [4] require a long time (hours or days) to converge. Large amounts of recent works target faster training and rendering speed and have explored different techniques ([42]–[47]), such as utilizing sparse voxel grids ([42], [47]), hash encoding [45], tensor decomposition [43], and light field networks [6].

Among those, Plenoctrees [42], [48] predicts radiance spherical harmonic coefficients instead of density and uses plenoxels to improve efficiency and learn view-independent radiance features. Instant NGP [45] incorporates hash encoding into NeRF's representation and accelerates training with multi-GPUs. TensoRF [43] models the radiance field as a 4D tensor and factorizes it into multiple compact low-rank tensor components for real-time training and rendering. These methods improve training and inference speed and offer better rendering quality than original NeRF [4]. However, most of those methods

still concentrate on representing a single scene and require dense, well-calibrated views for training. More and more research may be needed to explore how to render novel views across scenes with higher efficiency, such as feature domain adaptation, multi-task learning, or meta-learning, to better generalize to unseen scenes and data. Our work explores feed-forward optimization-free inference of a generalizable model while maintaining high computational efficiency.

Light Field Network Light field networks have shown great promise in generating high-quality 3D visualizations, particularly for applications such as virtual reality, augmented reality, and image and video processing ([63]–[67]). There are types of light field networks, including convolutional neural networks (CNNs) ([63], [65]) and recurrent neural networks (RNNs) ([66], [68]–[70]). Recent works Combine light field networks with neural rendering, which could lead to improved rendering quality, as it allows for more accurate modeling of complex scenes and better handling of reflections. However, challenges still remain, particularly in dealing with large-scale scenes and generating high-quality, photo-realistic images.

To address these challenges, researchers continue to progress in the cross-domain of neural light fields and rendering. For instance, Wang et al. [64] distill a neural radiance field to a neural light field to produce a compact and lightweight model that can generate high-quality light field data with fewer computational resources. However, the quality of the generated light field data may not be as high as that produced by the original NeRF [4], especially in complex lighting conditions and scene geometries. To model complex scenes' appearance and geometry accurately, Suhail1 et al. [70] combines the light field with epipolar geometry to represent a novel view dependently, and Attal et al. [69] propose a ray-space embedding network to map the 4D ray-space into an intermediate interpolable latent space to learn a neural light field. While generating novel views across different scenes still remains a challenge for the above works ([64], [69], [70]). Sitzmann et al. [6] introduced a new implicit representation for modeling multi-view appearance, which uses a neural light field function to directly map rays to their colors without the need for physical rendering such as volumetric rendering, i.e., volumetric rendering [4]. Though this method has shown promising results on synthetic data, it has not yet been demonstrated on data containing large images and real-world scenes, such as those found in the DTU and LLFF datasets. Furthermore, it was implemented in the auto-decoding setup, which requires test-time optimization. It uses a hyper-network for conditioning, making it expensive to

scale to larger images in computing and memory.

The later work [68] learns a light field representation across different scenes by leveraging an epipolar geometry transformer and directly predicting the color of a target patch at the expense of many training views. Specifically, MVSNeRF [55] demonstrate realistic rendered novel views by using only 3 reference views as inputs, whereas [68] requires 10 input views for its best performance demonstrated in their paper. Unlike the above method, our approach achieves competitive results with fewer inputs, which utilizes a deep convolutional network to extract features, aggregate those features with learnable weights, and composite and render those feature volumes to achieve generalizable representation across scenes.

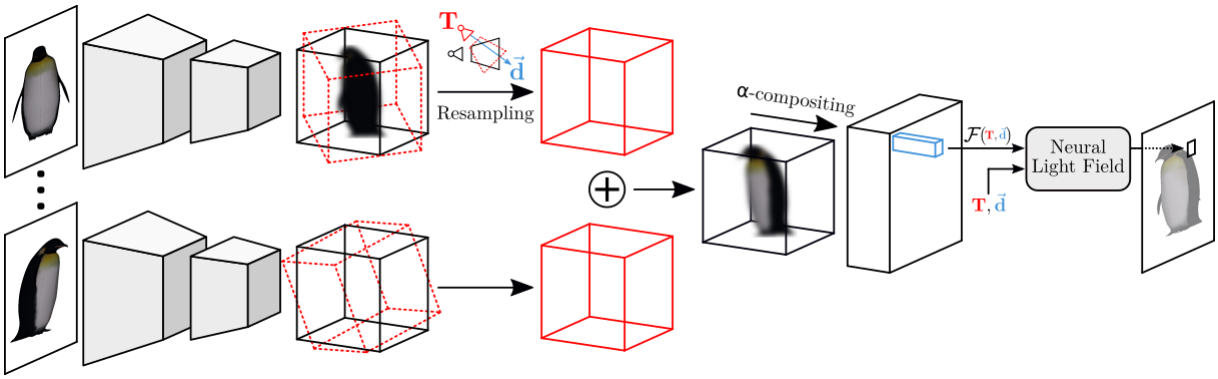


Figure 4.2 – Overview of our method. Given an input image, a 3D feature volume is built with a series of convolutional neural networks. The volume represents features inside the input view frustum. Given a targeted view, these features are resampled into a volume representing the target view frustum. Target feature volumes originating from different input views are aggregated using learnable weights. An image of ray features is produced by rendering the target aggregated feature volume with α-compositing. Finally, the light field network maps a ray stemming from a target camera origin T and spanning a direction d, along with its convolutional feature $\mathcal{F}$, to the corresponding pixel color of the target image.

4.3 Methodology

A summary of our method is illustrated in figure 4.2. Given one or few images $\{I_i\}$ of a scene or an object with their known camera parameters, i.e. camera poses $\{R_i, T_i\}$, $R_i \in SO(3)$, $T_i \in \mathbb{R}^3$, and intrinsics $K \in \mathbb{R}^{3\times3}$, our goal is to generate images $\{I_t\}$ for novel target views, i.e. new camera poses $\{R_t, T_t\}$. We are interested in generalization to scenes and objects unseen at training, and target views beyond input view interpolation,

using merely sparse input views. We also seek fast rendering time, and we do not assume the availability of any segmentation masks neither at training or testing.

To this end, we propose a single forward pass inference deep learning method, that uses a deep neural network to map a ray r in a projective pinhole camera model, to its desired color in the target view image c_r, using an implicit neural representation i.e. a neural light field network f. This network is conditioned with ray features $\mathcal{F}_r$, i.e. $c_r = f(r, \mathcal{F}_r)$. The ray features are generated through the volumetric rendering of explicit 3D convolutional features built from the input images. In the remaining section, we present the components of the two stages of our method, namely the convolutional stage, and the neural light field network.

4.3.1 Feature volume

Following seminal work (e.g. [27], [28], [55]), we build an explicit volume of features from an input image I_i using a fully convolutional neural network E consisting of a succession of a 2D convolutional U-Net and several 3D convolutional blocks:

$$F_i = E(I_i), \tag{4.1}$$

where $I_i \in \mathbb{R}^{H \times W \times 3}$, H and W are the height and width of the input RGB image, and $F_i \in \mathbb{R}^{H_V \times W_V \times D \times C}$, H_V, W_V, D and C being respectively the height, width, depth, and the number of channels of the 3D feature volume. The feature volume F_i is expected to encode 3D shape and appearance information of the captured object or scene in the view frustum associated with the input image and is hence aligned pixel-wise with the latter. As we will show in the following sections, this volume will encode prediction confidence, volume density [4], colors, and more generic appearance features. One limitation of these features being modeled explicitly and not implicitly as in NeRF [4] based methods is that they cannot be view direction dependent.

4.3.2 Feature resampling

Using the input feature volume F_i aligned with the input image, we would like to create a feature volume $F_{t/i}$ aligned to the target image, that could be used subsequently to render a target feature image given the target camera pose $\{R_t, T_t\}$. Following the principles of volumetric rendering ([4], [142]), in order to recreate a target image of dimensions

$H_V \times W_V$, we need to evaluate N points $\{p^z_{u,v}\}_{z=1}^N$ along each ray $r_{u,v}$ with direction $d_{u,v}$, where $u \in [\![1, H_V]\!]$ and $v \in [\![1, W_V]\!]$:

$$d_{u,v} = R_t K^{-1} \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} + T_t, \quad p^z_{u,v} = T_t + t_z \frac{d_{u,v}}{||d_{u,v}||}, \tag{4.2}$$

where $t_z \sim \mathcal{U}\left[z_n + \frac{z-1}{N}(z_f - z_n), z_n + \frac{z}{N}(z_f - z_n)\right]$ following [4], z_n and z_f being the depth near and far bounds of the visual frustum. K is the intrinsic camera matrix. The target volume $F_{t/i}$ is obtained then as the resampling of input volume F_i with trilinear interpolation, using points $\{p^z_{u,v}\}$ aligned rigidly to the input camera coordinate frame:

$$F_{t/i}(u, v, z) = F_i(R_i^\top (p^z_{u,v} - T_i)), \tag{4.3}$$

where $F_{t/i} \in \mathbb{R}^{H_V \times W_V \times N \times C}$ and $\{R_i, T_i\}$ is the input camera pose. In practice, we normalize the aligned points' coordinates prior to sampling as F_i is assumed to represent features in the input view normalized device coordinate (NDC) space. We use the NDC parametrization for optimal spatial exploitation of the input feature volume F_i and generalization across objects and scenes with different scales and datasets with different camera settings (e.g. intrinsics, z_n, z_f).

4.3.3 Feature aggregation

As different input views provide different information about the observed scene, we merge subsequently the 3D features obtained from the various inputs. We note that all target feature volumes $\{F^k_{t/i}\}_k$ provided by input images $\{I^k_i\}_k$ are represented in the same target view camera coordinate frame. A naive merging strategy would be to simply average these volumes element-wise. However, for a given 3D location in the target view frustum, different input views contribute appearance information with varying confidence, based on the visibility/occlusion of this spatial location in the input views. In order to emulate this principle, and inspired by attention mechanisms, we propose to learn a 3D confidence measure per input view in the form of a weight volume $W_i \in \mathbb{R}^{H_V \times W_V \times D}$.

This volume is obtained as one of the channels of the input volume features $W_i = F_i(1)$ (i.e. $W_{t/i} = F_{t/i}(1)$). As this confidence volume depends naturally on the input image and the relative camera pose of the target with respect to the input, similarly to [24], we

append these relative poses to the input image pixel values as additional input to the encoder E. After resampling the input features $\{F_i^k\}_k$ into the target ones $\{F_{t/i}^k\}_k$, we use the resampled weights $\{W_{t/i}^k\}_k$ normalized with Softmax across the input views to compute a weighted average of the target volumes:

$$F_t = \sum_k \underset{k}{\text{Softmax}}(W_{t/i}^k) F_{t/i}^k([\![1, C]\!]), \tag{4.4}$$

where index k is over the number of input views, and $F_t \in \mathbb{R}^{H_V \times W_V \times N \times C-1}$. Let us recall that this tensor represents features of N points, within $[z_n, z_f]$ depth-wise, for all rays associated with a target image of dimension $H_V \times W_V$. This aggregation allows our method to use an arbitrary number of input views at both training and testing. In the single input case, we note that $F_t = F_{t/i}([\![1, C]\!])$.

4.3.4 Feature rendering

Following volumetric rendering ([4], [142]), we generate a target feature image $\tilde{\mathcal{F}}$ for a given target view differentiably using α-compositing of the target feature volume F_t along the depth dimension. To this end, we assume one of the target feature channels to represent volume density $\sigma = F_t(1) \in \mathbb{R}^{H_V \times W_V \times D}$ [142]. We recall that the dimensions of tensor F_t span the pixels of the target feature resolution $H_v \times W_v$ in the first two dimensions, and N points sampled along each ray for the third dimension. The rendered target feature image then writes:

$$\tilde{\mathcal{F}} = \sum_{z=1}^{N} T_z \alpha_z F_t([\![1, C-1]\!]), \tag{4.5}$$

$$T_z = e^{-\sum_{j=1}^{z-1} \sigma(j)\delta_j}, \quad \alpha_z = 1 - e^{\sigma(z)\delta_z}, \tag{4.6}$$

where T represents transmittance, $\delta_z = t_{z+1} - t_z$ and $\tilde{\mathcal{F}} \in \mathbb{R}^{H_V \times W_V \times C-2}$. In order to reduce the memory cost and increase the rendering speed of our method, the size of the rendered feature image is chosen to be lower than the size of the target image resolution, i.e. $H_V = H/4$ and $W_V = W/4$.

4.3.5 Neural light field

At this stage, the convolutional rendered features produce a low-resolution feature image representative of all rays making up the target view. We propose to learn a light field function f, which performs both upsampling and refinement of the result of the convolutional first stage of our method. This implicit neural network maps rays of the target image to their colors, while being conditioned on ray features extracted from the convolutional rendered features.

Given a ray $r_{u,v}$ with direction $d_{u,v}$ corresponding to the target image pixel coordinates (u, v), with $(u, v) \in [\![1, H]\!] \times [\![1, W]\!]$, we encode rays using Plücker coordinates similarly to Sitzmann et al. [6]:

$$r_{u,v} = \frac{(d_{u,v}, T_t \times d_{u,v})}{||d_{u,v}||}, \tag{4.7}$$

where $r_{u,v} \in \mathbb{R}^6$. This representation ensures a unique ray encoding when the origin T_t moves along direction $d_{u,v}$. We recall that the expression of $d_{u,v}$ as a function of the target camera pose $\{R_t, T_t\}$ can be found in equation 4.2.

The feature $\mathcal{F}_{u,v}$ of a ray $r_{u,v}$ at the final image resolution $H \times W$ is obtained from the lower resolution rendered feature image $\tilde{\mathcal{F}} \in \mathbb{R}^{H_V \times W_V \times C-2}$ through a learned up-samplings. Specifically, the rendered feature image undergoes two successive 2D convolutions and up-samplings to produce a feature image at the desired resolution $\mathcal{F} \in \mathbb{R}^{W \times H \times C-2}$. The final target RGB image $I_t = \{c_{u,v}\}_{u \in [\![1,H]\!], v \in [\![1,W]\!]}$ is predicted then from the concatenation of the ray coordinate and its feature with an MLP f accordingly:

$$c_{u,v} = f(r_{u,v}, \mathcal{F}_{u,v}). \tag{4.8}$$

Notice that while convolution equipped NeRF [4] methods (e.g. PixelNeRF [56], MVS-NeRF [55], GRF[61]) require querying $H \times W \times N$ 3D points through their implicit neural radiance fields, our light field network only needs to evaluate $H \times W$ rays, which enables our method offering faster rendering speed.

4.3.6 Network structure

Table 4.1 describes the detailed architecture of our convolutional network E. The main structure follows [28], we use a much smaller feature cube and add two extra up-sampling

layers. Table 4.2 shows the detailed architecture of our network f introduced in this section. Please note that the baseline work PixelNeRF [56] uses MLP with 512 channels in each layer.

Input Shape	**Output shape**	**Operation**	
$(3, H, W)$	$(52, H, W)$	1×1 Conv	Image
$(12, H, W)$	$(12, H, W)$	1×1 Conv	Relative Pose
$(52+12, H, W)$	$(64, H, W)$	1×1 Conv	2D Conv
$(64, H, W)$	$(64, H, W)$	2× ResBlock	
$(64, H, W)$	$(128, H/2, W/2)$	4×4 Conv,Stride2	
$(128, H/2, W/2)$	$(128, H/2, W/2)$	1× ResBlock	
$(128, H/2, W/2)$	$(128, H/4, W/4)$	4×4 Conv,Stride2	
$(128, H/4, W/4)$	$(128, H/4, W/4)$	1× ResBlock	
$(128, H/4, W/4)$	$(256, H/8, W/8)$	4×4 Conv,Stride2	
$(256, H/8, W/8)$	$(256, H/8, W/8)$	1× ResBlock	
$(256, H/8, W/8)$	$(128, H/4, W/4)$	4×4 Conv.T,Stride2	
$(128, H/4, W/4)$	$(128, H/4, W/4)$	2× ResBlock	
$(128, H/4, W/4)$	$(256, H/4, W/4)$	1×1 Conv	2D to 3D
$(256, H/4, W/4)$	$(512, H/4, W/4)$	1×1 Conv	
$(512, H/4, W/4)$	$(2048, H/4, W/4)$	1×1 Conv	
$(2048, H/4, W/4)$	$(32, 64, H/4, W/4)$	Reshape	
$(32, 64, H/4, W/4)$	$(32, 64, H/4, W/4)$	1×1 Conv	3D Conv
$(32, 64, H/4, W/4)$	$(32, 64, H/4, W/4)$	2× ResBlock	
$(32, 64, H/4, W/4)$	$(64, 32, H/8, W/8)$	4×4 Conv,Stride2	
$(64, 32, H/8, W/8)$	$(64, 32, H/8, W/8)$	2× ResBlock	
$(64, 32, H/8, W/8)$	$(32, 64, H/4, W/4)$	4×4 Conv.T,Stride2	
$(32, 64, H/4, W/4)$	$(32, 64, H/4, W/4)$	2× ResBlock	
$(31, 64, H/4, W/4)$	$(31, H/4, W/4)$	-	Rendering
$(30, H/4, W/4)$	$(30, H/2, W/2)$	1×1 Conv,Upsampling	Upsampling
$(30, H/2, W/2)$	$(30, H, W)$	1×1 Conv,Upsampling	

Table 4.1 – The architecture of network E used in this section.

Layer	Channels	Inputs
LR_0	30/256	$\mathcal{F}_{u,v}$
PE	6/78	$r_{u,v}$
LR_1	78/256	PE
LR_2	256/256	$LR_1 \odot LR_0$
LR_3	256/256	$LR_2 \odot LR_0$
LR_4	256/256	$LR_3 \odot LR_0$
LR_5	256/256	$LR_4 \odot LR_0$
LR_6	256/256	$LR_5 \odot LR_0$
$c_{u,v}$	256/3	LR_6

Table 4.2 – The structure of MLP is shown here. $r_{u,v}$ is the Plücker coordinate of the ray. $\mathcal{F}_{u,v}$ is the feature of the ray. $c_{u,v}$ is the target pixel color. PE is the positional encoding in [4]. $\odot$ is the element-wise product. All layers are linear with Relu activation, except the last layer which uses a Sigmoid.

4.3.7 Novel structure for feature extraction

In this subsection, we show an improved version of the framework to improve the generalization ability of light field networks on real-world datasets. A summary of our method is illustrated in Figure 4.3.

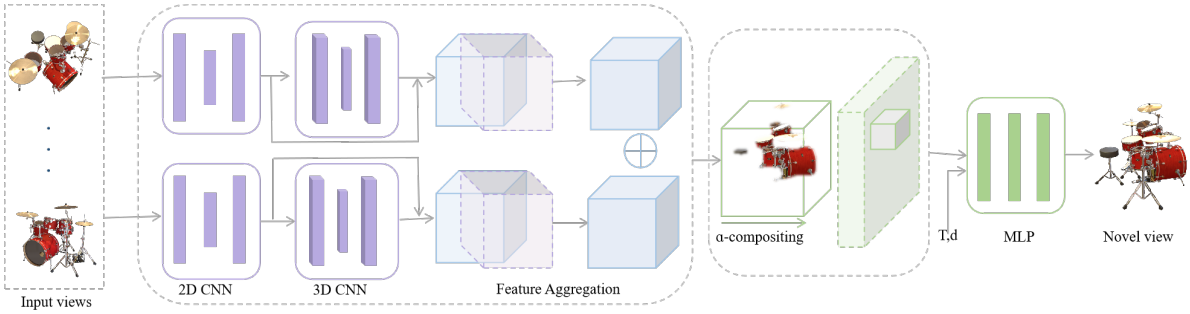


Figure 4.3 – Overview of our method. When presented with an input image, a 3D feature volume is constructed using a sequence of convolutional neural networks, visualized as a purple cube. This volume encapsulates the features found within the input view frustum. These features are then transformed into a volume representing the desired target view frustum, illustrated as a blue cube. These target feature volumes, originating from various input perspectives, are combined utilizing adaptable weights. The next step involves generating an image of ray features by rendering the aggregated target feature volume through an alpha-compositing process. To complete the process, the light field network is responsible for mapping a ray, emerging from the viewpoint of a target camera located at point T and extending along a specific direction d, along with its associated convolutional feature, to the corresponding pixel color within the target image.

Following similar work (e.g. [16], [27], [28], [55], [143]), our methods build multi-level features volume from an input image using a fully convolutional neural network 2D UNet networks U [144]:

$$F_i = U(I_i) \tag{4.9}$$

where $I_i \in \mathbb{R}^{h \times w \times 3}$, h and w being the height and width of the scale images. The feature extractor U used in this work produces three different scale feature volumes with scale value l, $l = 1, 2, 4$. In this case, for an original input RGB image size with width W and height H, the scaled input size is $h = \frac{H}{l}, w = \frac{W}{l}$. $F_i \in \mathbb{R}^{h_V \times w_V \times C}$, h_V, w_V, D and C being respectively the height, width, and the number of channels of multi-scale feature volumes. The feature maps with a different resolution provide different frequency information, i.e. with higher resolution capture more detailed information about the image, while the feature maps with lower resolution capture more global information.

After obtaining the input feature volume F_i aligned with the input image, the next step is to create a feature volume $F_{t/i}$ aligned with the target image. Following MVS stereo methods [16], [37], [143], we construct the 3D feature volume by warp extracted

2D feature volume:

$$F_{t/i}(u, v, z) = F_i \left(H_i(z) \begin{bmatrix} u & v & 1 \end{bmatrix}^T \right) \tag{4.10}$$

where $F_{t/i}(u, v, z)$ is the feature value at a pixel location (u, v) in the target image, obtained by warping the corresponding pixel in the source image at depth z using the homography warping $H_i(z)$ and extracting the feature value F_i at that location in the source image. The homography warping $H_i(z)$ is defined as:

$$H_i(z) = K_i R_i \left(I + \left(R_i^{-1} t_i - R_t^{-1} t_t \right) n^T R_t^{-1} / z \right) R_t^{-1} K_t^{-1} \tag{4.11}$$

where $[R_i, T_i, K_i]$ and $[R_t, T_t, K_t]$ are the camera intrinsic, rotation, and translation of the input view and target view, respectively, n that represents the normal direction of the plane on which the image lies,(u, v) is a pixel location in the reference view. In practice, we normalize the aligned points' coordinates prior to sampling as F_i is assumed to represent features in the input view normalized device coordinate (NDC) space. We use the NDC parametrization for optimal spatial exploitation of the input feature volume F_i and generalization across objects and scenes with different scales and datasets with different camera settings (e.g. intrinsics, z_n, z_f).

Next, we encode the warped feature volumes using several 3D convolutional blocks. It allows the network to capture both 2D and 3D features [16]. Specifically, following [143], the 3D convolutional blocks also predict per-pixel depth probability distributions, which are used to predict more accurate depth maps for finer feature volume sampling and rendering. As we have showed above, this feature volume will encode prediction confidence, volume density [4], colors, and more generic appearance features. This target feature volume can be subsequently used to render a target feature image given the target camera pose R_t, T_t.

As different input views provide different information about the observed scene, we merge subsequently the 3D features obtained from the various inputs. We note that all target feature volumes $\{F_{t/i}^k\}_k$ provided by input images $\{I_i^k\}_k$ are represented in the same target view camera coordinate frame. A naive merging strategy would be to simply average these volumes element-wise. However, for a given 3D location in the target view frustum, different input views contribute appearance information with varying confidence based on the visibility/occlusion of this spatial location in the input views. In order to emulate this principle, and inspired by attention mechanisms, we propose to learn a 3D

confidence measure per input view in the form of a weight volume $W_i \in \mathbb{R}^{H_V \times W_V \times D}$. This volume is obtained by encoding the input volume features with 3D conv ϕ:

$$\{W_{t/i}\}_k = \phi(\{F_{t/i}\}_k, \delta_P), \tag{4.12}$$

where $F_i \in \mathbb{R}^{H_V \times W_V \times D \times C}$, $W_i \in \mathbb{R}^{H_V \times W_V \times D \times 1}$ is the learnable weights of the target volumes. As this confidence volume depends naturally on the input image and the relative camera pose of the target with respect to the input δ_P, similarly to [24], we append these relative poses to the input image pixel values as additional input to the encoder U.

After re-sampling the input features $\{F_i^k\}_k$ into the target one $\{F_{t/i}^k\}_k$, we follow the method illustrated in the above subsection (4.3.2, 4.3.3, 4.3.4) to do feature aggregation, feature rendering and learning the neural light field.

4.3.8 Training objective

Our model is fully differentiable and we optimize all parameters of the model together, namely the convolutional network E and the light field network f jointly, by back-propagating a combination of a reconstruction loss L_r, perception loss L_p and depth loss L_d:

$$L = L_r + \lambda_p * L_p + \lambda_d * L_d. \tag{4.13}$$

where λ_p and λ_d are hyper parameters and set to 0.01 in experiments.

L_r is a L2 reconstruction loss between the final image I_t predicted by the light field network and the ground-truth I_t^{gt}:

$$L_r = ||I_t - I_t^{gt}||_2^2. \tag{4.14}$$

Except for the reconstruction loss L_r for each rendered pixel, we supervise the rendered image patches with perceptual loss:

$$Ł_p = ||\phi(I_t) - \phi(I_t^{gt})||, \tag{4.15}$$

where ϕ is the definition of VGG network(we use VGG16 in this work).

Additionally, we regularize the gradient of the low-resolution depth image $\tilde{d}_t$ rendered from the density volume σ of the first stage. Similarly to [145], we weight this cost with

an edge-aware term using the ground-truth image gradient:

$$\tilde{L}_d = \frac{1}{H_V \times W_V} \sum_{u,v} |\partial_u \tilde{d}_t| e^{-||\partial_u \tilde{I}_t^{gt}||} + |\partial_v \tilde{d}_t| e^{-||\partial_v \tilde{I}_t^{gt}||}. \tag{4.16}$$

The depth image $\tilde{d}_t$ can be expressed as a function of transmittance T and α values as follows:

$$\tilde{d}_t = \frac{1}{\sum_{z=1}^{N} T_z \alpha_z} \sum_{z=1}^{N} T_z \alpha_z t_z. \tag{4.17}$$

We note that the expressions of T and α are detailed in equation 5.4.

Figure 4.4 – The left side shows inputs and synthesized novel views on DTU dataset [16]. For given three input images, our method could generate novel views using a pre-trained model trained on different scenes of that dataset. The right side shows inputs and synthesized novel views on LLFF dataset [10]. For given three input images from a new dataset(e.g. LLFF dataset [10]), our method could generate novel views using a pre-trained model trained on a totally different dataset(e.g. DTU dataset [16]).

4.4 Experiments

4.4.1 Implementation details

We implemented our method with the PyTorch [146] framework on a Quadro RTX 5000 gpu. We optimize with the Adam [147] solver using learning rate 10^{-4} in training and 10^{-5} in fine-tuning. The depth of the convolutional feature volume is set to $D = 32$, and the number of channels $C = 32$. For the MLP of the light field network, we use 5 layers with a hidden dimension of 256, similar to NeRF [4]. For volumetric feature resampling, we use the coarse and fine sampling strategy similarly to previous work (e.g. [4], [56]), with $N = 64$ coarse samples and $N = 32$ fine samples.

4.4.2 Dataset

ShapeNet V2 Following the settings in GRF [61] and PixelNeRF[56], we evaluate first our synthesis results on the car and chair classes of dataset ShapeNet-V2[33]. Precisely, the cars amount to 2151 training objects, 352 validation objects and 704 testing objects, while the chairs count 4612 training objects, 662 validation objects, and 1317 testing objects. Hence the testing objects are not seen in training. In the training split, each object has 50 images with size 128×128. For testing, there are 251 views per object. Our model takes 1 or 2 fixed views as input and infers novel views for evaluation.

DTU dataset We demonstrate our method for novel view synthesis from sparse inputs using real-world multi-view datasets DTU benchmark [32]. Following the PixelNeRF [56] and MVSNeRF [55] experimental settings, the data is split into 88 training scenes and 16 testing scenes, each scene including 49 images with original resolution of 1600×1200. During testing, MVSNeRF selects 16 views as seen views and the remaining 4 views as test views. We note that this is a challenging scenario due to the complex illumination, geometry, and so on. In fact, the training scenes are limited, and the training and testing scenes do not share any semantic similarities as can be seen in figure 4.14. The lighting and backgrounds are also inconsistent between the scenes. Hence, this is a few-shot novel view synthesis task that demands considerable scene category generalization as well.

Real Forward-facing dataset The real forward-facing data (LLFF [10]) contains 8 scenes and each scene has 20-62 images with a resolution of 1008×756. This dataset has different camera distribution from the DTU dataset. We follow MVSNeRF's [55] experimental setup and select 3 center views as input.

Synthetic NeRF dataset The synthetic NeRF dataset [4] also has 8 scenes and images in each scene has the same resolution 800×800. We follow [55] for evaluation and fine-tuning settings.

4.4.3 Generalization on synthetic data

We first demonstrate our proposed first framework on synthetics dataset [33]. Table 4.3 shows a quantitative comparison of our method with the recent state-of-the-art in few-shot view synthesis. We report the peak signal-to-noise ratio (PSNR) and structural

similarity (SSIM) reconstruction metrics. We relay the numbers for methods TCO [20], WRL [21], dGQN [148] SRN [33] and GRF [61] as they were reported in [61]. We report the numbers of ENR [28] and PixelNeRF [56] from [56], and the numbers for LFN [6] from their paper. Figure 4.9 shows a qualitative comparison of these methods. We obtain the visualizations for PixelNeRF [56] and LFN [6] using their publicly available codes and models.

The results confirm that the 3D-aware ones (e.g. ENR, SRN, etc) outperform the 2D-based image-to-image novel view methods (e.g. TCO). Furthermore, 3D aware methods that use implicit 3D representations (e.g. PixelNeRF, GRF, SRN) outperform generally their counterparts relying on explicit 3D latent (e.g. ENR). Our method is hybrid, in that it uses an explicit 3D latent, combined with a 2D implicit representation.

Method	**PSNR(Cars)↑**		**SSIM(Cars)↑**		**PSNR(Chairs)↑**		**SSIM(Chairs)↑**	
	1-view	2-view	1-view	2-view	1-view	2-view	1-view	2-view
SRN[33]	20.72	22.94	0.85	0.88	22.89	24.48	0.91	0.92
LFN[6]	22.42	–	0.89	–	22.26	–	0.90	–
TCO[20]	18.15	18.41	0.79	0.80	21.27	21.33	0.88	0.88
WRL[21]	16.89	17.20	0.77	0.78	22.11	22.28	0.90	0.90
dGQN[148]	18.19	18.79	0.78	0.79	21.59	22.36	0.87	0.89
PixelNeRF[56]	23.17	25.66	0.90	0.94	23.72	26.20	0.91	0.94
GRF[61]	20.33	22.34	0.82	0.86	21.25	22.65	0.86	0.88
ENR[28]	22.26	–	–	–	22.83	–	–	–
Ours	22.31	23.82	0.87	0.91	22.52	24.10	0.90	0.92

Table 4.3 – Comparison of the average PSNR and SSIM of reconstructed images in the ShapeNet-V2 [33] dataset. The higher the better for both PSNR and SSIM. SRN [33] and LFN [6] require test time optimization.

Figure 4.5 and 4.5 show 360-degree novel view synthesis results in the ShapeNet-V2 [33] dataset. Figure 4.7 shows qualitative comparison with LFN [6] in the ShapeNet-V2 [33] dataset. As seen in the 4.7, compared with the baseline [6] on ShapeNet-V2 [33] dataset, our method could generate an object with fewer artifacts and more details based on only single input view. Moreover, Figure 4.10 and figure 4.11 show 360-degree novel view synthesis results on a single input view in the ShapeNet-V2 [33] dataset.

Figure 4.5 – Novel view synthesis of chairs from ShapeNet-V2 [33] given 2 input views using our method.

Figure 4.6 – Novel view synthesis of cars from ShapeNet-V2 [33] given 2 input views using our method.

Figure 4.7 – Qualitative comparison to [6] on novel view synthesis of chairs from ShapeNet-V2 [33] using a single input view. Compared with the baseline [6] on ShapeNet-V2 [33], our method could generate object with less artifacts and more details based on only single input view.

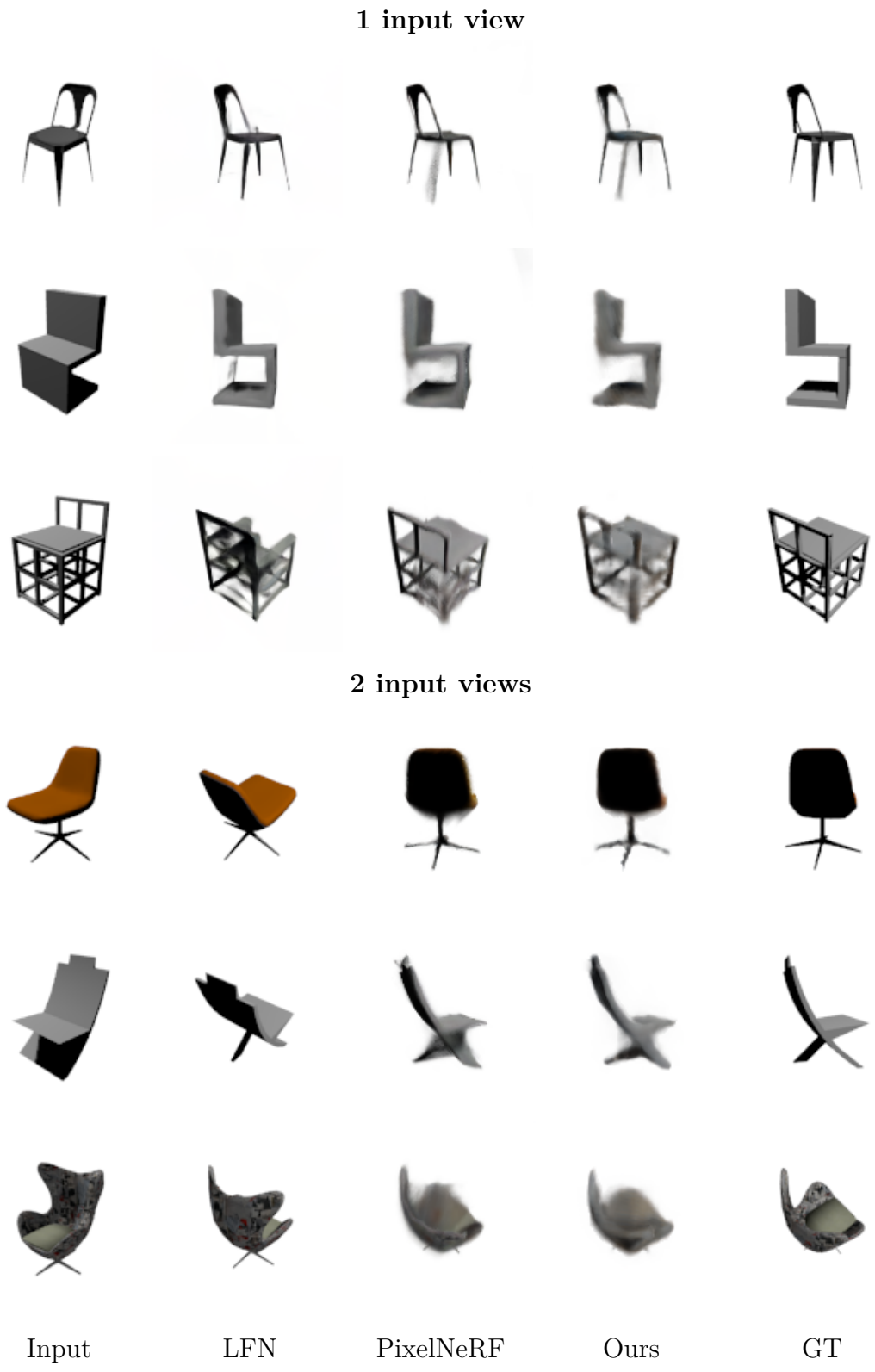


Figure 4.8 – Qualitative comparison of novel view synthesis of unseen chairs from a single and two input views on ShapeNet-V2 [33].

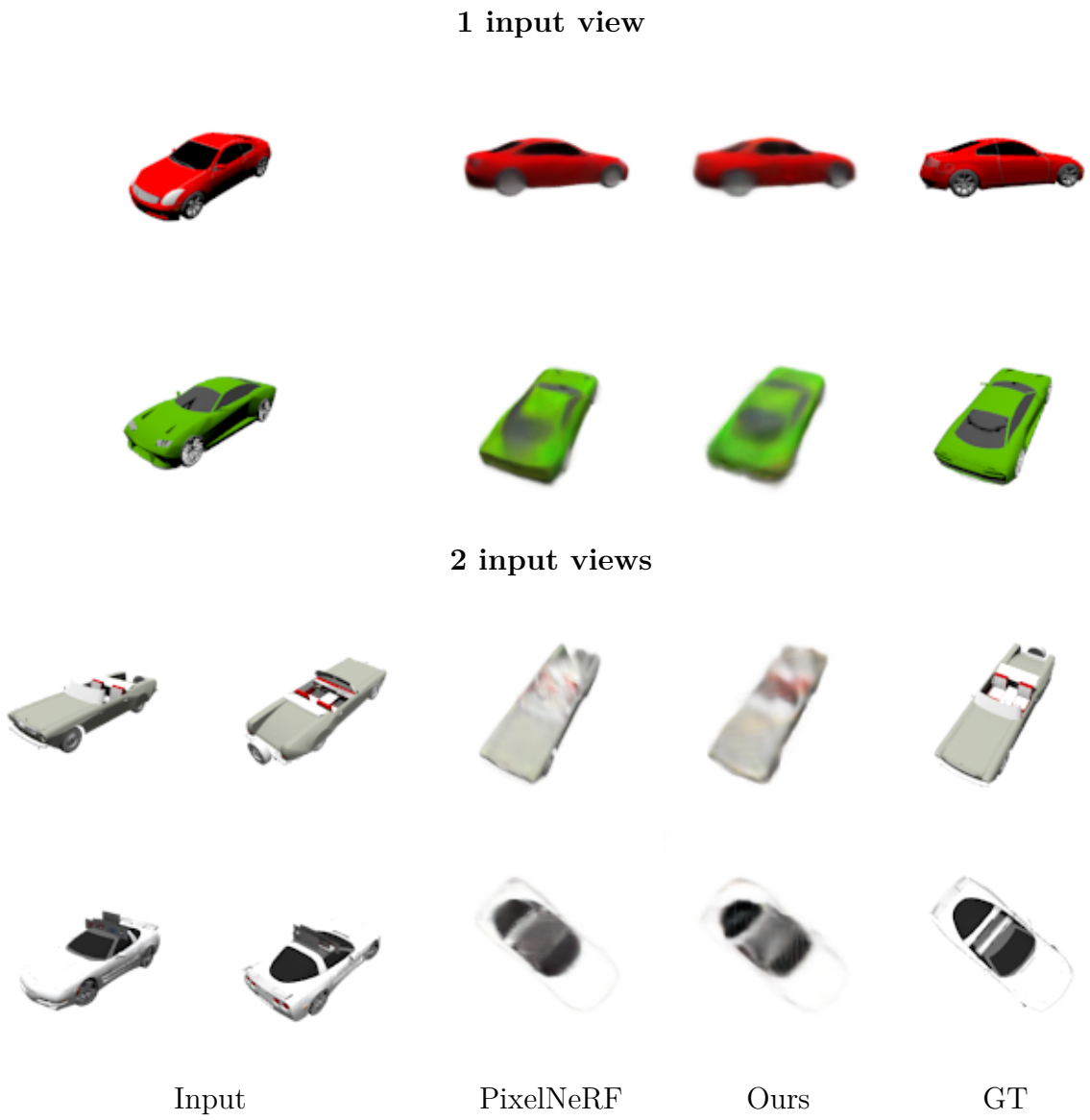


Figure 4.9 – Qualitative comparison of novel view synthesis of unseen cars from a single and 2 input views on ShapeNet-V2 [33]. Compared with the baseline [6] on ShapeNet-V2 [33], our method could generate an object with fewer artifacts and more details.

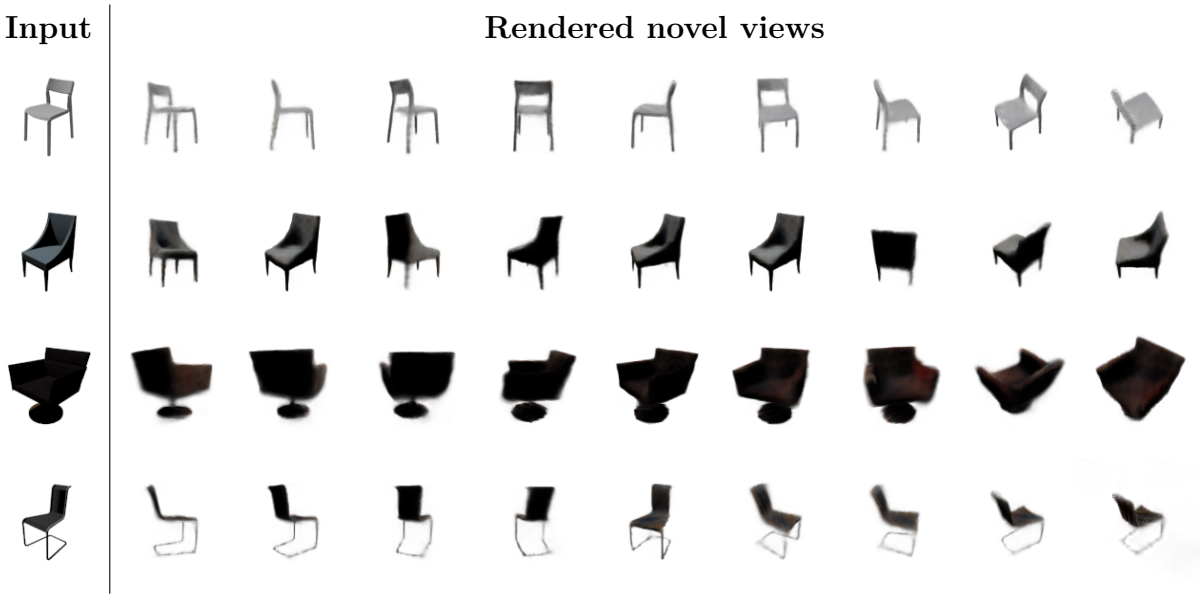


Figure 4.10 – Novel view synthesis of chairs from ShapeNet-V2 [33] given a single input view using our method.

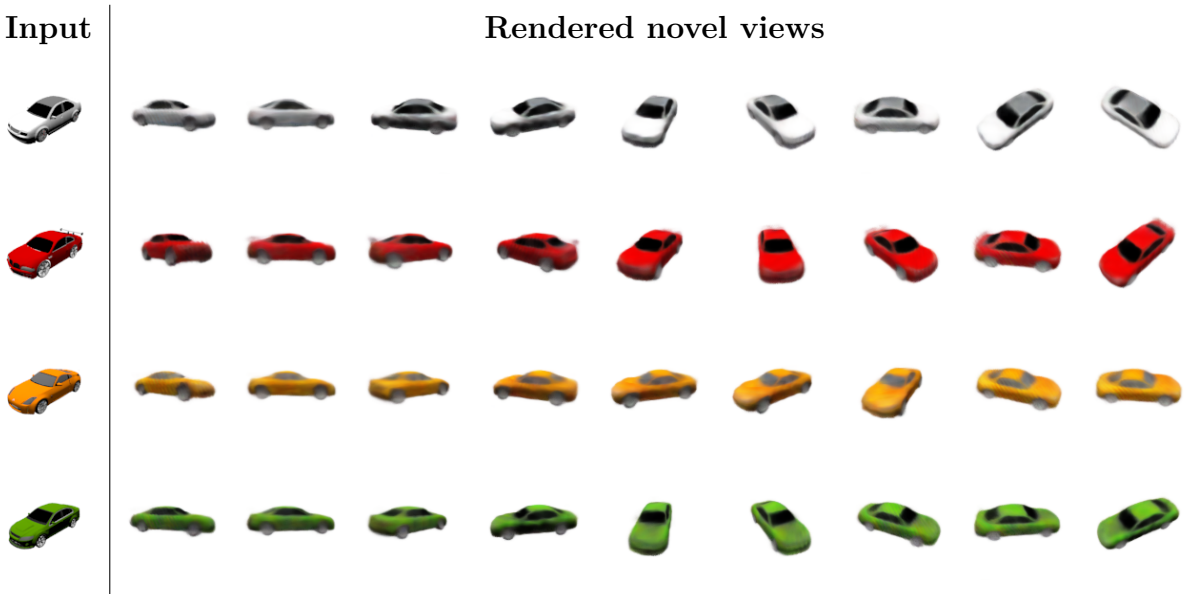


Figure 4.11 – Novel view synthesis of cars from ShapeNet-V2 [33] given a single input view using our method.

We then show the visual comparison in the ShapeNet-V2 [33] dataset. While our performance is generally close to LFN across the benchmark, we note that LFN requires auto-decoding test time optimization. It also requires training hypernetworks, which are

prohibitively expensive in computing and memory, thereby limiting the resolution of the reconstructed images. This hinders in turn the applicability of this method to real datasets with images larger than 128×128. Conversely, we demonstrate the ability of our method to model complex real scenes with bigger images using moderate computational resources, while providing optimization-free single forward pass prediction. We note that SRN requires test-time optimization as well.

4.4.4 Generalization on real data

In this section, we demonstrate that our method is capable of reconstructing novel views for real-world scenes unseen at training using the DTU dataset [32]. Following the PixelNeRF [56] experimental settings, the data is split into 88 training scenes and 16 testing scenes, each scene including 49 images with resolution 300×400. We note that the training scenes are limited, and the training and testing scenes do not share any semantic similarities as can be seen in figure 4.14. The lighting and backgrounds are also inconsistent between the scenes. Hence, this is a few-shot novel view synthesis task that demands considerable scene category generalization as well.

Table 4.6 shows a quantitative comparison of our method with the recent state-of-the-art in optimization-free few-shot view synthesis. We report the peak signal-to-noise ratio (PSNR), structural similarity (SSIM), and learned perceptual image patch similarity (LPIPS) reconstruction metrics, for the same 3 and 6 view inputs averaged across the same testing scenes. For a fair comparison, we report the performance of PixelNeRF [56] from their paper, and the numbers of methods MVSNeRF [55] and SRF [129] from RegNeRf [54], as the authors in the latter reproduce the performance of these methods in PixelNeRF's DTU setup. Figure 4.7 shows a qualitative comparison between our method and methods PixelNeRF [56] and MVSNerf [55] on synthesized views from testing scenes given the same inputs. We produce the results of PixelNeRF using their publicly available code and DTU model. For MVSNeRF, as their original model was trained on a different DTU setup, we fine-tune their model on the PixelNeRF DTU setup similarly to RegNeRF [54].

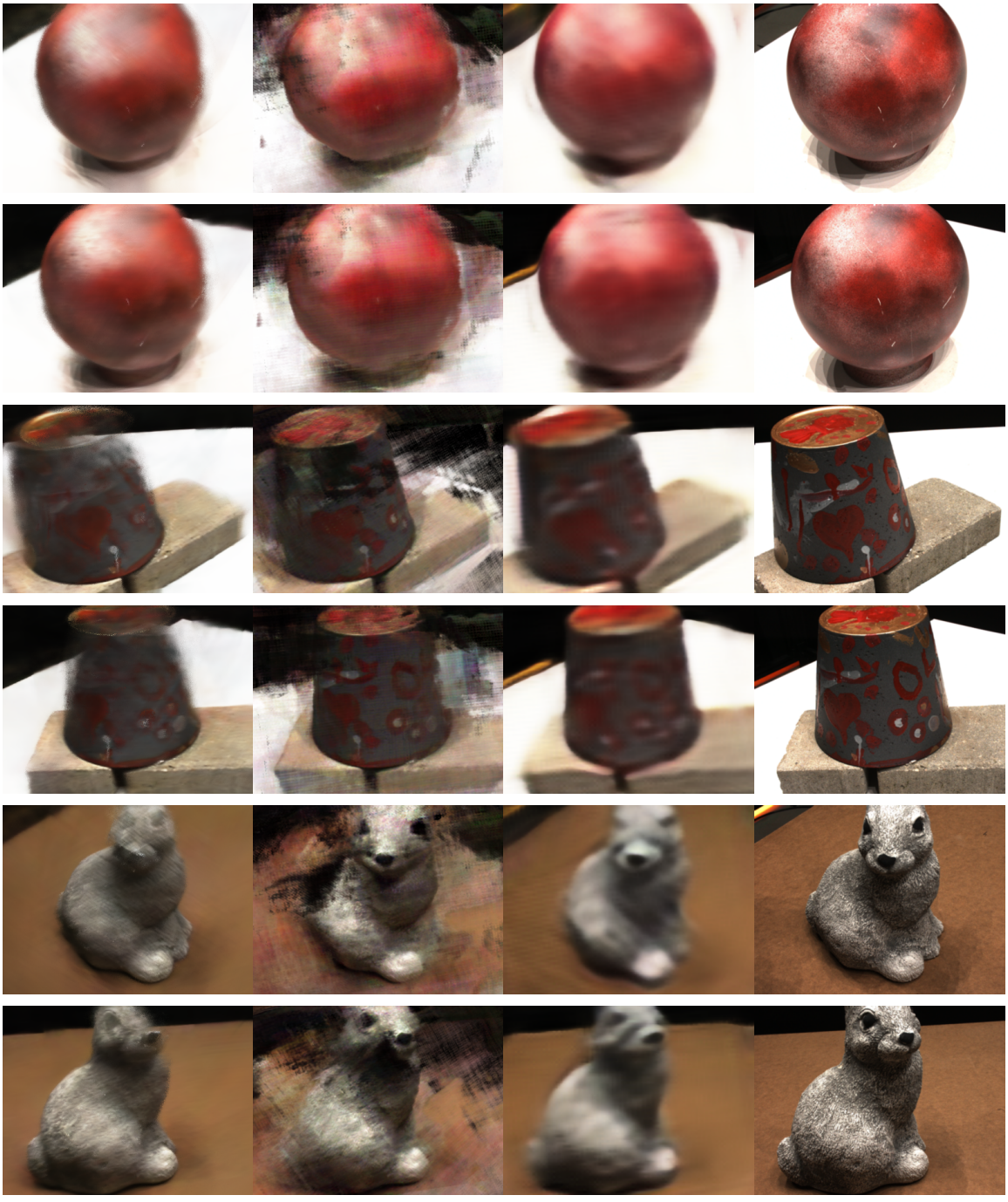
Method	**PSNR↑**		**SSIM↑**		**LPIPS↓**	
	3	6	3	6	3	6
PN[56]	18.74	21.02	0.618	0.684	0.401	0.340
MN[55]	16.33	18.26	0.602	0.695	0.385	**0.321**
Ours	**19.86**	**21.36**	**0.657**	**0.697**	**0.382**	0.355

Table 4.4 – Quantitative comparison of reconstructed images in the DTU [32] dataset without test time optimization.

Table 4.9 shows a quantitative comparison of our method with the recent few-shot novel view synthesis state-of-the-art with test time optimization. We outperform all methods in the PSNR and SSIM metrics, including conditional baseline PixelNeRF [56] and MVSNeRF [55], and unconditional baselines DietNeRF [50] and RegNeRF [52].
Figure 4.12 shows a qualitative comparison to MVSNeRF and PixelNeRF with 6 input views after finetuning. We obtain overall comparable performances with generalizable methods [55], [56]. We recall again that competition methods here require renderings that are orders of magnitude slower than ours.

Method	**PSNR↑**		**SSIM↑**		**LPIPS↓**	
	3	6	3	6	3	6
PixelNeRF [56]	17.33	21.52	0.548	0.670	0.456	0.351
MVSNeRF [55]	16.26	18.22	0.601	0.694	0.384	0.319
DietNeRF [50]	10.01	18.70	0.354	0.668	0.574	0.336
RegNeRF [52]	15.33	19.10	0.621	0.757	**0.341**	**0.233**
Ours	**20.72**	**22.60**	**0.677**	**0.786**	0.376	0.335

Table 4.5 – Quantitative comparison of reconstructed images in the DTU [32] dataset with test time optimization.



PixelNeRF MVSNeRF Ours GT

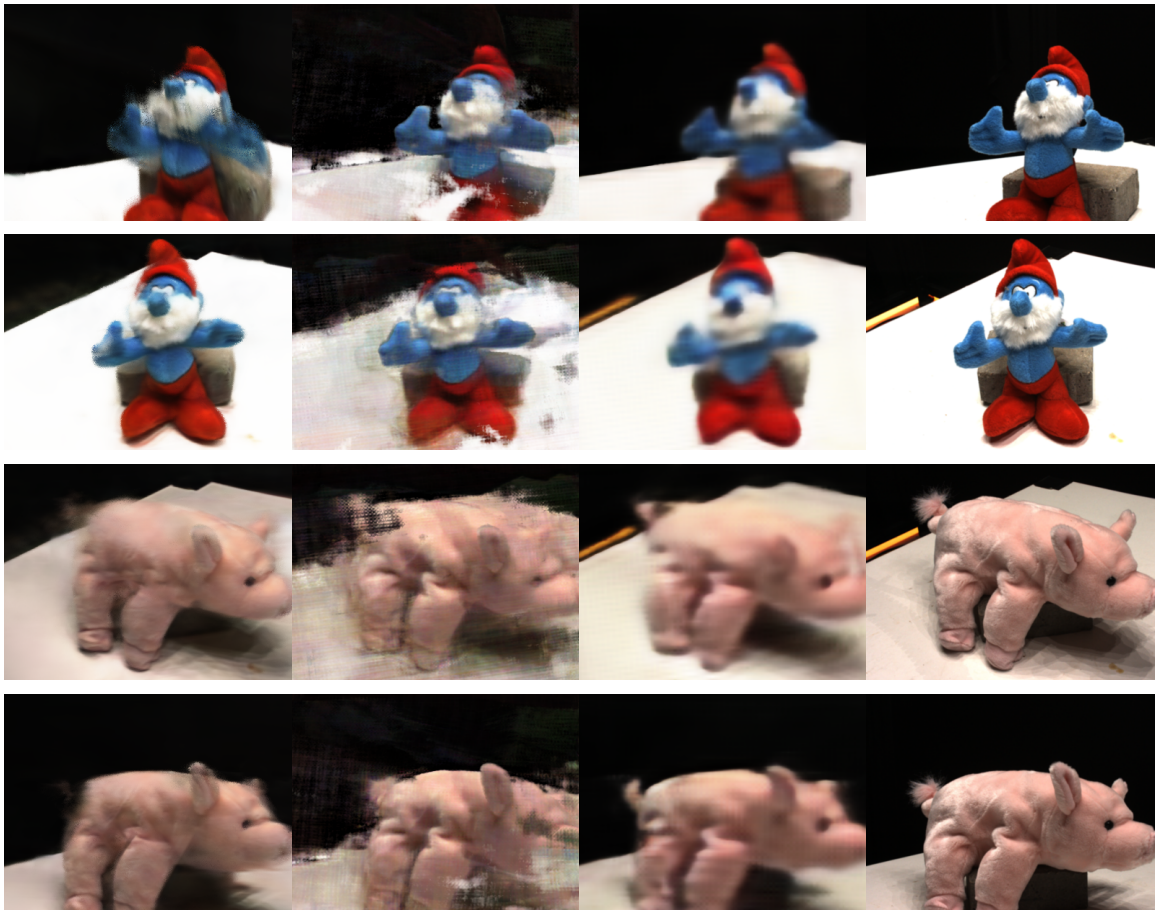


Figure 4.12 – Qualitative comparison of novel view synthesis of unseen scenes without test time optimization from 6 input views on the DTU dataset [32].

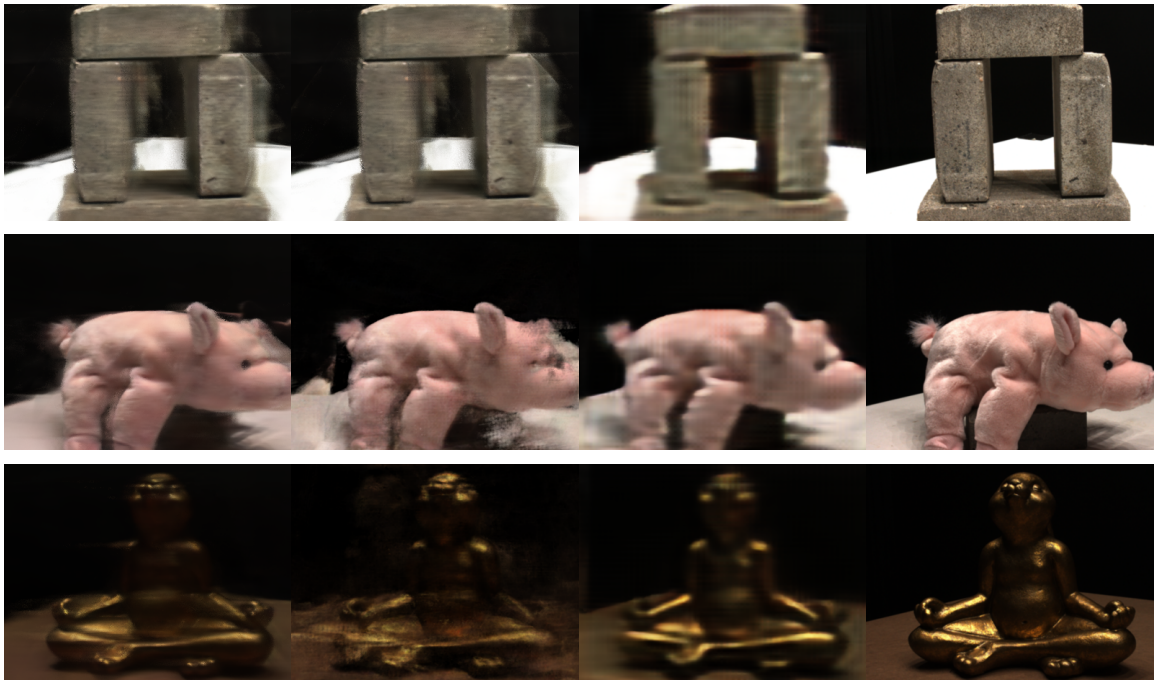
Although PixelNeRF and MVSNeRF are based on implicit radiance fields and hence require more evaluations and time for rendering, our implicit light field-based method provides competitive performances in comparison. In the single forward prediction setting, our method is overall second to PixelNeRF in PSNR, while providing much faster inference speed. As illustrated in the visual comparison in figure 4.12, we manage to reproduce the shape and appearance of the scene to a good extent and also recover from some of the competition's failures. Our method appears to be better in fact at preserving the coarser structure of the scene. Specifically, some elements of the ground truth that we manage to reproduce are not recovered by the competition, such as the pigtail in the last row, the background table's yellow lines, and the rabbit's eyes. Although we found MVSNeRF encounters multiple failures compared to PixelNeRF, it is apparent that NeRF

base methods (PixelNeRF and MVSNeRF) are able to produce some relatively higher frequency details, albeit at a considerably higher rendering cost.

Table 4.9 shows a quantitative comparison of our method with the recent few-shot novel view synthesis state-of-the-art with test time optimization. We report PSNR, SSIM, and LPIPS for the same 3 and 6 input views averaged over the same testing objects. All the method's numbers on the PixelNeRF DTU setup are reported as reproduced in RegNeRF [54]. Methods mip-NeRF [126], DietNeRF [50] and RegNeRF [54] are optimized per scene only, while PixelNeRF [56], MVSNeRF [55] and ours are trained on the DTU training set then finetuned per scene. Following the experimental setting in RegNeRF, only the input views were used for fine-tuning. Similarly to the finetuning of MVSNeRf and PixelNeRF, we reduce the learning rate from 10^{-4} to 10^{-5} and constrain the finetuning within 10k iterations for better performance. Figure 4.13 shows a qualitative comparison to MVSNeRf and PixelNeRF given 6 input views after finetuning.

In the 3-input view case, we outperform all methods in the PSNR and SSIM metrics. We obtain overall comparable performances with generalizable (PixelNeRF and MVS-NeRF) and single scene optimization (RegNeRF) NeRFs. Figure 4.13 shows that we can achieve relatively comparable results to the encoder-endowed NeRF approaches after optimization. We recall again that competition methods here require renderings that are orders of magnitude slower than ours.

PixelNeRF MVSNeRF Ours GT



PixelNeRF MVSNeRF Ours GT

Figure 4.13 – Qualitative comparison of novel view synthesis with test time optimization using 6 input views on the DTU dataset [32].

4.4.5 Generalization across different datasets

In this section, we demonstrate the results of our proposed second framework. Table 4.6, Table 4.7, and Table 4.8 shows a quantitative comparison of our method with the recent state-of-the-art on optimization free few-shot view synthesis from different datasets, respectively. We report PSNR, SSIM, and LPIPS for the 3 inputs averaged across the same testing scenes.

Specifically, we trained our model on the DTU dataset following MVSNeRF [55] and PixelNeRF [56] experimental settings. Table 4.6 demonstrate quantitative evaluations of PixelNeRF [56], IBRNet [62] and MVSNeRF [55] from their paper, and the numbers of SRF [129] from RegNeRF [54] paper, as the authors in the latter reproduce the performance of these methods in DTU setup. Although the above baselines are based on implicit radiance fields and hence require more evaluations and time for rendering, our implicit light field-based method provides more competitive performances in comparison. In the single forward prediction setting, our method achieves state of art performance, while providing faster inference.

Method	PSNR↑	SSIM↑	LPIPS↓
SRF [129]	15.32	0.671	0.730
PixelNeRF [56]	19.31	0.789	0.38 2
MVSNeRF [55]	26.63	0.931	0.168
IBRNet [62]	26.04	0.917	0.191
Ours	**27.29**	**0.966**	**0.092**

Table 4.6 – Comparison of the average PSNR, SSIM, and LPIPS of reconstructed images on the DTU [32] dataset. The higher the better for both PSNR and SSIM. The lower the better for LPIPS. The bold represents the best performance.

To further evaluate our proposed methods on generalizable tasks, we follow MVSNeRF [55] protocol, i.e. trained on DTU data while tested on different Real Forward-facing dataset (LLFF) [10] and NeRF synthetic dataset [4]. Table 4.7 reports evaluations on Table 4.8 reports evaluations on NeRF synthetic dataset [4] and Real Forward-facing dataset (LLFF) [10]. Except for evaluations on PixelNeRF and MVSNeRF, we added evaluations with recent light field network GPNR [68] on LLFF data. We report GPNR evaluation results from GPNR paper and supplementary file. Please noted that even though GPNR [68] reaches higher evaluations on the DTU dataset (28.50/0.932/0.167 in PSNR/SSIM/LPIPS), it requires 10 views inputs, while other baselines and our methods only take 3 views as inputs. Table 4.7 and Table 4.8 demonstrate that our method reaches the best performance when evaluated on NeRF synthetic dataset [4] and Real Forward-facing dataset (LLFF) [10].

Method	PSNR↑	SSIM↑	LPIPS↓
PixelNeRF[56]	11.24	0.486	0.671
MVSNeRF [55]	21.93	0.795	0.252
IBRNet [62]	21.79	0.786	0.279
GPNR [68]	20.69	0.808	0.281
Ours	**23.38**	**0.858**	**0.203**

Table 4.7 – Comparison of the average PSNR, SSIM, and LPIPS of reconstructed images on Real Forward-facing dataset (LLFF)[10] without test time optimization using a model trained on DTU dataset. The higher the better for both PSNR and SSIM. The lower the better for LPIPS. The bold represents the best performance.

Table 4.9, Table 4.11 and Table 4.10 show quantitative comparisons between our

Method	PSNR↑	SSIM↑	LPIPS↓
PixelNeRF[56]	7.39	0.658	0.411
MVSNeRF [55]	23.62	0.897	0.176
IBRNet [62]	22.44	0.874	0.195
GPNR [68]	24.10	0.933	0.097
Ours	**25.89**	**0.948**	**0.090**

Table 4.8 – Comparison of the average PSNR, SSIM, and LPIPS of reconstructed images on NeRF synthetic dataset [4] without test time optimization using the model trained on the DTU dataset. The higher the better for both PSNR and SSIM. The lower the better for LPIPS. The bold represents the best performance.

method and the recent state-of-the-art method with test time optimization. In the fine-tuning stage, we reduce the learning rate from 10^{-4} to 10^{-5}.

We first evaluate our per-scene fine-tuning results on DTU data [16]. Except for comparison with generalizable methods ([55], PixelNeRF [56], IBRNet [62]), we also compare with per-scene fine-tuning methods, including NeRF [4], DietNeRF [50] and RegNeRF [54]. All experiments follow MVSNeRF protocol to perform fine-tuning experiments and evaluation experiments. Specifically, our approach is fine-tuned within a minimal time (15 minutes). Table 4.9 demonstrates quantitative evaluations on the DTU dataset with per-scene finetuning. We reported numerical results of NeRF, MVSNeRF, and IBRNet from MVSNeRF paper [55], where NeRF is fine-tuned with 10 hours, IBRNet is fine-tuned with 1 hour and MVSNeRF is fine-tuned with 15 minutes. In addition, we report fine-tuning results of MipNeRF [126], DietNeRF [50], PixelNeRF [56] and RegNeRF [54] from RegNeRF paper. As we can see in Table 4.9, for a given task on novel view synthesis from few shot inputs, generalizable method (e.g. PixelNeRF, IBRNet, and MVSNeRF) usually outperforms single scene optimization methods (e.g. NeRF, DietNeRF and RegNeRF) on average. Moreover, our approach obtains overall comparable performances with generalizable methods (SRF, IBRNet, and MVSNeRF) and single scene optimization methods (NeRF, DietNeRF and RegNeRF) on the DTU dataset, see Table 4.9.

In addition, Table 4.10 and Table 4.11 demonstrate our fine-tuning results on NeRF synthetic dataset [4] and Real Forward-facing dataset (LLFF)[10]. Noted that all methods in Table 4.10 and Table 4.11 are fine-tuned using a model pre-trained on DTU data. Compared with generalizable methods, our methods achieve better performance with test time optimization.

Method	PSNR↑	SSIM↑	LPIPS↓
SRF ft[129]	15.68	0.698	0.281
MVSNeRF ft[55]	28.51	0.933	0.179
IBRNet ft [62]	**31.35**	0.956	0.131
PixelNeRF ft[56]	18.95	0.710	0.269
NeRF ft[4]	27.01	0.902	0.263
MipNeRF ft[126]	8.68	0.571	0.353
DietNeRF ft[50]	11.85	0.633	0.314
RegNeRF ft[54]	18.89	0.190	0.112
Ours ft	27.86	**0.967**	**0.043**

Table 4.9 – Comparison of the average PSNR, SSIM and LPIPS for synthesized novel views on the DTU [32] dataset with test time optimization. The higher the better for both PSNR and SSIM. The lower the better for LPIPS. The bold represents the best performance.

Method	PSNR↑	SSIM↑	LPIPS↓
MVSNeRF ft [55]	25.45	0.877	0.192
IBRNet ft [62]	24.88	0.861	0.189
Ours	**26.55**	**0.904**	**0.111**

Table 4.10 – Comparison of the average PSNR, SSIM, and LPIPS for synthesized novel views on the Real Forward-facing dataset [10] with test time optimization. The higher the better for both PSNR and SSIM. The lower the better for LPIPS. The bold represents the best performance.

Method	PSNR↑	SSIM↑	LPIPS↓
MVSNeRF ft [55]	27.07	0.931	0.168
IBRNet ft [62]	25.62	0.939	0.111
Ours	**27.77**	**0.956**	**0.054**

Table 4.11 – Comparison of the average PSNR, SSIM and LPIPS of reconstructed images on the NeRF synthetic dataset [4] with test time optimization. The higher the better for both PSNR and SSIM. The lower the better for LPIPS. The bold represents the best performance.

GT PixelNeRF MVSNeRF Ours MVSNeRF-ft Ours-ft

Figure 4.14 – Qualitative comparison of novel view synthesis of unseen scenes with and without test time optimization from 3 input views on the DTU dataset [32]. For 3 input views, our method could generate more accurate sharp details.

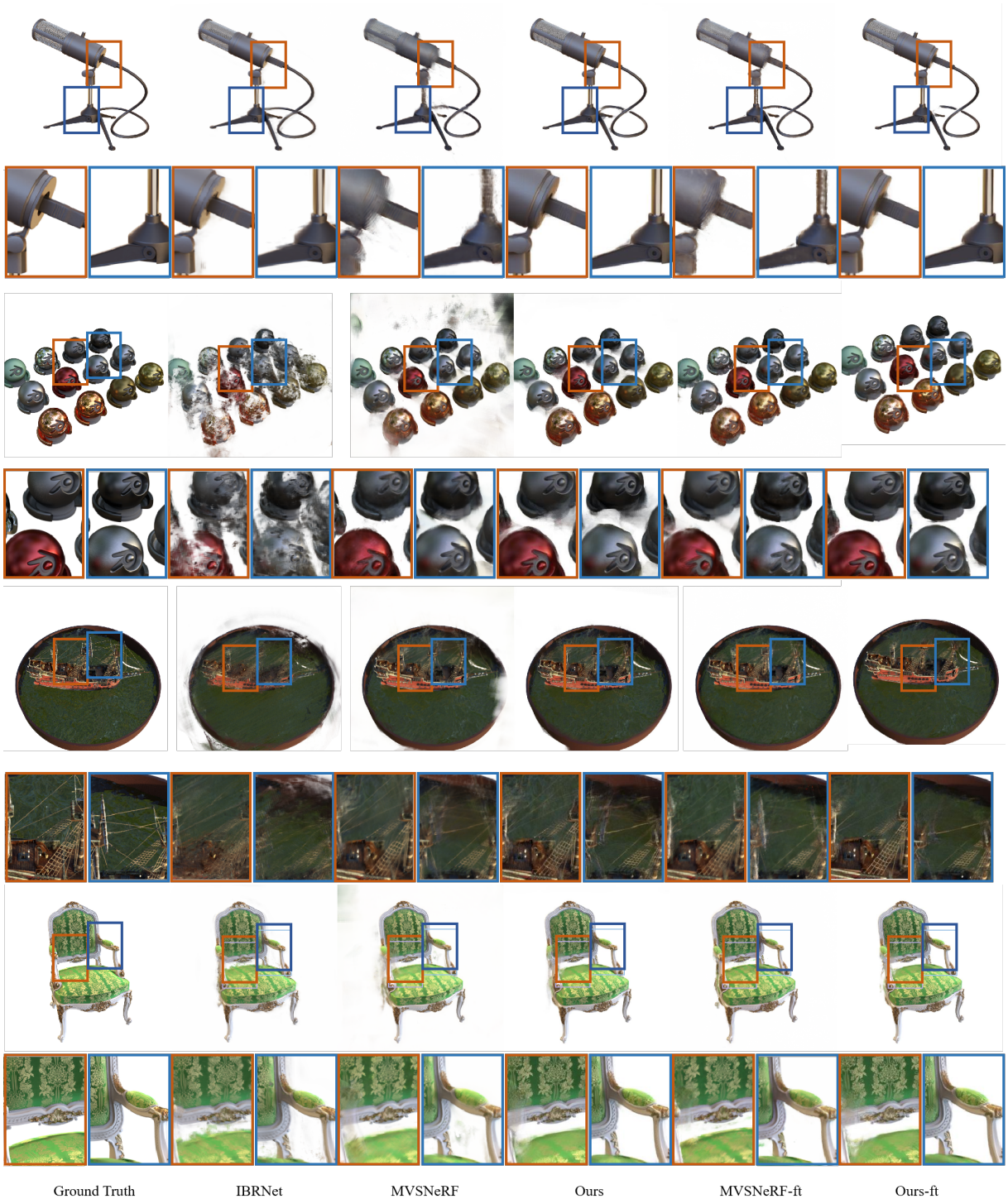


Figure 4.15 – Qualitative comparison of rendered novel view on unseen scenes from the NeRF synthetic dataset [4]. From the left to the right are the ground truth, novel view rendered by IBRNet without test time optimization, novel view rendered by MVSNeRF without test time optimization, our result without test time optimization, novel view of MVSNeRF and our result with test time optimization, respectively.

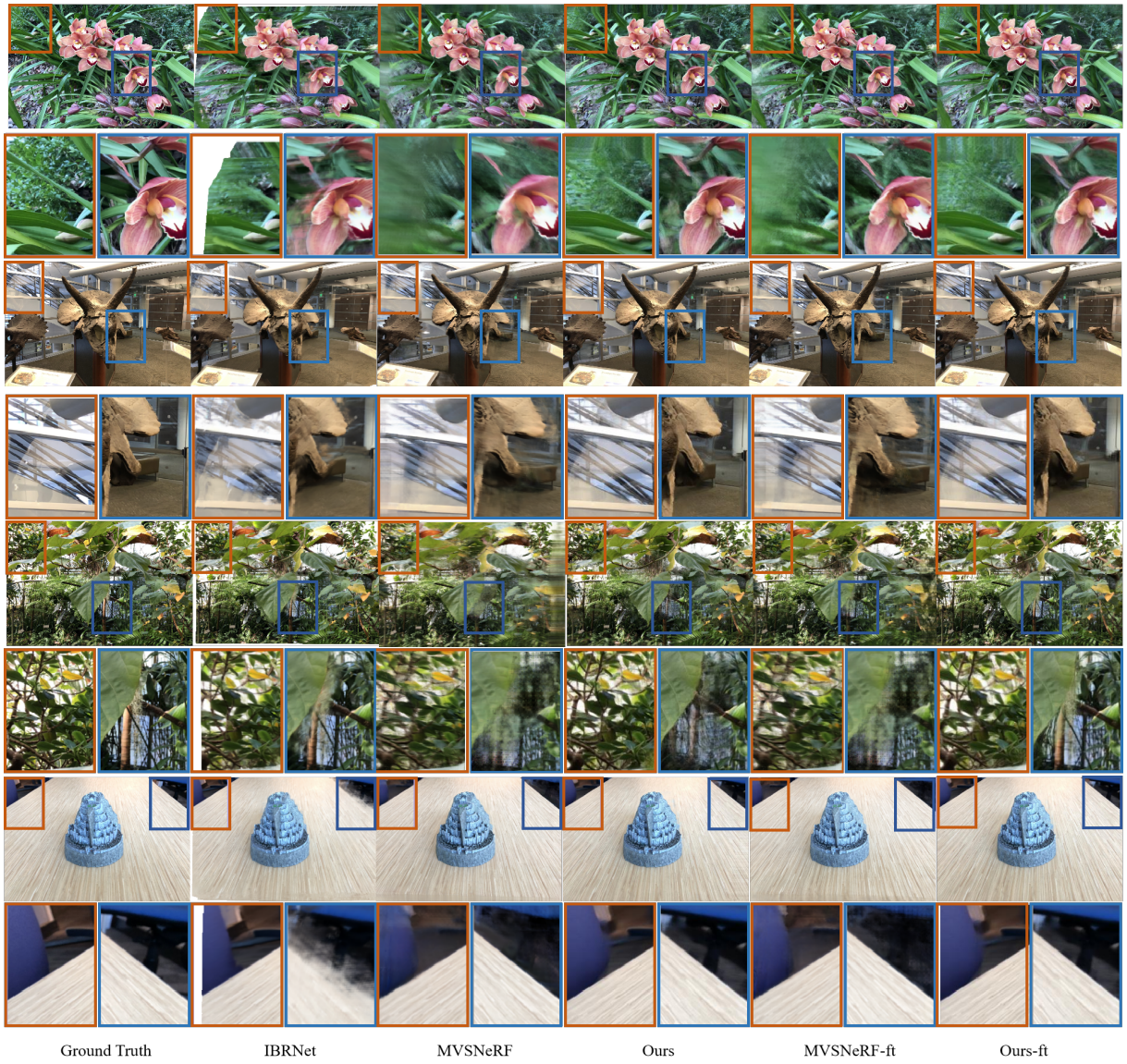


Figure 4.16 – Qualitative comparison of rendered novel view synthesis on unseen scenes from the Real Forward-facing dataset [10]. From the left to the right are the ground truth, novel view rendered by IBRNet without test time optimization, novel view rendered by MVSNeRF without test time optimization, our result without test time optimization, novel view of MVSNeRF and our result with test time optimization, respectively.

For qualitative comparison, we reproduce baseline methods' results with their provided code and pre-trained model. Figure 4.14 demonstrates a comparison with PixelNeRF and MVSNeRF on the DTU dataset. Our method could achieve much better results before test time optimization, i.e. more accurate shapes and edges, clear texture details, etc.. On per-scene fine-tuning comparison, we show the qualitative evaluation of MVSNeRF [55]

since it achieves better performance than the other one [56]. Figure 4.14 demonstrates that MVSNeRF achieves much better performance after test time optimization than before on the testing scene. Our results also improve after a short time(15 minutes) of fine-tuning and achieve relatively comparable results to the encoder-endowed NeRF approaches after optimization. We recall again that competition methods here require renderings that are orders of magnitude slower than ours.

Figure 4.15 demonstrates the qualitative comparison between our method and IBRNet [62] and MVSNeRF [55] on the NeRF synthetic dataset [4]. Even though only using a pre-trained model on the DTU dataset, our method achieves much better visual results than baselines without test-time optimization [55], [62], especially at preserving the structure of the scene and more robust to the shiny parts. When given short-time optimization(e.g. 15 minutes), our method could render less noisy novel views and outperform the baseline methods.

Figure 4.16 shows more visual results on the Real Forward-facing dataset [10]. Before fine-tuning, our methods manage to reproduce the shape and appearance of the scene to a good extent and also recover from some of the competition's failures, i.e. the leaves in Figure 4.16. As illustrated in the visual comparison in Figure 4.16, the fine-tuning process will recover more details in rendered novel view, i.e. the color of the trunk in the third row.

4.4.6 Computation complexity

As shown in Table 4.12, compared with PixelNeRF[56] and MVSNeRF [55], our method requires less inference time on the DTU dataset with 3 input views. Table 4.12 also demonstrates model sizes to complete the complexity comparison. We note that for our generalizable NeRF competition (e.g. PixelNeRF), the main computational bottleneck is the radiance field inference (MLP querying 192 points per ray to render volumetrically). Hence, our model circumvents this bottleneck by modeling a neural light field instead of a radiance field (our MLP only needs to query a ray once).

	PixelNeRF[56]	MVSNeRF[55]	**Ours**
Clock time	27.01	10.43	**0.2**
Param	28.162M	**0.34M**	0.57M
Flops	123T	61.74G	**39.86G**

Table 4.12 – Comparison of model complexity on DTU dataset [32].

4.4.7 Ablations and analysis

We propose here an ablative analysis of our method from the DTU [32] and shapeNet-V2 [33] datasets. Specifically, we disable the light field function (ours w/o light field), and we render the final image directly from the target view aligned convolutional feature volume. We also reproduce our method without using the ray coordinates (ours w/o ray coordinates).

w/o lf **Ours** w/o lf **Ours**

Figure 4.17 – Qualitative ablation of our method on unseen DTU [32] scenes (6 input views).

w/o lf **Ours** w/o lf **Ours**

Figure 4.18 – Qualitative ablation of our method on unseen ShapeNet-V2 [33] cars (1 input view).

4.5 Conclusion

We proposed a method for generating novel views from a few input-calibrated images with a single forward pass prediction deep neural network. We learn an implicit neural light field function that models ray colors directly. In comparison to [6], we proposed a more efficient local ray conditioning and an optimization-free inference. Our method combines the advantages of 3D-aware convolutional approaches and implicit representations and requires only image data in training. We demonstrated our method successfully on synthetic and real benchmarks for few-shot novel view synthesis. Our method outperforms the convolutional baselines (see Table 4.3) and provides competitive performances compared to locally conditioned radiance fields (e.g. PixelNeRF [56]) while being much faster to render.

Chapter 5

Few Shot Multi-human Reconstruction and Novel View Synthesis

5.1 Introduction

Human reconstruction and rendering is a fundamental problem in computer vision and graphics enabling various applications, e.g. human modeling, behavior analysis etc. Recent research has explored generating human shapes and novel views from diverse data, such as single images ([79], [80], [82]), multiple images ([90], [91]), RGB videos ([84], [149]) or RGB-D data ([86], [87]). However, these tasks are mainly focused on a single human or generation from videos. Generating multiple human shapes and appearances from sparse multi-view images is much less explored, which exhibits significant potential by providing 3D cues and requires less memory in computation and transmitting.

Despite the exhibited great potential, generating 3D shapes and radiance of multiple humans from sparse images remains challenging. One part of challenge comes from the complexity and variability of multiple human appearances, poses, occlusions, and interactions. One line of existing methods targeting this require segmentation masks and a pre-scanned template mesh ([119], [121]), rely on a coarse body model ([122], [123]), or require temporal information ([3], [122]). However, most of those work has been leveraged to obtain geometry and appearance from monocular video ([150], [151]), RGB-D video [152], and sparse multi-view video ([3], [99], [101]–[103], [153]–[155]). None of them were designed to handle multi-human's increased geometric complexity and occlusion from sparse static multi-view images. The other line of solutions simultaneously tackle the novel-view-synthesis and geometry-reconstruction problems by combining implicit signed distance functions (SDFs) ([74]), with differentiable rendering ([4], [34], [71], [77]). This approach has the advantage of producing geometry with renderings from novel viewpoints

that could capture complex surface/light interactions, increasing the scope of applications. We take an insight from both the human reconstruction methods and hybrid surface and volumetric reconstruction approach, make an assumption that utilizing human templates could provide 3D geometry cues for higher quality surface reconstruction and appearance rendering, which allows modeling the complexity and variability of multiple human appearances.

Except the challenges in multi-human settings, the other part of challenge lies in the static sparse inputs setting lacking sufficient geometry and appearance information. Recent approaches tackle sparse inputs utilizing image encoder [56], depth regularization ([51], [52], [59]), ray density regularization [53], etc. However, those methods are mainly focused on general objects, either requiring pre-training on a large-scale dataset ([50], [56]) or ignoring the training efficiency [52]. When generating 3D shapes and radiance of multi-human from sparse images, the human templates(e.g. A Skinned Multi-Person Linear Model (SMPL) [106]) becomes essential to enhance the geometry estimation in surface and volume rendering.

In this work, we address the problem of generating 3D shapes and radiance of multiple humans from sparse multi-view images. Our key insight is that human-specific geometric constraints can be leveraged to tackle the challenging sparse-view setting. Specifically, we first obtain an SMPL body model from the input data and use this to train a geometry-only implicit SDF network, where we define the multi-human surface as the zero-level set of the SDF. The geometry network is optimized using multi-view images by leveraging hybrid surface and volume rendering [34] along with uncertainty estimation ([51], [59]), where the SMPL meshes are treated as noisy estimations. To achieve higher rendering quality from sparse inputs, we additionally propose a patch-based regularization that guarantees consistency across different rays and a saturation regularization that ensures consistency for variable image illuminations within the same scene.

We evaluate our method on both real-world multiple human datasets (CMU Panoptic [1], [2]) and synthetic datasets (MultiHuman [3]) both quantitatively and qualitatively. We demonstrate results on 5,10,15 and 20 training views and achieve state-of-the-art performance in terms of surface reconstruction and image quality in all settings.

In summary, our contributions include:

— We propose the first neural implicit surface and volume rendering for multiple humans using a sparse set of static images;

— To address the problem of occlusion, we propose the use of SMPL for geometric

regularization;

— we propose a patch-based ray consistency regularization and an image saturation regularization that ensures illumination consistency across views.

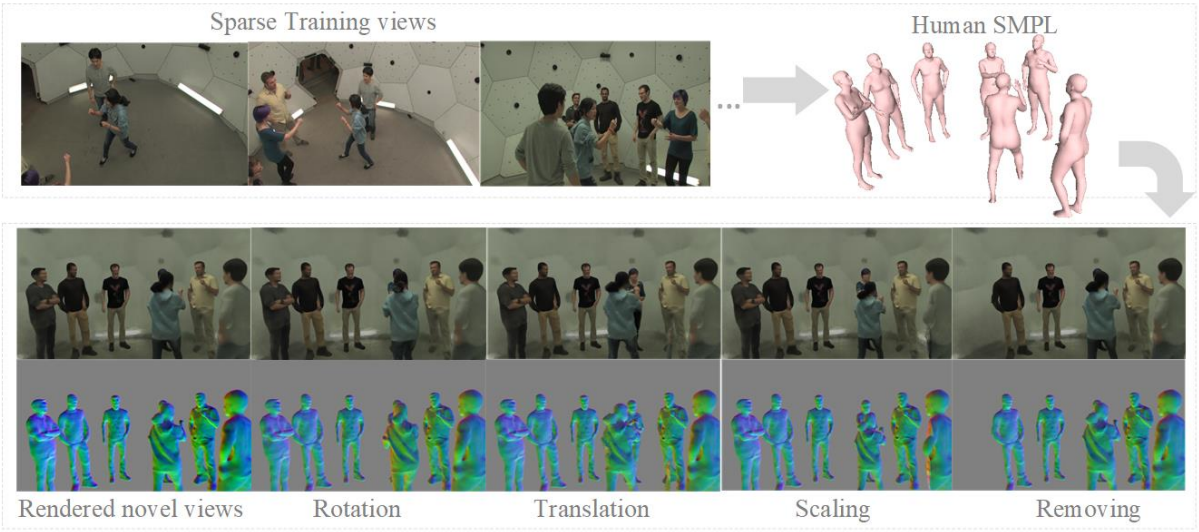


Figure 5.1 – With the SMPL initialization, our method could reconstruct high-quality 3D shapes and appearances of multi-humans by training only on sparse input views. It also enables editing applications on 3D space during rendering, including rotation, translation, scaling, and removal.

5.2 Related work

Single-Human Reconstruction. There is a vast amount of work on reconstructing 3D humans from single images ([78]–[82]), monocular video ([83]–[85]), RGB-D data ([86]–[88]) and multi-view data ([89]–[92]). We concentrate here on the multi-view setting. High-end multi-view capture systems can achieve reconstructions of outstanding quality ([2], [90], [91], [156]–[158]), but require a complex studio setup that is expensive to build and not easily accessible. To alleviate this, numerous works have been proposed that use instead a sparse set of RGB cameras (e.g. between 2 and 15), where the lack of views and presence of wide baselines is compensated by tracking a pre-scanned template ([93]–[96], [159]), using a parametric body model ([97], [98]), or more recently, by the use of deep learning ([92], [99]–[103], [153]–[155]).

Multi-Human reconstruction. In contrast, a limited number of works have addressed the problem of multiple human reconstruction. This is a difficult task since the presence of several people increases the geometric complexity of the scene, introduces occlusions, and amplifies ambiguities such that commonly used features like color, edges, or key points cannot be correctly assigned.

For single images and video, the problem has been mainly tackled by regressing the parameters of the SMPL [106] body model ([105], [107]–[117]). Although this can work robustly with as little as one view, the reconstructions are very coarse and cannot explain hair, clothing, and fine geometric details. The only exception is the work of Mustafa et al. [118], which performs model-free reconstruction of multiple humans by combining an explicit voxel-based representation with an implicit function refinement. However, the method requires training on a large synthetic dataset of multiple people which hinders generalization. Our work, on the other hand, performs 3D reconstructions, produces renderings of novel views, and can generalize to arbitrary multi-human scenes.

Multi-view capture setups can help resolve depth ambiguities and some of the occlusions. Classic methods for estimating multiple humans rely heavily on segmentation masks and template mesh tracking ([119]–[121]). We avoid the use of segmentation masks by adopting volumetric rendering for implicit surfaces [34]. More recently, deep learning-based approaches were proposed, but they either require temporal information ([3], [122]–[124]), pre-training on a large dataset ([3]) which cannot work on general scenes, or a coarse body model ([122]–[124]) which lacks geometric detail. Here, we focus on the multi-human setting on static scenes and propose a method that recovers accurate reconstructions and at the same time produces renderings of novel viewpoints.

Neural surface and radiance rendering. When the goal is to generate free-viewpoint video, image-based rendering has been considered as an alternative or complement to 3D reconstruction ([95], [102], [103], [153]–[155], [160]).

Recently, NeRF ([4]) demonstrated impressive rendering results by representing a 3D scene as a neural radiance field, trained only with calibrated multi-view images through the use of volume rendering. However, due to the unconstrained volumetric representation and self-supervised training on RGB values, reconstructed geometries tend to be too noisy to be useful for 3D applications.

To recover more accurate 3D geometry along with appearance, DVR [72], IDR [71], and NLR [35] propose to learn an implicit representation directly from multi-view images

but require accurate object masks to work. To avoid the need for segmentation masks, recent works propose to combine implicit representations with volume rendering ([34], [76], [77]).

These methods show remarkable reconstruction results but struggle when the number of input views is low. Implicit neural representations from sparse views can be obtained by using pre-trained pixel-aligned features ([56], [161]–[166]), but this requires ground-truth geometry and is limited by the training data, struggling to generalize to new scenes. Sparse variants that do not require pixel-aligned features were proposed in ([52], [53], [167]). InfoNeRF [53] regularizes sparse views by adding an entropy constraint on the density of the rays, RegNeRF [52] uses a patch-based regularizer over generated depth maps, and SparseNeuS [167] uses a multi-scale approach along with learned features that are fine-tuned on each scene.

Our approach builds on NeuS [34], and tackles the sparse view challenge by adding human-specific geometric priors [78] and novel regularizations.

5.3 Methodology

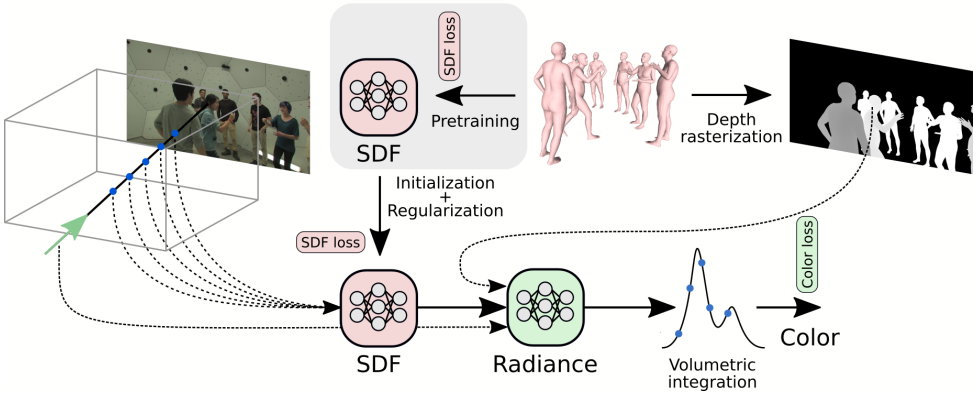


Figure 5.2 – **Overview.** We address the multi-human reconstruction problem by regularizing the geometry using SMPL, along with uncertainty-based SDF training and novel photo-metric regularizations designed to compensate for the lack of views.

Given a sparse set of views $\{I_i\}_{i=1}^{N}$ of a multi-human scene with camera intrinsics and extrinsic $\{K_i, [R|t]_i\}$, our goal is to reconstruct geometry and synthesize the appearance of multiple humans from arbitrary viewpoints. The pipeline is illustrated in Fig. 5.3. Our approach builds on NeuS [34], which combines an implicit signed distance representation for geometry with volumetric rendering.

In order to solve the challenging case of multiple humans occluding each other, we hypothesize that a naive RGB reconstruction loss is insufficient and propose to use a strong geometric prior before training with multi-view images. Towards this, we first train the implicit SDF network independently by leveraging off-the-shelf SMPL estimations (Sec. 5.3.2). To handle details and represent appearance, the geometry network is then fine-tuned considering foreground and background objects. Moreover, we propose the use of hybrid bounding box rendering to handle the multi-human setting (Sec. 5.3.3).

Additionally, we define an explicit SDF constraint based on the uncertainty of the SMPL estimations, together with a ray consistency loss, and a saturation loss to improve image rendering quality for sparse views (Sec. 5.3.4).

5.3.1 Scene representation and rendering

We define a multi-human surface $\mathcal{S}$ as the zero-level set of a signed distance function (SDF) $f_{\theta_0} : \mathbb{R}^3 \rightarrow \mathbb{R}$, encoded by a Multilayer Perceptron (MLP) f_{θ_0} with parameters θ_0:

$$\mathcal{S} = \left\{ p \in \mathbb{R}^3 | f_{\theta_0}(p) = 0 \right\}. \tag{5.1}$$

Following NeuS ([34]), we train the geometry network f_{θ_0} along with a color network c_{θ_1}, with parameters θ_1, mapping a point p to color values (more details in Sec. 5.3.3). Combining the SDF representation with volume rendering, we approximate the color along a ray r by:

$$C(r) = \sum_{i=1}^{N} w(p_i) c_{\theta_1}(p_i), \tag{5.2}$$

$$w(p_i) = T(p_i)\alpha(p_i), \tag{5.3}$$

$$T(p_i) = \prod_{j}^{i-1} (1 - \alpha(p_j)), \tag{5.4}$$

where $p_i = o + t_i v$ is a sampled point along the ray r starting at camera center o with direction v; $c_{\theta_1}(p_i)$ is the predicted color at p_i, $w(p_i)$ is the weight function, $T(p_i)$ is the accumulated transmittance, and $\alpha(p_i)$ is the opacity value. Following NeuS, $\alpha(p_i)$ is defined as a function of the signed distance representation:

$$\alpha(p_i) = max\left(\frac{\Phi(f_{\theta_0}(p_i)) - \Phi(f_{\theta_0}(p_{i+1}))}{\Phi(f_{\theta_0}(p_i))}, 0 \right) \tag{5.5}$$

where $f_{\theta_0}(p_i)$ is the signed distance of p_i, $\Phi(f_{\theta_0}(x)) = (1 + e^{-sx})^{-1}$ is the cumulative distribution function (CDF) of the logistic distribution, and s is a learnable parameter (see [34] for more details).

5.3.2 Geometric prior

Typically, the SDF function f_{θ_0} and the color function c_{θ_1} are simultaneously optimized by minimizing the difference between the rendered and ground-truth RGB values ([4], [34], [71]). While this allows for training without the need for geometric supervision, it has been noted that a photometric error alone is insufficient for the challenging sparse-view setting ([51], [59]), since there are not enough images to compensate for the inherent ambiguity in establishing correspondences between views. For the *multi-human* setting this becomes more problematic, as correspondences are even more ambiguous due to clutter.

To address this, we propose to regularize using geometric information by first independently training f_{θ_0} using off-the-shelf SMPL fittings, which can be robustly computed from the input data. We train this network in a supervised manner by sampling points with their distance values as in [74].

Given that SMPL can only coarsely represent the real surface, we treat this geometry as a 'noisy' estimate that will be later improved upon using the multi-view images. Preparing for this, and inspired by ([51], [59]), we model the SMPL "noise" as a Gaussian distribution $\mathcal{N}(0, s_{noise}(p_j)^2)$ with standard deviation $s_{noise}(p_j)$, and train f_{θ_0} to output an estimate of the uncertainty $s_{noise}(p_j)$ along with the distance value; that is, $f_{\theta_0}(p_j) = (d_j, s_{noise})$. The geometry network f_{θ_0} is then optimized by minimizing the negative log-likelihood of a Gaussian:

$$\mathcal{L}_s = \frac{1}{n} \sum_{j=1}^{n} \left(log(s_{noise}(p_j)^2) + \frac{(d_j - d'_j)^2}{s_{noise}(p_j)^2} \right), \tag{5.6}$$

where n is the number of sampled points, d_j is the predicted SDF value for point p_j, and d'_j is the signed distance sampled directly from the SMPL meshes.

5.3.3 Hybrid rendering with geometry constraints

To work with unbounded scenes, NeRF++ proposed to separately model the foreground and background geometries using an inverted sphere parameterization, where the foreground is parameterized within an inner unit sphere, and the rest is represented by

an inverted sphere covering the complement of the inner volume. We follow this and train separate models for foreground and background. Specifically, we use a simple NeRF [4] architecture for the background and train the foreground model using f_{θ_0} and the color network c_{θ_1}, where the output color $C(p_i)$ is predicted as:

$$C(p_i) = c_{\theta_1}(\gamma(p_i), \gamma(v_i), f_0, f_1). \tag{5.7}$$

Here, $\gamma(p_i)$ and $\gamma(v_i)$ are the positional encodings [4], [168] of the sampled point p_i and its ray direction v_i, and f_0 includes the gradients of predicted SDF and predicted feature from the geometry network f_{θ_0} [71]. Additionally, to inject geometric prior knowledge into the appearance network we condition c_{θ_1} on the rasterized depth feature from the corresponding SMPL mesh.

For reconstructing multiple humans, one difficulty in modeling the foreground as in NeRF++ is that the bounding sphere will contain a large empty space, making it costly to search for the surface during hierarchical sampling and adding non-relevant points to the training. To resolve this, we propose to use instead multiple 3D bounding boxes as the foreground volume. Specifically, we define a bounding box B^j for the $j-$th human using the SMPL fittings, with minimum and maximum coordinates $[B^j_{min} - \delta, B^j_{max} + \delta]$, where B^j_{min} and B^j_{max} are the minimum and maximum coordinates of SMPL along the x, y, z axes respectively, and δ is a spatial margin (here we set to 0.1). The foreground volume is then defined as $B = \cup_{j=1..M} B^j$, and we define $b(p_i)$ as:

$$b(p_i) = \begin{cases} 1, p_i \in B, \\ 0, p_i \notin B \end{cases} \tag{5.8}$$

For points that fall inside the foreground, $p \in B$, we calculate the opacity value $\alpha^{FG}(p_i)$ using the predictions of $f_{\theta_0}(p_i)$ according to Equation 5.5, and the color $C(p_i)^{FG}$ using c_{θ_1}. The points that fall outside the bounding box are modeled as background using a NeRF model, where the opacity is calculated as $\alpha^{BG}(p_i) = 1 - e^{\sigma(p_i)\delta(p_i)}$, with δ and σ defined as in [4], and the color C^{BG} is predicted using α^{BG}. Given a point p_i, its color and opacity values are updated as follows:

$$C(p_i) = b(p_i)C^{FG}(p_i) + (1 - b(p_i))C^{BG}(p_i) \tag{5.9}$$

$$\alpha(p_i) = b(p_i)\alpha^{FG}(p_i) + (1 - b(p_i))\alpha^{BG}(p_i) \tag{5.10}$$

Finally, following [169], given a ray r with n sampled points $\{p_i = o + t_i v\}_{i=1}^n$, the

color is approximated as:

$$C(r) = \frac{\sum_{i=1}^{N} W(p_i)C(p_i)}{\sum_{i=1}^{N} W(p_i)}, \tag{5.11}$$

where $W(p_i) = T(p_i)\alpha(p_i)$, $T(p_i) = \prod_{j}^{i-1}(1-\alpha(p_j))$. This function allocates higher weights to points near the surface and lower weights to points away from the surface and is used to improve the rendering quality.

5.3.4 Optimization

Given a set of multi-view images, and a pre-trained SDF network f'_{θ_0} (Sec. 5.3.2), we minimize the following objective:

$$\mathcal{L} = \mathcal{L}_{\text{r}} + \lambda_{eik}\mathcal{L}_{\text{eik}} + \lambda_{sdf}\mathcal{L}_{\text{sdf}} + \lambda_r\mathcal{L}_{\text{r}} + \lambda_s\mathcal{L}_{\text{s}}, \tag{5.12}$$

where $\mathcal{L}_{\text{r}}$ is a L1 reconstruction loss between the rendered image I_r and the ground-truth I'_r and $\mathcal{L}_{\text{eik}}$ is the Eikonal loss [170].

Additionally, we propose to use an uncertainty-based SDF loss $\mathcal{L}_{\text{sdf}}$, a novel ray consistency loss $\mathcal{L}_{\text{r}}$, and saturation loss $\mathcal{L}_{\text{s}}$ which are explained in the following. Fig. 5.3 illustrates the losses involved in the training of our method. Rays with available ground-truth pixels are supervised with pixel colors. Sub-pixel rays without available ground truth are supervised using color and density pseudo-ground-truth from neighboring rays.

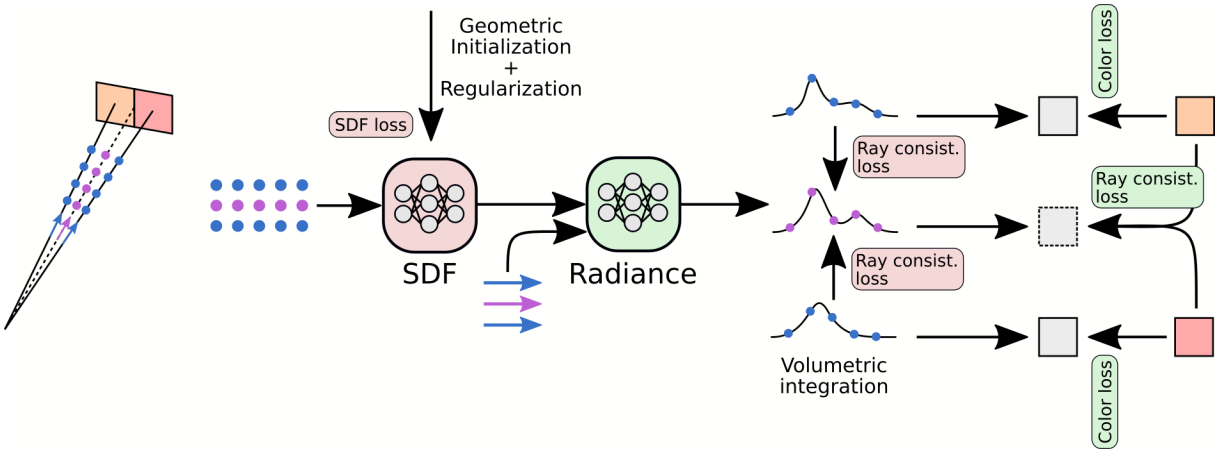


Figure 5.3 – Illustration of our losses. Rays without ground truth are supervised using our ray consistency loss. For rays corresponding to pixels in the training data, we supervise points using a combination of SDF losses and color losses.

SDF Loss. As detailed in Sec. 5.3.2, we treat the SMPL mesh as a noisy estimate of the real surface. When the sampled points are not within the foreground box B, or the absolute sdf value predicted by the geometry network f_{θ_0} is greater than a pre-defined threshold ξ_0, or the standard deviation $s_j = s_{noise}(p)_j$ is bigger than the threshold ξ_1, we use the following loss:

$$\mathcal{L}_{\text{sdf}} = \begin{cases} \frac{1}{n}\sum_{j=1}^{n}(log(s_j^2) + \frac{(d_j^{'}-d_j)^2}{s_j^2}), \\ s.t.(p_i \notin B, |d_j| > \xi_0 \quad or \quad s_j > \xi_1) \\ 0, otherwise \end{cases} \tag{5.13}$$

where d_j and $d_j^{'}$ are the SDF predictions from the final f_{θ_0} and initial network $\tilde{f_{\theta_0}}$, and ξ_0, ξ_1 are set to 0.2 and 0.5, respectively. This function encourages the network to maintain geometry consistency during learning while allowing some freedom to learn the details encoded in the images.

Ray Consistency Loss. We introduce the following ray consistency loss $\mathcal{L}_{\text{r}}$ to ensure photometric consistency across all images under sparse views:

$$\mathcal{L}_{\text{r}} = ||C(r_i) - C(r^*)||_1 + D_{KL}(P(r_i)||P(r^*)) \tag{5.14}$$

where $C(r_i)$ is the ground truth color of a randomly sampled ray r_i on a small patch and $C(r_p^*)$ denotes the rendered color of an interpolated ray on a small patch. Inspired by [53], we introduce a KL-divergence regularization for the ray density, where $P(r_i) = \frac{\alpha_i}{\sum_{i=1}^{N} a_i}$.

The goal of this loss is to ensure consistency and smoothness of unseen rays by constraining the interpolated rays on a small patch to have a similar distribution, both for color and density.

Saturation loss. Finally, we observe that real-world images might contain variable illumination or transient occluders among different views (this is the case for example in the CMU Panoptic dataset ([1], [2])), which can degrade the rendering quality due to inconsistency across views. Instead of learning complex transient embeddings as in [171], we propose to convert the RGB image into the HSV space and calculate the L1 reconstruction loss of the saturation value between the rendered image and the ground truth: $\mathcal{L}_{\text{s}} = ||I_s - I_s^{gt}||_1$.

5.3.5 Network structure

Fig. 5.4 shows the architecture of our network in more detail. The geometry MLP has 8 layers of width 256, with a skip connection from the input to the 4th layer. The radiance MLP consists of an additional 4 layers of width 256 and receives as input the positional encoding of the point $\gamma(p)$, positional encoding of the view direction $\gamma(v)$, rasterized depth feature f_1, and gradient of the SDF $n(p)$. All layers are linear with ReLU activation, except for the last layer which uses a sigmoid activation function. During training, we sample 512 rays per batch following the coarse and fine sampling strategy of ([4], [34]). For a fair comparison, we unified the number of sampled points on each ray for all methods, namely, each ray with $N = 64$ coarsely sampled points and $N = 64$ finely sampled points for the foreground, and $N = 32$ for the background.

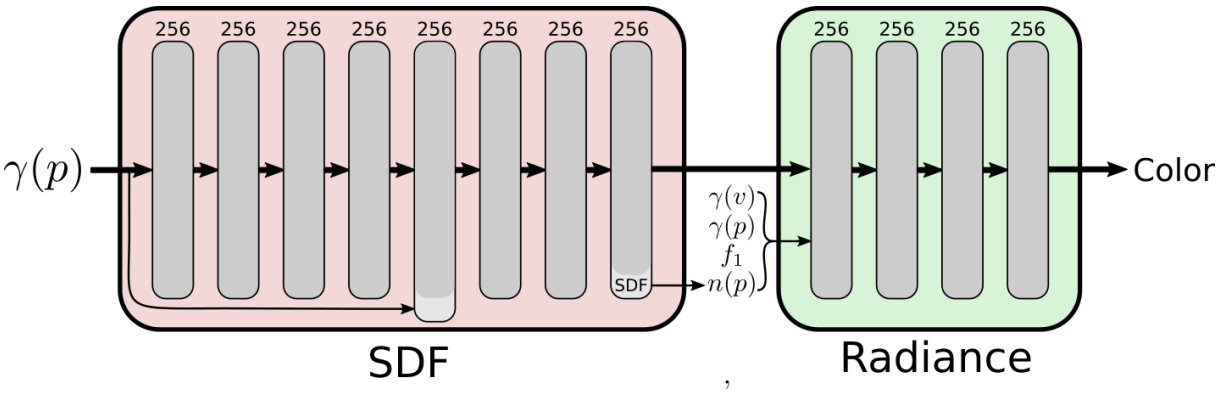


Figure 5.4 – **Network architecture**. p is a sampled point along a ray. γ is the positional encoding [4], [168]. $n(p)$ is the gradient of predicted sdf w.r.t the input point p. v is the direction of the ray, and f_1 the rasterized depth feature.

5.4 Experiments

5.4.1 Implementation details

In this section, we first provide implementation details (Sec. 5.4.1), and next demonstrate the performance against baselines on real (Sec. 5.4.3) and synthetic (5.4.4) datasets, where both are tested in terms of novel-view synthesis and visual reconstructions and the latter is used to quantify geometry error. Finally, we show ablation studies (Sec. 5.4.5) that demonstrate the importance of the proposed components.

Our method was implemented using PyTorch [146], and trained on a Quadro RTX

5000 GPU. We use ADAM optimizer [147] with a learning rate ranging from 5×10^{-4} to 2.5×10^{-5}, controlled by cosine decay schedule. Our network architecture follows ([4], [71]). For a fair comparison, we sample 512 rays per batch and follow the coarse and fine sampling strategy of [34].

5.4.2 Dataset

CMU panoptic dataset We first evaluate our approach on the CMU Panoptic Dataset ([1], [2]). This dataset contains multiple humans containing 3D key points, which are used to fit human SMPL. Our experiments were performed on five different scenes, where each scene originally included 30 views containing 3/4/5/6/7 people. The camera of each scene includes 30 views located on a spherical spiral. The training views were randomly extracted from the HD sequences 'Ultimatum' and 'Haggling'. Specifically, we used static frames from those sequences, including frame 9200 from the Video 'Haggling' and frames 5500,7800,9200,22900 from 'Ultimatum'. We uniformly sampled 5, 10, 15, and 20 views as training and we used the remaining 25, 20, 15, and 10 views respectively as testing. The image resolution in training and testing is 1920×1080. We compare with two major baselines: NeuS [34] and VolSDF[77], both in terms of novel-view synthesis and geometry reconstructions (qualitatively). For quantitative evaluation, we report three commonly used image metrics: peak signal-to-noise ratio (PSNR) [172], structural similarity index (SSIM) [173] and learned perceptual image patch similarity (LPIPS) [174]. For qualitative comparison, both rendered images and rendered normal images are shown.

Synthetic dataset Based on the multi-human dataset ([3], [175]), we used Unity 3D to create a synthetic dataset with 29 cameras arranged in a great circle. This includes three scenes with similar backgrounds but different camera locations and orientations. Each of the scenes contains 1/5/10 humans respectively. The image resolution is 1920×1080.

5.4.3 Generalization on real multi-Human dataset

Table 5.1 demonstrates novel view synthesis results with different training views (5/10/15/20) compared to the baselines. Our proposed method outperforms these in PSNR and SSIM in all the scenes and consistently performs better or equal in terms of LPIPS. For qualitative comparison, we demonstrate both rendered novel views and normal images in Fig. 5.7. As depicted in the figure, when only given 5/10 training views,

Scene	Method	PSNR↑				SSIM↑				LPIPS↓			
		5	10	15	20	5	10	15	20	5	10	15	20
1	NeuS	17.83	18.84	19.39	21.97	0.62	0.67	0.69	0.55	**0.74**	0.51	**0.49**	**0.45**
	VolSDF	17.50	18.08	19.51	22.31	0.64	0.61	0.67	0.71	0.61	0.54	0.51	0.48
	Ours	**18.41**	**20.32**	**21.60**	**23.19**	**0.67**	**0.73**	**0.73**	**0.74**	**0.55**	**0.50**	0.50	0.49
2	NeuS	16.87	18.51	19.40	21.05	0.60	0.65	0.70	0.71	0.57	0.53	0.51	0.49
	VolSDF	16.36	17.52	19.40	21.60	0.57	0.59	0.67	0.70	0.62	0.53	0.49	0.47
	Ours	**19.72**	**21.15**	**21.40**	**23.12**	**0.70**	**0.73**	**0.73**	**0.74**	**0.50**	**0.49**	**0.48**	**0.47**
3	NeuS	16.03	17.39	19.17	21.21	0.56	0.61	0.70	0.73	0.62	0.54	**0.47**	**0.46**
	VolSDF	16.36	18.21	19.56	21.06	0.57	0.59	0.64	0.68	0.62	0.52	0.48	0.47
	Ours	**18.57**	**20.94**	**21.86**	**23.16**	**0.66**	**0.73**	**0.74**	**0.74**	**0.52**	**0.48**	**0.47**	0.47
4	NeuS	14.16	17.14	19.87	21.37	0.49	0.51	0.70	0.72	0.60	0.57	0.48	0.46
	VolSDF	13.51	17.07	18.68	20.89	0.50	0.57	0.65	0.68	0.64	0.54	0.53	0.46
	Ours	**19.54**	**20.94**	**21.35**	**23.29**	**0.69**	**0.72**	**0.73**	**0.75**	**0.50**	**0.47**	**0.47**	**0.45**
5	NeuS	17.69	18.60	20.03	21.50	0.57	0.62	0.69	0.70	0.55	0.54	0.50	**0.47**
	VolSDF	14.85	17.32	19.04	20.91	0.53	0.57	0.66	0.68	0.63	0.58	0.53	0.48
	Ours	**19.34**	**20.55**	**21.08**	**22.55**	**0.67**	**0.70**	**0.72**	**0.72**	**0.51**	**0.47**	**0.47**	**0.47**
Average	NeuS	16.52	17.79	19.57	21.42	0.57	0.62	0.69	0.72	0.58	0.54	0.49	**0.47**
	VolSDF	15.81	17.68	19.23	21.35	0.56	0.59	0.66	0.69	0.62	0.54	0.50	**0.47**
	Ours	**19.12**	**20.78**	**21.46**	**23.06**	**0.68**	**0.72**	**0.73**	**0.74**	**0.52**	**0.48**	**0.48**	**0.47**

Table 5.1 – Comparison against NeuS [34] and VolSDF [77] on the CMU Panoptic dataset [1], [2], using 5/10/15/20 views for training.

the baseline methods fail to reconstruct a good geometry or render a realistic appearance. Although the quality of the geometries improves with 15/20 training views, the results still exhibit missing body parts or can mix the background with the subjects. In contrast, our method can reconstruct a complete geometry for all humans in all sparse-view cases. The following figures demonstrate qualitative comparison against NeuS [34] and VolSDF [77] of synthesized novel views and reconstructed normal images from varying different training views.

Comparison to single human NeRF. We compare our method to the single human nerf state-of-the-art method ARAH [101]. We note that adapting such methods to our setup requires tedious manual pre-processing (detecting and segmenting people, associating detections across views), which is not required by our approach. We run a separate ARAH model for each person in the scene using 5 training images. Figure 5.8 shows the training images used in this experiment. It also shows the segmentation masks used for ARAH for 3 people in the scene, which we built using a state-of-the-art method. Figure 5.8 shows additional comparative results for reconstructed appearance and geometry. Fig. 5.9 shows novel view and reconstruction results. Learning for each person separately implies providing erroneous supervision to the model whenever the person is occluded in

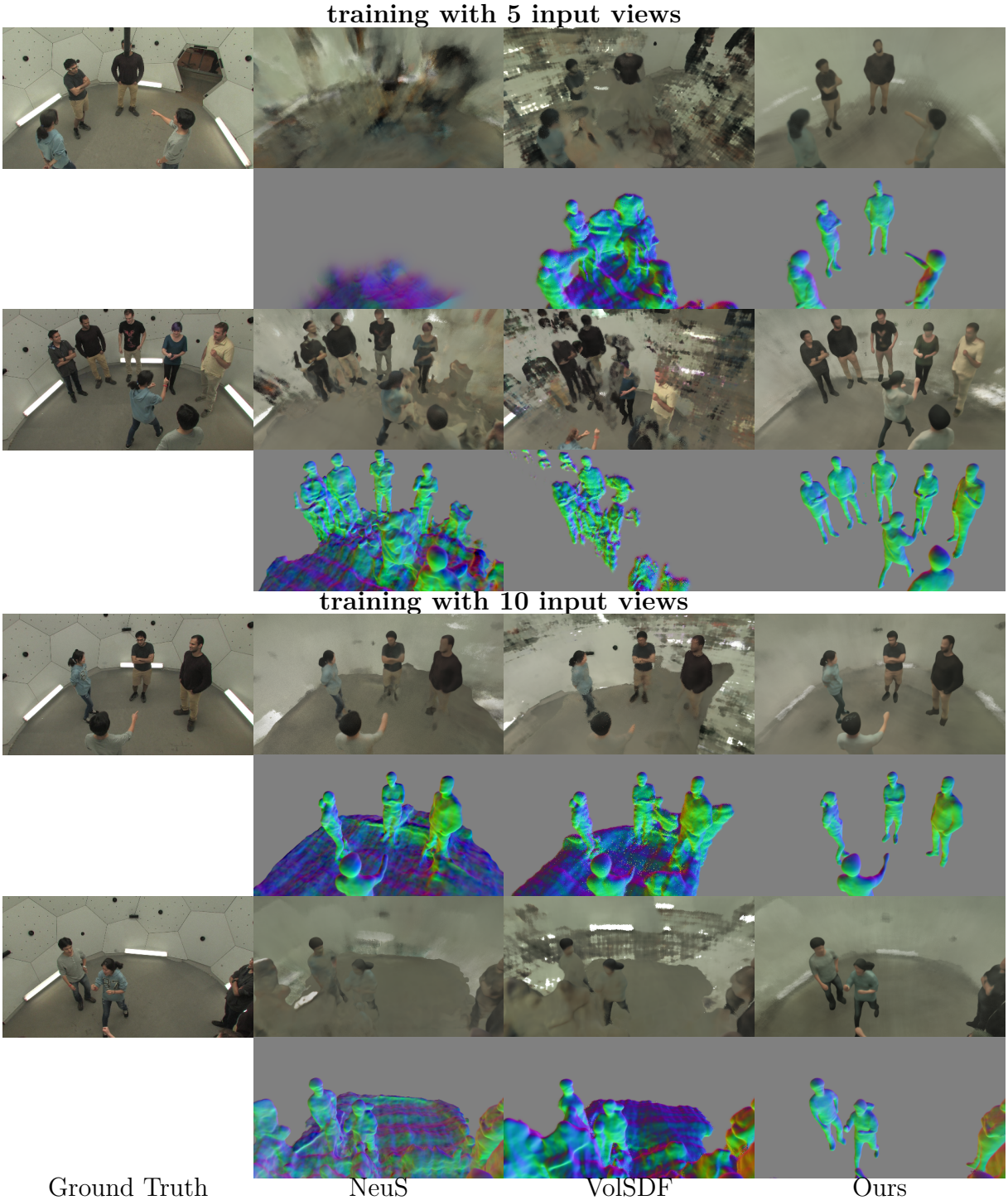


Figure 5.5 – Qualitative comparison against NeuS [34] and VolSDF [77] of synthesized novel views and reconstructed normal images of multiple humans on CMU Panoptic dataset [1], [2], using 5/10 training views.

Figure 5.6 – Qualitative comparison against NeuS [34] and VolSDF [77] of synthesized novel views and reconstructed normal images of multiple humans on CMU Panoptic dataset [1], [2], using 15/20 training views.

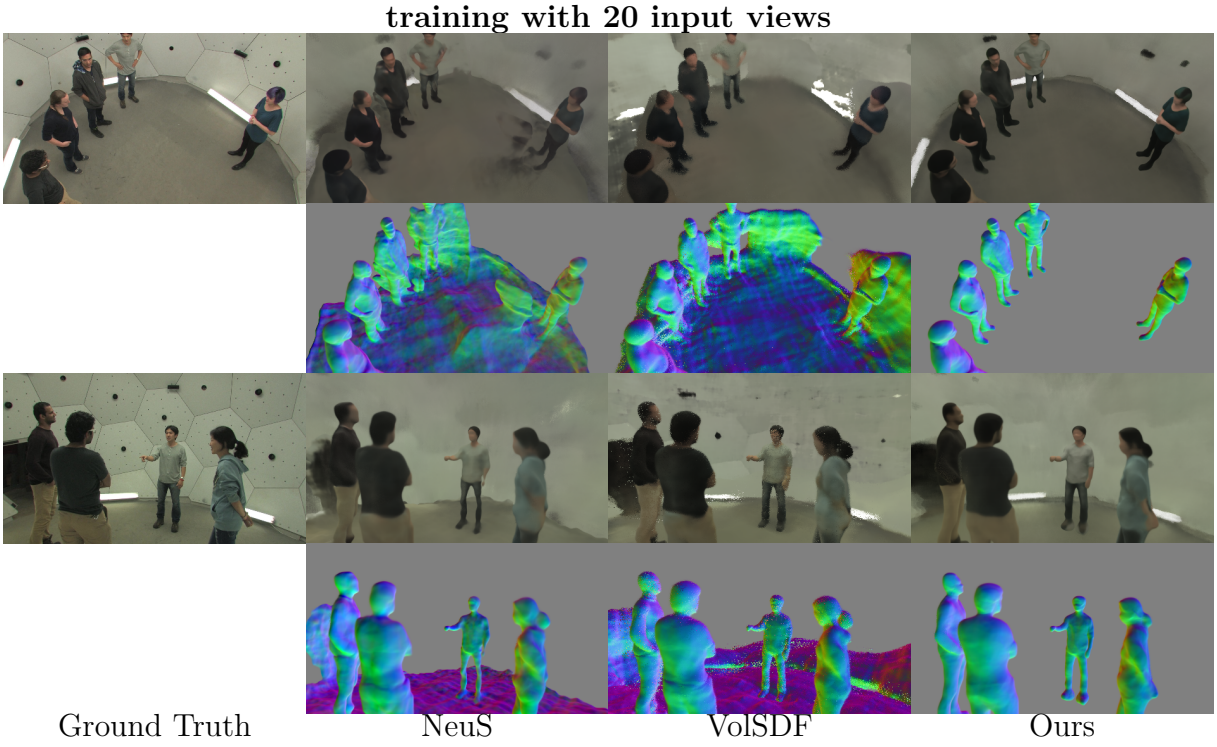


Figure 5.7 – Qualitative comparison against NeuS [34] and VolSDF [77] of synthesized novel views and reconstructed normal images of multiple humans on CMU Panoptic dataset [1], [2], using 20 training views.

the scene or segmentation masks are not accurate. As a result, ARAH's renderings and geometry display many artifacts compared to our results. Our method is more robust to sparser information and occlusions due to the proposed hybrid box-based rendering with geometry constraints. Further, ARAH fails to converge on some persons with a large area of occlusions. Conversely, our method avoids this by learning through rendering the union of SMLP bounding boxes conjointly. We also noticed that ARAH's results are very sensitive to the sparsity and choice of the training views.

Figure 5.8 – The five training views and person segmentations used to produce results of single human.

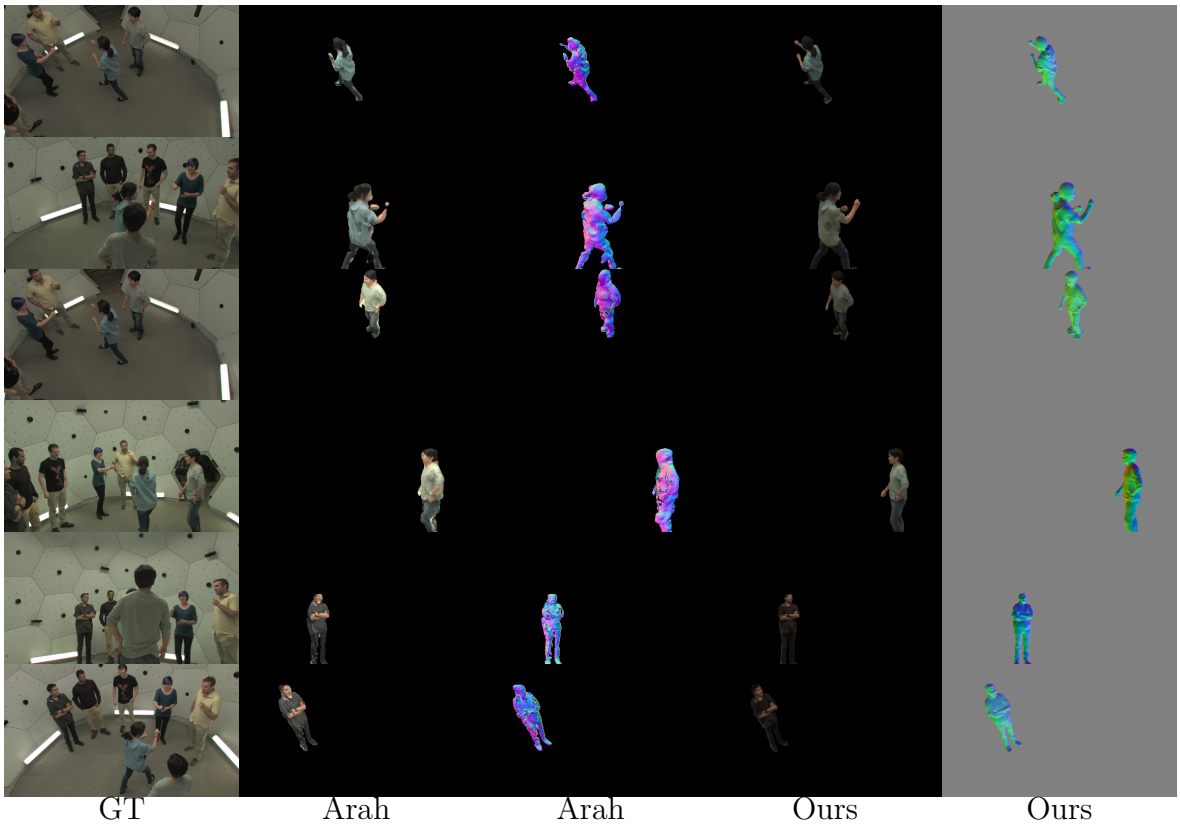


Figure 5.9 – Comparison against single human method ARAH [101] using 5 training views. The average PSNR in these examples is 24.11/**27.40** (ARAH/**Ours**).

Comparison to SparseNeRF. We further compare with a recent NeRF method that was specifically designed to handle sparse views, namely InfoNeRF [53]. We compare both against the original InfoNeRF, and a version of NeuS trained with InfoNeRF's regularization. For this experiment, we use again the CMU Panoptic dataset [1], [2] with five training views. Tab. 5.12 shows that, compared to InfoNeRF and NeuS with InfoNeRF's regularization, our method improves the rendering quality in all of the scenes.

Scene	Method	**PSNR↑**	**SSIM↑**	**LPIPS↓**
1	InfoNeRF	14.64	0.50	0.64
	NeuS w/ info	17.98	0.65	0.58
	Ours	**18.41**	**0.67**	**0.55**
2	InfoNeRF	14.21	0.49	0.63
	NeuS w/ info	18.21	0.64	0.57
	Ours	**19.72**	**0.70**	**0.50**
3	InfoNeRF	13.78	0.45	0.63
	NeuS w/ info	16.31	0.59	0.60
	Ours	**18.57**	**0.66**	**0.52**
4	InfoNeRF	12.26	0.41	0.68
	NeuS w/ info	14.42	0.51	0.60
	Ours	**19.54**	**0.69**	**0.50**
5	InfoNeRF	12.17	0.45	0.63
	NeuS w/ info	17.89	0.60	0.61
	Ours	**19.34**	**0.67**	**0.51**
Ave	InfoNeRF	13.61	0.46	0.64
	NeuS w/ info	16.96	0.60	0.59
	Ours	**19.12**	**0.68**	**0.52**

Table 5.2 – Comparison against sparse-view NeRF approaches: InfoNeRF [53] and NeuS with InfoNeRF's regularizations, on the CMU Panoptic dataset [1], [2] using 5 training views.

Comparison of different input numbers. Fig. 5.10 additionally shows the relationship between the number of training views and the quality of the synthesized images. The fewer the number of views, the harder it is for all methods to reconstruct high-quality images, whereas our approach is more robust to fewer training views. For denser inputs (e.g. more than 20 views), our method reaches a similar albeit slightly better performance than the baselines, since the proposed work focuses on sparse scenarios.

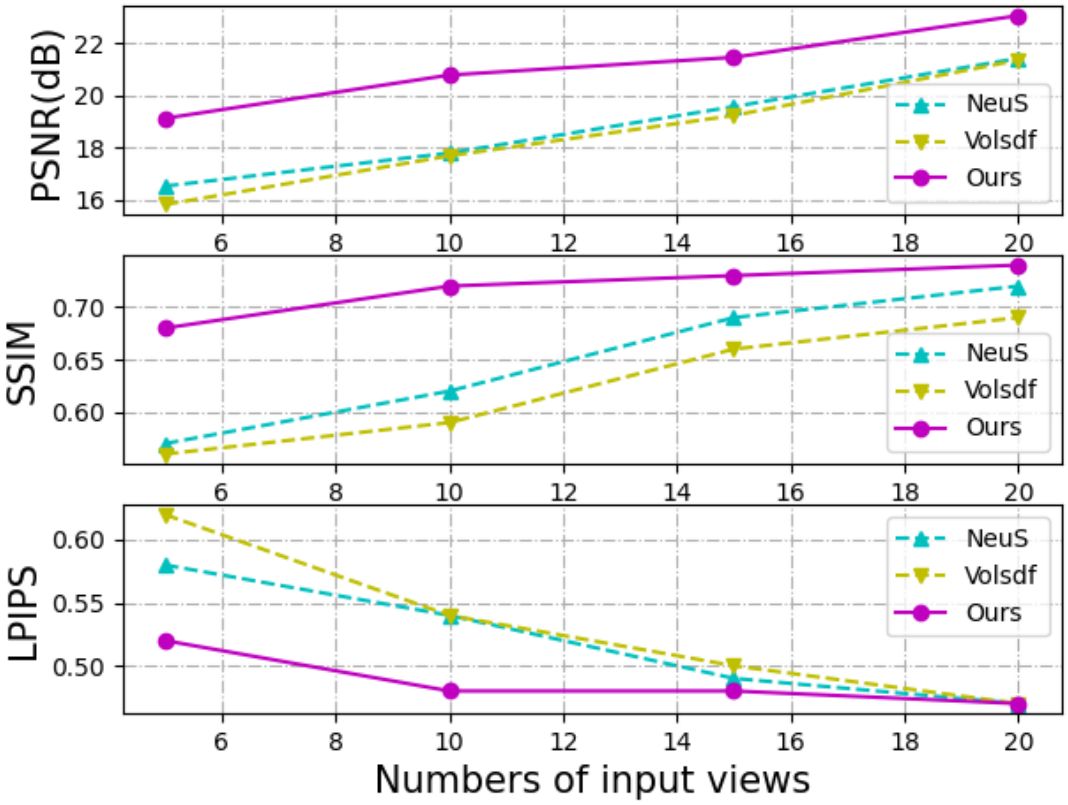


Figure 5.10 – Quantitative comparison of average PSNR (↑), SSIM (↑), and LPIPS (↓) with increased number of training views.

5.4.4 Generalization on synthetic dataset

On the synthetic dataset, we train with 5/10/15 views on each scene and test with 14 fixed views. Table 5.3 reports the average error for all testing views in PSNR, SSIM, and LPIPS metrics. Our method reaches state-of-the-art performance on synthesized novel-view results. Figure 5.11 shows generated novel views and corresponding normal images using 10/15 training images. Our approach can reconstruct the complete geometry of all humans in the scene, while the baseline methods might miss some of the people when they have similar color with the background, e.g. the shadow area in Fig. 5.11.

In the 5/10 input views case, the baseline methods usually fail to reconstruct the full geometry of humans due to the sparse inputs. Thus, we report Chamfer distance in Tab. 5.3 only for the 15-views case. Since the baseline methods usually contain extra floor, for

# Humans	Method	PSNR↑			SSIM↑			LPIPS↓			Chamfer ↓
		5	10	15	5	10	15	5	10	15	15
1	NeuS	14.04	17.89	23.25	0.63	0.72	0.84	**0.55**	0.53	0.44	0.308
	VolSDF	13.93	21.75	25.89	0.61	0.81	0.86	**0.55**	0.51	0.44	0.019
	Ours	**15.36**	**23.85**	**26.28**	**0.65**	**0.84**	**0.87**	**0.55**	**0.43**	**0.41**	**0.018**
5	NeuS	14.15	18.14	18.54	0.61	0.72	0.72	0.54	0.46	0.44	0.321
	VolSDF	12.97	15.11	18.59	0.58	0.63	0.73	0.56	0.55	0.47	0.151
	Ours	**17.63**	**20.10**	**20.33**	**0.71**	**0.79**	**0.77**	**0.47**	**0.40**	**0.40**	**0.020**
10	NeuS	14.09	15.69	19.27	0.58	0.65	0.75	0.52	0.48	0.42	0.383
	VolSDF	12.66	16.99	19.30	0.56	0.70	0.77	0.56	0.50	0.41	0.248
	Ours	**16.52**	**18.39**	**21.01**	**0.65**	**0.71**	**0.80**	**0.50**	**0.44**	**0.37**	**0.043**
Average	NeuS	14.09	17.24	20.35	0.60	0.70	0.77	0.54	0.49	0.43	0.337
	VolSDF	13.18	17.95	21.26	0.58	0.71	0.79	0.56	0.52	0.44	0.139
	Ours	**16.50**	**20.78**	**22.54**	**0.67**	**0.78**	**0.81**	**0.51**	**0.42**	**0.39**	**0.026**

Table 5.3 – Comparison against NeuS [34] and VolSDF [77] on the synthetic dataset, for different numbers of humans in the scene. We measure novel-view synthesis quality in terms of PSNR, SSIM, and LIPIS, as well as geometry error in terms of Chamfer distance.

a fair comparison, we sample points from ground-truth meshes and compute the distance towards the reconstructed mesh for all methods. We also report the bi-directional Chamfer distance. Table 5.3 shows that, with an increasing number of humans in the scene, the quality of the reconstructed geometry of all methods decreases. However, compared with the baselines, our method can better handle multiple human scenes, achieving an order of magnitude less error.

Quantitative results. Table 5.4 provides a full Chamfer distance comparison in the synthetic data setup. In the 5/10 input views case, the baseline methods usually fail to reconstruct the full geometry of humans due to the sparse inputs. Symbol '−' represents cases where the baselines fail to reconstruct a meaningful geometry, and hence the error is too large. To favor the baselines NeuS [34] and VolSDF [77], we computed the uni-directional Chamfer distance from ground-truth to source, as the baselines reconstructed the ground of the scene in addition to the people. For a more standard evaluation, we additionally show here the bi-directional Chamfer distance after removing the floor for the competing methods. Table 5.3 shows that, with an increasing number of humans in the scene, the quality of the reconstructed geometry of all methods decreases. However, Compared with the baselines, our method can better handle multiple human scenes, achieving an order of magnitude less error.

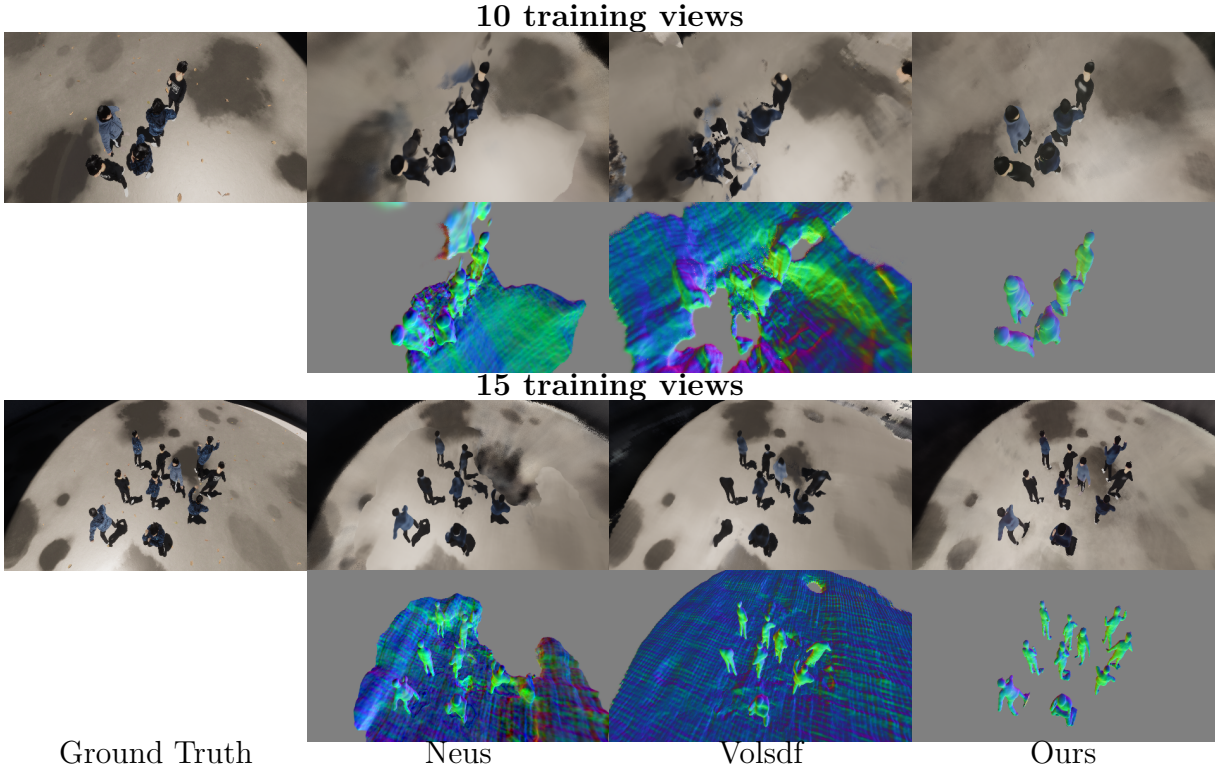


Figure 5.11 – Qualitative comparison of synthesized novel views and reconstructed normal images on the synthetic dataset (MultiHuman-Dataset [3]) with 10 and 15 training views respectively.

People	Method	one-way Chamfer ↓			bidirectional Chamfer ↓		
		5	10	15	5	10	15
1	NeuS	-	-	0.308	-	-	3.026
	VolSDF	-	0.020	0.019	-	**0.039**	0.167
	Ours	0.025	**0.019**	**0.018**	0.271	0.211	**0.154**
5	NeuS	-	-	0.321	-	-	3.044
	VolSDF	-	-	0.151	-	-	1.478
	Ours	0.025	0.023	**0.020**	0.391	0.289	**0.138**
10	NeuS	-	-	0.383	-	-	4.639
	VolSDF	-	-	0.248	-	-	1.579
	Ours	0.082	0.063	**0.043**	0.111	0.085	**0.081**

Table 5.4 – Geometry reconstruction error under a different number of people, compared to NeuS [34] and VolSDF [77] using the synthetic dataset, and 5/10/15 views for training. Symbol '–' represents cases where the baselines fail to reconstruct a meaningful geometry.

5.4.5 Ablation and analysis

Provided loss function. To prove the effectiveness of our proposed components we performed ablation studies on the CMU Panoptic dataset [1], [2]. We demonstrate quantitative comparisons in Table 5.5 and qualitative results in Fig. 5.12. We test the following settings: **Without geometry regularization ("w/o geometry").** We compare our full model against the model without geometry regularization (Sec. 5.3.2) and SDF uncertainty regularization (Eq. 5.13). We can see here that, although the method is still capable of isolating humans thanks to the bounding box rendering, both geometry and novel views are much less accurate, and the rendered images exhibit background artifacts and overly smooth results.

Without ray consistency loss ('w/o ray loss'). Here we remove the proposed ray consistency loss, without which the average rendering quality also degrades.

Without saturation loss. Finally, we remove the saturation loss from our methods, which decreases by about 0.5 in PSNR on average. Fig. 5.12 shows that, without this, the image tone can contain artifacts due to changes in lighting (see for example the back of the rightmost subject).

Method	PSNR↑		SSIM↑		LPIPS↓	
	5	15	5	15	5	15
Neus [34]	16.87	19.40	0.60	0.70	0.51	0.53
Volsdf[77]	16.03	19.40	0.53	0.67	0.60	0.49
w/o Geometry	17.54	20.28	0.60	0.70	0.53	0.48
w/o Ray loss	19.07	20.95	0.67	0.72	0.52	0.47
w/o Saturation	18.95	20.92	0.65	0.72	0.54	0.49
Ours(Full)	**19.72**	**21.40**	**0.70**	**0.73**	**0.50**	**0.48**

Table 5.5 – Ablation study on the CMU Panoptic dataset [1], [2] with 5/15 training views respectively. Comparison against our method without geometric regularization (w/o Geometry), without ray consistency regularization (w/o Ray loss), and without saturation regularization (w/o Saturation).

Comparisons with a varying number of people. In Fig. 5.7 we provide additional qualitative comparisons against NeuS [34] and VolSDF [77], where we show results on the CMU Panoptic dataset [1], [2] with a varying number of people in the scene (3-7). Note here how increasing the number of people causes the baselines to reduce the quality of

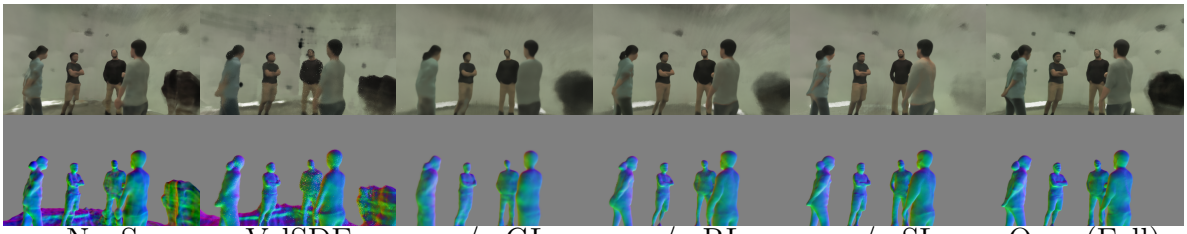


Figure 5.12 – Ablation study on CMU Panoptic dataset [1], [2]. Comparison against our method without geometric regularization (w/o geometry), our method without ray consistency regularization (w/o ray loss), and our method without saturation regularization (w/o saturation).

the results, mixing background with humans or generating noisy geometries. Meanwhile, our method performs consistently, independently of the number of people.

5.5 Additional applications

We show here how our method can be used to perform post-learning scene editing without any additional training. Thanks to the human bounding-box-based modeling of the foreground scene, it is straightforward to rigidly transform or omit each person by simply applying, before rendering, the corresponding manipulation to the points sampled inside the defined bounding box. Figure 5.15 shows qualitative results of such application, trained on scene #5 from the CMU Panoptic dataset [1], [2] using 20 training views. We can see here that our approach can generate realistic new scenes as well as plausible paintings of the missing regions.

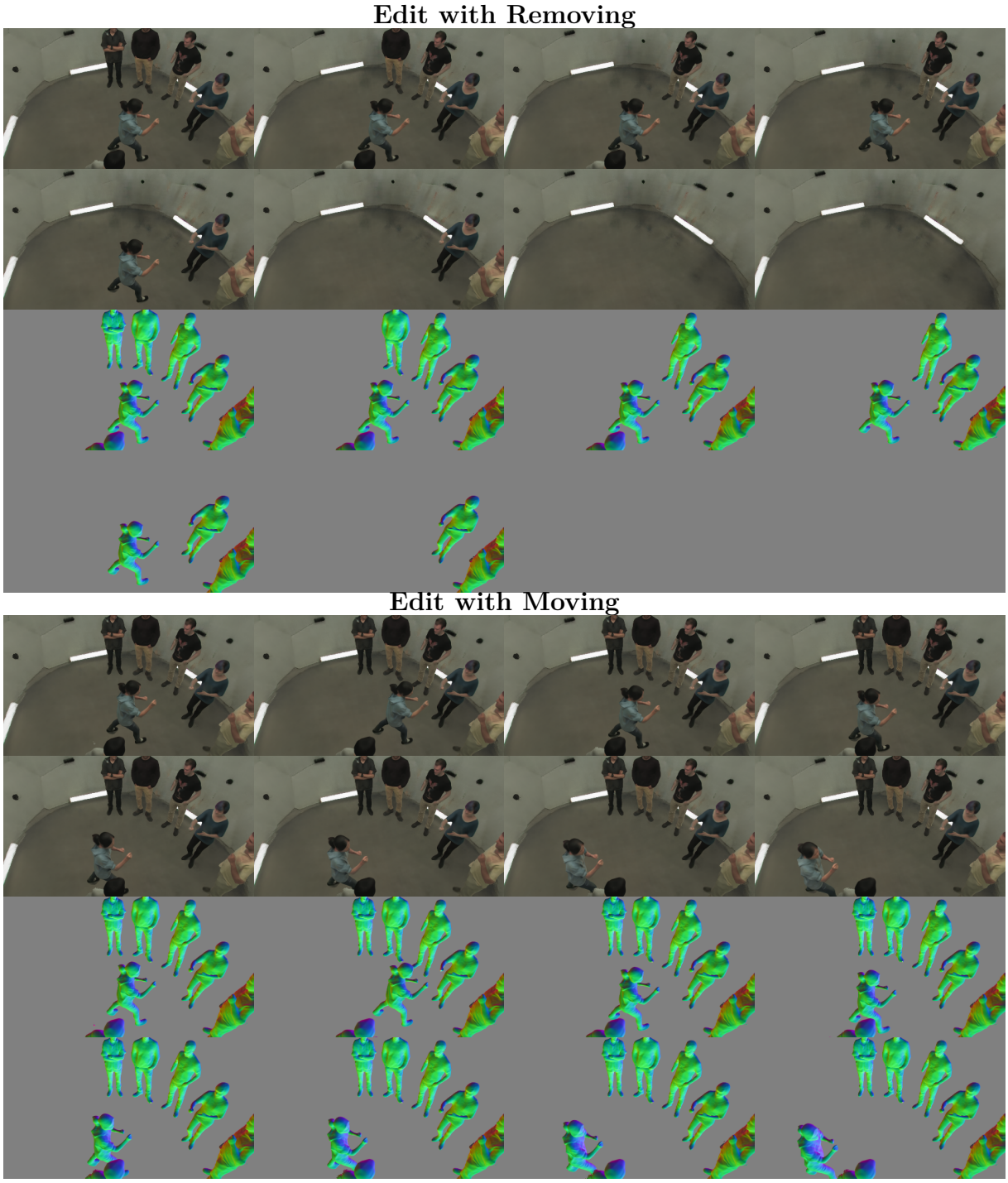


Figure 5.14 – Qualitative results for the editing application. We show synthesized novel views and reconstructed normal images of multiple humans when (1) removing, (2) translating, (3) rotating, and (4) scaling subjects in the scene.

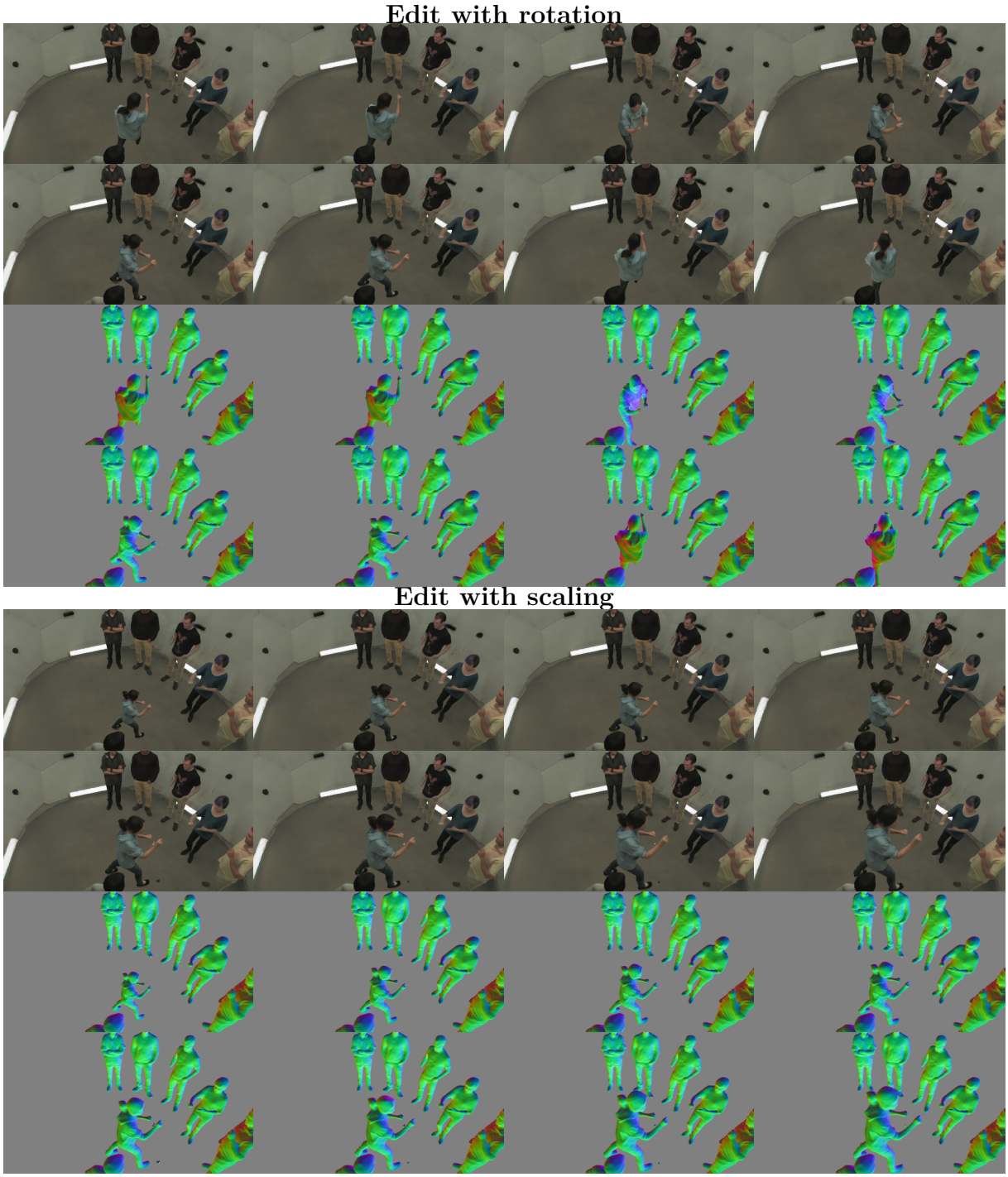


Figure 5.15 – Qualitative results for the editing application. We show synthesized novel views and reconstructed normal images of multiple humans when (1) removing, (2) translating, (3) rotating, and (4) scaling subjects in the scene.

5.6 Conclusion

We presented an approach for novel view synthesis of multiple humans from a sparse set of input views. To achieve this, we proposed geometric regularizations that improve geometry training by leveraging a pre-computed SMPL model, along with a patch-based ray consistency loss and a saturation loss that help with novel-view renderings in the sparse-view setting. Our experiments showed state-of-the-art performance for multiple human geometry and appearance reconstruction on real multi-human datasets (CMU Panoptic [1], [2]) and on synthetic data (MultiHuman-Dataset [3]). Our method still has several limitations. First, we rely on SMPL fittings which might not always be accurate, particularly for scenes with a very large number of humans. A possible solution is to improve the SMPL reconstructions while training the geometry and appearance networks. Second, our method does not model close human interactions, as this is a much more challenging case. Addressing this is an interesting direction for future work.

Chapter 6

Conclusion

6.1 Summary

Novel view synthesis has been one of the most popular research fields recently. Though novel view synthesis from sparse views has shown great potential in various computer graphics and computer vision applications, it remains challenging to infer the underlying 3D structure of the object or scene and infer the appearance of unseen views from a limited number of inputs. This thesis has addressed these challenges regarding novel view synthesis from sparse inputs, making the process more practical in real-world scenarios where obtaining massive datasets may be challenging. We have investigated those problems and proposed different solutions for three important techniques regarding novel view synthesis from sparse inputs: neural radiance fields, light field networks, and novel view synthesis of multi-humans, which are introduced in three chapters (Ch.2, Ch.3, Ch.4) respectively. Let's summarize our work:

Neural radiance field from sparse inputs: We have explored neural radiance fields suffering from quality degradation from sparse views. We have presented a novel approach to improve neural radiance field (NeRF) from sparse inputs to address this challenge. Our proposed methods include a global sampling strategy, geometry regularization using augmented pseudo-views, and a local patch sampling scheme with patch-based regularization. We introduced the use of depth information for explicit geometry regularization. The proposed approach outperformed several baselines on real benchmarks and achieved state-of-the-art results. However, one of the limitations is that it needs accurate depth information from sparse views. In this thesis, we have chosen to use sensor depth from the dataset, while future studies could explore how using estimated depth from the network. Moreover, future research could incorporate improving the neural radiance field and implicit surface reconstruction from sparse RGBD inputs.

Neural Light field from sparse inputs: We proposed a novel approach based on a neural light field representation for a few-shot novel view synthesis. Our proposed method

employs an implicit neural network conditioned on local ray features generated from a coarse volumetric rendering. We explored different convolutional neural network architectures. With the depth-based sampling and MVS network, our methods could generalize realistic appearance across scenes. The proposed method achieves competitive performance across different datasets and offers a much faster rendering speed. The proposed methods allow us to generalize well to novel views of seen and unseen scenes from a few-shot input. Meanwhile, our approach significantly reduces the computational cost of rendering while maintaining complex relationship learning. While our method offers an efficient rendering, it still has difficulties reproducing the highest level of details in real large images as the NeRF-based methods. This is due to the reduced resolution of our feature volume and our coarse feature rendering, which contributes to reducing the memory footprint.

3D shape and radiance rendering of multi-Human from sparse inputs: We proposed a learning-based method for generating multiple humans from sparse images. Our approach addressed the challenges of occlusion and clutter in multi-human scenes by incorporating geometry constraints using pre-computed meshes, patch-based ray regularization for appearance consistency, and saturation regularization for robust optimization. Extensive experiments on real and synthetic data demonstrated the benefits of our method and its state-of-the-art performance against existing neural reconstruction methods on real multi-human datasets (CMU Panoptic [1], [2]) and on synthetic data (MultiHuman-Dataset [3]). Our approach still has several limitations. First, we rely on SMPL fittings, which might only sometimes be accurate, particularly for scenes with many humans. A possible solution is to improve the SMPL reconstructions while training the geometry and appearance networks. Second, our method does not model close human interactions, as this is a much more challenging case.

To sum up, our thesis has focused on an important topic regarding novel view synthesis from sparse inputs. Since most novel view synthesis algorithms require a large amount of data to train accurately, our research aims to develop an algorithm that can learn from limited or sparse data. Our investigated algorithms include neural radiance fields, light field networks, and multi-human rendering and reconstruction. We have proposed solutions to improve those novel view synthesis algorithms from sparse views. Our proposed methods have advanced the state-of-the-art in each specific topic. We wish this work could push the boundaries further and help researchers for future advancements in this research area.

6.2 Future work and perspectives

In the future, there are several potential avenues for further exploration and improvement based on the findings and methodologies presented in this thesis:

In Chapter 3, our method regularizes the neural radiance fields via depth regularization and image-based warping. More solutions for improving the neural radiance fields from sparse inputs can be explored, such as leveraging additional modalities such as surface normal, incorporating temporal information for video-based synthesis, or using unsupervised or self-supervised learning approaches to reduce the reliance on labeled data. One limitation of our method is that it depends on accurate depth information. Future work can explore how to replace the sensor depth with a monocular depth estimated by neural networks or use some depth information using structure from motion. Also, depth information could improve the quality of estimated surface and rendered appearances. More work could be explored here.

In Chapter 4, our methods utilize pluckier coordinates and condition these ray representations with extracted feature volumes. Future work to enhance rendering quality includes leveraging more advanced conditioning methods. For example, incorporating attention or diffusion models into the feature volume extraction process may improve results. Attention models help the network to learn more relevant information, allowing it to capture more intricate details and improve overall rendering quality. Diffusion models, on the other hand, can facilitate the transfer of information across different regions, enabling the network to understand better and model complex relationships within the scene. Additionally, advanced pre-trained models on big datasets could help the network better capture intricate scene details, resulting in higher-quality renderings.

In Chapter 5, our method of rendering multi-humans by utilizing SMPL from sparse inputs opens up possibilities for more complex scene generation. Since one limitation is that we focus on static scenes, we could explore how to improve video rendering results by considering temporal information. Future work can focus on extending the proposed approach to multi-human rendering from monocular videos or multi-view videos, enabling the synthesis of dynamic multi-human scenes. Additionally, addressing challenges posed by diverse objects, varying backgrounds, and challenging lighting conditions in more complex scenes would be an interesting direction for further exploration.

Despite the above possibilities in each topic, some common extensions regarding our current works exist. For instance, integrating neural radiance and light fields might yield

even more powerful few-shot novel view synthesis techniques. Integrating user interaction and feedback mechanisms into our frameworks would enable users to participate actively in the synthesis or reconstruction process, improving the quality and personalizing the results. In addition, estimation of scene geometry, such as depth maps and surface normals, is crucial for synthesizing novel views. Future work will focus on improving the robustness and accuracy of geometry estimation methods, leveraging deep learning approaches, and incorporating semantic scene understanding. Moreover, novel view synthesis techniques often struggle to generalize well to unseen or diverse scenes or objects. Future research will aim to improve the generalization capabilities of these methods, enabling them to work effectively across different domains, lighting conditions, and object categories.

In conclusion, novel view synthesis has gained significant attention recently due to its applications in various fields. This thesis has made valuable contributions by addressing the challenges of synthesizing novel views from sparse inputs. Advancements in these areas will contribute to the broader adoption and practical application of novel view synthesis technologies in various industries and academic domains. The future work and perspectives outlined above aim to push the boundaries further, fostering progress in this exciting area of research and benefiting the broader community working in related domains, such as virtual reality, augmented reality, and computer graphics.

BIBLIOGRAPHY

[1] T. Simon, H. Joo, and Y. Sheikh, « Hand keypoint detection in single images using multiview bootstrapping », *CVPR*, 2017.

[2] H. Joo, T. Simon, X. Li, *et al.*, « Panoptic studio: a massively multiview system for social interaction capture », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[3] Y. Zheng, R. Shao, Y. Zhang, *et al.*, « Deepmulticap: performance capture of multiple characters using sparse multiview cameras », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6239–6249.

[4] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, « Nerf: representing scenes as neural radiance fields for view synthesis », *in European conference on computer vision*, Springer, 2020, pp. 405–421.

[5] K. Rematas, C. H. Nguyen, T. Ritschel, M. Fritz, and T. Tuytelaars, « Novel views of objects from a single image », *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, *8*, pp. 1576–1590, 2016.

[6] V. Sitzmann, S. Rezchikov, B. Freeman, J. Tenenbaum, and F. Durand, « Light field networks: neural scene representations with single-evaluation rendering », *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[7] W. Xia and J.-H. Xue, « A survey on 3d-aware image synthesis », *arXiv preprint arXiv:2210.14267*, 2022.

[8] R. Tucker and N. Snavely, « Single-view view synthesis with multiplane images », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 551–560.

[9] M.-L. Shih, S.-Y. Su, J. Kopf, and J.-B. Huang, « 3d photography using context-aware layered depth inpainting », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8028–8038.

[10] B. Mildenhall, P. P. Srinivasan, R. Ortiz-Cayon, *et al.*, « Local light field fusion: practical view synthesis with prescriptive sampling guidelines », *ACM Transactions on Graphics (TOG)*, vol. 38, *4*, pp. 1–14, 2019.

[11] S. Niklaus, L. Mai, J. Yang, and F. Liu, « 3d ken burns effect from a single image », *ACM Transactions on Graphics (ToG)*, vol. 38, *6*, pp. 1–15, 2019.

[12] C. Fehn, « Depth-image-based rendering (dibr), compression, and transmission for a new approach on 3d-tv », *in Stereoscopic displays and virtual reality systems XI*, SPIE, vol. 5291, 2004, pp. 93–104.

[13] S. Zinger, L. Do, and P. De With, « Free-viewpoint depth image based rendering », *Journal of visual communication and image representation*, vol. 21, *5-6*, pp. 533–541, 2010.

[14] S. B. Kang, Y. Li, X. Tong, H.-Y. Shum, *et al.*, « Image-based rendering », *Foundations and Trends® in Computer Graphics and Vision*, vol. 2, *3*, pp. 173–258, 2007.

[15] Y. Furukawa, C. Hernández, *et al.*, « Multi-view stereo: a tutorial », *Foundations and Trends® in Computer Graphics and Vision*, vol. 9, *1-2*, pp. 1–148, 2015.

[16] Y. Yao, Z. Luo, S. Li, T. Fang, and L. Quan, « Mvsnet: depth inference for unstructured multi-view stereo », *in Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 767–783.

[17] S. Albawi, T. A. Mohammed, and S. Al-Zawi, « Understanding of a convolutional neural network », *in 2017 international conference on engineering and technology (ICET)*, Ieee, 2017, pp. 1–6.

[18] I. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, « Generative adversarial networks », *Communications of the ACM*, vol. 63, *11*, pp. 139–144, 2020.

[19] C. Doersch, « Tutorial on variational autoencoders », *arXiv preprint arXiv:1606.05908*, 2016.

[20] M. Tatarchenko, A. Dosovitskiy, and T. Brox, « Single-view to multi-view: reconstructing unseen views with a convolutional network », *CoRR abs/1511.06702*, vol. 1, *2*, p. 2, 2015.

[21] D. E. Worrall, S. J. Garbin, D. Turmukhambetov, and G. J. Brostow, « Interpretable transformations with encoder-decoder networks », *in Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5726–5735.

[22] T. Zhou, S. Tulsiani, W. Sun, J. Malik, and A. A. Efros, « View synthesis by appearance flow », *in European conference on computer vision*, Springer, 2016, pp. 286–301.

[23] E. Park, J. Yang, E. Yumer, D. Ceylan, and A. C. Berg, « Transformation-grounded image generation network for novel 3d view synthesis », *in Proceedings of the ieee conference on computer vision and pattern recognition*, 2017, pp. 3500–3509.

[24] S.-H. Sun, M. Huh, Y.-H. Liao, N. Zhang, and J. J. Lim, « Multi-view to novel view: synthesizing novel views with self-learned confidence », *in Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 155–171.

[25] S. Lombardi, T. Simon, J. Saragih, G. Schwartz, A. Lehrmann, and Y. Sheikh, « Neural volumes: learning dynamic renderable volumes from images », *arXiv preprint arXiv:1906.07751*, 2019.

[26] O. Wiles, G. Gkioxari, R. Szeliski, and J. Johnson, « Synsin: end-to-end view synthesis from a single image », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7467–7477.

[27] K. Olszewski, S. Tulyakov, O. Woodford, H. Li, and L. Luo, « Transformable bottleneck networks », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7648–7657.

[28] E. Dupont, M. B. Martin, A. Colburn, A. Sankar, J. Susskind, and Q. Shan, « Equivariant neural rendering », *in International Conference on Machine Learning*, PMLR, 2020, pp. 2761–2770.

[29] R. Hu, N. Ravi, A. C. Berg, and D. Pathak, « Worldsheet: wrapping the world in a 3d sheet for view synthesis from a single image », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 12 528–12 537.

[30] X. Chen, J. Song, and O. Hilliges, « Monocular neural image based rendering with continuous view control », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4090–4100.

[31] X. Shen, J. Plested, Y. Yao, and T. Gedeon, « Pairwise-gan: pose-based view synthesis through pair-wise training », *in International conference on neural information processing*, Springer, 2020, pp. 507–515.

[32] R. Jensen, A. Dahl, G. Vogiatzis, E. Tola, and H. Aanæs, « Large scale multi-view stereopsis evaluation », *in Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 406–413.

[33] V. Sitzmann, M. Zollhöfer, and G. Wetzstein, « Scene representation networks: continuous 3d-structure-aware neural scene representations », *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[34] P. Wang, L. Liu, Y. Liu, C. Theobalt, T. Komura, and W. Wang, « Neus: learning neural implicit surfaces by volume rendering for multi-view reconstruction », *arXiv preprint arXiv:2106.10689*, 2021.

[35] P. Kellnhofer, L. C. Jebe, A. Jones, R. Spicer, K. Pulli, and G. Wetzstein, « Neural lumigraph rendering », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4287–4297.

[36] K. Rematas, R. Martin-Brualla, and V. Ferrari, « Sharf: shape-conditioned radiance fields from a single view », *arXiv preprint arXiv:2102.08860*, 2021.

[37] Y. Wei, S. Liu, Y. Rao, W. Zhao, J. Lu, and J. Zhou, « Nerfingmvs: guided optimization of neural radiance fields for indoor multi-view stereo », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5610–5619.

[38] G. Riegler and V. Koltun, « Free view synthesis », *in European Conference on Computer Vision*, Springer, 2020, pp. 623–640.

[39] G. Riegler and V. Koltun, « Stable view synthesis », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12 216–12 225.

[40] J. Thies, M. Zollhöfer, and M. Nießner, « Deferred neural rendering: image synthesis using neural textures », *ACM Transactions on Graphics (TOG)*, vol. 38, *4*, pp. 1–12, 2019.

[41] K.-A. Aliev, A. Sevastopolsky, M. Kolos, D. Ulyanov, and V. Lempitsky, « Neural point-based graphics », *in European Conference on Computer Vision*, Springer, 2020, pp. 696–712.

[42] A. Yu, R. Li, M. Tancik, H. Li, R. Ng, and A. Kanazawa, « Plenoctrees for real-time rendering of neural radiance fields », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5752–5761.

[43] A. Chen, Z. Xu, A. Geiger, J. Yu, and H. Su, « Tensorf: tensorial radiance fields », *arXiv preprint arXiv:2203.09517*, 2022.

[44] P. Hedman, P. P. Srinivasan, B. Mildenhall, J. T. Barron, and P. Debevec, « Baking neural radiance fields for real-time view synthesis », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5875–5884.

[45] T. Müller, A. Evans, C. Schied, and A. Keller, « Instant neural graphics primitives with a multiresolution hash encoding », *arXiv preprint arXiv:2201.05989*, 2022.

[46] C. Sun, M. Sun, and H.-T. Chen, « Improved direct voxel grid optimization for radiance fields reconstruction », *arXiv preprint arXiv:2206.05085*, 2022.

[47] S. Fridovich-Keil, A. Yu, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa, « Plenoxels: radiance fields without neural networks », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5501–5510.

[48] A. Yu, S. Fridovich-Keil, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa, « Plenoxels: radiance fields without neural networks », *arXiv preprint arXiv:2112.05131*, 2021.

[49] S. J. Garbin, M. Kowalski, M. Johnson, J. Shotton, and J. Valentin, « Fastnerf: high-fidelity neural rendering at 200fps », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 14 346–14 355.

[50] A. Jain, M. Tancik, and P. Abbeel, « Putting nerf on a diet: semantically consistent few-shot view synthesis », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5885–5894.

[51] B. Roessle, J. T. Barron, B. Mildenhall, P. P. Srinivasan, and M. Nießner, « Dense depth priors for neural radiance fields from sparse input views », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 892–12 901.

[52] M. Niemeyer, J. T. Barron, B. Mildenhall, M. S. Sajjadi, A. Geiger, and N. Radwan, « Regnerf: regularizing neural radiance fields for view synthesis from sparse inputs », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5480–5490.

[53] M. Kim, S. Seo, and B. Han, « Infonerf: ray entropy minimization for few-shot neural volume rendering », *in CVPR*, 2022.

[54] M. Niemeyer, J. T. Barron, B. Mildenhall, M. S. Sajjadi, A. Geiger, and N. Radwan, « Regnerf: regularizing neural radiance fields for view synthesis from sparse inputs », *in CVPR*, 2022.

[55] A. Chen, Z. Xu, F. Zhao, *et al.*, « Mvsnerf: fast generalizable radiance field reconstruction from multi-view stereo », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 14 124–14 133.

[56] A. Yu, V. Ye, M. Tancik, and A. Kanazawa, « Pixelnerf: neural radiance fields from one or few images », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4578–4587.

[57] L. Liu, J. Gu, K. Zaw Lin, T.-S. Chua, and C. Theobalt, « Neural sparse voxel fields », *Advances in Neural Information Processing Systems*, vol. 33, pp. 15 651–15 663, 2020.

[58] C. Reiser, S. Peng, Y. Liao, and A. Geiger, « Kilonerf: speeding up neural radiance fields with thousands of tiny mlps », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 14 335–14 345.

[59] K. Deng, A. Liu, J.-Y. Zhu, and D. Ramanan, « Depth-supervised nerf: fewer views and faster training for free », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 882–12 891.

[60] A. Cao, C. Rockwell, and J. Johnson, « Fwd: real-time novel view synthesis with forward warping and depth », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 15 713–15 724.

[61] A. Trevithick and B. Yang, « Grf: learning a general radiance field for 3d representation and rendering », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 15 182–15 192.

[62] Q. Wang, Z. Wang, K. Genova, *et al.*, « Ibrnet: learning multi-view image-based rendering », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4690–4699.

[63] Y. Yoon, H.-G. Jeon, D. Yoo, J.-Y. Lee, and I. So Kweon, « Learning a deep convolutional network for light-field image super-resolution », *in Proceedings of the IEEE international conference on computer vision workshops*, 2015, pp. 24–32.

[64] H. Wang, J. Ren, Z. Huang, *et al.*, « R2l: distilling neural radiance field to neural light field for efficient novel view synthesis », *in Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXI*, Springer, 2022, pp. 612–629.

[65] G. Wu, M. Zhao, L. Wang, Q. Dai, T. Chai, and Y. Liu, « Light field reconstruction using deep convolutional network on epi », *in Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6319–6327.

[66] Z. Shi, S. Zheng, X. Huang, M. Xu, and L. Han, « Light-field depth estimation using rnn and crf », *in 2022 7th International Conference on Image, Vision and Computing (ICIVC)*, IEEE, 2022, pp. 725–729.

[67] Z. Hu, H. W. F. Yeung, X. Chen, Y. Y. Chung, and H. Li, « Efficient light field reconstruction via spatio-angular dense network », *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–14, 2021.

[68] M. Suhail, C. Esteves, L. Sigal, and A. Makadia, « Generalizable patch-based neural rendering », *in Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXII*, Springer, 2022, pp. 156–174.

[69] B. Attal, J.-B. Huang, M. Zollhöfer, J. Kopf, and C. Kim, « Learning neural light fields with ray-space embedding », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 19 819–19 829.

[70] M. Suhail, C. Esteves, L. Sigal, and A. Makadia, « Light field neural rendering », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 8269–8279.

[71] L. Yariv, Y. Kasten, D. Moran, *et al.*, « Multiview neural surface reconstruction by disentangling geometry and appearance », *Advances in Neural Information Processing Systems*, vol. 33, pp. 2492–2502, 2020.

[72] M. Niemeyer, L. Mescheder, M. Oechsle, and A. Geiger, « Differentiable volumetric rendering: learning implicit 3d representations without 3d supervision », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3504–3515.

[73] K. Genova, F. Cole, A. Sud, A. Sarna, and T. Funkhouser, « Local deep implicit functions for 3d shape », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 4857–4866.

[74] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, « Deepsdf: learning continuous signed distance functions for shape representation », *in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 165–174.

[75] S. Peng, M. Niemeyer, L. Mescheder, M. Pollefeys, and A. Geiger, « Convolutional occupancy networks », *in Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, Springer, 2020, pp. 523–540.

[76] M. Oechsle, S. Peng, and A. Geiger, « Unisurf: unifying neural implicit surfaces and radiance fields for multi-view reconstruction », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5589–5599.

[77] L. Yariv, J. Gu, Y. Kasten, and Y. Lipman, « Volume rendering of neural implicit surfaces », *Advances in Neural Information Processing Systems*, vol. 34, pp. 4805–4815, 2021.

[78] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black, « Keep it smpl: automatic estimation of 3d human pose and shape from a single image », *in European conference on computer vision*, Springer, 2016, pp. 561–578.

[79] V. Choutas, G. Pavlakos, T. Bolkart, D. Tzionas, and M. J. Black, « Monocular expressive body regression through body-driven attention », *in European Conference on Computer Vision*, Springer, 2020, pp. 20–40.

[80] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, « End-to-end recovery of human shape and pose », *in Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7122–7131.

[81] L. Muller, A. A. Osman, S. Tang, C.-H. P. Huang, and M. J. Black, « On self-contact and human pose », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9990–9999.

[82] W. Liu and T. Mei, « Recent advances of monocular 2d and 3d human pose estimation: a deep learning perspective », *ACM Computing Surveys (CSUR)*, 2022.

[83] Y. Yuan, U. Iqbal, P. Molchanov, K. Kitani, and J. Kautz, « Glamr: global occlusion-aware human mesh recovery with dynamic cameras », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[84] M. Kocabas, N. Athanasiou, and M. J. Black, « VIBE: video inference for human body pose and shape estimation », *in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020)*, Piscataway, NJ: IEEE, Jun. 2020, pp. 5252–5262. DOI: 10.1109/CVPR42600.2020.00530.

[85] T. Alldieck, M. Magnor, W. Xu, C. Theobalt, and G. Pons-Moll, « Video based reconstruction of 3d people models », *in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8387–8397.

[86] T. Yu, K. Guo, F. Xu, *et al.*, « Bodyfusion: real-time capture of human motion and surface geometry using a single depth camera », *in Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 910–919.

[87] T. Yu, Z. Zheng, K. Guo, *et al.*, « Doublefusion: real-time capture of human performances with inner body shapes from a single depth sensor », *in Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7287–7296.

[88] A. Burov, M. Nießner, and J. Thies, « Dynamic surface function networks for clothed human bodies », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 754–10 764.

[89] J. Starck and A. Hilton, « Surface capture for performance-based animation », *IEEE computer graphics and applications*, vol. 27, *3*, pp. 21–31, 2007.

[90] A. Collet, M. Chuang, P. Sweeney, *et al.*, « High-quality streamable free-viewpoint video », *ACM Transactions on Graphics (ToG)*, vol. 34, *4*, pp. 1–13, 2015.

[91] K. Guo, P. Lincoln, P. Davidson, *et al.*, « The relightables: volumetric performance capture of humans with realistic relighting », *ACM Transactions on Graphics (ToG)*, vol. 38, *6*, pp. 1–19, 2019.

[92] Z. Huang, T. Li, W. Chen, *et al.*, « Deep volumetric video from very sparse multi-view performance capture », *in Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 336–354.

[93] J. Gall, C. Stoll, E. De Aguiar, C. Theobalt, B. Rosenhahn, and H.-P. Seidel, « Motion capture using joint skeleton tracking and surface estimation », *in 2009 IEEE Conference on Computer Vision and Pattern Recognition*, Ieee, 2009, pp. 1746–1753.

[94] D. Vlasic, I. Baran, W. Matusik, and J. Popović, « Articulated mesh animation from multi-view silhouettes », *in ACM SIGGRAPH 2008 papers*, 2008, pp. 1–9.

[95] J. Carranza, C. Theobalt, M. A. Magnor, and H.-P. Seidel, « Free-viewpoint video of human actors », *ACM transactions on graphics (TOG)*, vol. 22, *3*, pp. 569–577, 2003.

[96] E. De Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H.-P. Seidel, and S. Thrun, « Performance capture from sparse multi-view video », *in ACM SIGGRAPH 2008 papers*, 2008, pp. 1–10.

[97] Y. Huang, F. Bogo, C. Lassner, *et al.*, « Towards accurate marker-less human shape and pose estimation over time », *in 2017 international conference on 3D vision (3DV)*, IEEE, 2017, pp. 421–430.

[98] A. O. Balan, L. Sigal, M. J. Black, J. E. Davis, and H. W. Haussecker, « Detailed human shape and pose from images », *in 2007 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2007, pp. 1–8.

[99] S. Peng, Y. Zhang, Y. Xu, *et al.*, « Neural body: implicit neural representations with structured latent codes for novel view synthesis of dynamic humans », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9054–9063.

[100] J. Liang and M. C. Lin, « Shape-aware human pose and shape reconstruction using multi-view images », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4352–4362.

[101] S. Wang, K. Schwarz, A. Geiger, and S. Tang, « Arah: animatable volume rendering of articulated human sdfs », *in European Conference on Computer Vision*, 2022.

[102] L. Liu, M. Habermann, V. Rudnev, K. Sarkar, J. Gu, and C. Theobalt, « Neural actor: neural free-view synthesis of human actors with pose control », *ACM Transactions on Graphics (TOG)*, vol. 40, *6*, pp. 1–16, 2021.

[103] C.-Y. Weng, B. Curless, P. P. Srinivasan, J. T. Barron, and I. Kemelmacher-Shlizerman, « Humannerf: free-viewpoint rendering of moving people from monocular video », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 210–16 220.

[104] J. Zhang, X. Liu, X. Ye, *et al.*, « Editable free-viewpoint video using a layered neural representation », *ACM Transactions on Graphics (TOG)*, vol. 40, *4*, pp. 1–18, 2021.

[105] Z. Dong, J. Song, X. Chen, C. Guo, and O. Hilliges, « Shape-aware multi-person pose estimation from multi-view images », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 11 158–11 168.

[106] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, « Smpl: a skinned multi-person linear model », *ACM transactions on graphics (TOG)*, vol. 34, *6*, pp. 1–16, 2015.

[107] J. Y. Zhang, S. Pepose, H. Joo, D. Ramanan, J. Malik, and A. Kanazawa, « Perceiving 3d human-object spatial arrangements from a single image in the wild », *in European conference on computer vision*, Springer, 2020, pp. 34–51.

[108] J. Zhang, D. Yu, J. H. Liew, X. Nie, and J. Feng, « Body meshes as points », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 546–556.

[109] N. Ugrinovic, A. Ruiz, A. Agudo, A. Sanfeliu, and F. Moreno-Noguer, « Body size and depth disambiguation in multi-person reconstruction from single images », *in 2021 International Conference on 3D Vision (3DV)*, IEEE, 2021, pp. 53–63.

[110] R. A. Guler and I. Kokkinos, « Holopose: holistic 3d human reconstruction in-the-wild », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 884–10 894.

[111] H. Choi, G. Moon, J. Park, and K. M. Lee, « Learning to estimate robust 3d human mesh from in-the-wild crowded scenes », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 1475–1484.

[112] Y. Sun, W. Liu, Q. Bao, Y. Fu, T. Mei, and M. J. Black, « Putting people in their place: monocular regression of 3d people in depth », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 13 243–13 252.

[113] Y. Sun, Q. Bao, W. Liu, Y. Fu, M. J. Black, and T. Mei, « Monocular, one-stage, regression of multiple 3d people », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 11 179–11 188.

[114] A. Zanfir, E. Marinoiu, M. Zanfir, A.-I. Popa, and C. Sminchisescu, « Deep network for the integrated 3d sensing of multiple people in natural images », *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[115] A. Zanfir, E. Marinoiu, and C. Sminchisescu, « Monocular 3d pose and shape estimation of multiple people in natural scenes-the importance of multiple scene constraints », *in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2148–2157.

[116] M. Fieraru, M. Zanfir, E. Oneata, A.-I. Popa, V. Olaru, and C. Sminchisescu, « Three-dimensional reconstruction of human interactions », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7214–7223.

[117] W. Jiang, N. Kolotouros, G. Pavlakos, X. Zhou, and K. Daniilidis, « Coherent reconstruction of multiple humans from a single image », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5579–5588.

[118] A. Mustafa, A. Caliskan, L. Agapito, and A. Hilton, « Multi-person implicit reconstruction from a single image », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 474–14 483.

[119] Y. Liu, C. Stoll, J. Gall, H.-P. Seidel, and C. Theobalt, « Markerless motion capture of interacting characters using multi-view image segmentation », *in CVPR 2011*, Ieee, 2011, pp. 1249–1256.

[120] Y. Liu, J. Gall, C. Stoll, Q. Dai, H.-P. Seidel, and C. Theobalt, « Markerless motion capture of multiple characters using multiview image segmentation », *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, *11*, pp. 2720–2735, 2013.

[121] C. Wu, C. Stoll, L. Valgaerts, and C. Theobalt, « On-set performance capture of multiple actors with a stereo camera », *ACM Transactions on Graphics (TOG)*, vol. 32, *6*, pp. 1–11, 2013.

[122] B. Huang, Y. Shu, T. Zhang, and Y. Wang, « Dynamic multi-person mesh recovery from uncalibrated multi-view cameras », *in 2021 International Conference on 3D Vision (3DV)*, IEEE, 2021, pp. 710–720.

[123] Y. Zhang, Z. Li, L. An, M. Li, T. Yu, and Y. Liu, « Lightweight multi-person total motion capture using sparse multi-view cameras », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5560–5569.

[124] Q. Shuai, C. Geng, Q. Fang, *et al.*, « Novel view synthesis of human interactions from sparse multi-view videos », *in ACM SIGGRAPH 2022 Conference Proceedings*, 2022, pp. 1–10.

[125] A. Radford, J. W. Kim, C. Hallacy, *et al.*, « Learning transferable visual models from natural language supervision », *in International Conference on Machine Learning*, PMLR, 2021, pp. 8748–8763.

[126] J. T. Barron, B. Mildenhall, M. Tancik, P. Hedman, R. Martin-Brualla, and P. P. Srinivasan, « Mip-nerf: a multiscale representation for anti-aliasing neural radiance fields », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5855–5864.

[127] M. Levoy and P. Hanrahan, « Light field rendering », *in Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 1996, pp. 31–42.

[128] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, « Mip-nerf 360: unbounded anti-aliased neural radiance fields », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5470–5479.

[129] J. Chibane, A. Bansal, V. Lazova, and G. Pons-Moll, « Stereo radiance fields (srf): learning view synthesis for sparse views of novel scenes », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 7911–7920.

[130] B. Koonce and B. Koonce, « Vgg network », *Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization*, pp. 35–50, 2021.

[131] J. L. Schonberger and J.-M. Frahm, « Structure-from-motion revisited », *in Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4104–4113.

[132] J. Johnson, A. Alahi, and L. Fei-Fei, « Perceptual losses for real-time style transfer and super-resolution », *in European conference on computer vision*, Springer, 2016, pp. 694–711.

[133] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, « The unreasonable effectiveness of deep features as a perceptual metric », *in CVPR*, 2018.

[134] B. Mildenhall, P. P. Srinivasan, R. Ortiz-Cayon, *et al.*, « Local light field fusion: practical view synthesis with prescriptive sampling guidelines », *ACM Transactions on Graphics (TOG)*, 2019.

[135] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger, « Occupancy networks: learning 3d reconstruction in function space », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4460–4470.

[136] J. Chibane, T. Alldieck, and G. Pons-Moll, « Implicit functions in feature space for 3d shape reconstruction and completion », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6970–6981.

[137] V. Sitzmann, E. Chan, R. Tucker, N. Snavely, and G. Wetzstein, « Metasdf: meta-learning signed distance functions », *Advances in Neural Information Processing Systems*, vol. 33, pp. 10 136–10 147, 2020.

[138] A. X. Chang, T. Funkhouser, L. Guibas, *et al.*, « Shapenet: an information-rich 3d model repository », *arXiv preprint arXiv:1512.03012*, 2015.

[139] D. Chen, Y. Liu, L. Huang, B. Wang, and P. Pan, « Geoaug: data augmentation for few-shot nerf with geometry constraints », *in European Conference on Computer Vision*, Springer, 2022, pp. 322–337.

[140] M. M. Johari, Y. Lepoittevin, and F. Fleuret, « Geonerf: generalizing nerf with geometry priors », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 365–18 375.

[141] Y. Liu, S. Peng, L. Liu, *et al.*, « Neural rays for occlusion-aware image-based rendering », *in CVPR*, 2022.

[142] J. T. Kajiya and B. P. Von Herzen, « Ray tracing volume densities », *in Proceedings of the 11th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '84, 1984, pp. 165–174.

[143] S. Cheng, Z. Xu, S. Zhu, *et al.*, « Deep stereo using adaptive thin volume representation with uncertainty awareness », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2524–2534.

[144] O. Ronneberger, P. Fischer, and T. Brox, « U-net: convolutional networks for biomedical image segmentation », *in Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, Springer, 2015, pp. 234–241.

[145] P. Heise, S. Klose, B. Jensen, and A. Knoll, « Pm-huber: patchmatch with huber regularization for stereo matching », *in Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 2360–2367.

[146] A. Paszke, S. Gross, F. Massa, *et al.*, « Pytorch: an imperative style, high-performance deep learning library », *Advances in neural information processing systems*, vol. 32, 2019.

[147] D. P. Kingma and J. Ba, « Adam: a method for stochastic optimization », *arXiv preprint arXiv:1412.6980*, 2014.

[148] S. A. Eslami, D. Jimenez Rezende, F. Besse, *et al.*, « Neural scene representation and rendering », *Science*, vol. 360, *6394*, pp. 1204–1210, 2018.

[149] T. Alldieck, M. Magnor, W. Xu, C. Theobalt, and G. Pons-Moll, « Detailed human avatars from monocular video », *in 2018 International Conference on 3D Vision (3DV)*, IEEE, 2018, pp. 98–109.

[150] B. Jiang, Y. Hong, H. Bao, and J. Zhang, « Selfrecon: self reconstruction your digital avatar from monocular video », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5605–5615.

[151] J. Chen, Y. Zhang, D. Kang, *et al.*, « Animatable neural radiance fields from monocular rgb videos », *arXiv preprint arXiv:2106.13629*, 2021.

[152] Z. Dong, C. Guo, J. Song, X. Chen, A. Geiger, and O. Hilliges, « Pina: learning a personalized implicit neural avatar from a single rgb-d video sequence », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 20 470–20 480.

[153] Y. Kwon, D. Kim, D. Ceylan, and H. Fuchs, « Neural human performer: learning generalizable radiance fields for human performance rendering », *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[154] S. Peng, J. Dong, Q. Wang, *et al.*, « Animatable neural radiance fields for modeling dynamic human bodies », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 14 314–14 323.

[155] H. Xu, T. Alldieck, and C. Sminchisescu, « H-nerf: neural radiance fields for rendering and temporal reconstruction of humans in motion », *Advances in Neural Information Processing Systems*, vol. 34, pp. 14 955–14 966, 2021.

[156] V. Leroy, J.-S. Franco, and E. Boyer, « Shape reconstruction using volume sweeping and learned photoconsistency », *in Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 781–796.

[157] M. Dou, S. Khamis, Y. Degtyarev, *et al.*, « Fusion4d: real-time performance capture of challenging scenes », *ACM Transactions on Graphics (ToG)*, vol. 35, *4*, pp. 1–13, 2016.

[158] D. Vlasic, P. Peers, I. Baran, *et al.*, « Dynamic shape capture using multi-view photometric stereo », *in ACM SIGGRAPH Asia 2009 papers*, 2009, pp. 1–11.

[159] C. Wu, K. Varanasi, and C. Theobalt, « Full body performance capture under uncontrolled and varying illumination: a shading-based approach », *in European Conference on Computer Vision*, Springer, 2012, pp. 757–770.

[160] M. Wu, Y. Wang, Q. Hu, and J. Yu, « Multi-view neural human rendering », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1682–1691.

[161] S. Saito, Z. Huang, R. Natsume, S. Morishima, A. Kanazawa, and H. Li, « Pifu: pixel-aligned implicit function for high-resolution clothed human digitization », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 2304–2314.

[162] S. Saito, T. Simon, J. Saragih, and H. Joo, « Pifuhd: multi-level pixel-aligned implicit function for high-resolution 3d human digitization », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 84–93.

[163] T. Alldieck, M. Zanfir, and C. Sminchisescu, « Photorealistic monocular 3d reconstruction of humans wearing clothing », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 1506–1515.

[164] T. He, J. Collomosse, H. Jin, and S. Soatto, « Geo-pifu: geometry and pixel aligned implicit functions for single-view human reconstruction », *Advances in Neural Information Processing Systems*, vol. 33, pp. 9276–9287, 2020.

[165] Z. Huang, Y. Xu, C. Lassner, H. Li, and T. Tung, « Arch: animatable reconstruction of clothed humans », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3093–3102.

[166] T. He, Y. Xu, S. Saito, S. Soatto, and T. Tung, « Arch++: animation-ready clothed human reconstruction revisited », *in Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 11 046–11 056.

[167] X. Long, C. Lin, P. Wang, T. Komura, and W. Wang, « SparseNeuS: fast generalizable neural surface reconstruction from sparse views », *ECCV*, 2022.

[168] M. Tancik, P. Srinivasan, B. Mildenhall, *et al.*, « Fourier features let networks learn high frequency functions in low dimensional domains », *Advances in Neural Information Processing Systems*, vol. 33, pp. 7537–7547, 2020.

[169] D. Azinović, R. Martin-Brualla, D. B. Goldman, M. Nießner, and J. Thies, « Neural rgb-d surface reconstruction », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 6290–6301.

[170] A. Gropp, L. Yariv, N. Haim, M. Atzmon, and Y. Lipman, « Implicit geometric regularization for learning shapes », *arXiv preprint arXiv:2002.10099*, 2020.

[171] R. Martin-Brualla, N. Radwan, M. S. Sajjadi, J. T. Barron, A. Dosovitskiy, and D. Duckworth, « Nerf in the wild: neural radiance fields for unconstrained photo collections », *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 7210–7219.

[172] A. Hore and D. Ziou, « Image quality metrics: psnr vs. ssim », *in 2010 20th international conference on pattern recognition*, IEEE, 2010, pp. 2366–2369.

[173] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, « Image quality assessment: from error visibility to structural similarity », *IEEE transactions on image processing*, vol. 13, *4*, pp. 600–612, 2004.

[174] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, « The unreasonable effectiveness of deep features as a perceptual metric », *in Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.

[175] T. Yu, Z. Zheng, K. Guo, P. Liu, Q. Dai, and Y. Liu, « Function4d: real-time human volumetric capture from very sparse consumer rgbd sensors », *in IEEE Conference on Computer Vision and Pattern Recognition (CVPR2021)*, Jun. 2021.

COLLEGE DOCTORAL BRETAGNE
MATHS, TELECOMS
INFORMATIQUE, SIGNAL
SYSTEMES, ELECTRONIQUE

Université de Rennes

Titre : Synthèse de nouvelles vues à partir d'entrées limitées

Mot clés : Synthèse de nouvelles vues, champs de rayonnement neuronal, réseau de champ lumineux, génération de formes multi-humaine, entrées éparses

Résumé : Malgré le potentiel important de la synthèse de nouvelles vues à partir d'entrées éparses dans les applications d'infographie et de vision par ordinateur, plusieurs défis subsistent dans ce sujet. Cette thèse étudie trois aspects concernant la synthèse de nouvelles vues. Tout d'abord, nous avons présenté une nouvelle approche pour améliorer les NeRF à partir d'entrées éparses. Les méthodes proposées comprennent échantillonnage global avec régularisation, l'augmentation des données, l'échantillonnage de patchs locaux avec régularisation basée sur les patchs et la régularisation de profondeur explicite. Des évaluations approfondies démontrent que notre méthode surpasse les performance de référence. Deuxièmement, nous avons proposé d'améliorer le champ lumineux neuronal à partir d'entrées éparses. Nous utilisons un réseau neuronal implicite conditionné sur les caractéristiques des rayons locaux d'un encodeur à convolutions. Nous atteignons des performances compétitives et offrons une vitesse de rendu beaucoup plus rapide. Troisièmement, nous avons introduit une nouvelle approche pour génération de forme et rayonnement 3D d'un scène contenant plusieurs personnes à partir d'images éparses. Notre approche intègre des contraintes géométriques à l'aide de maillages pré-calculés, de la régularisation des rayons basée sur les patchs et de la régularisation de la saturation. Nous atteignons des performances de pointe sur des données réelles et synthétiques.

Title: Novel View Synthesis from Sparse Inputs

Keywords: neural radiance fields, light field network, multi-human generation, sparse inputs

Abstract: Despite the significant potential of novel view synthesis from sparse inputs in computer graphics and computer vision applications, several challenges remain in this topic. This thesis investigates and provides solutions in three aspects regarding novel view synthesis. Firstly, we presented a novel approach to improve NeRF from sparse inputs. The proposed methods include global sampling with regularization, data augmentation, local patch sampling with patch-based regularization, and explicit depth regularization. Extensive evaluations demonstrate our method outperforms baselines. Secondly, We proposed to improve the neural light field from sparse inputs. It employed an implicit neural network conditioned on local ray features from the convolutional encoder. It achieved competitive performance and offered a much faster rendering speed. Thirdly, We introduced a novel 3d shape and radiance generation approach for multiple humans from sparse images. Our approach incorporates geometry constraints using pre-computed meshes, patch-based ray regularization, and saturation regularization. It achieved state-of-the-art performance on real and synthetic data.