Improving Mixed Reality with Multi-task Scene Understanding and Data Augmentation

マルチタスク学習を用いたシーン理解とデータ拡張による複合現実感の向上

July, 2022

Qi FENG
馮　起

Improving Mixed Reality with Multi-task Scene Understanding and Data Augmentation

マルチタスク学習を用いたシーン理解とデータ拡張による複合現実感の向上

July, 2022

Waseda University Graduate School of Advanced Science and Engineering

Department of Pure and Applied Physics, Research on Image Processing

Qi FENG
馮　起

Contents

Chapter 1

Introduction

1.1 Background

Mixed reality as a broad concept describes a process of blending the physical world and the digital environment computed and rendered in real-time. This new hybrid reality that extends both ways from a completely virtual world and our perceived reality enables numerous potential interactions and applications. When emphasizing more on the virtual environment, mixed reality allows users to enter an entirely different environment with intuitive interaction. Such unique strength over traditional user interface sparks a wide range of studies including human-computer interaction, psychology, cognition, etc., and inspires plentiful practical applications in training, entertainment, education, and medical [1]. When the focus is put on the real world instead of complete disconnection from reality, mixed reality shows its capability to aid real-world tasks effectively and efficiently by presenting computer-generated visualizations as an overlay. For instance, Google Translate uses augmented reality technology to overlay the translated scripts onto the physical texts in real-time through the camera-equipped hand-held device [2]. With the ability to seamlessly superimpose virtual objects onto user's observations, it motivates abundant research topics such as haptics, collaboration, and robotics, and is broadly adapted into manufacturing, design, and modeling.

To seamlessly merge the digital and the real world, correctly understanding both the virtual scene and the physical environment is crucial to every mixed reality application. Irrelevant to the specific function of mixed reality applications, the entire process often incorporates the following steps, which are demonstrated in Figure 1.3. After sensors and imaging devices are successfully calibrated, systems execute multiple computer vision tasks to establish and maintain a correct understanding of the scene for the later augmentation process. The subsequent step is mapping and registering the environment, which enables the device to establish a reliable correspondence between the virtual environment and the real world. This is usually done with vision-based methods such as using fiducial markers or recognizing features of the physical surroundings [3]. During the usage of each application, the system receives continuous real-time inputs with different modalities (e. g. motion controllers or ray-traced touch input with a touchscreen [4]) and updates the stored correspondence iteratively.

After the application finished processing the information with renders being available, the final step requires the system to understand the mapping between virtuality and reality to correctly visualize the result and superimpose it onto physical objects.

When compared to the first barebone and bulky mixed reality device designed in 1994 [5], modern mixed reality has been experiencing dramatically increased popularity with more affordable hardware and polished software in recent years, owing to the progress in computer graphics, display technology, input systems, cloud computing, and other research fields. Nevertheless, for sophisticated applications under diverse conditions, providing an immersive mixed reality experience is still challenging. One crucial issue is the difficulty of accurately and instantaneously understanding the complex relationship between the virtual environment and the physical world. As we can observe that throughout a typical mixed reality process, correctly understanding both the virtual scene and the physical environment is of great importance.

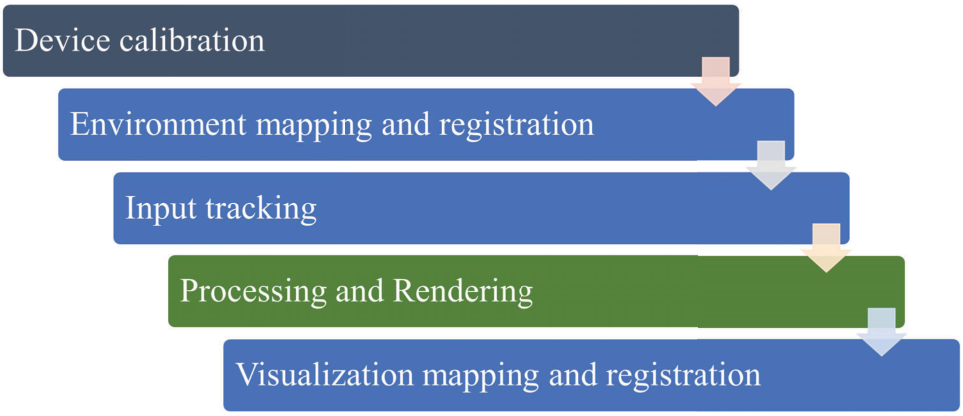

Device calibration
Environment mapping and registration
Input tracking
Processing and Rendering
Visualization mapping and registration

Figure 1.1: Different steps to process mixed reality.

To satisfy the requirements of real-time efficiency and highly accurate efficacy, scene understanding, as a fundamental component of mixed reality technologies, has seen great progress over recent years with more advanced deep learning algorithms. Among a wide range of topics in scene understanding, the following ones are frequently involved in the multiple steps of a mixed reality application that are marked in blue (Figure 1.3). Object recognition enables the system to identify a physical object in the real world with high accuracy, facilitating a museum guide application [6] that recognizes the artworks with the camera and provides the user with related information to aid the tour. Semantic segmentation provides each pixel with correct labels, aiding collision detection [7] in the mixed reality environment. Depth prediction helps understand the scale and distance of a physical object, enabling occlusion calculations to correctly render the virtual augmentation for an immersive experience [8].

While traditional deep learning-based scene understanding algorithms that accomplish respective objectives of object recognition, semantic segmentation, and depth estimation in isolation have shown great performance, mixed reality applications often involve multi-modal data and several pipelines running simultaneously in real-time.

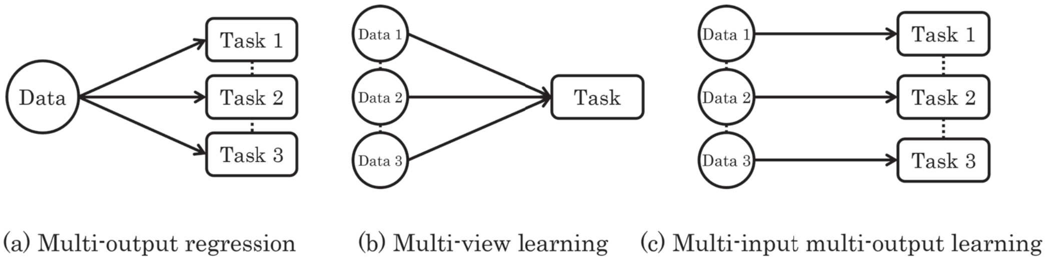
For instance, tracking and predicting the displacement of the controller using acceleration and visual data, while detecting nearby dynamic objects with visual and depth information, in addition to localizing and mapping the main device in the physical environment, are all carried out in real-time. Motivated by biology that human tackles similar tasks with shared knowledge and experience, multi-task learning has gained growing attention in the past few years. With the ability to learn and solve multiple tasks concurrently, multi-task learning is well-suited for mixed reality applications. However, improving mixed reality with multi-task scene understanding algorithms is a topic that is rarely investigated.

Multi-task learning excels in improving the generalization and the accuracy of each task, optimizing the overall model size and inference time, and alleviating the scarcity problem of training data. A multi-task learning model receives different domains of input and aims to learn multiple objectives simultaneously. First, by sharing knowledge across relevant tasks, the model usually learns a more robust and accurate representation of the training data. This is achieved through different training schemes and network architectures. For instance, hard parameter sharing incorporates sharing several hidden layers across every task to capture a better representation of features and have several branched layers at the end to yield respective outputs for each task. One task eavesdropping on another can let the model leverage the features advised by different tasks, which are otherwise hard to find when learning individually. As a consequence, each task shows better generalization with a lower risk of overfitting. When tasks are related and similar, multi-task models usually show improved performance. Second, due to inherent network design with shared layers, a portion of calculation and corresponding memory requirements can be greatly reduced, along with optimized training and inference time when compared to single-task learning. Third, multi-tasking learning can aggregate the training data across different tasks as implicit data augmentation, alleviating the scarcity problem of annotated data for some applications. By passing the learned knowledge from different supervisory signals to a task with scarce samples, multi-task learning can utilize the information to guide the training process and reduce the necessity of laborious manual annotation.

Despite the great capabilities of multi-task learning, multi-modal annotated databases are still indispensable for supervised learning approaches. While there are abundant samples for traditional computer vision tasks such as object recognition, in the context of scene understanding in mixed reality, many tasks suffer from low-quality databases. For instance, while hand posture samples are usually captured and annotated using a nearby webcam or Kinect, hand-object interactions in mixed reality are often viewed with an egocentric perspective. Moreover, captured photos that are intended for mixed reality usage are usually stored with an entirely different projection, rendering traditional perspective training databases less effective. As a result, to take advantage of multi-tasking in mixed reality applications, an effective data augmentation method is less studied but very much desired.

1.2 Research Scope and Objectives

To achieve immersive mixed reality with improved realism, a more effective and efficient understanding of both the virtual environment and the real world is essential. In this thesis, we aim to investigate the great potential of multi-task learning-based scene understanding algorithms for mixed reality applications. We categorize multi-task learning into three different paradigms with respective strengths and advisable applications in mixed reality: (1) multi-output regression, (2) multi-view learning, and (3) multi-input multi-output learning. Fig. 1.2 overviews the structural difference between them. We will focus on several topics of scene understanding that are particularly crucial in mixed reality: object recognition, semantic segmentation, and depth prediction. It is well-known that labeled training data are crucial for learning-based approaches, and great performance high-capacity deep learning models need an adequate number of annotated samples. We aim to propose multi-task learning algorithms to solve persistent problems in mixed reality by combining different data augmentation techniques to obtain high-quality large-scale databases that are currently scarce or not available.



(a) Multi-output regression (b) Multi-view learning (c) Multi-input multi-output learning

FIGURE 1.2: Different multi-task learning paradigms.

In this thesis, we follow the order of different spatial scopes and choose three unique challenges in mixed reality to demonstrate the capability of each multi-task learning paradigm. We start from a smaller scope in mixed reality by trying to understand egocentric hand-object interactions. Next, we aim to understand a larger scope: estimating the correct depth of the foreground objects in a mixed reality environment. Finally, we further zoom out to observe the global context and aim to comprehend the entire 360-degree scene with depth prediction. On the local scope, we design a multi-output regression network that receives single domain input and yield multiple domain output to showcase the better capability and efficiency of multi-task learning in mixed reality. Due to scarce egocentric training samples and strong motion, previous semantic segmentation methods yield sub-optimal results in interactive applications. We demonstrate that learning different tasks in parallel and yielding predictions at the same time with a multi-output regression network (Fig. 1.2(a)) can help achieve higher accuracy and real-time efficiency.

On the regional scope, we design a multi-view learning network that receives multiple domain inputs and yields single domain outputs to achieve higher accuracy with multi-task learning. The network aims to model diverse views with different features with a single function to achieve improvements in performance for the same task. Combining the equirectangular view which is consistent yet has distortion with the cubemap view which has no distortion but is inconsistent at boundaries, the proposed multi-view learning network outperforms a similar multi-output regression network and verifies the capability of multi-view learning for mixed reality applications of this scope.

On the global scope, we propose to achieve a good understanding of the entire scene in mixed reality through depth estimation. Since multi-view learning showed insufficient ability to comprehend the global information in our experiments, we design a multi-input multi-output learning architecture that instructs different views to respectively accomplish depth estimation and semantic segmentation. With a fusion scheme that shares information with the other branch, we successfully demonstrate the strength of multi-task learning in challenging mixed reality problems.

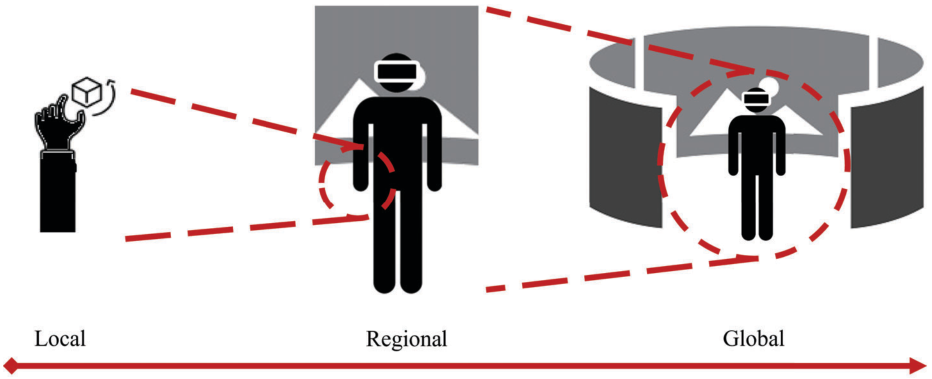


FIGURE 1.3: Understanding local, regional, and global scope of mixed reality scenes.

1.3 Thesis Organization

In this thesis, we follow the order of different spatial scopes to understand scenes in mixed reality. After a thorough review of the background and literature on mixed reality and relevant multi-task learning scene understanding approaches in Chapter 2, we start from a smaller scale of understanding users' hand-object interactions to resolve occlusions. We then focus on the foreground objects of mixed reality scenes. We choose humans as an example to demonstrate the capabilities of predicting depth and semantic segmentation with different network designs. Later, we propose to comprehend the global scene through depth prediction and showcase its usage in mixed reality. Finally, we explore employing existing scene understanding algorithms for practical mixed reality applications. In the remainder of the thesis, we will discuss limitations and give potential directions for future computer vision for mixed reality.

Background (Chapter 2). In Chapter 2, we first review the definition and important applications of state-of-the-art mixed reality. We then continue to review the relevant scene understanding topics including object recognition, semantic segmentation, and depth prediction. We finally examine the literature on multi-task learning and data augmentation methods.

Grasping the Local: Solving Hand-object Occlusion in Mixed Reality (Chapter 3). The hand is one of the key components in mixed reality, and hand-object interactions are critical to a wide range of MR applications such as surgery simulations. However, their practicality and immersive experiences are severely limited by occlusions. In Chapter 3, we first revisit existing occlusion solutions, followed by explaining the proposed RGBD database generated with data augmentation, and then a novel joint learning process to predict hand postures and masks. We finally present our novel two-step approach to resolving the occlusions in mixed reality with implementation details, evaluations, and a user study. This research can be applied to egocentric mixed reality applications that include hand-object interactions such as apparatus-involved training.

Observing the Regional: Foreground-aware 360° Depth Prediction (Chapter 4). Although the ability to predict depth from a single 360-degree image can benefit plentiful applications, existing approaches produce sub-optimal results for foreground objects. In this chapter, we propose to augment databases with realistic foregrounds with an image-based approach and design a novel auxiliary deep neural network to predict depth and semantic segmentation simultaneously. We further design a bi-projection-based network to improve the capability of understanding the foreground object. We demonstrate the system using humans as the foreground due to its complexity and contextual importance and show consistent and accurate local estimations compared with state-of-the-arts.

Comprehending the Global: 360° Depth Prediction in the Wild (Chapter 5). Although data-driven learning-based methods demonstrate significant potential in understanding the entirety of 360-degree images, scarce training data and ineffective 360-degree estimation algorithms are still two key limitations hindering accurate estimation across diverse domains. In this chapter, we first establish a large-scale database by exploring the use of a plenteous source of data, 360-degree videos from the internet, using a test-time training method. We then propose an end-to-end two-branch multi-task learning network, SegFuse, that mimics the human eye to effectively learn from the dataset and estimate high-quality depth maps from diverse monocular RGB images. We showcase that our method has a great understanding of the global mixed reality scene under arbitrary conditions.

Employing Scene Understanding in Immersive Mixed Reality (Chapter 6). With the established understanding of different scales of scenes, we explore the practical applications of mixed reality in Chapter 6. We propose two applications: editing foreground objects of interest in pre-captured 360-degree videos and consistent artistic stylization for pre-captured videos. We expect this application-focused chapter can shed more light on more practical employments of newer scene understanding algorithms in the modern virtual/augmented reality era.

Conclusion (Chapter 7). In Chapter 7, we start with a summary of the work. We then discuss the limitations of current scene understanding in upcoming mixed reality, and try to explore some promising directions for alike future research that focus on solving the computer vision aspect of mixed reality technologies.

Chapter 2

Literature Review

In this chapter, we present the necessary literature and background for the overall goal of this thesis, improving the immersive mixed reality experience with multi-task learning-based scene understanding algorithms. First, we review the broad concepts and theoretical background related to mixed reality technology. Considering that convincing visuals require coherent registrations between the physical and the virtual environment, we then briefly discuss the significance of key scene understanding abilities in typical mixed reality processes. Next, we review the background and capabilities of multi-tasking learning in effective and efficient scene understanding. To facilitate an effective learning process, we also briefly describe popular data augmentation approaches that are relevant to this research.

2.1 Mixed Reality

The term mixed reality was firstly introduced in 1994, defines as the blended reality that extends from the extremes of a completely virtual environment rendered within computers and the actual reality the users stay in [5]. Mixed reality is realized through constructing a three-dimensional virtual environment and merging the virtual world with the physical environment. Allowing the user to interact with the co-existed physical and digital environments intuitively, greatly expands the ability of human beings to simulate and understand the world. With more matured mixed reality technology, a wide range of industrial applications such as entertainment [9], product design and modeling [10], military training [11], education [12], have been implemented, and cross-field research topics including human-computer interaction [13], computer vision [8], cognition and emotion [14], medical and healthcare [15] are gaining more attention in recent years. Figure 2.1 showcases an increasing amount of industrial interest since 2004, while Figure 2.2 illustrates the steady growth of academic attention in related research fields of mixed reality.

To further clarify the scope of this research, we utilize the widely accepted definition of the mixed reality continuum to briefly explain the different research problems under mixed reality. As shown in Figure 2.3, the traditional graphical user interface that is presented through a flat-screen is defined at the right end of the spectrum as complete virtuality, and the real-world environment without any computer-generated

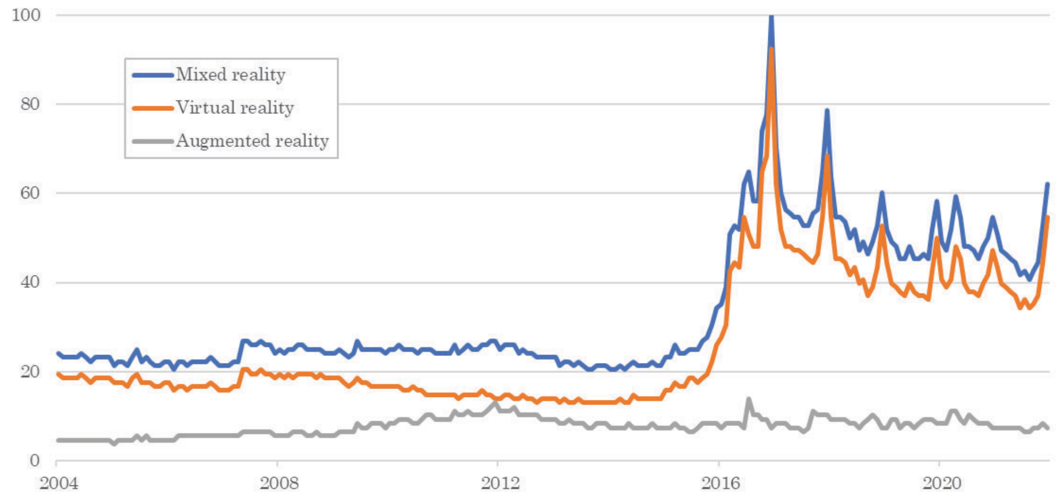


FIGURE 2.1: The search interest of mixed reality related keywords. (Source: Google Trends [16], from Jan. 2004 to Dec. 2021.)

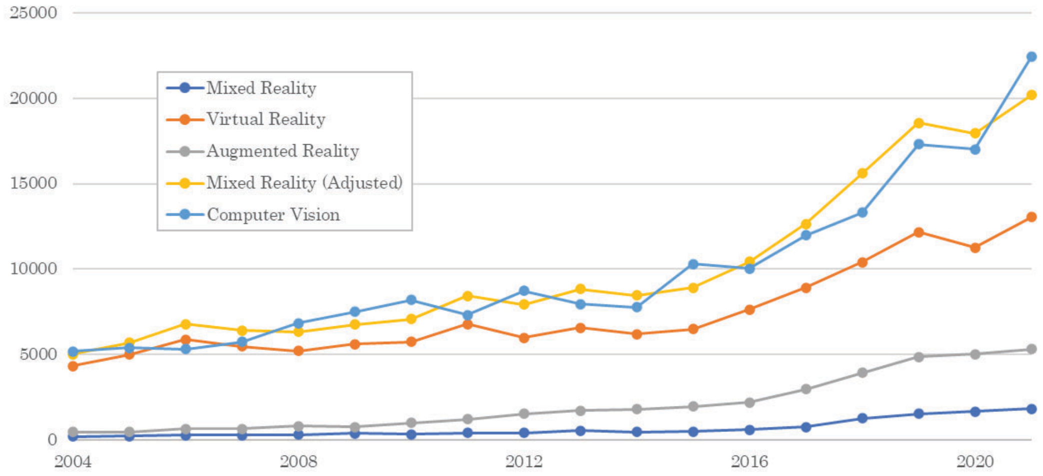


FIGURE 2.2: The statistic of publications with mixed reality related topics. (Source: Scopus [17], from Jan. 2004 to Dec. 2021.)

elements is defined as reality. On the one hand, when an increasing number of physical elements are incorporated into the three-dimensional virtual environment, we will meet with virtual reality on the spectrum. Virtual reality describes a predominantly virtual environment with a rather small number of physical elements, such as physical objects and movements of the user. On the other hand, when we augment the visual of the real environment with additional visualization of virtual objects, we meet with augmented reality on the other side of the spectrum. However, some research topics, such as rendering pre-captured real-world environments in virtual worlds [13] and projecting virtual environments to real screens surrounding the user [18], are more difficult to determine whether they belong to the virtual reality or the augmented reality domain, therefore sometimes researchers use mixed reality as a broad and inclusive term to cover the entirety of blended physical and virtual environments.

To achieve an immersive mixed reality experience, two elements are crucial: (1)

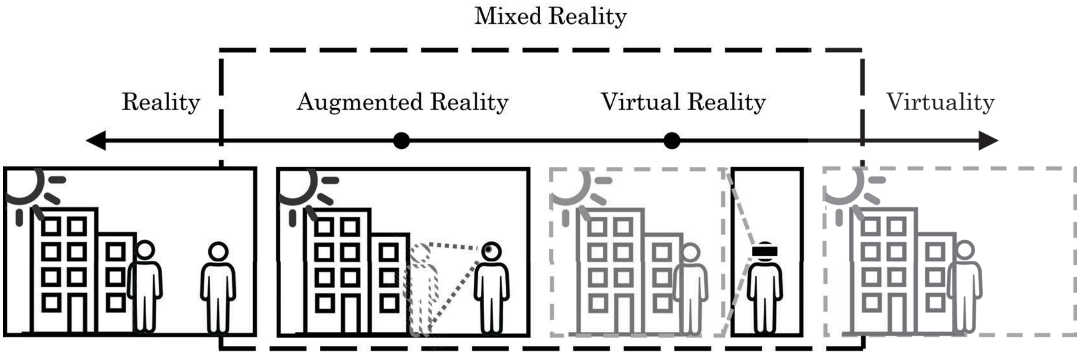


FIGURE 2.3: The relationship between mixed reality, virtual reality, and augmented reality.

accurate understanding of the scene, as an input stage, for correct geometric registration and seamless fusion of the virtual environment and real-world counterparts; (2) real-time visualization and interaction, as an output stage, for a natural and responsive experience. These are realized through different hardware and software for virtual reality and augmented reality, and each has its practical applications in a variety of disciplines. Therefore, I briefly review them separately in the following sections.

2.1.1 Virtual Reality

The introduction of the concept of virtual reality can be dated back to 1986 [19], which aimed to create a synthetic environment created by computers and achieve the illusion for the user that he/she is located in a different reality. While multi-sensory signals such as haptic feedback and olfactory stimuli [20] are being studied for a more immersive sense of presence, this illusion is predominantly achieved through visual cues.

Applications. With the advance in computer graphics and computational power, modern virtual reality can render objects and environments with an excellent degree of realism. As a result, virtual reality has gained popularity over the recent years in both industry and academia, ranging in a great variety of disciplines. There are abundant cases of adopting virtual reality in educational settings to enhance the learning process. For instance, employing virtual reality in medical education can make the process of learning human anatomy effective and enjoyable [15]. It is also widely adopted in training processes due to its proven potential for skill acquirement, such as learning to drive [21], performing surgery [15], and refining performance in sports [22]. Recently, one of the most prevalent uses of virtual reality is entertainment. With an increasing number of polished software and affordable hardware being released, virtual reality has seen a great increase in sales and acceptance over the past five years [23].

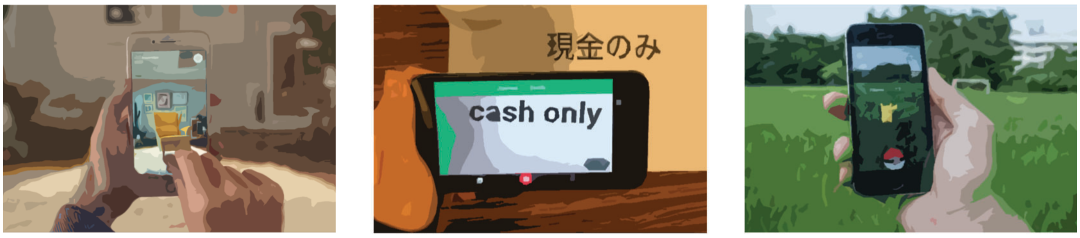
Hardware. In most cases, virtual reality creates a completely enclosed environment through a head-mounted display. Being one of the most important factors for depth perception and immersive experience, stereoscopic vision is achieved through two identical high-resolution screens positioned right in front of each eye. To simulate the realistic visual feedback according to the user's movements, real-time tracking of the headset is achieved through acceleration data and visual inputs gathered by inbuilt gyroscopes and vision-based algorithms. To improve realism, display resolution, field-of-view, and display refresh frequency are all key factors. The recent commercial success of Oculus Quest and Valve Index [23] attributes to their powerful yet compact hardware.

Media creation in virtual reality, including 360-degree photography and videos, has also gained great attention in recent years. With affordable commercial omnidirectional cameras being researched and developed, capturing omnidirectional media and playback with virtual reality devices can provide unparalleled immersion when compared to traditional media formats [13]. At the same time, online video sharing platforms such as YouTube also readily promote the progress of omnidirectional contents with efficient compression and transmission algorithms [24], making real-time online streaming possible.

Limitations. Although great progress has been made in theory, technology, and application, one of the major scientific issues of virtual reality is still real-time computation and high-fidelity render of both the virtual and the physical world. Compared to traditional two-dimensional displays, a close distance between the screens in the head-mounted displays and the eyes of the user means that artifacts and delays would inevitably disrupt the immersion. Even worse, the unsynchronized visual feedback shown at a sub-optimal frequency will result in disorientation, severe fatigue, or even severe discomfort. Currently, the industrial standard for virtual reality fidelity is usually 90 frames per second with a per-eye resolution of 1440 by 1600 [23]. However, a rather low field of view around 100 degrees is still limiting the immersion of the current generation of virtual reality. Moreover, to simulate and render a virtual environment with a combination of high refresh rates and large amounts of pixels, the computational cost usually demands powerful equipment, further limiting the progress of virtual reality. As a result, the efficiency of the algorithm is highly desired and has become an important criterion for virtual reality studies and applications.

2.1.2 Augmented Reality

To alleviate such limitations, augmented reality greatly reduces the amounts of virtual objects that are required to be simulated and rendered in real-time and lowers the heavy workload of three-dimensional modeling. The concept of augmented reality was firstly introduced in 1992, defined as extending the spatial human perception of the world in three dimensions with computer-generated objects [25]. Instead of computing every detail for the entirety of the surroundings to completely immerse the user, it relies on compositing the digital visualizations onto the reality through accurate mapping and registration processes. The two key features of augmented reality are accurate correspondence between the physical and the virtual objects, and real-time interactions between the user and the blended reality.



(a) Visual design

(b) Translation

(c) Entertainment

FIGURE 2.4: Examples of augmented reality applications in different disciplines [9] [12] [10].

Applications. As a result, the usage of augmented reality dramatically diverges from virtual reality: by enhancing the sensory signals of the real environment with virtual counterparts, the goal of augmented reality is primarily to support task completion by providing a more natural integration of digital contents and the real-world environment. In visual design and planning, computer-generated three-dimensional can be superimposed onto the view of the real world with see-through devices before the actual plan is carried out. For instance, in Figure 2.4 (a), augmented reality allows users to preview the results before purchasing a piece of furniture [10]. In healthcare and medical training, augmented reality is widely adopted to provide crucial contexts spatially close to the patient and training medical professionals [15]. Figure 2.4 (b) provides an example of an augmented reality application to seamlessly acquire the foreign texts and display the translation in place of the original texts [12], which was a laborious and offline task in the past. In other areas such as entertainment, augmented reality has also seen considerable commercial and industrial successes. For instance, in live broadcasting of sports events and weather forecasts, overlaying trajectories of graphics symbols and intuitive effects onto the traditional streaming feed can provide an improved viewing experience. In Figure 2.4 (c), we showcase a very popular entertainment application that has grabbed a considerable amount of industrial attention to augmented reality technology when released [9]. Details of interesting applications across plentiful disciplines including collaboration, social interaction, robotics, and psychology will be omitted for brevity.

Hardware. To correctly merge the virtual environment with the physical world and provide a natural and faithful visualization of both worlds, augmented reality hardware functions as the gateway between the human body and the blended reality. The core components are (1) sensors, for taking input, tracking the physical objects, and registering the environment in real-time, and (2) displays, for rendering the composition with convincing quality. The first fully functional optical see-through head-mounted display was built by Sutherland in 1968 [26] for proof-of-concept augmented reality applications. Although being bulky and barebone with limited functionality, the essence of this mechanical contraption is still receiving the movement of the user's perspective as an input, and rendering a three-dimensional wireframe that corresponds to the correct perspective (please refer to Figure 2.5). Since then, researchers have been keeping improving both the registration algorithms and display technology through computer vision and computer graphics.

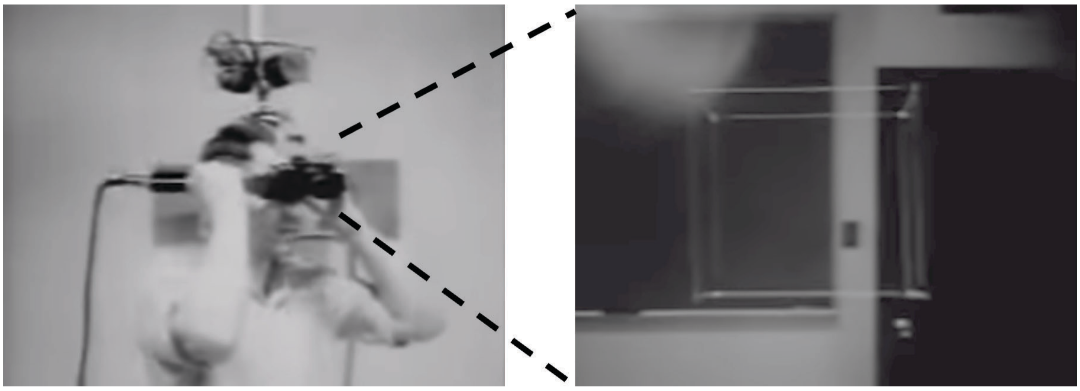


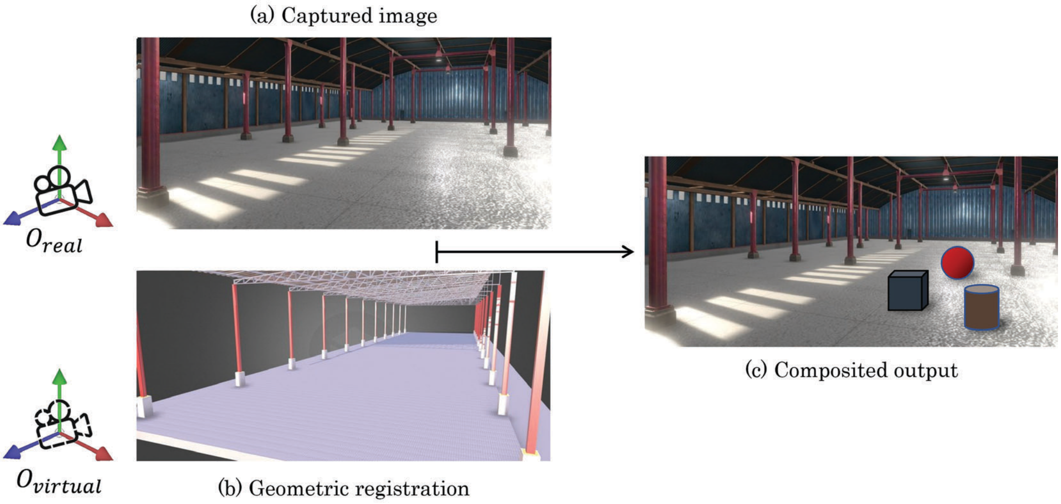
Figure 2.5: An early optical see-through device [26] running a proof-of-concept augmented reality application.

Modern augmented reality devices can be divided into two different categories: (1) optical see-through head-mounted displays and (2) video see-through head-mounted displays. Optical see-through devices allow users to directly view the real environment through transparent displays, such as a pair of glasses, with additional digital information being reflected or projected onto users' view. With a less demanding rendering workload and lower power consumption, optical see-through devices have the benefit of more compact sizes and a more natural view of the real world. However, while the process of addition is streamlined, hiding physical objects is an extremely challenging task for this type of augmented reality. On the other hand, Video see-through devices utilize single or multiple cameras to capture external images, then compute a convincing output with virtual objects, and finally display the synthesized composition in front of users' eyes in a similar fashion to virtual reality. While it is easy to manipulate the entirety of visual information, video see-through devices require heavier computations, better cameras, and high-fidelity displays. Although handheld mobile devices satisfy the criteria and see popular adoption in augmented reality applications, lack of depth perception and limited interaction greatly limit the immersion of flat screen-based augmented reality.

Limitations. Regardless of the type of device being used, the most crucial and fundamental capability of augmented reality is to accurately understand the physical environment and establish the mapping between the computer-generated environment and the reality in real-time. To derive the correct displacement of physical objects that is robust to camera poses and external conditions is a challenging task. Recent progress made in computer vision research, especially in visual odometry, allows applications to acquire the three-dimensional models and understand the lighting of the real world [27]. However, existing solutions still require specific setups sensors such as LiDAR and depth sensors to achieve a robust experience. With limited application scope and costly and bulky hardware, more work are remained to be done to further push augmented reality towards higher realism, better immersion, and wider adoption.

2.1.3 Geometric Registration

In this subsection, I will present the main concepts and approaches for establishing a consistent spatial correspondence between the virtual environment and the real world. This process is illustrated in Figure 2.6.



(a) Captured image

(b) Geometric registration

(c) Composited output

FIGURE 2.6: Augmented reality registers the geometric correspondence between the virtual and the real environment through captured images, and then augment virtual objects with correct displacement.

Within the physical environment, augmented reality systems first use a real camera O_{real} to capture a series of images $I = \{I_k, k = 1, 2, 3 \ldots, n\}$ of the real world given that the real camera is centered at O in real coordinate, and only pinhole camera model is considered for simplicity. In this sense, we view the captured image I as a projection $R_3 \rightarrow \mathrm{R}_2$ with a fixed perspective transformation T given the consistent camera intrinsic. After the perspective transformation T is determined, the system calculates the displacement of a virtual camera $O_{virtual}$ in a virtual world coordinate, that when projecting the virtual environment in a similar fashion with transformation T, the displacement of a projected feature point $\hat{X}_i$ in squared difference D is minimized to make sure that the feature point X_i in the real-world coordinate is as close as possible to $\hat{X}_i$. After iterative tracking of feature points across multiple frames, the augmented reality system can determine a more confident correspondence between the virtual and the real environment, so that it can composite digital contents onto the image I, as shown in Figure 2.6 (c).

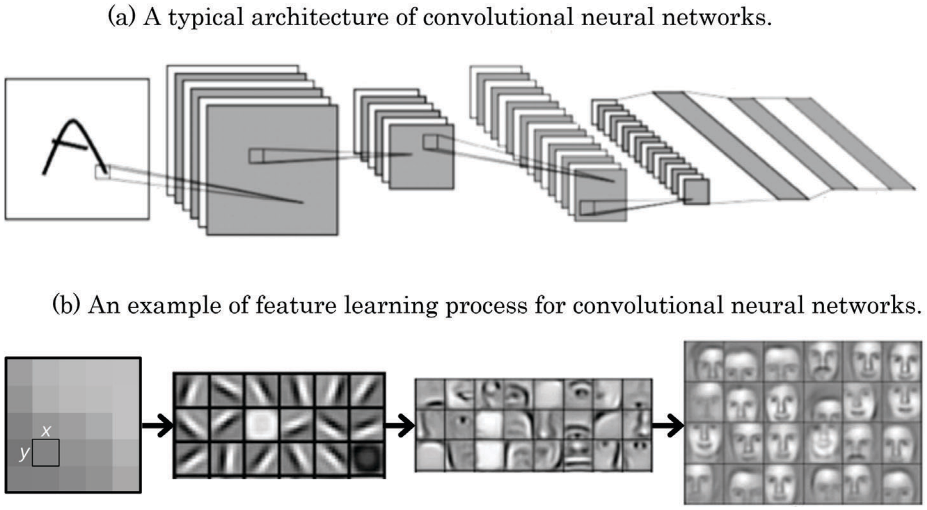
$$D = \sum_i (\hat{X}_i - MX_i)^2, \tag{2.1}$$

In the past, fiducial markers are widely adopted in augmented reality systems [28]. When there is no object with known geometry present in the scene, fiducial markers can provide robust features for the tracking process. However, marker-based systems require the easily identifiable and always visible placement of the markers in the scene, and this greatly limits the practicality of augmented reality applications. In recent years, natural feature-based registration can achieve localization of the camera by extracting and tracking two- or three-dimensional features in the scene like edge detection. In the field of monocular camera-based three-dimensional geometric registration, Davison [29] propose to use simultaneous localization and map construction algorithms to achieve real-time update of the camera pose in an indoor environment using a hand-waved camera. Later, they refine the process by simultaneously modeling the camera pose and features as a probabilistic state [30]. However, the computational cost of simultaneous localization and map construction is extremely high with a $O(N^3)$ (N is the number of features), greatly limiting the application and performance of augmented reality applications using this method. Later, Klein and Murray [31] propose to use parallel tracking and mapping algorithms to separate the process of feature tracking and map construction. After the initialization is done, the system tracks feature points with the optical flow with keyframes. Dense tracking and mapping-based methods [32] calculate depth based on image pairs with narrow baselines instead of matching the in-scene features. By minimizing the difference in depth of each point in relation to the reference keyframes, the system can achieve an accurate estimation of the camera pose. Recent simultaneous localization and map construction methods detect the structural changes of the environment and update the keyframes accordingly to achieve some tolerance against dynamic environments [33]. While state-of-the-art simultaneous localization and map construction approaches can achieve robust tracking to construct an accurate mapping, the major weakness is still modeling dynamic scenes with strong motions. Furthermore, when keyframes have larger baselines, the computational cost of alike methods dramatically increases and disqualifies for augmented reality applications. Therefore, instead of designing entirely markerless solutions to arbitrary scenes, some methods focus on certain targets to ensure a better performance. By using facial landmarks and matching with a general three-dimensional model, this type of approach estimates the camera extrinsic and determines the transformation for geometric registration [34]. The study in Chapter 3 uses a similar idea to effectively and efficiently register the geometry of the scene through hand pose estimation.

Compared to monocular camera-based registration, stereo-vision-based methods and depth-based methods all show great potential in augmented reality applications. Both methods are capable of achieving higher accuracy due to their accurate tracking of three-dimensional feature points. For instance, Zhu et al. [35] use image pairs of forward and backward observations to construct a landmark database to achieve reduced long-term error. KinectFusion [36] matches the depth of each pixel to reconstruct small-scale indoor scenes with greatly improved robustness. While it is a promising direction to utilize multi-modal information to refine the registration process, however, current commercial depth sensors usually suffer from noises and a short effective range, limiting the usability in augmented reality applications. In the next sections, I will discuss recent advance in learning-based scene understanding that can help with more accurate and efficient registration, which possess great potential in providing an immersive mixed reality experience.

2.2 Scene Understanding

Although state-of-the-art augmented reality systems have shown great potential in geometric registration, the process of visualizing the virtual objects with a high degree of realism still requires understanding multiple knowledge of the physical environment. For instance, without understanding the boundaries of different objects, the depth relation between foreground and background, or the correct lighting of the scene, there are a variety of challenges that need to be solved for full immersion. As videos and images captured by the camera are still two-dimensional, without fully functional reconstruction, virtual objects can hardly interact with the physical world or the user in an intuitive way. To circumvent the heavy computational cost of reconstruction and achieve real-time performance, recent deep-learning-based scene understanding greatly helps solve mixed reality problems. In this section, I will first briefly overview the most prominent deep neural networks that are frequently used for scene understanding tasks, then I will go over three topics, object recognition, semantic segmentation, and depth estimation, as they are crucial components of my following research presented in Chapter 3, 4, 5, and 6.


(a) A typical architecture of convolutional neural networks.

(b) An example of feature learning process for convolutional neural networks.

FIGURE 2.7: A typical architecture of convolutional neural networks with an example feature learning process.

Convolutional neural networks. Convolutional neural networks are one of the most popular choices for computer vision tasks for their ability to capture image features by simulating the process of the biological human brain [37]. The human brain starts from the sensory signal of individual pixels, followed by preliminary processing to find edges and features as the function of certain cells of the cerebral cortex, finally further abstracting the information to obtain a high-level understanding. A typical structure of convolutional neural networks is shown in Figure 2.7. While the features extracted at the bottom layers look similar to each other, more distinct features are extracted in higher layers, finally, different high-level features are combined at the top layers, enabling accurate high-level understanding. Convolutional neural networks achieve high accuracy of scene understanding when compared to traditional algorithms.

Convolutional neural networks often consist of three different components, the convolutional layer, which is responsible for extracting local features from the input image, pooling layers, which reduce the dimensionality to significantly reduce the number of parameters with statistical information, and the fully connected layer, which enables output of the desired result. Some of the most popular architectures include VGGNet [38], AlexNet [39], DenseNet [40], and GoogLeNet [41].

Encoder-decoder networks. Encoder-decoder architecture is a very popular network design in deep learning. An encoder $f(x)$ usually consists of a deep neural network (e.g. fully convolutional networks, convolutional neural networks, recurrent neural networks, etc.) that receives input and encodes the information into latent features $v = f(x)$ with reduced dimensions. Such latent features function to retain meaningful hidden semantic information of the input for the process of regressing the output. The decoder $g(v)$ also consists of a type of neural network, usually having the exact or similar structure as the encoder but processes feature in the opposite direction. Taking latent vectors as input, the decoder learns to map the feature from the input domain to the same domain as the desired output $\hat{x} = g(v)$. By minimizing the distance of $\hat{x}$ with the output for supervised learning, an encoder-decoder architecture can effectively learn to solve image-to-image translation problems. The architecture is explained in Figure 2.8. Popular applications of this type of network designs include semantic segmentation [8], super-resolution [42], translation [43], etc.

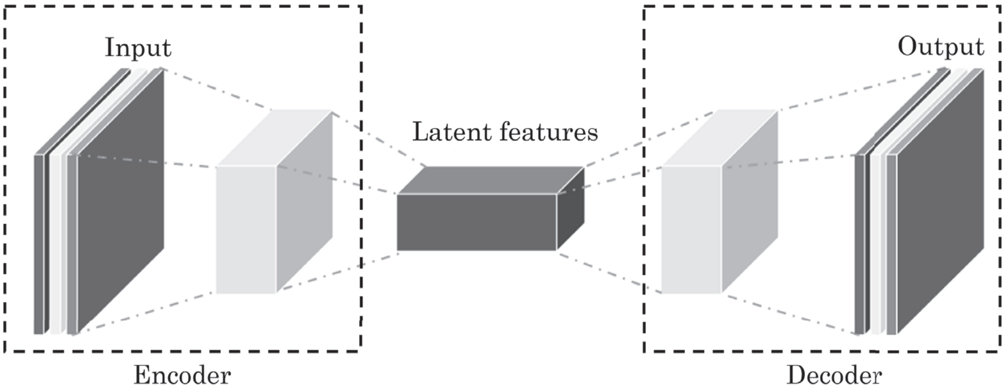


FIGURE 2.8: The architecture of encoder-decoder networks.

Generative adversarial networks. Generative adversarial networks [44] are a group of deep learning networks that consist of two distinct components, a generator, and a discriminator. When noises or latent features are input into the generator, the generator G learns a mapping to synthesize output that is close to the real samples as possible. Afterward, the discriminator D learns to evaluate the output distribution and distinguish the synthesized samples from the ground truth by maximizing the difference between samples generated by G and real samples. The process iterates until the generator is good enough to synthesize convincing results that resemble the real data distribution. The network structure is illustrated in Figure 2.9, and the process can be expressed as

$$\min_{G}\max_{D} V(D,G) = E_{x\sim p_{data}(x)}[logD(x)] + E_{z\sim p_{z}(z)}[log(1-D(G(z))], \tag{2.2}$$

where x represents real samples and z represents the input noise.

Since this interesting idea of the neural network is proposed, researchers have been making different improvements to barebone generative adversarial networks. Conditional generative adversarial network [45] synthesizes samples with additional class labels, Wasserstein distance-based loss function improves the performance for cases with non-overlapping distributions between the real and synthesized samples [46], deep convolutional generative adversarial network [47] introduces convolutional layers and batch normalization instead of fully connected layers and achieves better image-to-image translation. For a larger collection of generative adversarial networks, we suggest reading a more comprehensive survey [48].

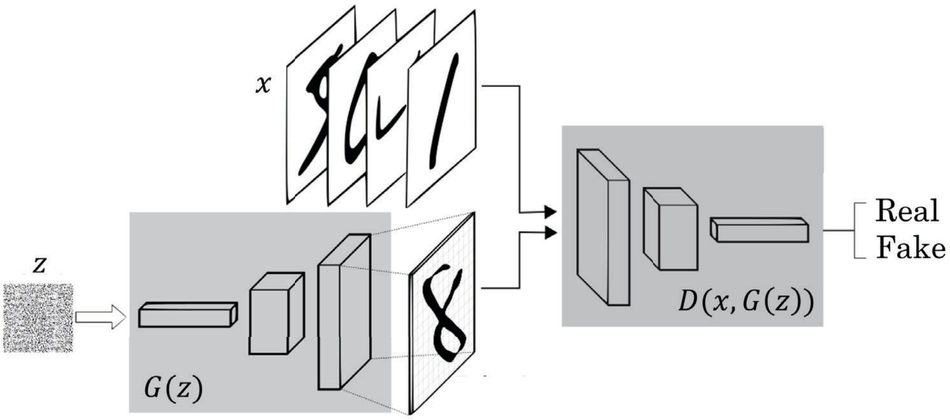


FIGURE 2.9: The architecture of generative adversarial networks.

2.2.1 Object Recognition

As one of the most fundamental and crucial components of computer vision problems, object recognition usually describes the task of identifying objects in images or videos. Tasks including image classification, object localization, and object detection, all belong to this category. When given an image with multiple foreground objects, image classification yields multiple labels that are presented in the given image, usually with integers and confidence. Object detection and localization locate the presence of the interesting objects and regress bounding boxes for each class label. By providing a high-level understanding of the image, object recognition can be applied to a wide range of real-world applications. As the cornerstone for complex computer vision problems such as segmentation and tracking, recent object recognition algorithms see great progress with the deep learning techniques mentioned above. Traditional object recognition approaches utilize hand-crafted features to detect and local interesting objects. In image classification and object detection and localization, calculating scale-invariant feature transform (SIFT) features [49] and histogram of oriented gradients (HOG) features [50] and matching the feature points was widely adopted. However, both the accuracy and the efficiency of non-learning-based methods are sub-optimal for augmented reality applications.

With the development of a deep neural network, the performance of object recognition has been dramatically improved. Deep learning-based object recognition can be categorized into two different types. One proposes plausible region divisions of the input image followed by classifying each patch afterwards, such as region-based convolutional neural network (R-CNN) [51], mask R-CNN [52], fast R-CNN [53], feature pyramid network [54]. R-CNN first proposes to extract features from input images using convolutional neural networks for object recognition tasks. It was later improved with a fully convolutional structure [55], a region proposal network [56], and a multi-scale pyramid representation of features in convolutional layers [57]. The other directly regress the label and the location of interesting objects without a region proposal stage, such as single-shot multi-box detector [58], You Only Look Once (YOLO) [59], deconvolutional single shot detector [60]. This type of end-to-end approach regresses the bounding boxes and the confidence of different labels simultaneously. To further facilitate real-time augmented reality applications, template-based object detection shows great potential [61].

2.2.2 Semantic Segmentation

Semantic segmentation is a computer vision task that aims to understand every pixel of the input image. By classifying each pixel with a certain label, it can provide a high-level understanding of the image and facilitate a wide range of applications. In robotics and autonomous driving, mounted cameras capture the front view to predict the location of obstacles and decide which lane is safe to drive [62]; In medical diagnosis, semantic segmented X-ray images significantly reduce the time for the radiologist to perform an accurate analysis [63]; In augmented reality, semantic segmentation can be used to separate foreground objects and background environment, making it possible to composite virtual visualizations with correct occlusions [8].

Traditional semantic segmentation approaches utilize Markov Random fields [64], random decision forests [65], and support vector machines [66] to achieve more accurate results over naïve solutions such as thresholding, clustering, edge detection, region growing. These methods usually require additional pre-processing and post-processing to maintain robustness.

As of now, learning-based approaches have made great progress in semantic segmentation with the advantages of convolutional neural networks: shallower layers have

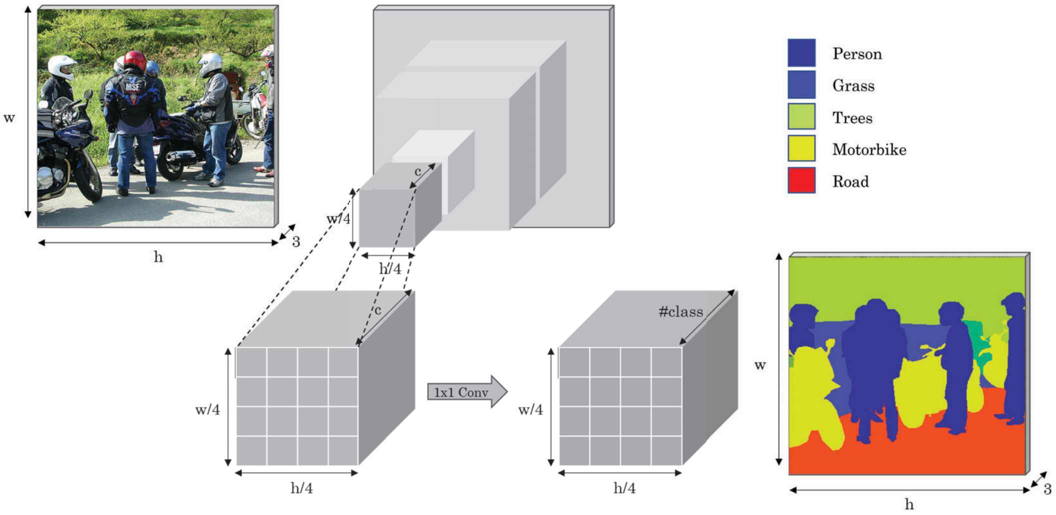


Figure 2.10: An example of a deep learning approach to semantic segmentation using an encoder-decoder network.

a smaller perceptual window to learn features from local regions, while the deeper layers understand abstract features. Initial solutions are patch classification and each pixel is classified independently using patches around the pixel. This type of method is modified from existing convolutional neural network architectures such as VGGNet and GoogLeNet and requires a fixed-size image with its fully connected layer. Later, a fully convolutional neural network [55] allows predicting dense per-pixel labels for images with arbitrary sizes and improved efficiency. Besides the fully connected layer, the pooling layer also causes the loss of location information during the process of convolution. To solve this issue, conditional random fields and Markov random fields are further incorporated into deep learning-based methods [67] to further refine the ability of localization. Encoder-decoder structured network is a popular choice for end-to-end training, realized by gradually reducing and restoring the spatial dimension to retain the spatial information of the original image, as shown in Figure 2.10. However, this structure is prone to loss of details for high-resolution images due to its encoding-decoding process. To alleviate this disadvantage, U-Net [63] adds skip connections to further improve the restoration of the spatial information. While the performance of these models greatly surpasses conventional algorithms, the efficiency is still not enough for real-time inference, which is vital in applications like autonomous driving and augmented reality. Point-wise convolutions and dilated convolutions are later proposed [68] to achieve low latency with comparable accuracy.

2.2.3 Depth Estimation

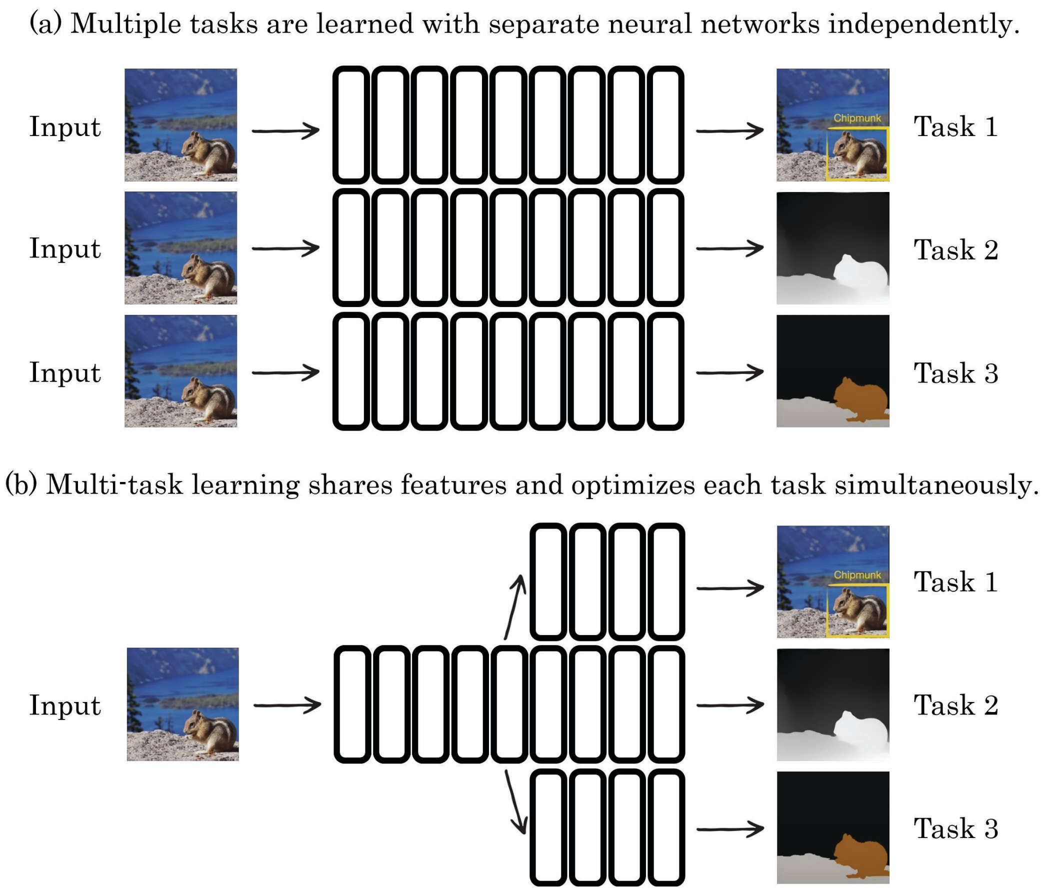
Depth information provides valuable hints for computers to understand the geometry of the scene. While it is possible to directly acquire the depth map with structured light cameras, such as Microsoft Kinect with infrared sensors, limited effective range, and sensor noises are problematic in a wide range of applications. In addition, due to the interference of sunlight, it is also less reliable in outdoor environments. Light detection and ranging (LiDAR) devices are robust in outdoor environments, but the cost dramatically increases with their resolution. Stereovision can calculate disparity based on a geometry model, however, different intrinsic and extrinsic of camera setups need to be accounted for to minimize the reconstruction error. To facilitate ubiquitous augmented reality systems, monocular depth estimation is a crucial yet challenging task. With accurate depth information, it is possible to enhance larger scale geometry-aware mixed reality experiences such as interactive physics of virtual objects [69]. In chapter 5, we showcase the implementation of using monocular depth estimation to enable rendering visual effects with correct occlusions. Other applications include robot vision, three-dimensional reconstruction, photo enhancement, etc.

Early literature uses Markov random fields-based probability model [70] and non-parametric depth sampling [71] to synthesize plausible depth maps. In recent years, learning-based approaches have demonstrated their capability of learning the mapping between color images and depth maps and generating depth prediction with improved accuracy and high efficiency. Supervised methods [72] [73] [74] [24] directly regresses the depth value of each pixel and minimize the distance between the prediction and the ground truth using dataset acquired with depth sensors. Similar to semantic segmentation, fully convolutional neural networks [73] have also seen considerable improvement in accuracy. However, considering high-quality depth dataset is expensive to obtain, unsupervised methods are also a popular research topic for depth estimation. Stereo-view-based [75] and multi-view-based methods [76] projects multiple views according to predicted depth maps and minimize the reconstruction error. However, some methods require stereo inputs [75] and some methods require reliable matching between images [77], further limiting their practicality of them in augmented reality applications. The more detailed and specific background will be addressed in individual studies described in Chapter 4 and Chapter 5.

2.3 Multi-task Learning

Mixed reality applications usually require an accurate and efficient understanding of the scene with multiple modalities. Besides predicting the pose of the camera with a reliable geometric registration process, recognizing in-scene objects of interest, understanding their high-level meaning of them, or even reconstructing three-dimensional models are all important capabilities of an immersive mixed reality application. Although deep learning has pushed computer vision forward greatly, maintaining a satisfying accuracy while keeping computational costs low to facilitate real-time usage is still a challenging goal.

Motivated by the learning process of human beings, using the knowledge from one task to assist other tasks is an intuitive and straightforward idea. For instance, when humans learn to recognize the shapes of a certain type of object, it is intuitive to them to accomplish a related but different task, such as drawing them out. When



(a) Multiple tasks are learned with separate neural networks independently.

(b) Multi-task learning shares features and optimizes each task simultaneously.

FIGURE 2.11: A comparison between the traditional single-task based learning designs and the multi-task learning designs.

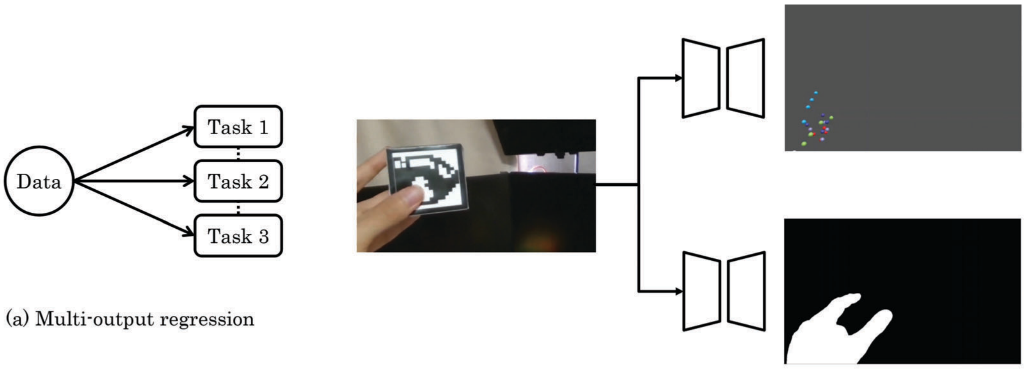
it comes to deep learning, traditional approaches learn a latent representation of the same scene independently with multiple neural networks, as illustrated in Figure 2.11 (a), and predict in isolation for tasks such as object recognition, depth estimation, and semantic segmentation. Multi-task learning with a schematic diagram of Figure 2.11 (b), proposes to learn the underlying features that are shared across different representations of the same input so that each model can better understand the context, which is otherwise difficult to comprehend independently. In the past few years, multi-task learning has gained popularity in computer vision, computer graphics, natural language processing, and many other research communities. In this section, we first explain the definition of multi-task learning. We then categorize multi-task learning approaches into three different learning paradigms based on the input and output, and briefly review the strength and applications of each of them.

Given n multiple tasks $T_i, i = 1, 2, 3, \dots, n$ where all tasks are weakly or strongly related, let L_i denote the loss of each specific task to optimize, the objective of the entire multi-task learning model is to optimize

$$L_{multi-task} = \sum_i w_i \cdot L_i, \tag{2.3}$$

where w_i defines the weight of a single task. By learning multiple tasks together, this strategy usually improves the performance of every task. Homogeneous-feature multi-task learning methods usually receive different inputs to optimize the same task, while the heterogeneous-feature ones learn to optimize different types of tasks such as depth estimation and semantic segmentation. Both methods can be applied to supervised, unsupervised, and semi-supervised learning processes. Without special explanation, we focus on supervised settings for the studies in this thesis.

The advantages of multi-task learning can be concluded into three different aspects. First, to ensure a good result for different tasks, the model usually needs to gain “true” knowledge of the input samples, and make the hidden representation more robust and accurate. A great improvement in generalization is achieved through “eavesdropping”, leveraging the useful features of other tasks during the training process. A better generalization will result in higher accuracy and robustness when compared to learning the same task independently. Second, with shared layers in the multi-task learning model, the overall model sizes are usually reduced. This inherent feature of multi-task learning networks greatly benefits mobile mixed reality applications with decreased memory usage. At the same time, by eliminating the process of encoding features repeatedly, the inference time can also be optimized with multi-task learning, further facilitating the real-time demand of mixed reality. Finally, different training data are implicitly aggregated during the multi-task training process, alleviating the problem of insufficient training data. For certain mixed reality applications such as omnidirectional format media, quality training data is indispensable yet expensive to acquire. The ability to accomplish several tasks at the same time with improved effectiveness and efficiency, while reducing the combined size of models and requirement for the large-scale training dataset, makes multi-task learning perfect for solving issues in mixed reality. I believe research in multi-task scene understanding can help better understand the physical and virtual environments and improve an immersive mixed reality experience in the future.



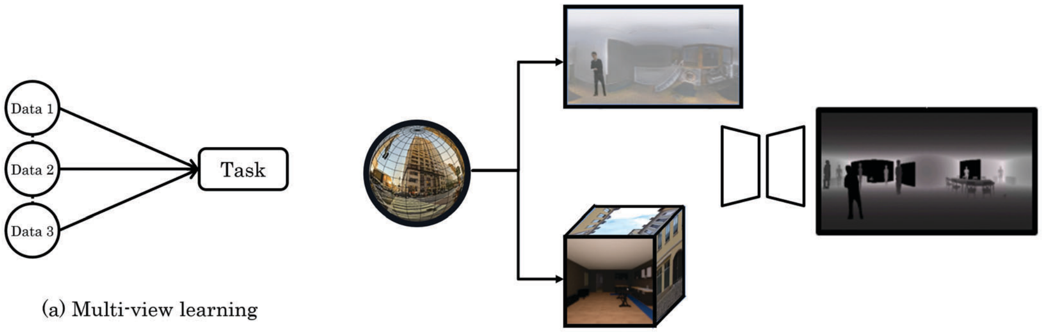
(a) Multi-output regression

(b) An example of multi-out regression used in the study explained in Chapter 3

FIGURE 2.12: An example of using multi-output regression for learning hand pose and semantic segmentation simultaneously [8].

2.3.1 Multi-output Regression

Multi-output regression, or multi-label learning, describes a multi-task learning paradigm in that every single input corresponds to multiple modalities of ground truth labels. An example is shown in Figure 2.12. Assuming that a training dataset of multiple modalities is available, solving all the subproblems in parallel as a multi-output regression problem is advantageous [78]. Using the example from the study described in Chapter 3 [8], if I want to train a neural network to learn hand semantic segmentation, a model that is only trained with the color to semantic segmentation independently might fail in real-world scenarios with different lighting and perspectives. Instead, a multi-output regression design that estimates the hand pose simultaneously can give hints and advise a more accurate latent representation of hands.



(a) Multi-view learning

(b) An example of multi-view learning used in the study explained in Chapter 4

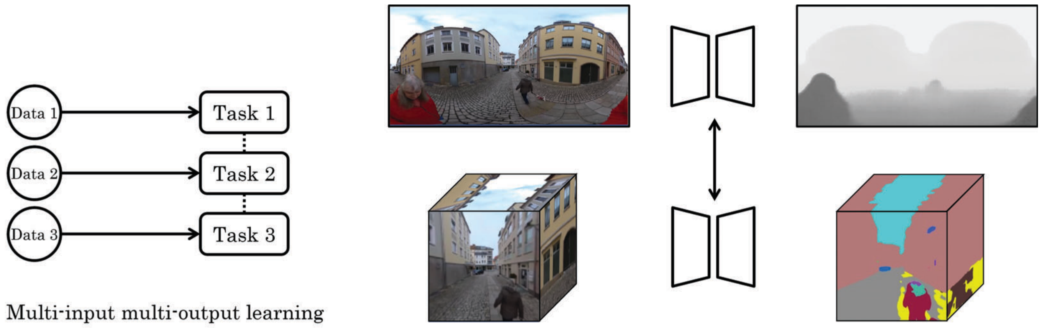
FIGURE 2.13: An example of using multi-view learning for learning depth estimation with equirectangular view and cubemap view simultaneously [79].

2.3.2 Multi-view Learning

Contrary to multi-output regression with single input and multiple outputs, multi-view learning aims to learn a single task from different inputs. Considering that multi-view data are widely available in different applications across diverse domains, learning with different views can improve the generalization for solving each view independently. Here, views are not limited to different positioned cameras capturing the same scene, instead, they can be sampled with different modalities such as video stream and auditory signals [80], words and images [81], etc. The additional information from other views provides insight to learn better features when analyzed simultaneously. A straightforward approach is to directly apply different views to a single-view network. However, it is prone to overfitting for datasets with a relatively small size. Recent multi-view learning learns the correspondence that can models the features from different views to each other, showing an advantage of better accuracy. An example of multi-view learning is shown in Figure 2.13. As shown in the Figure 2.13 (b), since quality omnidirectional training data is not widely available, another major advantage of multi-view training is that manually synthesizing new views can still improve the performance of the network. By projecting spherical information to an equirectangular formant and cubemap format, the approach described in Chapter 4 achieves improved accuracy over traditional deep learning methods.

2.3.3 Multi-input Multi-output Learning

Due to that there is an underlying correlation between multiple tasks, and multiple views with different modalities also share overlapping features, it is intuitive to combine multi-output regression and multi-view learning to benefit from the advantages of both. One example is shown in Figure 2.14, the study described in Chapter 5 learns depth estimation from an equirectangular view and semantic segmentation from a cubemap view simultaneously to achieve higher accuracy. The challenge of jointly learning multiple tasks is to find a good balance between each task. Previous methods manually refine the weight by comparing the difficulty of each task [82], using homoscedastic uncertainty to determine the weight of each task [83], and using gradient normalization to dynamically adjust weights [84].



(a) Multi-input multi-output learning

(b) An example of multi-input multi-output learning used in the study explained in Chapter 5

FIGURE 2.14: An example of using multi-input multi-output learning for learning depth estimation and semantic segmentation simultaneously [24].

Datasets play an important role in multi-task learning and computer vision tasks. For supervised training, it is challenging to capture a great diversity of objects with high quality, in addition to multiple modalities that pair with each other. Popular datasets for multi-task learning include the NYUv2 dataset [85], which has color, annotated semantic segmentation, and depth maps for indoor scenes, the CelebA dataset [86] contains color and multiple attribute annotations, the Cityscapes dataset [87] consists of color, classifications, and instance segmentation. In the context of mixed reality applications, many complex computer vision tasks are hindered by scarce and expensive labeled data. For instance, egocentric hand-object is an important feature of immersive mixed reality, however, it exceeds the effective range of commercial depth sensors, making such training data difficult to acquire. Moreover, 360-degree images that are frequently used in mixed reality have severe distortions when projected to a two-dimensional plane, rendering traditional perspective datasets such as NYUv2 less effective. Therefore, an effective data augmentation method is crucial for using multi-task learning in mixed reality. Previously, learning-based data augmentation has seen great success to reduce the cost of data collection. The generative adversarial network can effectively synthesize new views, modalities, and domains [88]. In this thesis, we use multiple different learning-based data augmentation methods to facilitate effective multi-task learning to understand the images in the context of mixed reality.

Chapter 3

Grasping the Local: Solving Hand-object Occlusion in Mixed Reality

In this chapter, we start from a smaller spatial scale by focusing on the interaction between users' hands and in-hand objects in a mixed reality environment. While a similar idea has already been discussed in the previous thesis [89] [8], we further polished the idea in later research by successfully utilizing the latest data augmentation and learning-based scene understanding algorithms. The originality and contributions of this published work [90] provide valuable insights into understanding one of the most crucial scopes of mixed reality technology: local interaction.

3.1 Introduction

Over the last few years, the concept of mixed reality, usually comprising virtual reality and augmented reality, has been drawing a growing amount of research interest from many for its plentiful capabilities and applications [91]. Instead of rendering an entire virtual environment from scratch, recent mixed reality highlights its capability of aiding users to accomplish various tasks [92] and providing them with a seamless and immersive experience through overlaying rendered virtual objects onto their visuals of surroundings [93].

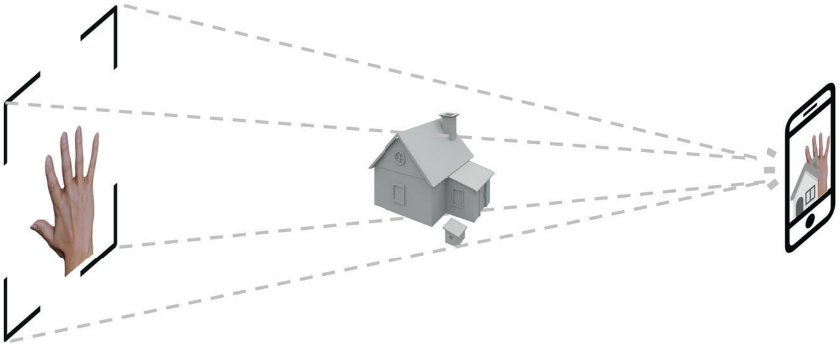


FIGURE 3.1: Webcam-based augmentation process in a mixed reality application without occlusion handling.

This augmentation process is usually realized with more advanced head-mounted displays, either optical see-through types or streaming the real world with mounted webcams. Thanks to highly accurate and efficient computer vision algorithms and polished applications, mixed reality hardware has recently seen increasing commercial acceptance for assisting real-world tasks [94]. A wide range of research fields has been incorporating mixed reality as a feasible component to their research to further their understandings of human-computer interactions and better embedded systems. Numerous applications of education, healthcare, and entertainment verify the benefit of seamlessly augmenting real-world visuals with mixed reality objects.

For augmented objects that are virtually constructed and have 3-dimensional models available, when combined with computed 6 degree-of-freedom trajectories of the device, it is trivial to render occluded portions. However, the real world is usually observed by using RGB cameras or depth sensors. With noise and sparse tracking, it is challenging to determine the foreground-background relationship between the augmented objects and the reality (see Figure 3.1), resulting in incorrect occlusions for most of the rendered virtual objects' pixels. If we render the virtual object in absence of correct occlusions, an unrealistic "floating" illusion will lead to incorrect depth/distance perception, ruining an immersive mixed reality experience [95]. An example can be seen in Figure 3.2.

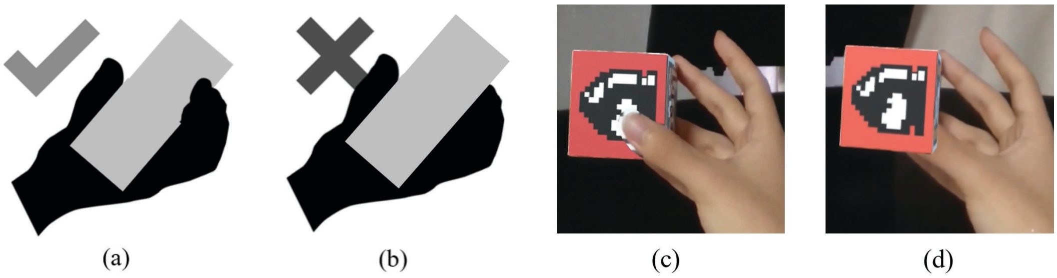


Figure 3.2: An example of hand-object interactions with incorrect occlusions in a mixed reality application.

The hand is one of the key components in mixed reality, and controller-free hand-object interactions are critical to a wide range of mixed reality applications such as surgery training [15], tangible interface [14], and driving simulations [21]. However, the feasibility and immersive experiences of interaction-involved applications are severely limited by false occlusions. When users are freehand interacting with objects, it is highly possible that their palm and fingers will partially occlude the object.

To resolve hand-object occlusions in mixed reality, previous literature proposed several methods to resolve occlusions for rendered objects. Nevertheless, the quality of the proposed approaches is sub-optimal when directly adopted in hand-object scenarios. Reconstruction-based methods utilize algorithms such as simultaneous localization and mapping to acquire the geometry of the entire environment and then render virtual appearances with Z-buffer [95]. However, these methods are not viable for hand-object interactions. With highly dynamic egocentric motions, it is difficult to achieve quality reconstruction in a timely manner. For tracking-based methods that trace the contour of targets with optical flow, their performance is restricted by the highly deformable hand structures and acute self-occlusions [96]. Despite the fact that depth-based methods can overcome previous challenges, shadows and noises induced by sensors, misaligned edges, and limited performance when utilized for close ranges all hinder the feasibility to be applied to egocentric scenarios [97].

In this chapter, we start from a smaller scope and try to better understand the user in mixed reality with scene understanding and data augmentation. We first present a photo-realistic and occlusion-aware hand-object database comprising both color and depth information. It aims to alleviate the ambiguity induced by occlusions and facilitate the following deep learning system. By synthesizing hand-object samples with occlusions and augmenting color images with photo-realistic appearance with a generative adversarial network, a large-scale multi-modal database accommodating accurate annotations of hand joints and semantic segmentation is augmented with minimal manual effort.

Taking advantage of the proposed database, we design a jointly trained deep learning network that shares knowledge between the tasks of predicting hand postures and generating semantic segmentation. By passing information between tasks, our system can predict more consistent results compared to existing single-task architectures. Making use of the occlusion-aware database, the jointly learned neural network shows robust performance in hand-object interactions. With accurate predictions of the hand posture and semantic segmentation, the framework provides valuable information for the following novel real-time optimization system to finally resolve the hand-object occlusions.

With robust posture and semantic segmentation being available, we design a novel real-time optimization system that computes valid occlusions when augmenting physical objects with the appearance of a virtual object. To overcome the static-scene-only constraint of reconstruction-based methods, the proposed system efficiently reconstructs the spatial area of interest by performing a two-step process: optimize and fit. By iteratively optimizing a parameterized virtual hand model with regard to the semantic segmentation result followed by fitting the optimized model back to the predicted postures instantaneously, the system eventually computes occlusion masks in real-time and renders the hand-object interaction with precise occlusions.

Experimental results highlight accurate and realistic overlays of rendered hand-object interactions. A quantitative benchmark shows improved performance over methods of state-of-the-art. A qualitative comparison shows more natural visuals. A comprehensive user study verifies more intuitive mixed reality experiences against existing methods. We expect the result of this chapter to be applied to egocentric mixed reality applications with a focus on hand-object interactions including simulations.

The contributions of this chapter are summarized as follows:

- A photo-realistic and occlusion-aware hand database with paired color and depth information that facilitates robust hand posture prediction and semantic segmentation. The database is available for future research through the script.

- An occlusion-aware jointly trained deep learning network that predicts hand postures and semantic segmentation simultaneously and instantaneously. With shared knowledge between two tasks, the network yields accurate results even under challenging hand-object interactions with severe occlusions.

- A novel real-time optimization system for computing correct hand-object occlusions. It spatially reconstructs the area of interest using iterative optimizing and fitting.

The rest of the chapter is organized as follows. We revisit existing occlusion solutions and hand posture prediction methods in Chapter 3.2. In Chapter 3.4 and 3.5, we elaborate on the proposed occlusion-aware RGBD database and the joint deep learning framework to predict hand postures and semantic segmentation in real-time. In Chapter 3.6, we present the novel two-step system to resolve hand-object occlusions. Experimental details, evaluations, and the user study are described in Chapter 3.7. Finally, Chapter 3.8 concludes this chapter.

3.2 Related Work

As the main objective of this work, we first review previous occlusion solutions for mixed reality. Since the hand-object database and the jointly trained deep learning system are both crucial components of our method, we also revisit existing deep learning posture prediction approaches and hand posture prediction databases.

3.2.1 Occlusion in Mixed Reality

A low-quality rendering in mixed reality, such as inaccurate occlusions and false lighting, will spoil the immersive experience [95] and introduce incorrect perception of the scene [98]. Methods in the following research compute occlusions to faithfully composite virtual objects onto the visuals of the real surroundings with no prior knowledge of the scene geometry.

Tracking-based solutions. Semi-automatic approaches, including handpicking the object boundary [99] and manually annotating the foregrounds and backgrounds [96], usually calculate occlusions with the traced contour of the foreground target. This type of method requires additional lengthy input and functions based on the implicit assumption of a finite and constant contour. Moreover, the performance of contour extraction algorithms can easily be affected by false initialization, inadequate resolutions, incorrect local minima, and so on.

Reconstruction-based solutions. Considering that the foreground-background relationship for multiple objects can be computed in a straightforward way given precise models of the current scene, utilizing a fast simultaneous localization and mapping algorithm and reconstructing the entire scene is a viable choice [95]. Nonetheless, this solution is extremely computationally expensive with strict requirements: an almost static and well-textured scene coupled with translating motions from the perspective. When it comes to hand-object interactions, it is difficult to establish a satisfying reconstruction. Another practice uses prepared 3-dimensional virtual models of the real-world targets to perform a fitting task [100] during usage. The accuracy of this approach highly depends on the robustness of tracing and often yields subpar results for deformable targets. Without an occlusion-robust tracking solution, optimization-based methods are prone to inconsistent results.

Depth-based solutions. Utilizing supplementary depth sensors to obtain the per-pixel depth information directly can serve the purpose as well. However, temporal shadows and noises, misaligned depth edges, and other underlying problems lead to low-quality results. A stream of research refines the yielded depth information at the boundaries [97] to improve the overall consistency and accuracy. However, most state-of-the-art algorithms are impractical when the computational cost is a trouble that cannot be overlooked in interactive mixed reality applications [101]. Refining the obtained depth in a "layered" fashion with cost-volume filtering [102] can help achieve real-time performance, but it generalizes poorly for complex scenes such as interactions. Besides, the hand being simultaneously foreground and background object would make color-based segmentation impractical.

By leveraging the efficiency of tracking-based and model-based methods, we propose a real-time approach that solves hand-object occlusions in mixed reality without introducing a lengthy initialization, additional sensors, or an expensive process of reconstructing the entire scene.

3.2.2 Occlusion-aware Hand Posture Prediction

Learning-based approaches. Vision-based 3-dimensional hand posture prediction is a demanding problem to solve due to its high degree of freedom articulations and severe self and hand-object occlusions. Marker-less approaches introduce generative components to improve the prediction between the simulation and the observation, such as consistencies between frames [103], iterative closest point [104], particle swarm optimization [105], etc. However, most methods require a lengthy initialization process, and their precision highly depends on quality observation. To address such restrictions, learning-based discriminative components have become a popular choice recently [106]. Although being beyond the scope of this study, adapting MANO [107] to solve interactions between hands [108], and joint tracking of hand and object [109] are all promising directions for improvements. In this work, we use a learning-based approach to precisely and efficiently predict hand postures without manual initialization.

Occlusion-aware databases. One of the major issues of learning-based hand tracking methods is difficulties in preparing training samples with correct 3-dimensional annotations of the joints. Recently, a handful of high-quality databases for 3-dimensional hand posture prediction are released [105]. Even databases constructed upon manual annotations exist [106], inaccuracy and insufficient size are problems that can hardly be dismissed. Multi-view approaches [110] suffer from the limitation of occlusions due to their outside-in setups. To obtain accurate paired data, some works render synthetic paired color and depth data for hand-object images with virtual hand models and cameras [106]. Nevertheless, existing CNNs-based approaches that are trained on synthetic data generalize poorly due to the domain gap between synthetic and real-world images. To improve the accuracy of occlusion-aware hand posture prediction from RGBD input, our method leverages a generative adversarial network and incorporates the geometric consistency loss [88] to synthesize a photo-realistic hand database that comprises paired color and depth information.

3.3 The Framework Overview

To achieve the goal of augmenting in-hand objects with correct occlusions in real-time, our approach consists of two major components (see Figure 3.3). The first one is an occlusion-aware joint learning framework for (a) hand posture prediction and (b) semantic segmentation. This involves building an occlusion-aware hand database, a joint posture prediction module, and a semantic segmentation module. The second one is a real-time optimization-based occlusion resolving system (c) for virtual object augmentation. This involves optimizing a hand model using the predicted semantic segmentation, fitting the model with the tracked hand posture from the joint learning step, and resolving occlusion masks for augmenting virtual objects.

3.4 Data Augmentation with Generative Adversarial Network

We proposed a photo-realistic and occlusion-aware hand-object database of paired color images and depth maps to facilitate learning-based hand posture prediction and semantic segmentation in interactions. This is motivated by the difficulty to annotate 3-dimensional hand joints and semantic segmentation in occluded samples. Capture-and-annotate methods are inadequate due to ambiguities, glove-based methods yield different appearances and are not suitable for bare-hand applications. To acquire accurate posture data and semantic segmentation for hand-object interactions, synthesizing samples and annotations is a more efficient and effective direction when compared to other methods.

To efficiently synthesize the photo-realistic and occlusion-aware RGBD database, we repurpose an existing synthetic RGBD hand database [106]. It contains samples with hand-object interactions and joint annotations. To adapt it to our use, we first re-render the hand into binary masks to facilitate the semantic segmentation task. Inspired by [88], we then use the generated semantic segmentation as a geometric constraint to transfer the photo-realistic appearance to synthetic samples by training

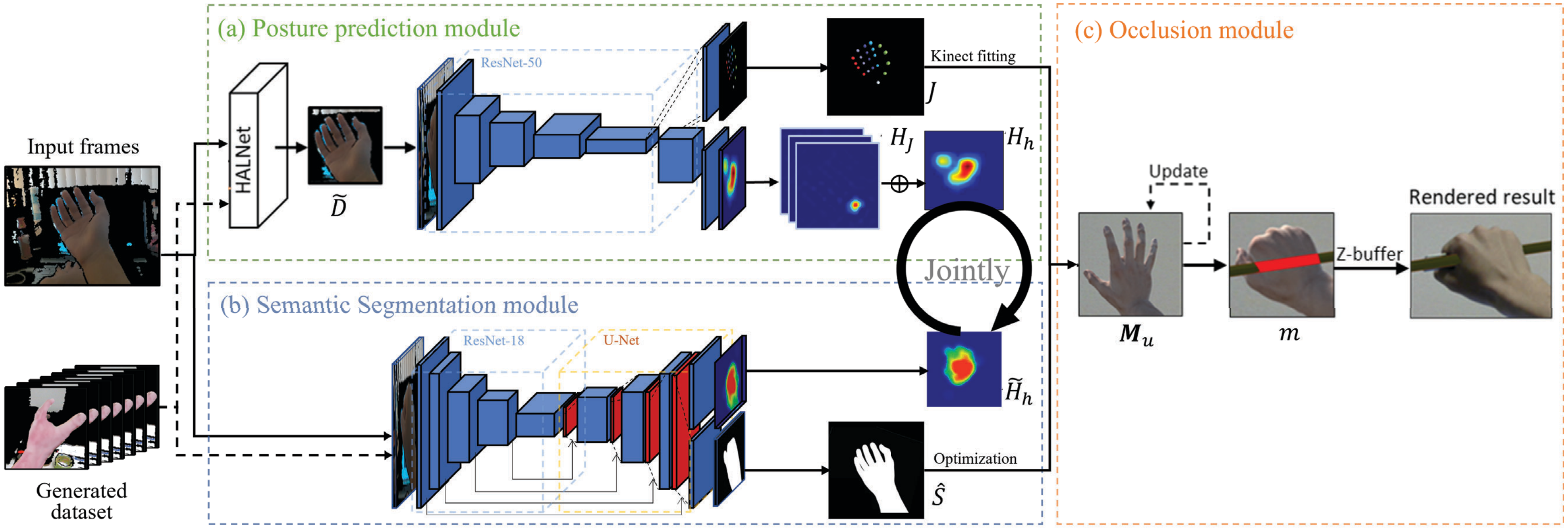


FIGURE 3.3: The architecture of our hand-object occlusion resolving system.

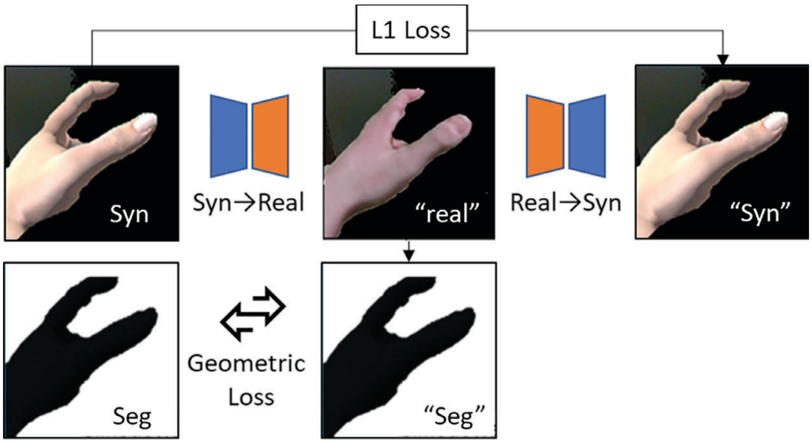


FIGURE 3.4: The CycleGAN architecture of our photorealistic RGBD data generating network.

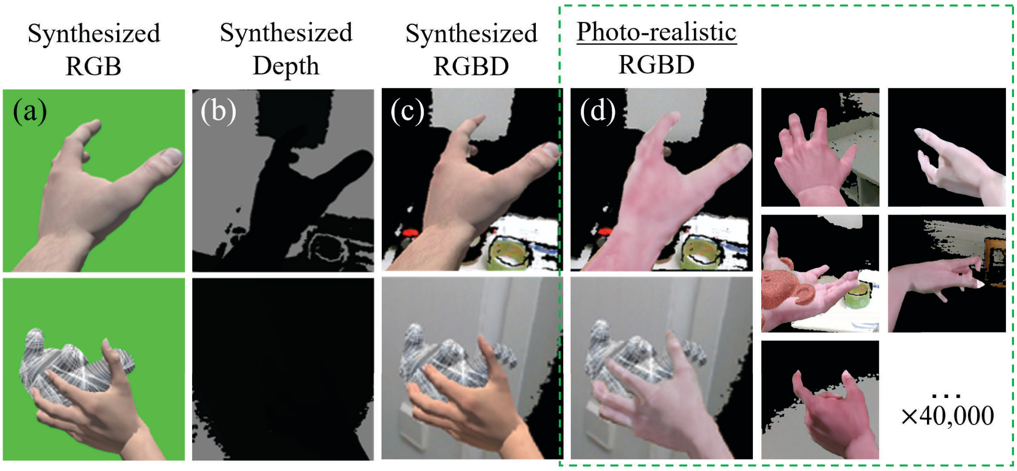


FIGURE 3.5: An example of generated photorealistic **RGBD** hand database. Images from left to right are (a) synthesized RGB; (b) synthesized Depth; (c) synthesized RGBD; (d) photorealistic RGBD.

a CycleGAN. To ensure annotations stay correct before and after the transfer, we calculate the geometric consistency loss from the predicted and real silhouettes:

$$L_{geo} = -\sum_{i}(S_i log\hat{S}_i + (1 - S_i)log(1 - \hat{S}_i)) \tag{3.1}$$

where S is the rendered segmentation and $\hat{S}$ is the mask of the generated sample. The pipeline is explained in Figure 3.4 and Figure 3.5 shows some samples of the database. We only show the synthetic-to-real half of components for simplicity.

By bridging the domain difference between synthesized and real samples, our approach greatly improves the accuracy of learning-based approaches. A photo-realistic RGBD hand-object database with occlusions that contains 40,000 accurate hand posture and semantic segmentation annotations is created to facilitate various applications.

3.5 Real-time Multi-output Regression Architecture

With precise hand posture annotations and semantic segmentation of photo-realistic hand data available, we propose a deep-learning system that simultaneously predicts semantic segmentation and hand posture. As illustrated in Figure 3.3 (a) and (b), we pass the input to our joint-learning system with a resnet-structured posture prediction module and a U-net-structured semantic segmentation module running parallelly to each other.

To achieve a more coherent prediction even under severe occlusions, we exploit the information of posture annotations and inform the other task of potential uncertainties with concatenated heatmaps of posture prediction. More specifically, in addition to predicting 3-dimensional coordinates of each joint, 2-dimensional Gaussian heatmaps of every joint are also created with the posture prediction module. We find that two tasks are complementary to each other since hand joints should always be located within the hand contour, hence we concatenate and convey heatmaps to the semantic segmentation module. With a heatmap loss calculated to reduce false-positive predictions, our joint learning system has improved accuracy compared to two separate modules without communications.

3.5.1 Posture Prediction Module

To predict hand postures with improved accuracy and robustness, we take advantage of our generated photo-realistic hand database and propose the posture prediction module to estimate hand postures. With the input of an RGBD image, the posture prediction module is trained to regress the 3-dimensional displacement of 21 hand joints. As additional information to be shared with the other task, 2-dimensional Gaussian heatmaps are also output in image space with the posture prediction module.

A two-step localizing-and-tracking method is used to improve the robustness of the network. We adapt the HALNet [106] and trained with the proposed database to localize the hand when an image is inputted. $\tilde{D}$ that contains the hand will be cropped from the input RGBD frame D and passed to the next step. We then propose a posture prediction network bases on a modified ResNet-50 structure with reduced layers to achieve real-time performance. By minimizing the Euclidean loss between predicted joints and ground truth $\hat{J}$, our hand posture prediction module can estimate 3-dimensional hand joints' coordination J in real-time during usage.

$$d_{pred} = \sum_{i=0}^{N} J - \hat{J}_2^2 \tag{3.2}$$

Since hand posture annotations being a high-level information is expensive to acquire but highly correlated to and beneficial for different tasks (model reconstruction, normal estimation, etc.), we exploit the learned image to pose mapping through heatmap representations. As 2-dimensional likelihood heatmaps H_j are regressed during pose estimation for each joint, we concatenate heatmaps of each joint to obtain hand heatmap H_h, and pass the information to the segmentation module during training.

3.5.2 Semantic Segmentation Module

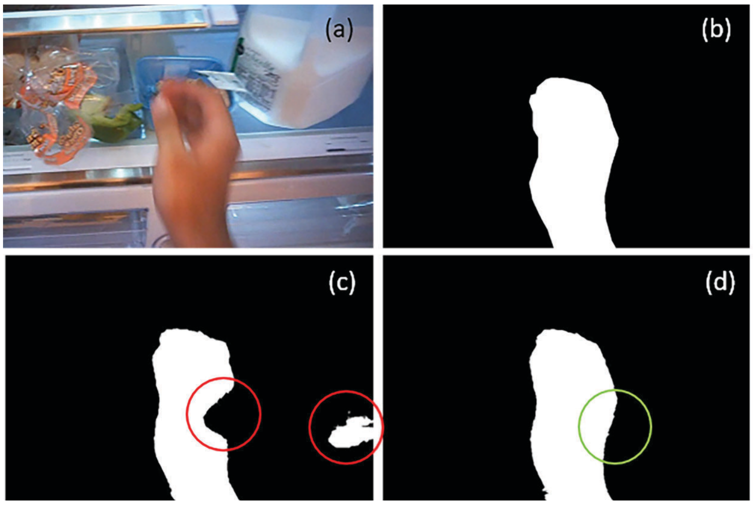


FIGURE 3.6: A comparison between segmentation masks estimated with and without heatmap loss. (a) Input color image. (b) Ground truth segmentation mask. (c) Estimated mask without the heatmap loss. (d) Estimated mask with the heatmap loss.

To facilitate the real-time optimization system in the next step, we propose a semantic segmentation module to estimate hand segmentation from an image input. To take advantage of hand posture knowledge, the segmentation module outputs intermediate heatmaps for mask estimations, and calculates an additional heatmap loss to ensure that the hand joints fall within the segmentation estimation. Combined with the synthesized occlusion-aware pairwise images and masks, this module can handle occluded scenes with improved performance.

Structure-wise, the segmentation module consists of a U-Net structure with the encoder part replaced with a ResNet-18 backbone. Considering the binary output mask, we choose the dice coefficient as our segmentation loss function.

$$L_{dice} = \frac{2|\hat{S} \cap S|}{|\hat{S} + S|} \tag{3.3}$$

The $\hat{S}$ in Eq. 3 is the estimated segmentation while S is the ground truth.

To reduce false positives in estimated masks, we leverage the information, heatmap of the hand H_h obtained from the posture prediction module. Apart from the main loss between the segmentation masks, with the average pooling, we create activation maps at the same time for calculating the complementary heatmap loss between the $\tilde{H}_h$ obtained by the segmentation module and the H_h with Euclidean loss. The weight of heatmap loss is set at 0.1 during training, and the resolution is downscaled to 640x360 to maintain a stable speed. As demonstrated in Figure 3.6, we can clearly see the effectiveness of guiding the semantic segmentation task through heatmaps passed by the posture prediction module.

3.6 Real-time Optimization System for Occlusion

In this section, we explain our novel real-time optimization system that resolves the occlusions in hand-object interactions through a 2-step optimizing-and-fitting method. With the hand posture and semantic segmentation information available, we spatially reconstruct the region of interest with high accuracy by first iteratively optimizing a virtual hand model based on the user's hand and then fitting the updated model to predicted hand postures.

This system design circumvents the limitations of previous occlusion resolving approaches effectively. For reconstruction-based methods, we overcome the constraint of only being able to recover a static scene by fitting the optimized model to estimated joints in real-time. For tracking-based methods, the problem of low-quality outcomes against changing shapes or under severe occlusions is solved by our occlusion-aware joint learning system. Instead of tracking contour directly, we calculate it through more occlusion-robust hand postures. With local models of the hand and the virtual object available, we then augment the object through an occlusion mask calculated in real-time.

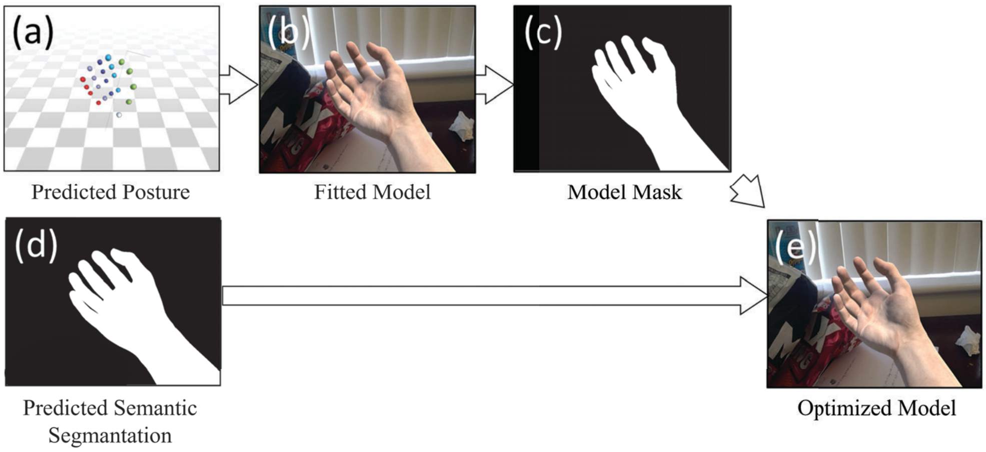


Figure 3.7: The process of updating the hand model during runtime with estimated hand poses and masks. The model is optimized by minimizing the distance between observed and model's rendered segmentation.

3.6.1 Model Optimization

With the current frame of hands available, a hand posture and semantic segmentation are estimated with the joint learning system and inputted to this occlusion module to optimize a virtual hand model M_d in real-time. This iterative process (Figure 3.7) is effective and efficient against hand-object interactions by only reconstructing models. More specifically, by fitting the (b) current model according to (a) the predicted hand posture J and projecting the model back to the image plane where the scene is rendered, we can obtain (c) a binary mask $\tilde{S}$. At the same time, we can acquire (d) an estimated hand segmentation mask $\hat{S}$ through our segmentation module. The hand model consists of finger-wise components and a palm component, and each has parameters of vertical and horizontal scale. We then update (e) the current model M_u to minimize the Euclidean distance $d_S = \|\hat{S} - \tilde{S}\|$ between observed and rendered hand masks.

To further enhance the stability of outputs, we take consistency into consideration and minimize the distance through a step-based iterative optimization. The initial step for updating the scale of the model is 0.2 for every 30 frames. When the model meets a plateau for successive 300 frames, we upscale/downscale the step by 50%. Since we want to achieve a more stable output, the optimization is stopped when the step size goes smaller than 0.02 to save computational power and prevent flickering effects in the implemented real-time application.

3.6.2 Model Fitting and Virtual Objects Augmentation

To cope with fine occluding edges between the user's hand and the in-hand objects, we propose a way to calculate occlusion masks through refined depth relations by comparing the reconstructed hands and objects to be augmented in a virtual environment. Existing depth-based methods suffer from problems including noise and misalignment, and their quality deteriorates when the distance from targets gets closer.

More specifically, we solve occlusions with the updated hand model by fitting it to the joints acquired through the hand posture prediction module in our joint learning framework. Our approach minimizes the fitting energy with regard to the optimized hand model. The updated hand model is displaced to minimize the distance d_j between the captured hand joints J_i and the current hand model $M_u(i)$:

$$d_j = \sqrt{\sum_{i=0}^{N} (d_J(i) - \hat{d}_{M_u}(i))^2} \tag{3.4}$$

where $d(i)$ is the normalized distance obtained by $d(i) = r_i/\sqrt{\hat{S}}$. The r_i is the distance between the feature joint $J_i(i = 0, 1, ..., N)$ and the root of the hand J_r. we fit the optimized hand model M_u back according to the acquired joint coordination J to calculate the occluding mask m, the spatial location is shown on the image plane to which the invisible part of the virtual object V corresponds.

We decide the label of each point as 'visible'or 'invisible'of V based on the comparison between the 3-dimensional displacement of V and the optimized hand M_u to determine the occluding mask m. During the rendering process, pixels of V' labeled with 'invisible'will not be rendered to represent the occlusion. Based on the acquired occlusion mask, some portions of the virtual object model will be masked invisible while other portions remain visible to the user. This process is done by frame and will remain robust even under strong motion.

3.7 Experimental Evaluation

3.7.1 Implementation Details

We implemented a complete table-top application (Figure 3.8) to showcase the idea, verify the quality of masks, and conduct a user study. This Unity3D application allows users to use their bare hands to interact with real objects augmented with virtual appearances. The frame rate was fixed at 30 fps with a resolution of 1440 by 1440 per eye using a PC with an Intel 7800X CPU and an NVIDIA RTX 2080Ti. Although a piece of video-see-through equipment (Intel RealSense SR300) is used during the experiment, our system also works with optical-see-through devices.

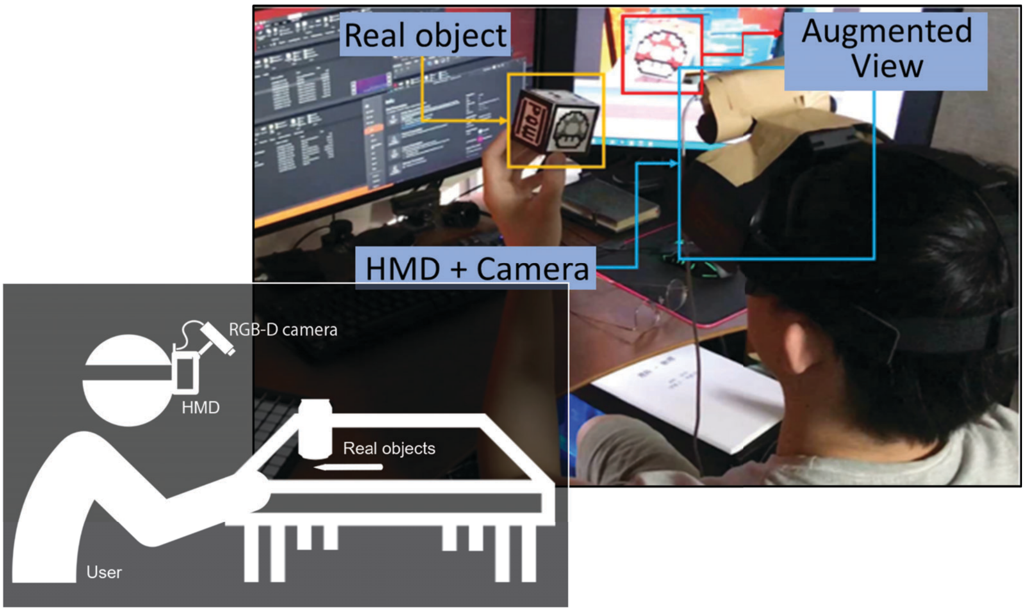


FIGURE 3.8: The configuration of the implemented application.

TABLE 3.1: A comparison between the proposed method and the previous methods

Method	Viewpoint	Scene	Mutual occlusion	Equipment
Lu [24] Depth based	Restricted	Static	No	Stereo cameras
Tian et al. [4] Contour based	**Arbitrary**	Static	No	**RGB camera**
Dong et al. [25] Depth based	Restricted	**Dynamic**	**Yes**	TOF camera
Tian et al. [5] Reconstruction based	**Arbitrary**	Static	**Yes**	RGB-D camera
Holynski et al. [1] Reconstruction based	**Arbitrary**	Static	**Yes**	**RGB camera**
Walton et al. [9] Depth based	**Arbitrary**	**Dynamic**	No	RGB-D camera
Our method	**Arbitrary**	**Dynamic**	**Yes**	RGB-D camera

* Bold texts in the table show the best capability for most applications/run most efficiently/require minimal setup among proposed approaches.

3.7.2 Experimental Results

Qualitative Results

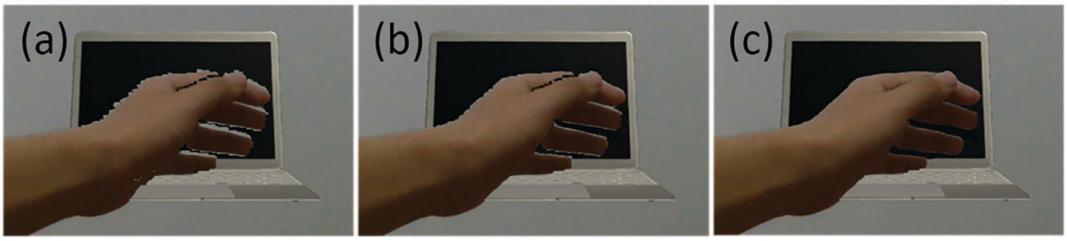


FIGURE 3.9: A mixed reality scene rendered with occlusions based on (a) naive approach that uses raw depth, (b) CVF occlusion [102] and (c) our approach.

To qualitatively verify the applicability of our proposed system when applied to hand-object interactions, we compared it to previous real-time occlusion solutions in the following five aspects. First, the system should be able to resolve the occlusion with a moving viewpoint. Restricting the viewpoint will significantly reduce the practicability. Second, the placement of in-scene objects will change constantly, and thus being able to handle dynamic scenes is critical. Moreover, additional equipment and complex implementations can limit usability. The detailed comparison is shown in Table 3.1. Our approach can handle dynamic scenes with moving objects and egocentric viewpoints in real-time with a simple setup.

We evaluate our system and verify this is a better method compared to the naive method that decides the visibility of each pixel based on the raw input of the RGBD camera, the state-of-the-art CVF occlusion approach [102]. We exclude simultaneous localization and mapping methods due to their unrealistic requirements of a stable and rigid environment in hand-object interactions. By placing virtual objects in the scene to interact with the scene geometry, we implemented a traditional mixed reality scenario of object insertion to evaluate the accuracy of the occlusion mask and rendered object. Direct results of rendered virtual objects can be observed in Figure 3.9. The readers are also referred to the supplementary video for further results.

Quantitative Results

Prediction under Occlusions To verify the effectiveness of the improved photorealistic hand-object RGBD database, our model is trained with a similar architecture to the JORNet [106] with Caffe framework. The weight of our network is initialized based on the original ResNet50 trained with ImageNet [111]. We use Percentage of Correct Keypoints (PCK) as the measure to evaluate the accuracy of our approach. After training 45,000 iterations with the same configuration based on the original SynthHands [106] and our improved database, we benchmark both approaches with the stereo tracking benchmark database [112], which consists of 12 sequences of paired RGBD images. Figure 3.10 presents the result that and our approach outperforms

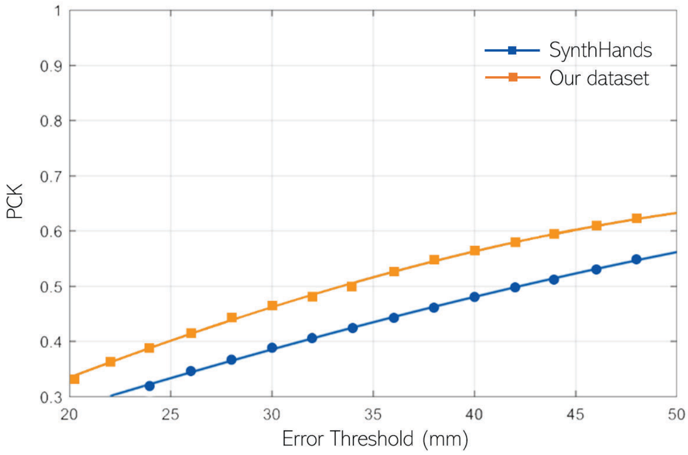


FIGURE 3.10: PCK benchmark with the Stereo database. The model trained with the improved database (orange) shows a higher prediction accuracy compared to the original approach (blue).

the original method trained with synthetic data. With a threshold set at 50mm, the accuracy is significantly improved from 0.55 to 0.63.

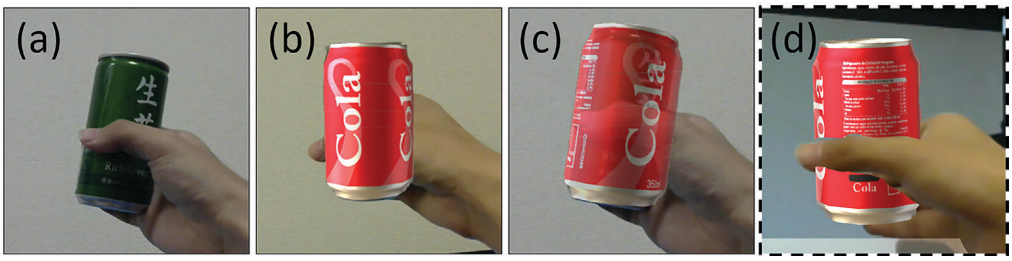


FIGURE 3.11: Results of three methods to augment a green can with a virtual Cola can: (a) the real scene without any overlay; (b) result without any occlusion handling; (c) result when applying the approach proposed by Liang et al. [113]. The transparency was adjusted to 70% according to the direction of the palm in this case; (d) result of our method.

3.7.3 Ablation Study

To validate the quality of the overlay, we mainly focus on the reprojection error in pixels of the rendered objects. Since the egocentric head-mounted display works differently from the traditional screens, the screens are positioned closer to the user and thus make the pixels easier to be identified. To evaluate the experimental results quantitatively, we multiply the factor of the pixels per degree of visual angle of the magnified headset screen with the measured length of the deviated position to obtain the reprojection error. Figure 3.12 presents an ablative analysis of the hand optimization step. With updated hand models, our combined approach shows the best performance with the lowest average reprojection errors.

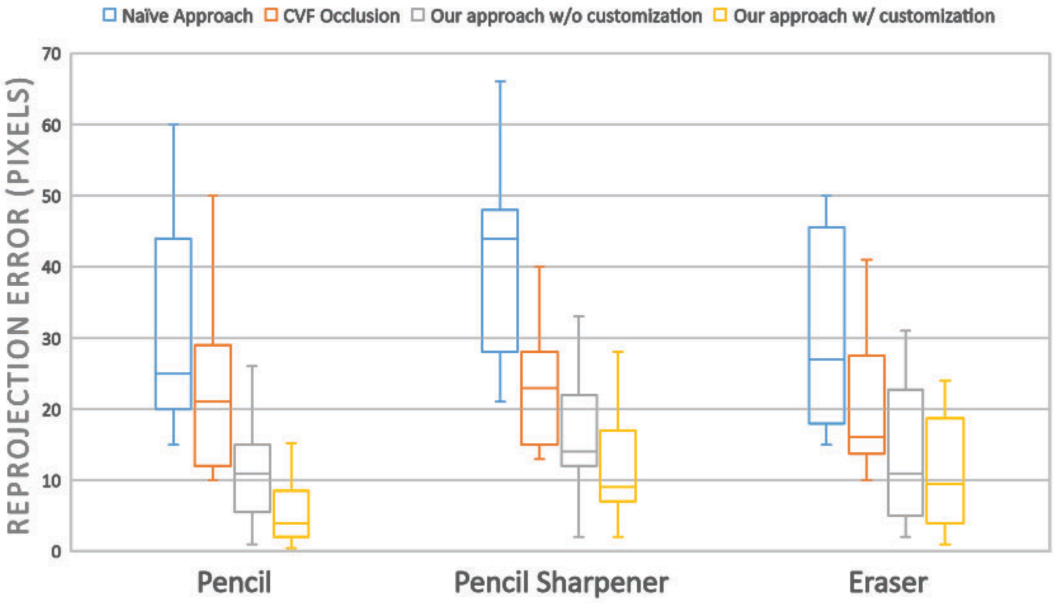


FIGURE 3.12: Reprojection errors of occlusion masks acquired by different approaches for three sequences, pencil, pencil sharpener, and eraser. While using cost volume filtering (orange) to improve the raw depth (brown) shows better accuracy, our approach (grey and blue) shows a further improvement.

While we outperform the previous approaches in the quantitative comparison (Figure 3.12), we emphasize that our approach can also handle complex scenes that the hand cannot be labeled as either foreground or background object.

3.7.4 User Study

We designed a participant-based cooperative qualitative evaluation to evaluate our application for real-object enhancement. Nine subjects (ages 18-26, average 21.7 years) without virtual reality/augmented reality experience participated in this study. The main goal of this study is to test the sense of presence, the realism of the experience, and stability, and to identify potential issues through interviews after each trial. This study compares experiences of using four different conditions (Figure 3.11): (a) without any occlusion handling; (b) with a naive approach [113] that adjusts the transparency of virtual objects based on the angle of the palm; (c) render occluded objects without the updated hand model using the proposed method and (d) render occluded objects with the updated hand model.

During the experiment with the configuration shown in Figure 3.8, each user went through 3 scenes interacting with a pencil, a box, and a card, with 4 different conditions. The sequence of trials in each scene was randomized to prevent bias. Users followed instructions to perform the simple task of interacting with objects with translating and rotating motions. After each trial, feedback was collected through a semi-structured interview.

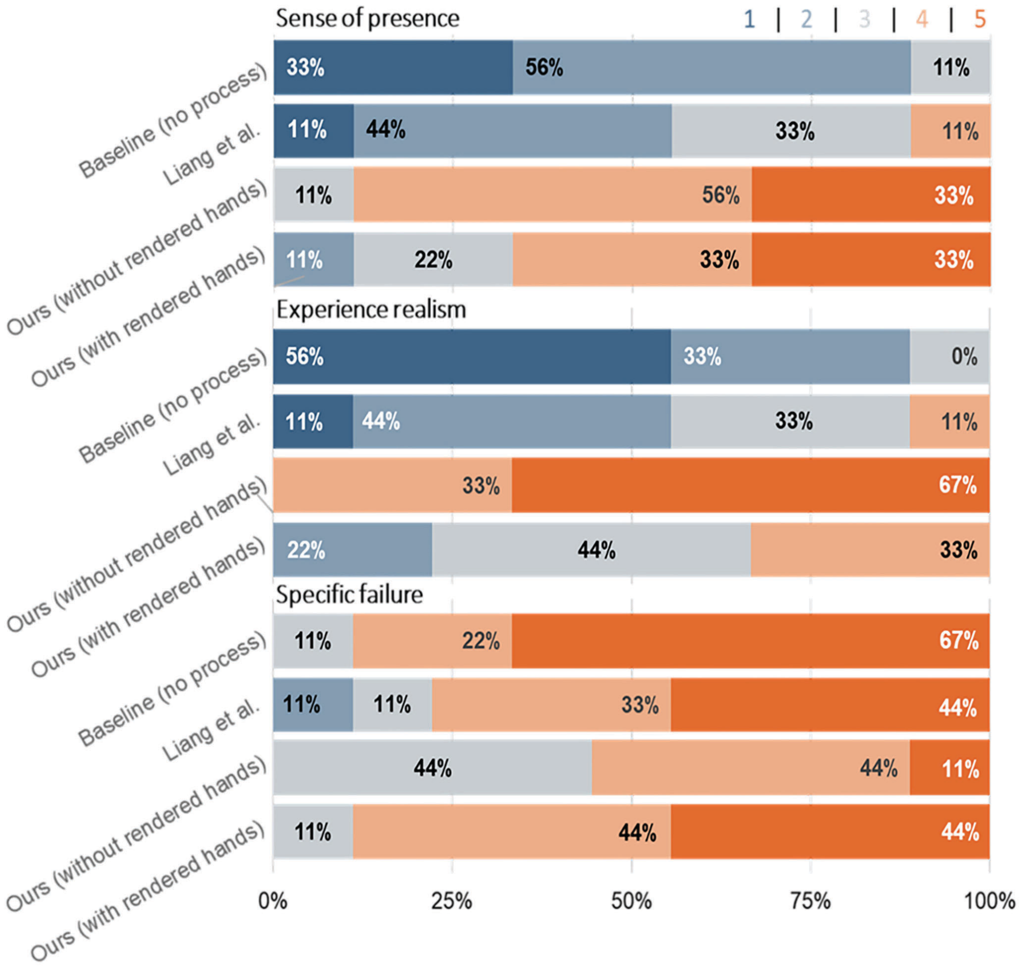


FIGURE 3.13: Likert-type survey result from the user study. The experience of each trial is rated from 1 as "Bad" to 5 as "Good" with a step of 1.

Figure 3.13 illustrates the results of the study. From both the results of the questionnaire as well as the comments in the subsequent open discussions, we confirmed a positive impression of our implementation. Since most objects are partly occluded by fingers during the interaction, and participants were actually holding realistic objects in hand, the naive solution of adjusting transparency to be fully opaque when participants flip their hands outward was highly problematic in this situation and resulted in a strange impression that virtual objects seemed to be fading away when they turned their arm. This problem can be observed in Figure 3.11. With a K-W test, we verified a more immersive mixed reality experience and a significantly more realistic feeling of interacting with virtual objects with our approach ($P <.001$ in scenes 1 and 3, $P = .022$ in scene 2). Some users reported that rendering model hands mitigates some latency and resulted in synchronization, thus giving them a more consistent feeling.

3.8 Conclusion of the Chapter

In this chapter, we have presented a real-time method to handle the hand-object occlusions in mixed reality. We propose a photo-realistic RGBD hand-object database with precise hand postures and semantic segmentation annotations to facilitate our occlusion-aware joint learning system. With a novel real-time optimization pipeline, we utilize the jointly predicted postures and segmentation to calculate occlusion masks and render objects with correct occlusions. The experimental results show better quantitative and qualitative performance than previous literature, and a user study verifies a more realistic mixed reality experience of hand-object interactions. The implementation shows good accuracy, robustness, and speed with the potential to be further adapted to other applications.

Since we are using a commercial implementation of object augmentation this time, there is a technical issue of misalignment when localizing the optimized hand model. We believe a re-implementation can solve this problem. As a general limitation of learning-based approaches, greatly changing the appearance of hands such as wearing gloves may reduce the robustness. In addition, there is no sophisticated occlusion-aware object tracking in the current implementation and this leads to losing augmentation of the object during experiments due to strong occlusions. Joint tracking of hand and object is a promising direction for future improvements.

Chapter 4

Observing the Regional: Foreground-aware 360° Depth Prediction

In this chapter, we extend the scope to a larger spatial scale: understanding the foreground objects, by employing scene understanding with data augmentation in mixed reality. Instead of local hand-object interactions, we investigate contextual information that requires better comprehension of omnidirectional images that are prevalent and essential for mixed reality applications.

We start with a novel data augmentation pipeline that generates a large-scale database with accurate and quality pairs of color and depth information. By repurposing existing perspective databases, the proposed database is the first to provide photo-realistic representations of foreground objects. It facilitates the following learning-based supervised algorithm of depth prediction and semantic segmentation. In the second half of this chapter, we propose two different designs to achieve the goal of outputting depth maps for omnidirectional images with accurate foreground predictions. The first uni-projection-based design [114] successfully verified the effectiveness of both the proposed database and a novel auxiliary network, MaskNet, while the second bi-projection-based architecture [79] further improved the accuracy by utilizing the equirectangular and cubemap projections at the same time.

4.1 Introduction

With commercial 360° cameras becoming widely available and highly efficient to capture surrounding environments with high fidelity, omnidirectional content has gained great popularity in education, entertainment, etc. As a result, the need for better visual reasoning algorithms in the context of omnidirectional media rises accordingly. One of the most important visual reasoning capabilities is to predict depth information from a single-color image as it provides structural clues of the surroundings, and thus facilitates a wide range of applications including navigation in robotics [115], stereoscopic rendering in graphics [75], augmenting virtual objects [116].

Recent advances in deep learning have even extended the capability from the domain of traditional 2-dimensional content to omnidirectional content [117]. However, existing omnidirectional approaches produce sub-optimal estimations on real-world scenarios due to their lack of consideration of dynamic foreground objects. Since obtaining omnidirectional RGBD data with dynamic foreground objects is a more challenging problem compared to traditional perspective data, previous researchers resort to different methods to synthesize high-quality paired omnidirectional color and depth samples. For captured-based approaches, using a stereo setup of two 360° cameras will inevitably include the other camera in the captured data [118]. While recent 360°-capable scanning devices [119] can acquire paired RGB and ground truth depth of scenes with improved quality, they are incapable of including any dynamic object as a result of scanning and stitching scheme (Figure 4.1) [120]. For synthesis-based approaches [121], although researchers attempt to solve this problem by inserting 3-dimensional models into the scene to improve the prediction (Figure 4.2), it is challenging to efficiently generate highly-realistic virtual foreground objects that resemble real-world ones [122], and non-photo-realistic data often lead to undesirable and inaccurate outputs.

In this chapter, we tackle the problem of foreground by first augmenting databases with realistic foreground representations. We observe that given the same object with a determined distance, its scale in spherical images should remain consistent. Taking advantage of it, we effectively composite color data of abundant and easily obtainable 2-dimensional databases and rendered omnidirectional images according to ground truth depth maps to ensure correct occlusion representations. To preserve correct distortions in equirectangular images, we project the data to cube maps before and after compositions.

We then propose two different network designs to effectively learn the depth estimation for foreground objects from a monocular omnidirectional image. The first one is a novel auxiliary deep neural network that estimates both the mask of the foreground objects and regresses the depth of the omnidirectional images. With the depth and segmentation estimations, we propose a new local depth loss of dynamic foreground objects to achieve more consistent depth predictions. This solves the problem that small areas with steep local gradients often got minimized when regressing

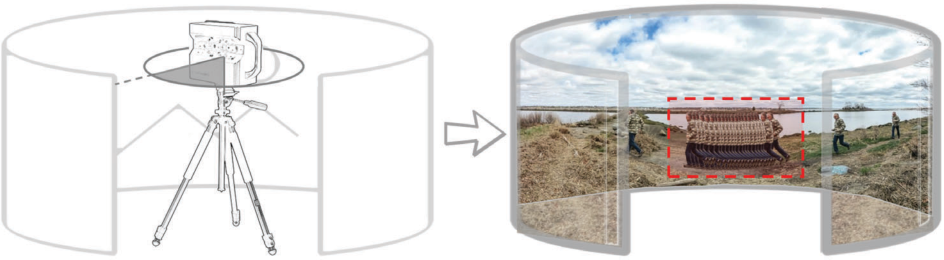


FIGURE 4.1: A demonstration of incorrect representations of dynamic objects (e.g. a running person) captured with an omnidirectional RGBD scanning device.

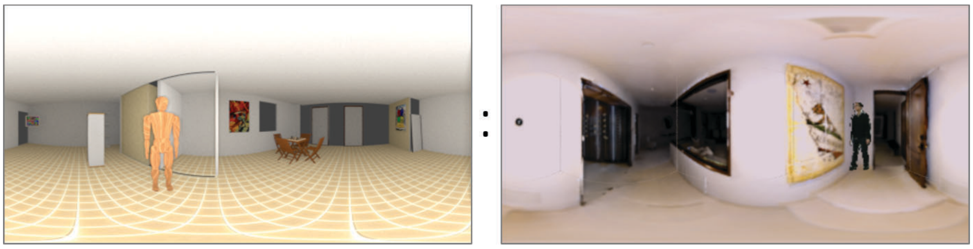


Figure 4.2: The previous approach of inserting human models introduces the problem of severe domain bias. This is demonstrated by comparing synthetic data (left) with captured data (right).

the global gradient of the prediction, resulting in areas of interest that are frequently smoothed out in existing work.

The second network is proposed to further improve upon the first work. It obtains accurate and sharp foreground depth prediction with consistent global predictions by exploring a bi-projection algorithm that consists of an equirectangular projection that predicts global depth information and a cubemap projection that simultaneously estimates the depth and the semantic segmentation of cube faces. While the equirectangular projection ensures a consistent and smooth global context, the cubemap faces provide insights regarding local details with a smaller FOV. By merging two projections together, we achieve better depth prediction for omnidirectional images with foreground objects.

During our experiments, we choose humans as the dynamic foreground object to show the efficacy of our approach. As a foreground object, human shares both a high complexity in deformation and non-uniform depths, and great importance being one of the most interested and common subjects to deliver the context of the image. By showcasing accurate estimations of humans, we demonstrate the ability of our method to be generalized to other foreground objects.

Experimental results show that both proposed methods yield more consistent global estimations and more accurate local estimations against contemporary state-of-the-art models quantitatively and qualitatively. Moreover, the bi-projection-based network provides more accurate results when compared to the first uni-projection-based approach, verifying the effectiveness of an improved design. This research is best applied in fields including occlusion-aware augment reality and stereoscopic rendering.

The technical contributions of this chapter are summarized as follows:

1. We propose a method to synthesize an RGBD omnidirectional database with dynamic foreground objects to tackle the challenge of estimating the depth of them in the context of spherical images. The database is offered to promote future research.

2. We first employ the proposed auxiliary network that estimates depth and segmentation masks to calculate a new local depth loss of dynamic foreground objects. This can resolve the issue of steep local gradients getting smoothed out

during optimization and improve the estimation results of local regions. The source code is publicly offered online.

3. We further propose a bi-projection-based network that can more effectively predict the depth of global input and the foreground objects. By concatenating learned depth representations from both equirectangular projection and cubemap projection, this foreground-aware design shows the superior performance when compared to the state-of-the-art methods.

The rest of the chapter is organized as follows: we revisit learning-based monocular depth estimation methods and methods for synthesizing training data in Chapter 4.2. In Chapter 4.3, we explain the novelty of our database and describe the generation framework. In Chapter 4.4, we first describe the uni-projection-based network architecture and the proposed loss function to leverage the database. Details of experiments are presented along with qualitative and quantitative evaluations. In Chapter 4.5, we explain the improved bi-projection-based network design and its implementation details. We then present benchmarks against the state-of-the-art methods and the first design to highlight the efficacy of our methods.

4.2 Related Work

4.2.1 Learning-based Depth Prediction

Estimating the depth given a monocular RGB image is one of the most fundamental capabilities in understanding the 3-dimensional geometry of the scene [123]. A wide range of applications in robotics, graphics, virtual reality, etc. can benefit from more accurate depth predictions. Owing to more established machine learning algorithms, learning an implicit relation between color and depth has seen significant progress recently.

A variety of algorithms [72] [73] have been proposed by training a model with collected color and ground truth depth images in a supervised fashion. Lately, numerous strategies have been proposed to achieve a more coherent and accurate monocular depth estimation. Multi-scale networks [124] make coarse global depth predictions and refine the local prediction. Multitask learning [83] [125] with multiple regression and classification objectives is also prevalent in understanding scene geometry and semantics due to their complementarity. A fully convolutional network architecture [73] that endows novel up-sampling blocks achieved impressive accuracy and efficiency.

Unsupervised methods focused on a stereo correspondence framework to cope with the need for an expensive secondary supervisory signal. This is either accomplished by synthesizing stereo-views with left/right consistency [75] to produce intermediary disparity map [126] [127], or multi-view consistency with structure-from-motion [76] to learn a dense disparity prediction.

To yield accurate estimations of both global and local objects in the context of the omnidirectional domain, lacking paired data with dynamic foreground objects and distortion introduced by equirectangular projection will result in poor outputs for supervised approaches. On the other hand, while some unsupervised approaches do not explicitly require paired databases, issues like distortions and occlusions still persist.

Therefore, predicting the depth of omnidirectional contents with the aforementioned 2-dimensional approaches often yields sub-optimal results [117]. Failing to learn feature representations in the equirectangular domain inevitably leads to inferior accuracy and coherency. To improve the performance of prediction in 360° contents, cubemap projection is one of the most popular choices. By projecting spherical signals onto the faces of a cube, six non-distorted square patches can still be processed with existing convolution techniques. Since projecting spherical contents onto six faces of a cube can eliminate distortion for each face to a great extent, it is made possible to adopt perspective-based methods with minimal effort. Moreover, as each face has a reduced FOV, cubemap projection puts more focus on local objects compared to equirectangular projection [74]. However, since each face is processed independently, the discontinuity along edges is problematic for many applications. A common approach to alleviate this problem is through padding edges during the process [128] of merging predictions back to a single output. However, while such an issue may not be critical in certain tasks such as stylization and classification, the lack of consistency between the output of each patch is more pronounced in depth regression. Recently, methods for enabling rotation-equivariance in CNNs were proposed by Cohen [118]. However, since such equivariant architectures provide a lower network capacity, only single variable regression problems were demonstrated. Inspired by [129], the state-of-the-art method [117] incorporated distorted CNN filters to improve the performance of fully convolutional networks with skip connections and showed impressive predictions of equirectangular images. However, without any consideration of foreground objects, the network will penalize small areas with a steep local gradient when regressing the global gradient of the prediction, resulting in areas of interest such as humans being frequently missing in the output. In our second improved network design, we try to incorporate both equirectangular and cubemap projection to complement each other, so that while the equirectangular prediction can provide a global context, the proposed network can still yield accurate results for foreground objects.

4.2.2 Databases for 360° Images

Since the standard method to approach monocular depth estimation is to train a model directly from paired RGB images and ground truth depth, the performance of such supervised approaches cannot produce better results than the limits of its training data. With advanced imaging devices and depth sensors, high-quality databases consisting of traditional perspective images are easily obtainable, for instance, KITTI [62], NYUv2 [85], Make3D [70], etc. However, obtaining paired 360° data is not as straightforward as using traditional imaging devices with calibrated color and depth sensors such as Kinect to capture 2-dimensional contents. Using a stereo setup of

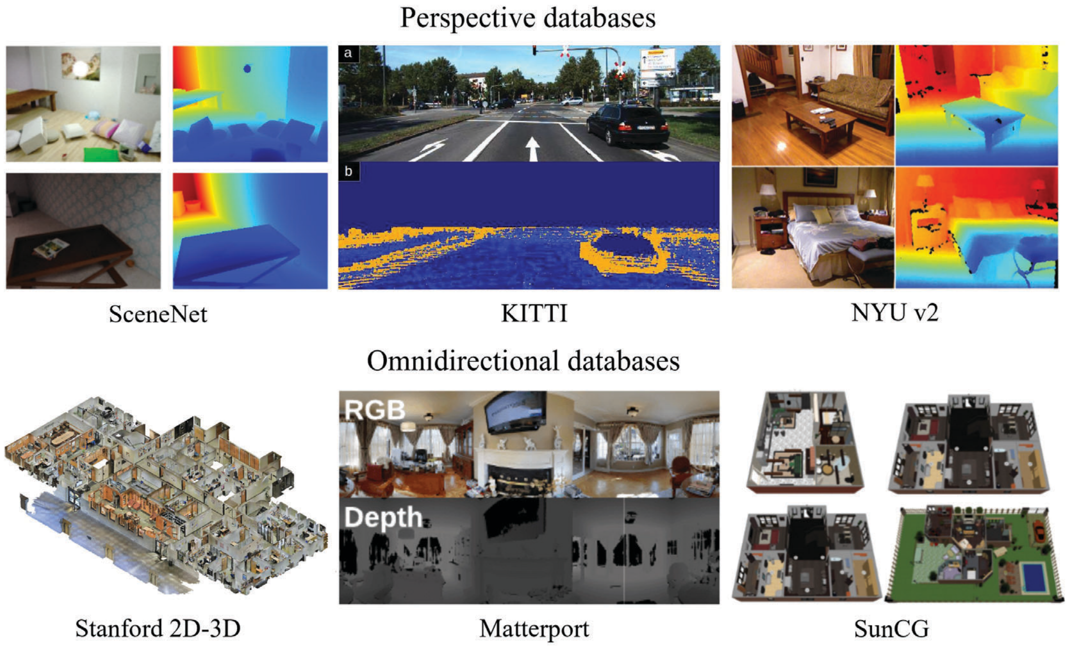


FIGURE 4.3: The most used databases for perspective and omnidirectional learning-based depth prediction.

two 360° cameras to calculate disparity is challenging due to the presence of occluded regions [130]. In the case of omnidirectional images, both cameras will inevitably include the other camera in the captured data. Recent scanning devices are capable of acquiring databases that consist of paired 360° RGB and ground truth depth of static scenes with improved quality, such as Stanford 2D-3D [120] and Matterport3D [119].

However, existing methods fail to include any dynamic object in the scene. As a result of a scanning and stitching scheme, trying to include dynamic foreground objects in the captured data [120] [119] will lead to distorted and incorrectly composited images, as shown in Figure 4.1. [117] repurposed 3-dimensional model databases, SunCG [121] and SceneNet [131], to render 360° synthesis-based RGBD images with virtual cameras. However, a model trained with synthetic data does not necessarily generalize well to real-world scenarios, due to database bias. As observable in Figure 4.2, a previous attempt to resolve this issue by inserting human models into the existing synthetic database suffers from a severe domain bias from the real-world scenarios. However, because of the inability to include realistic human representation in the existing omnidirectional RGBD database, the performance of all previous methods is greatly limited when applied to real-world scenarios with humans.

4.3 Foreground-aware Data Augmentation

To produce a foreground-aware photo-realistic database for machine learning algorithms, we explain our method of augmenting databases with realistic foreground objects using an image-based approach in this section. The pipeline of our method is visualized in Figure 4.5. As shown in Figure 4.5, based on the observation that 360° images can circumvent challenges brought by perspective transformations in the traditional 2-dimensional plane, we effectively composite color data of abundant and easily obtainable 2-dimensional databases and rendered omnidirectional images with z-buffer. We employ a Mask R-CNN network to predict pixel-perfect masks of the dynamic foreground objects. With the acquired masks of interest, we can obtain perspective paired color and depth batches. With cubemap projections done before and after compositions, we can composite with correct occlusions and distortions.

4.3.1 Scale-invariant Correspondence for 360° Images

In this section, we explain the novelty and feasibility of compositing existing 2-dimensional RGBD databases onto equirectangular images.

We observe that it is difficult to establish a correspondence between color and depth in the traditional 2-dimensional domain. We take perspective transformations as an example and demonstrate them with Figure 4.4. During the process of "zooming in" onto the target region (dashed box), the global color data changes continuously while the depth of the target area stays the same, forming a many-to-one mapping. It is particularly true in the real world: when we use binoculars to observe the same object, even given the prior knowledge of an object's average size, it is inherently harder to estimate the distance without knowing the magnification.

On the other hand, the relation between color and depth in 360° images is scale-invariant. While some perspective transformations such as cropping will make 360° images no longer spherical, rotation and zoom will not affect the global color representation of the original image after down-scaling. Therefore, given the same object with a determined distance, the appearance of the target region in 360° images should remain consistent.

Based on this observation, we exploit such an advantage of omnidirectional images by inversely compositing local regions onto them with regard to depth information. In this work, we choose z-buffer to composite owing to its simple implementation, high efficiency, and compatibility of occlusions.

4.3.2 Synthesizing RGBD Foregrounds

General Foreground Synthesis

To automatically acquire paired color and depth maps of a dynamic foreground object, we can either capture with sophisticated RGBD sensors or take advantage of abundant and easily obtainable existing databases in the traditional 2-dimensional domain. In order to efficiently acquire highly accurate segmentation masks of the input data, we adopt a Mask R-CNN model with a backbone of ResNet-101, trained with the COCO database to predict per-pixel label masks. The strengths of per-instance prediction and less complex post-processing are the main reason we choose Mask R-CNN over a

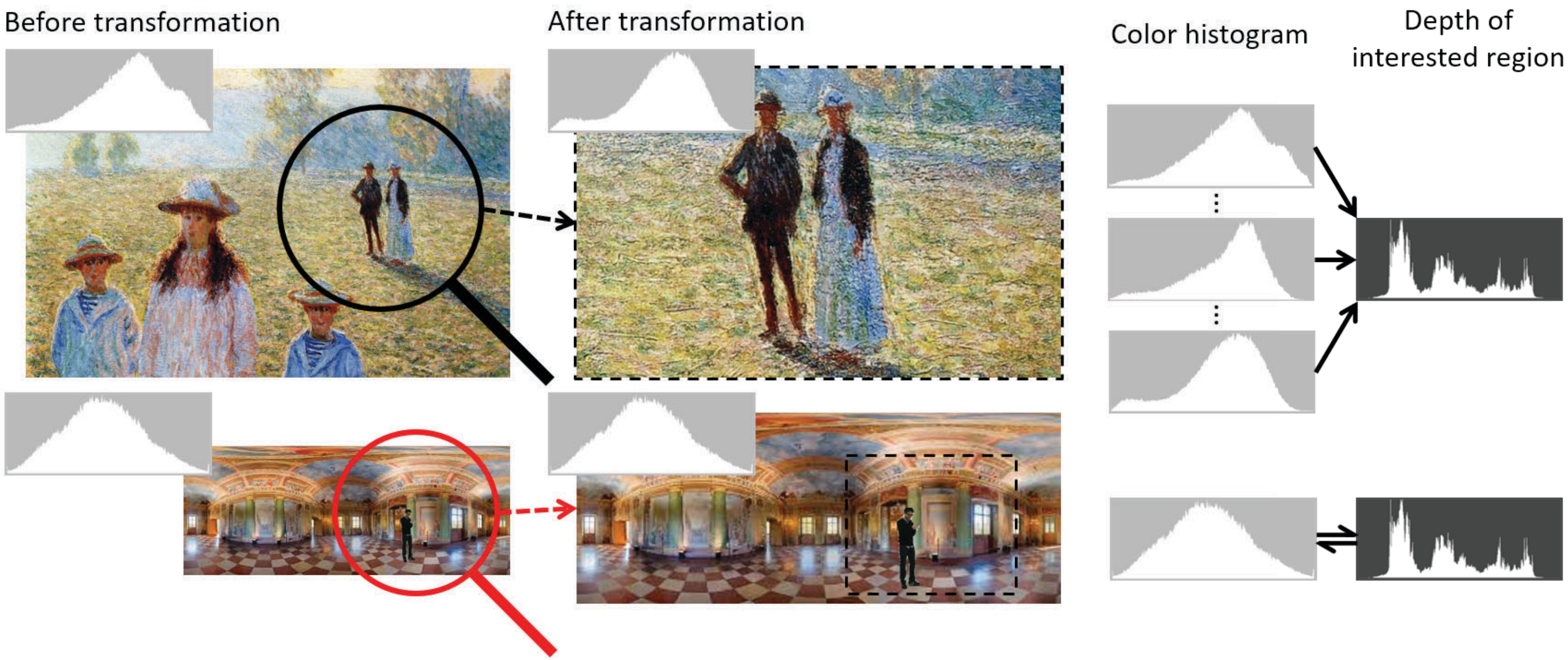


FIGURE 4.4: Since traditional 2-dimensional images can be processed with perspective transformations, it is difficult to establish a correspondence between color and depth information. In comparison, 360° images cannot be cropped or zoomed and hence have a scale-invariant RGBD correspondence. As shown on the right side, in the 2-dimensional plane, different color representations map to the same depth map of interested regions. One the other hand, 360° images share a one-to-one mapping between color and depth, and every object has a fixed scale.

simpler U-Net network. During prediction, our implementation predicts per-person-instance masks at a near-real-time speed (5 fps) with high accuracy. Some examples are shown in Figure 4.6. With acquired masks for areas of interest, we crop batches from the input RGBD data accordingly.

Human Batch Synthesis

Since human as a dynamic foreground object shares both a high complexity in deformation and non-uniform depths, we choose humans to show the efficacy of our approach. At the same time, humans have great importance in being one of the most interested and common subjects to deliver the context of the image. By showcasing accurate estimations of humans, we demonstrate the ability of our method to be generalized to other foreground objects. In this work, we repurpose the PKU-MMD database [132], which contains calibrated and synchronized RGBD video sequences. This large-scale database includes motions of 51 categories performed by 66 distinct subjects. It contains different views, sufficient intra-class variations, and adequate classes of motions to ensure a robust prediction result.

4.3.3 Augmenting 360° Databases

360° Background Synthesis

Since paired real-world 360° RGBD databases with humans are not available to our knowledge, to alleviate the difficulty of evaluating the accuracy between our and the-state-of-the-art approaches, we use a similar strategy matching with [117] to render paired and realistic omnidirectional RGBD images from the Stanford 2D-3D database and the Matterport3D database captured with professional 360°-capable scanning devices. Specifically, a path tracing renderer with a virtual omnidirectional camera is used to generate the samples. The light source is positioned identically with the virtual camera. Omnidirectional depth maps with linear distances of each pixel are generated with Z depth. To show the effectiveness of our method across different domains and to benchmark the accuracy with synthetic 360° databases, identical processes are brought out with the SunCG [129] and the SceneNet [131] as well.

Compositing Foregrounds and Backgrounds

Since the RGBD local batches are captured in the traditional 2-dimensional domain, a direct composition will lead to distorted and unrealistic appearances in the 360° context. To cope with this challenge, both the RGB and depth map of each rendered omnidirectional sample are projected onto a cube map through cubic projection. With ground truth depth information of both foreground batches and background faces, the composition is done through a highly efficient and effective Z-buffer, preserving correct depth annotations and in-scene occlusions. To simulate real-world scenarios, batches are randomly composited to lower halves of 4 surrounding cube faces, while faces of the ceiling and the floor are not used during composition. Finally, a reverse

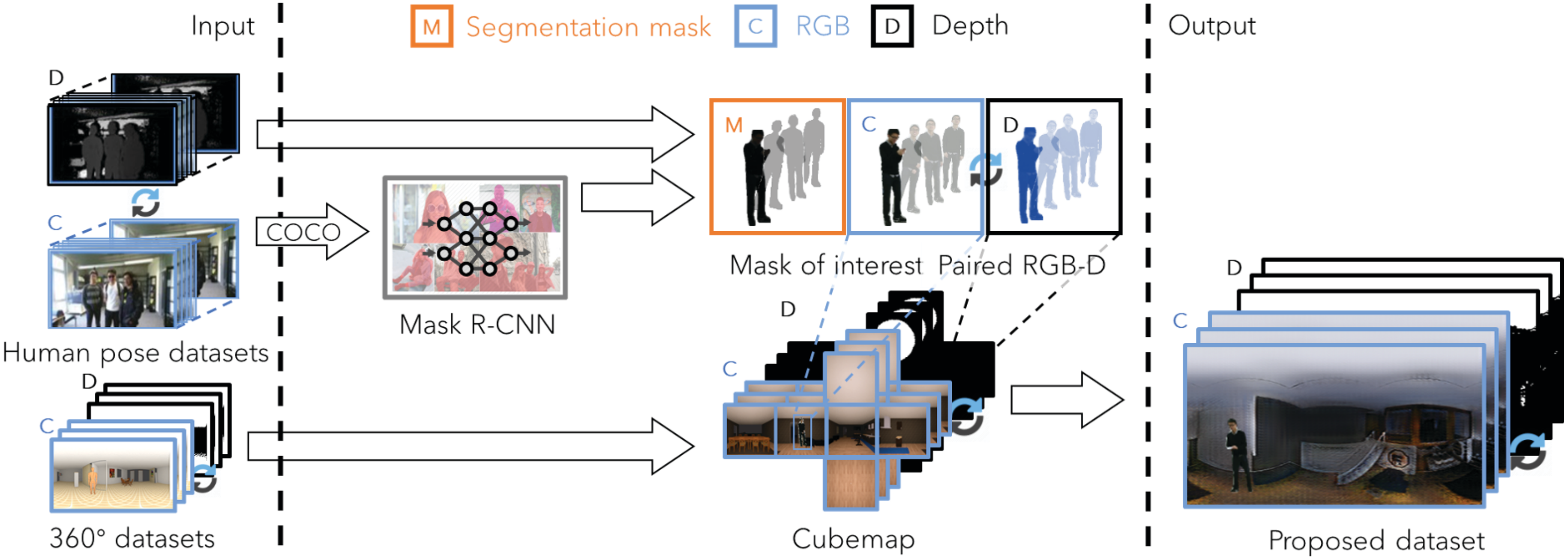


FIGURE 4.5: The pipeline of the proposed data synthesizing system. The left section shows the input databases, the middle section shows the intermediate results, and the right section shows the output. We generate masks of the interested region with mask R-CNN and corresponding RGBD batches from the input 2-dimensional database. The batches are then composited to the input 360° database with regard to the depth information.

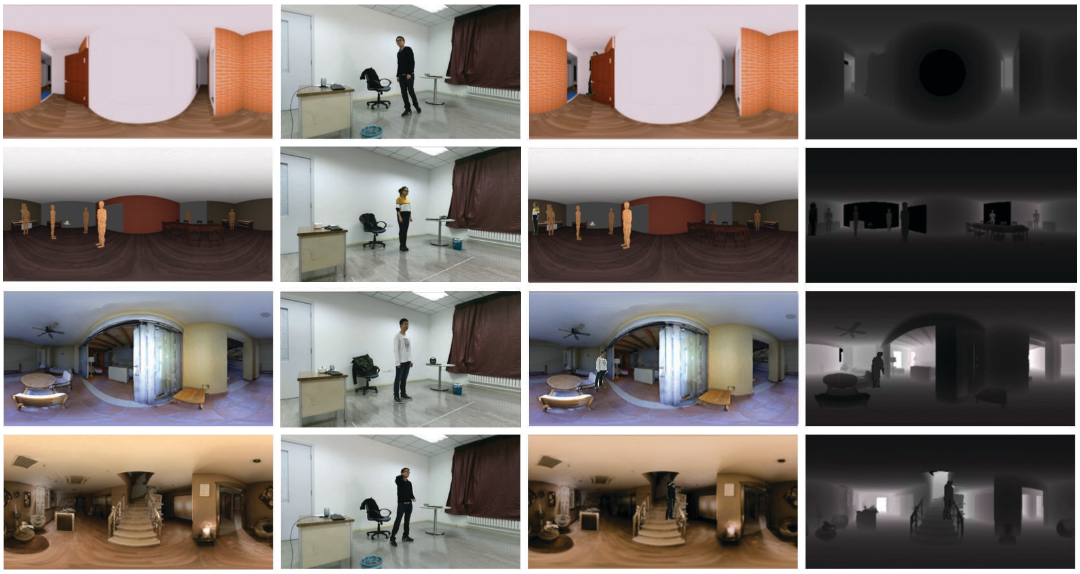


FIGURE 4.6: Generated examples with the proposed method. From left to right: rendered color images with original omnidirectional databases, samples from an input human pose database, generated omnidirectional images with humans, and corresponding depth maps.

cubemap projection is done to generate high-quality RGBD equirectangular samples with dynamic foreground objects.

In this work, our proposed database consists of 25,000 realistic and 25,000 synthetic equirectangular samples with synchronized color information and depth annotations. Abundant variation is achieved through a sufficiently wide range of indoor scenes as backgrounds, and a large human batch pool acquired in the previous step as foregrounds.

4.4 Multi-output Regression for Foreground-aware Depth Estimation

This section presents our proposed end-to-end learning model to estimate a depth map from an equirectangular image. As shown in Figure 4.7, We use two fully-convolutional encoder-decoder structured networks, RectNet and MaskNet to regress depth and predict masks of local regions respectively from a given RGB input. The RectNet that resembles the design in the literature can regress depth with changing filters in an omnidirectional context. To take advantage of the generated database with dynamic foreground objects, we leverage the generated masks of interesting areas to train the auxiliary MaskNet. By calculating both local depth loss and global loss, our network further improves the consistency of local predictions.

4.4.1 Network Structure

The proposed network approaches dense depth estimation from monocular RGB images and shares an encoder-decoder design that progressively downscales and upscales

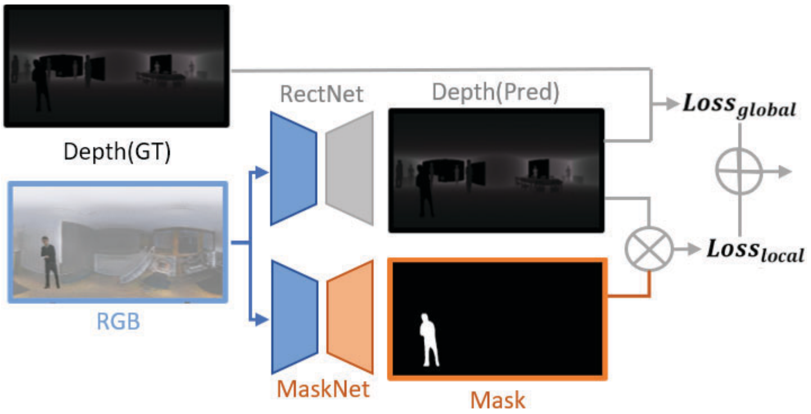


FIGURE 4.7: An overview of the proposed depth estimation network. The weight of the auxiliary MaskNet is fixed when training the depth estimation model RectNet [117].

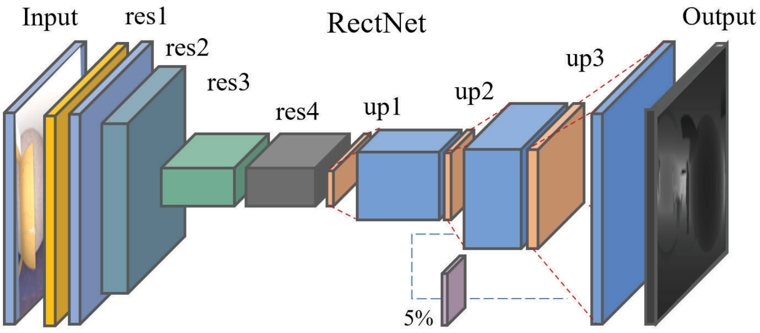


FIGURE 4.8: The architecture of the fully convolutional RectNet for depth regression. the encoder consists of two preprocessing blocks (yellow and blue) and a downscaling block (teal), followed by two increasing dilation blocks (green and grey), and the decoder contains three up-prediction blocks, followed by a prediction layer. With 360° degree color images in equirectangular format as the input, it predicts the corresponding depth map.

to the target representation through regression. Skip connections similar to ResNet structures can help to preserve the information from a higher level during regression while preventing vanishing gradient. When applied to equirectangular images, inspired by [129], we incorporate rectangular filters with changing sizes according to rows of the input to cope with the characteristic that the density of information, or namely the distortion level changes along the vertical axis but invariant along the horizontal axis. In addition to L2 depth loss to regress the prediction, a neighborhood smoothness regularization term [117] is also calculated to improve the global consistency of the output.

However, small regions with steep gradient changes usually got smoothed out during the regression and missing in the prediction. This can be observed in Figure 4.11. Predictions of humans severely suffer from this issue. To tackle this limitation, we introduce an auxiliary network, MaskNet, to calculate the local depth loss of humans. The MaskNet network that predicts masks of foreground objects from equirectangular

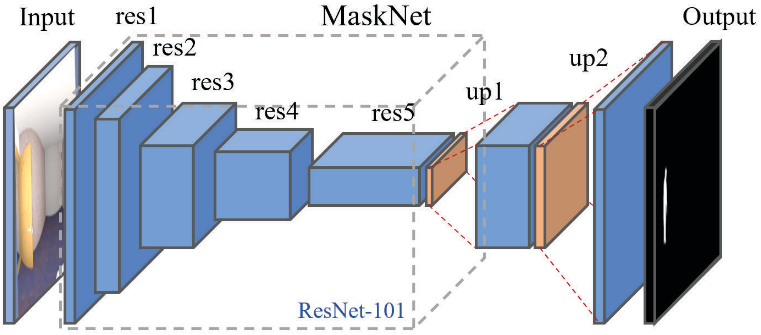


Figure 4.9: The architecture of the fully convolutional auxiliary MaskNet. The encoder of our network shares the same structure of ResNet-101 [133], followed by a decoding process with two upsampling layers to predict the mask of the target object.

RGB inputs has the architecture shown in Figure 4.9. It is trained with the COCO database and finetuned with generated equirectangular RGB images with foreground objects and corresponding segmentation masks to minimize a cross-entropy loss. The weight is fixed during training the depth estimation model.

4.4.2 Loss Functions

We train the depth estimating network in a completely supervised fashion with the input of the generated foreground-aware RGBD database. To address the problem of vanishing local gradients for areas of interest while keeping the desirable properties of the original RectNet like consistent global predictions, the total loss of our model consists of three different terms:

$$L_{total} = \sum_i (\alpha_i L_{depth} + \beta_i L_{smooth} + \gamma L_{local}),$$

while the α, β and γ are the weights for each loss term. Since the loss is calculated under different scales i, the estimations of lower scales are interpolated with nearest neighbors are concatenated together to form the final output. The depth loss L_{depth} is regressed by minimizing the least square errors between the groudtruth depth maps D_{gt} and the predicted depth maps D_{pred}:

$$L_{depth} = D_{gt} - D_{pred}{}^2.$$

The smoothness loss is calculated by $\nabla D_{pred}{}^2$ to minimize the gradient of the prediction. In order to calculate the local depth loss, we pass the equirectangular color image C_{input} through the trained auxiliary network $\mathbf{M}$ to obtain the mask of human instances $M_{human} = \mathbf{M}(C_{input})$, so we can calculate the local depth loss with

$$L_{local} = D_{pred} \otimes M_{human}{}^2.$$

By minimizing the local depth loss, we can ensure that spatially closer pixels within the same area of interest would have closer depth values.

4.4.3 Experimental Evaluation

In this section, we first evaluate our data augmentation method by presenting quantitative comparisons between models trained with existing omnidirectional databases and our generated databases. We then verify the performance of the proposed network by comparing it to the state-of-the-art omnidirectional depth estimation algorithm. Finally, to evaluate the effectiveness of our method in real-world scenarios with human objects, we offer comparative qualitative results of estimating unseen images by different methods.

Training Details

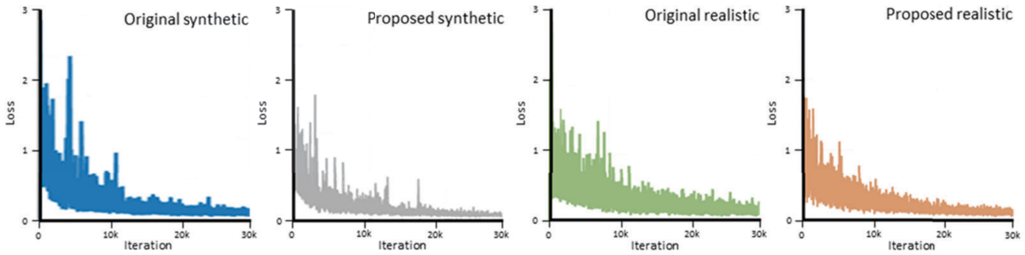


Figure 4.10: Learning curves of models respectively trained with original synthetic, original realistic, proposed synthetic and proposed realistic databases.

For fair comparisons, we randomly acquired 25,000 samples from existing synthetic omnidirectional databases to train models as the existing synthetic database, and then we acquired 25,000 samples from existing realistic omnidirectional databases to train models as the existing realistic database. We respectively generate 25,000 synthetic samples and realistic samples augmented with human objects to train models as our proposed databases. Each 512 x 256 sample has color information and corresponding ground truth depth annotation. We randomly split samples from each database into training and validation databases with a ratio of 80% and 20%. All networks in this paper are implemented with PyTorch [134] on an Nvidia RTX 2080Ti graphic card and trained with Adam optimizer [135], Xavier initialization [136], and a learning rate of 2e-4. Training parameters of our networks are $[\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma] = [0.482, 0.245, 0.121, 0.061, 0.090]$, while parameters of training previous RectNet models are $[\alpha_1, \alpha_2, \beta_1, \beta_2] = [0.535, 0.272, 0.134, 0.068]$. The same quantitative metrics from the literature [75] [117] are used for evaluation. During experiments, predicting a single image approximately costs 100 ms with the same setup.

TABLE 4.1: Quantitative results of different training databases. Error metrics are calculated on a global basis.

database	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Synthetic	0.4918	0.4133	0.8944	0.6550	0.4083	0.6806	0.8212
Proposed Synthetic	**0.3789**	**0.2893**	**0.6878**	**0.5225**	**0.4245**	**0.7926**	**0.9257**
Realistic	0.3765	0.3540	0.8864	0.5230	0.5907	0.7500	0.8926
Proposed Realistic	**0.3190**	**0.2180**	**0.5993**	**0.4788**	**0.6988**	**0.8454**	**0.9150**

For four error metrics, absolute relative difference (Abs Rel), squared relative difference (Sq Rel), root mean square error (RMSE) and RMSE log, lower values are better. For percentage of inliers under threshold $\delta < 1.25$, $\delta < 1.25^2$ and $\delta < 1.25^3$, higher values are better. Same for tables below.

TABLE 4.2: Quantitative evaluation against other models. Error metrics are calculated on a global basis.

Model	Training Set	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
RectNet [117]	Proposed Syn	0.3789	0.2893	0.6878	0.5225	0.4245	0.7926	0.9257
Proposed	Proposed Syn	**0.2895**	**0.2354**	**0.5957**	**0.4272**	**0.7440**	**0.8805**	**0.9284**
RectNet [117]	Proposed Real	0.3190	0.2180	0.5993	0.4788	0.6988	0.8454	0.9150
Proposed	Proposed Real	**0.1984**	**0.0817**	**0.3286**	**0.2608**	**0.7298**	**0.8984**	**0.9727**

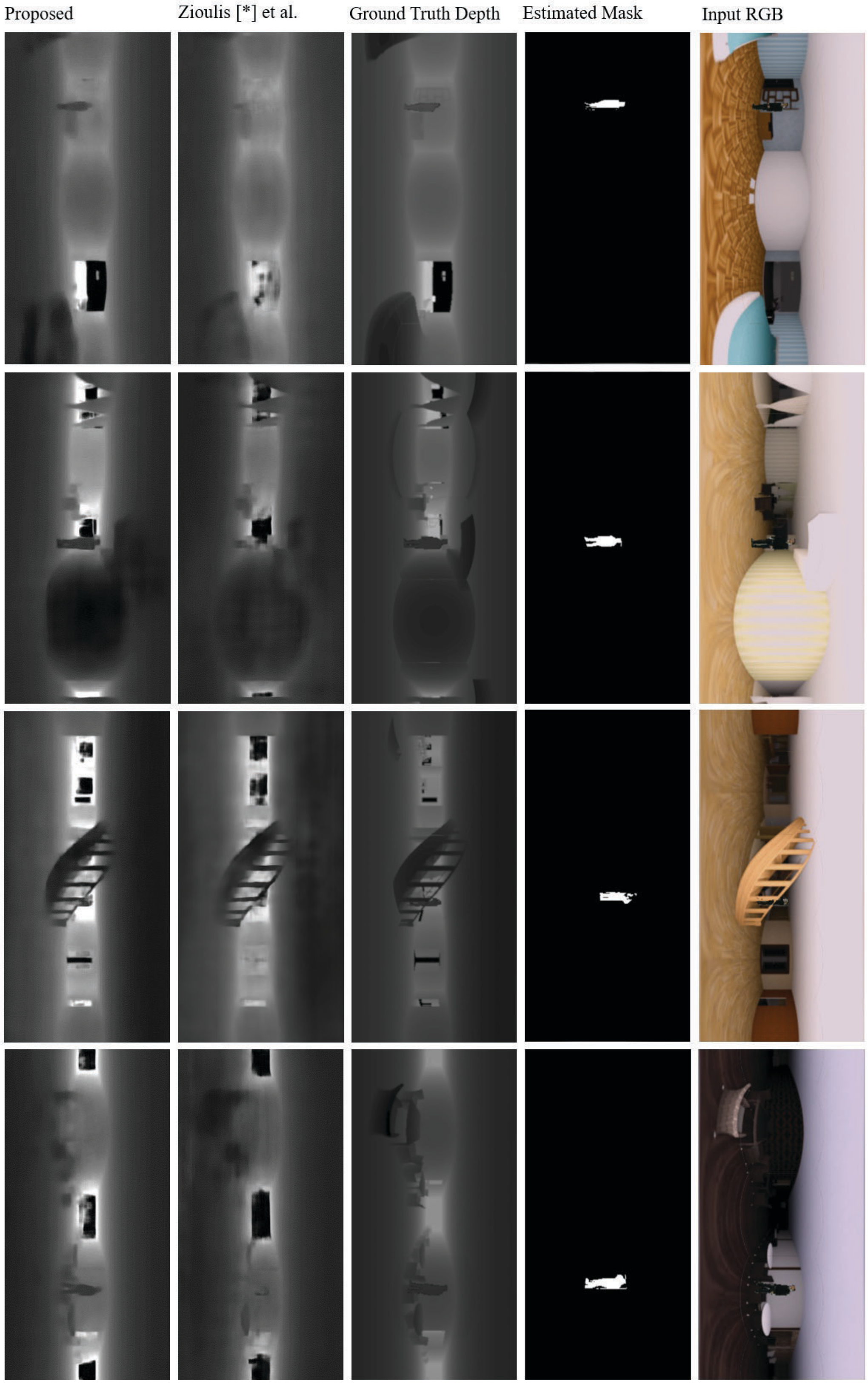


Figure 4.11: Qualitative comparison between each model when tested on synthetic images.

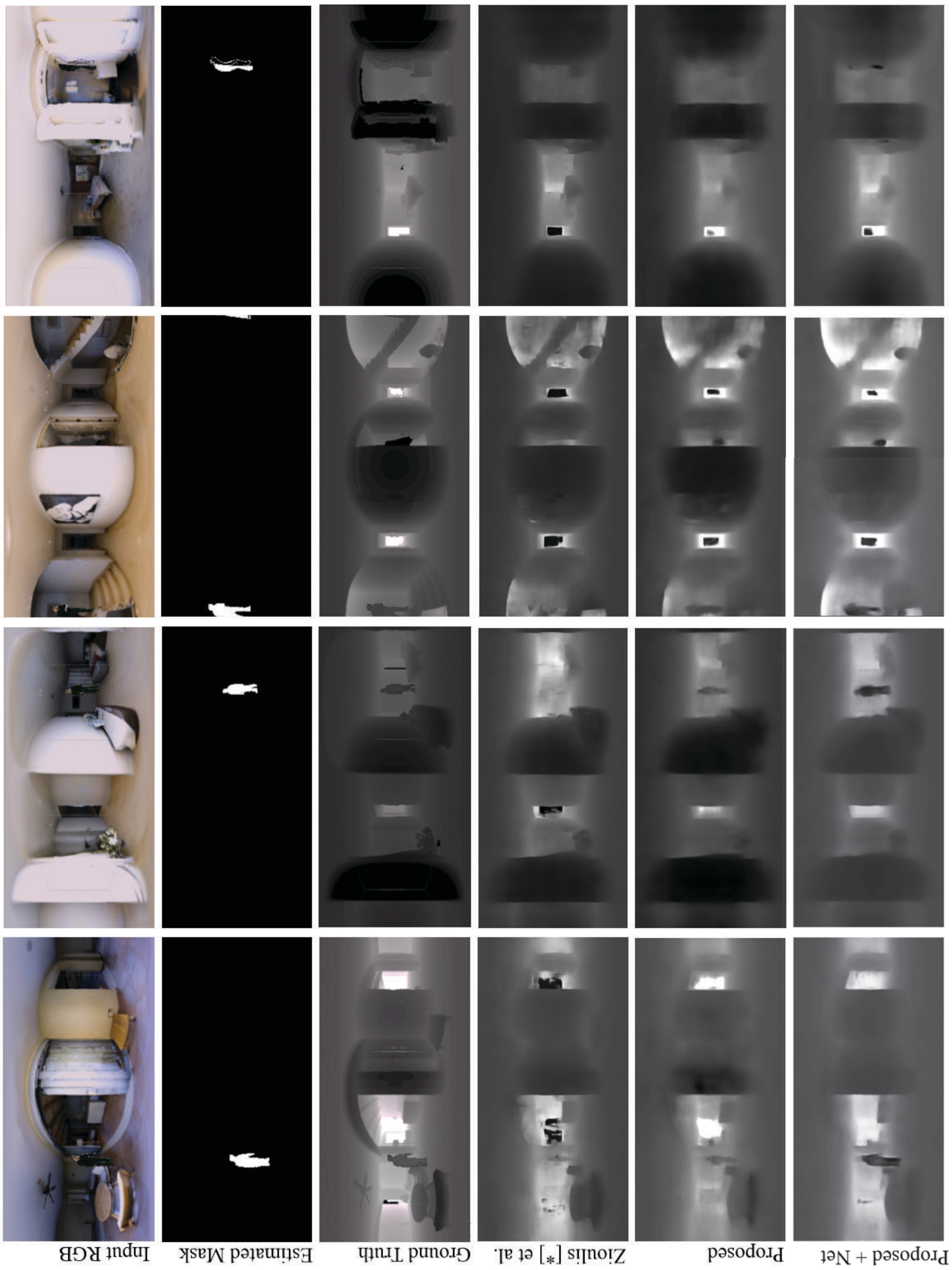


FIGURE 4.12: Qualitative comparison between each model when tested on realistic images.

Quantitative Results

Table 4.1 presents the results of the state-of-the-art models respectively trained with existing synthetic and realistic databases and our proposed databases. We observe that when tested on unseen samples with human objects, networks trained with our proposed databases outperform the existing ones. The increased performance in accuracy against previous methods attributes to more accurate estimations of local human regions, as can be observed in Figure 9. By further quantitatively evaluating the accuracy of estimations between our proposed network and the state-of-the-art models, we can observe the inferior performance of previous approaches as expected in Table 4.2. Depth estimations of local human regions are further refined with our proposed network.

Qualitative Results

To qualitatively evaluate our models' ability to generalize to unseen data, we further acquire and augment samples from the SunCG and the Matterport3D that come from other locations different from training databases. As we can observe in Figure 4.11 and Figure 4.12, our models perform better to estimate the depth of both synthetic and realistic scenes with a human. While previous models yield human depth estimations that are blended with the background and have a blurred edge, our models can predict much clearer and human-shaped results. It is worth mentioning that although all omnidirectional samples used in the experiment only cover indoor settings, our method works with outdoor cases as well.

After observing generated samples, we believe there are many challenges left to overcome. First, even though our method can augment foreground objects, we do not take lighting into consideration during the process. This unnaturalness may lead to less robust estimation in certain scenarios (e.g. scenes with very high brightness).

4.4.4 Ablation Study

In Figure 4.13, we compare the accuracy of depth estimations for local regions under different configurations. Specifically, we compare using original data and proposed data to train only the depth estimation network without the auxiliary MaskNet at first to validate the effectiveness of our data generation method. We then use augmented data to train depth estimation networks with the auxiliary MaskNet, and verified that the local depth loss can successfully improve the consistency of estimated depth within areas of interest. As we can observe in Figure 4.13, our method significantly outperforms the state-of-the-art in local depth estimation.

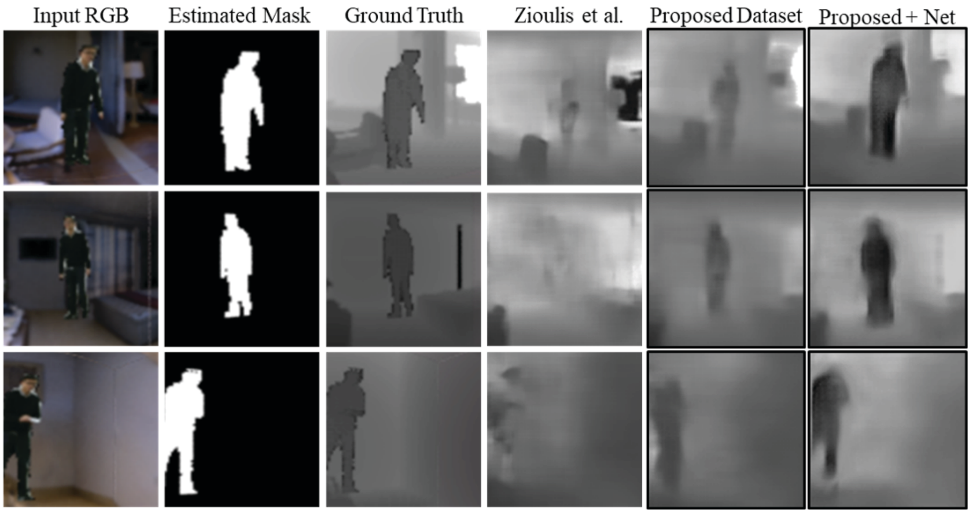


FIGURE 4.13: Estimated depth information of local regions with different configurations. An ablation study shows that using our augmented database can improve the accuracy of local regions, and the proposed network shows an improved consistency with clearer boundaries.

4.5 Multi-view Learning for Foreground-aware Depth Estimation

4.5.1 Network Structure

We explain the proposed foreground-aware bi-projection-based depth prediction method for omnidirectional images in this section. We use a multi-branch end-to-end structure that incorporates two different projections to achieve a more consistent global context and detailed local foreground object features. The proposed architecture is shown in Figure 4.14. In particular, the first branch learns regressing depth information from a single omnidirectional image in the format of equirectangular, providing surrounding information through a wider FOV. As directly using equirectangular images usually causes blurred prediction for local objects with steep gradient changes, the second branch uses cubemap projection to make it more effective to learn local features. With a narrower FOV, cube faces provide more insights into the shape and boundary of foreground objects. Since semantic segmentation and depth prediction are two tasks usually learned together to reveal the scene layout [53] [137] [138], we can improve the accuracy of depth prediction through this foreground-aware network.

For the equirectangular branch, it regresses dense depth information from omnidirectional images with an encoder-decoder structure by progressively downscales and upscales to the depth output. Since skip connections are used to preserve features from higher levels, we adopt Resnet as the encoder of the network. We take advantage of a distorted CNN filter [129] that changes filter sizes with regard to the coordinate on the equirectangular image to improve the effectiveness when training directly on spherical images. We use a traditional L2 loss to calculate the depth loss

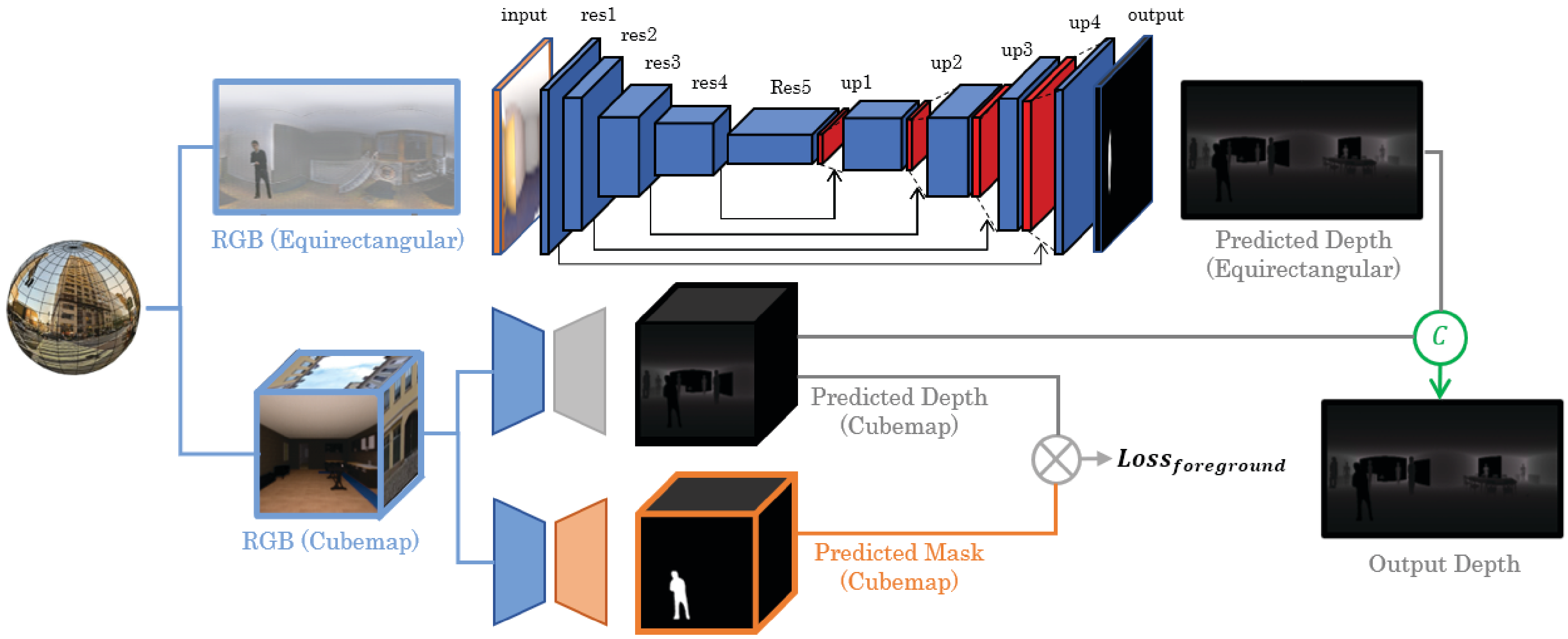


FIGURE 4.14: The proposed bi-projection-based foreground-aware dense depth prediction method for omnidirectional images. We first transform spherical contents into equirectangular and cubemap projection. For the equirectangular projection, we directly regress depth maps with a distorted CNN kernel. For the cubemap projection, we simultaneously predict the semantic segmentation and depth maps. After calculating an additional local loss for foreground objects, we merge the cubemap depth map with the equirectangular one to achieve consistent global prediction with sharp and detailed local regions.

and a smoothness regularization term [117] to improve the consistency of the output.

We further introduce spherical padding and a convolution module at the end of both branches to ensure a consistent merged output. While cubemap projection does not quite suffer from the distortion when projecting spherical information onto a 2-dimensional plane, it instead introduces discontinuity at the boundaries of each face. To alleviate this problem, we adopt a spherical padding technique [74] that increases the FOV when rendering each face and connects them afterward to address the consistency issue. After two branches produce respective dense depth predictions, we unify both branches by concatenating them together and pass through a convolution module described in [139].

4.5.2 Loss Functions

We further utilize the binary mask for foreground objects prepared in the previous step and propose a depth/semantic segmentation multi-task learning scheme for the cubemap branch to strengthen the loss for foreground objects with a foreground object loss.

$$L_{foreground} = D_{cubic\,depth} \otimes M_{foreground}{}^2,$$

and thus the overall loss function for the network is

$$L_{total} = \sum_i (\alpha_i L_{output\,depth} + \beta_i L_{smooth} + \gamma L_{foreground}),$$

while the α, β and γ are the weight coefficients for each loss term.

4.5.3 Experimental Evaluation

Implementation Details

For generating the foreground-aware database, we randomly selected 25,000 synthetic and 25,000 realistic omnidirectional image pairs from existing databases and split them into training and validation sets with a ratio of 80% and 20%. We then composite foreground objects (i.e. humans) onto the acquired samples with a resolution of 512 x 256. We implement the aforementioned network structure with PyTorch[134], Adam optimizer [135], Xavier initialization [136], and a learning rate of 2e-4. The training process is conducted on an Nvidia RTX 2080Ti graphic card. The parameters used for training are $[\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma] = [0.482, 0.245, 0.121, 0.061, 0.090]$. We use the same metrics from the previous work [75] to evaluate our method. At runtime, predicting images with the same resolution can achieve real-time performance.

Experimental Results

We quantitatively and qualitatively evaluate our proposed method in this section. In Table 4.3, we present the result of depth prediction when compared to the state-of-the-art omnidirectional method, [117]. The upper column showcases the effectiveness when

TABLE 4.3: Quantitative comparison against state-of-the-art methods.

Metrics	Database	OmniDepth [117]	Ours
Abs Rel ↓	Synthetic	0.3789	**0.2279**
Sq Rel ↓	Synthetic	0.2893	**0.2134**
RMSE ↓	Synthetic	0.6878	**0.5999**
RMSE log ↓	Synthetic	0.5225	**0.2257**
$\delta < 1.25$ ↑	Synthetic	42.45%	**78.41%**
$\delta < 1.25^2$ ↑	Synthetic	79.26%	**92.85%**
$\delta < 1.25^3$ ↑	Synthetic	92.57%	**97.13%**
Abs Rel ↓	Real	0.3190	**0.2246**
Sq Rel ↓	Real	0.2180	**0.1727**
RMSE ↓	Real	**0.5993**	0.6042
RMSE log ↓	Real	0.4788	**0.2427**
$\delta < 1.25$ ↑	Real	69.88%	**75.37%**
$\delta < 1.25^2$ ↑	Real	84.54%	**91.73%**
$\delta < 1.25^3$ ↑	Real	91.50%	**96.66%**

applied to the synthetic domain, while the bottom column demonstrates its efficacy in real-world scenarios. We can observe that our method shows favorable performance with improved accuracy across the board against the existing method when benchmarking with accuracy metrics. We believe that the increased accuracy attributes to the bi-projection network architecture in addition to the semantic segmentation task in the cubemap projection branch. This is qualitatively verified through Figure 4.15 and Figure 4.16, as we can observe that our model generalizes to unseen data with foreground objects and yield satisfying depth prediction.

4.6 Conclusion of the Chapter

We have presented a data augmentation method to generate high-quality equirectangular databases with paired color and ground-truth depth annotations by repurposing abundant and easily obtainable 2-dimensional RGBD databases. With this database, we further introduced and implemented an auxiliary network that calculates local depth loss to resolve an issue that small regions of interest are frequently smoothed out during optimizing global gradients. We take humans, a crucial subject in 360° contents, as an example to show the efficacy of our approach. We showed improved accuracy of our approach compared to the state-of-the-art technique. We then present a foreground-aware bi-projection-based depth prediction method for omnidirectional images. The proposed architecture produces consistent global depth prediction with the equirectangular projection while enforcing local detailed features through the cubemap projection. An additional foreground loss acquired through a multitask learning approach of semantic segmentation complementarily provides sharper boundaries of

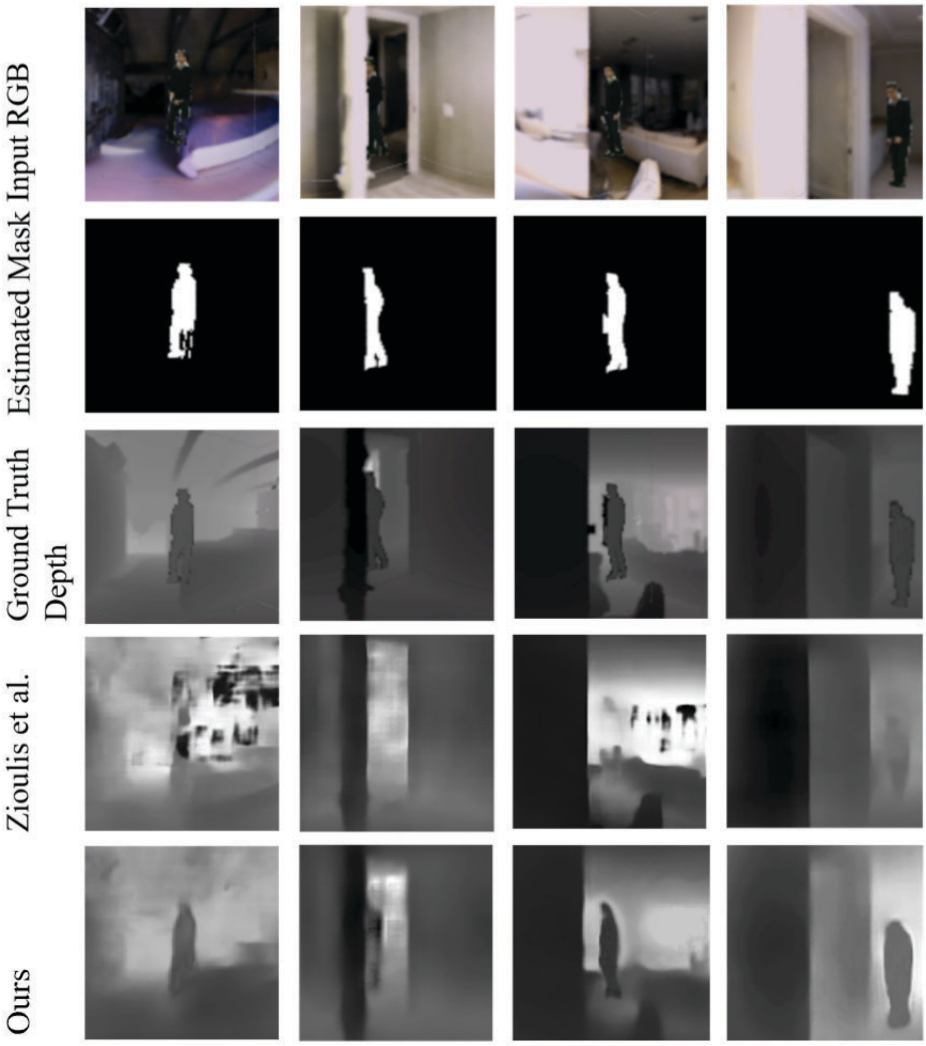


Figure 4.15: Qualitative comparison of foreground estimations against the state-of-the-art method when tested on realistic images.

predicted foreground objects. With quantitative and qualitative evaluation, we successfully verified the effectiveness of the proposed method. We believe the ability to accurately predict depth information for omnidirectional images can facilitate a wide range of applications such as 3-dimensional reconstruction and virtual reality. We believe that the ability to estimate depth for foreground objects in 360° images can benefit a wide range of applications such as navigation in robotics and augmenting virtual objects with occlusions.

Currently, our data augmentation method is based on the premise that both 2-dimensional and 360° data are captured with similar extrinsic parameters (e.g. cameras are aligned horizontally, positioned at average eye-level height) and lighting conditions, while it is true for most data captured in lab conditions, its application for in-the-wild images is limited. Furthermore, our approach works for both indoor and outdoor settings by compositing synthetic/captured omnidirectional databases. Nevertheless, for outdoor settings, a higher dynamic range of luminosity and sunlight's

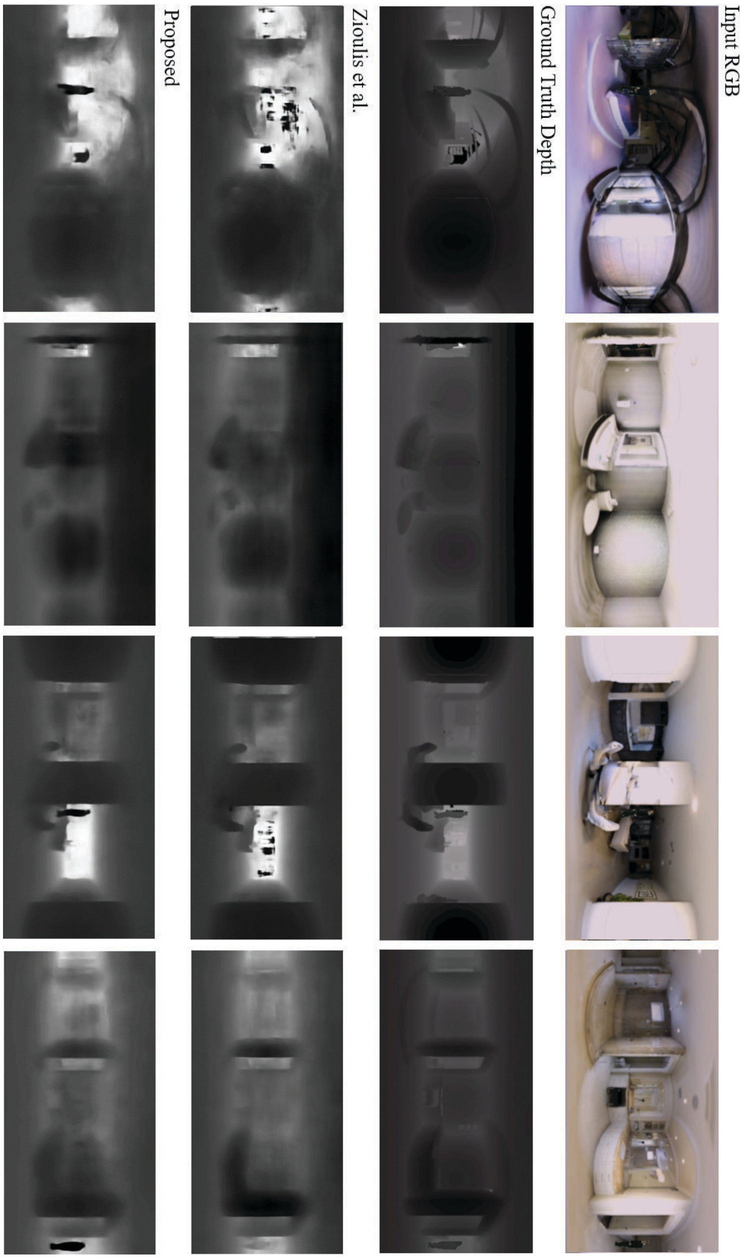


FIGURE 4.16: Qualitative comparison against the state-of-the-art method when tested on realistic images.

ambient IR will render capturing RGB and depth information inherently difficult. For future work, we aim to explore generating samples with different lighting conditions with GANs to improve the robustness of depth estimation.

Chapter 5

Comprehending the Global: 360° Depth Prediction in the Wild

In this chapter, we continue zooming out and further broaden the scope to understand the global environment of mixed reality. We propose a novel approach of data augmentation and depth estimation to extend the capability of 360° scene understanding from only indoor environments to all situations [24]. We are the first to propose utilizing abundant online 360° videos available on the internet to generate a large-scale database, Depth360, that comprises a wide range of conditions. We further proposed an end-to-end multi-task deep learning network to effectively learn from the proposed database. With the ability to estimate high-quality depth information of the global context of omnidirectional images, we implement an application to showcase how scene understanding can help improve mixed reality.

5.1 Introduction

Visual reasoning in the context of omnidirectional images has gained increasing popularity in both academic and industrial communities during the past few years. By providing rich information about the environment with a large field-of-view (FOV), predicting dense depth maps from a single 360° image shows wide applicability and facilitates applications that require accurate understandings of the context, such as scene reconstruction [140] and autonomous navigation [115]. However, inferring depth from a monocular image is a challenging and ill-posed problem due to uncontrolled extrinsic, ambiguous scales, and varied settings. Recently, data-driven deep learning methods [73] have presented significant potential in this field.

Despite learning-based methods having been extensively studied within the context of perspective images, omnidirectional format presents challenges in both aspects: data preparation and depth estimation algorithm. On the one hand, large-scale 360° training data is difficult to collect. For synthesis-based methods, the cost to create large-scale models that resemble real-world ones with abundant settings is excessively high [141], and the diversity gap between synthetic samples and real data leads to less accurate results [142]. For capturing-based methods, using dual 360° cameras for stereo-capturing will introduce mutual occlusion. Specialized scanning devices

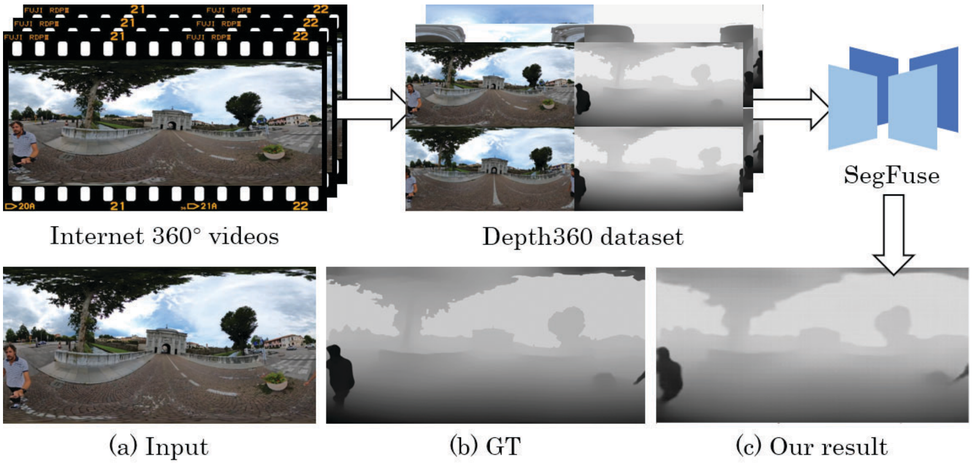


FIGURE 5.1: We present a method for generating large amounts of color/depth training data from abundant internet 360° videos. After creating a large-scale general omnidirectional dataset, Depth360, we propose an end-to-end two-branch multitasking network, SegFuse to learn single-view depth estimation from it. Our method shows dense, consistent and detailed predictions.

(e.g. Matterport [119]) produce dense datasets but are limited to indoor use due to their working principle. Depth maps produced with laser scanners (such as LIDAR [143]) suffer from self-occlusion albeit being the main source for outdoor settings. Most datasets are only captured under specific scenarios (e.g. atop a driving car [144]). On the other hand, existing learning-based approaches cannot effectively take advantage of 360° image datasets. The majority of depth estimation methods [73] are designed for perspective cameras with narrower FOV. Due to the spherical nature of the content, projecting to a 2D image introduces irregular distortions and thus hinders effective learning [145]. Even though there are a few methods [117] [74] proposed with distortions in mind, they only focus on indoor settings due to the unavailability of outdoor datasets. As a result, they show sub-optimal performance under general cases.

In this chapter, we first tackle the problem of limited datasets by exploring the use of the plenteous source of data: 360° videos from the internet that are captured with a moving hand-held omnidirectional camera. We propose a test-time training method that utilizes a learning-based prior to synthesizing plausible depth maps for each consistent 360° video. By leveraging the rich information that is only presented in omnidirectional formats, we propose to use the output of structure-from-motion (SfM) and multi-view stereo (MVS) methods to calculate a novel geometric consistency based on a geometric spherical disparity model. We also propose to use optical flow [146] to encourage temporal consistency and establish multiple constraints for each pixel that ensure a convincing output. With established constraints, we fine-tune a pre-trained model by updating the parameters according to the calculated geometric and temporal losses to produce a more consistent output for a particular sequence. During dataset creation, our test-time training method takes preprocessed video sequences as input and generates a geometrically and temporally consistent dense depth map for each frame. To our knowledge, our large-scale dataset, Depth360, is the first to use internet omnidirectional videos for achieving monocular depth estimation from single 360° images. To benchmark the accuracy of data generation, we propose using rendering-based methods to further generate a photorealistic synthetic dataset, SynDepth360. With the unlimited training data with diverse conditions, we seek to learn depth estimation with high accuracy and generalization.

We then propose an end-to-end neural network architecture, SegFuse, to learn the single-view depth estimation of omnidirectional images that generalize well with a wide range of settings by mimicking the human eye. While videos usually provide more cues for depth calculation and facilitate dataset creation, lengthy optimization for individual scenes does not achieve as good generalization and practicality compared to single-view depth estimation. We believe that compared to indoor depth maps with more uniform distributions and relatively universal ranges, more challenging variations of outdoor images, i.e. unsymmetrical depth distributions (sky and ground) and distinct depth ranges between different scenes, lead to ineffectively learning processes and generalization for existing methods. To cope with such problems, we propose a multi-task learning framework that adopts a bi-projection fusion scheme: a peripheral branch that uses equirectangular projection for depth estimation and a foveal branch that uses cubemap projection for semantic segmentation. While equirectangular projection can provide consistent global context, cubemap projection gives more local details with a narrower FOV. With the peripheral vision to perceive the depth of the scene and foveal vision to distinguish between different objects, our method can successfully learn a smooth global depth while maintaining details in local regions. Compared to the method [74] with a similar structure, SegFuse uses multi-task learning to exploit semantics in complex depth distributions, and achieve significantly improved performance in outdoor settings.

By applying the generated training data with diverse conditions to multiple state-of-the-art learning-based omnidirectional depth estimation methods, our experimental results show that our method outperforms existing methods with more consistent global results and sharper local estimations.

To summarize, our contributions are as follows:

1. To solve the unavailability of a general omnidirectional dataset with dense depth maps, we are first to propose to utilize omnidirectional video in the wild to generate a large-scale dataset, Depth360. By exploiting unique temporal and geometric consistencies of 360° videos with a spherical disparity model, we use test-time training to generate convincing depth maps.

2. We propose an end-to-end two-branch multi-task architecture called SegFuse that estimates depth from a single-view 360° image input by mimicking the human eye. The peripheral branch regresses global depth estimation while the

foveal branch estimates local semantic segmentation. By fusing the global context and local details, our design ensures a sharp and consistent depth prediction under challenging cases.

3. To validate the accuracy of the proposed dataset and evaluate the effectiveness of our multitasking method, we perform an extensive evaluation against state-of-the-art omnidirectional datasets and methods and present a better quantitative and qualitative performance.

5.2 Related Work

5.2.1 Monocular Depth Datasets

One of the major issues in learning-based single-view depth estimation is the unavailability of data. For perspective images, most supervised depth-estimation methods are trained on a few standard datasets (e.g. NYU [85]) due to the difficulty of acquiring ground truth depth maps. Capturing-based methods often utilize RGB-D sensors and laser scanning (e.g. LIDAR [144]). To improve data availability and ease of acquisition, several efforts have been made. Godard et al. [75] use multiple views of a scene as a supervisory signal, but these approaches usually require two input images at test time [147]. Mayer et al. [148] use a synthetic dataset, but the domain gap results in sub-optimal performance in real-world scenarios and requires further domain adaptation [122]. Using internet images [149] and videos [150] to calculate pseudo ground truth with structure-from-motion and multi-view stereo shows great performance but is only explored in perspective context.

When it comes to omnidirectional depth maps, not only capturing-based methods are greatly limited, but also the existing perspective-based approaches are less effective, resulting in the scarcity of outdoor datasets. Existing omnidirectional sensors with customized arrays suffer from strong self-occlusions, leading to missing or sparse information at the bottom of the sphere. Using multiple monocular cameras as a stereo setup (i.e., 3D VR cameras) to calculate disparity is also problematic due to mutual occlusion [151]. Using domain adaptation for synthetic data requires both large-scale 3D models with great variations and corresponding similar 360° color ground truth. Most concurrent works [117] [74] either use synthetic datasets (i.e., PanoSunCG [152]) or 3D scanned datasets (i.e., Matterport3D [119], Stanford 2D-3D [120], Pano3D [153]). The former is generated with 3D models and a virtual omnidirectional camera without domain adaptation, and the latter ones are captured with specialized equipment and post-processed. Both suffer from no dynamic foregrounds, further limiting their usefulness in real-world scenarios [114]. Zhu et al. [142] propose to use physics-based rendering to generate synthetic outdoor panoramas, but the diversity gap between synthetic samples and real data leads to less accurate results. Therefore, taking advantage of an increasing number of shared online omnidirectional

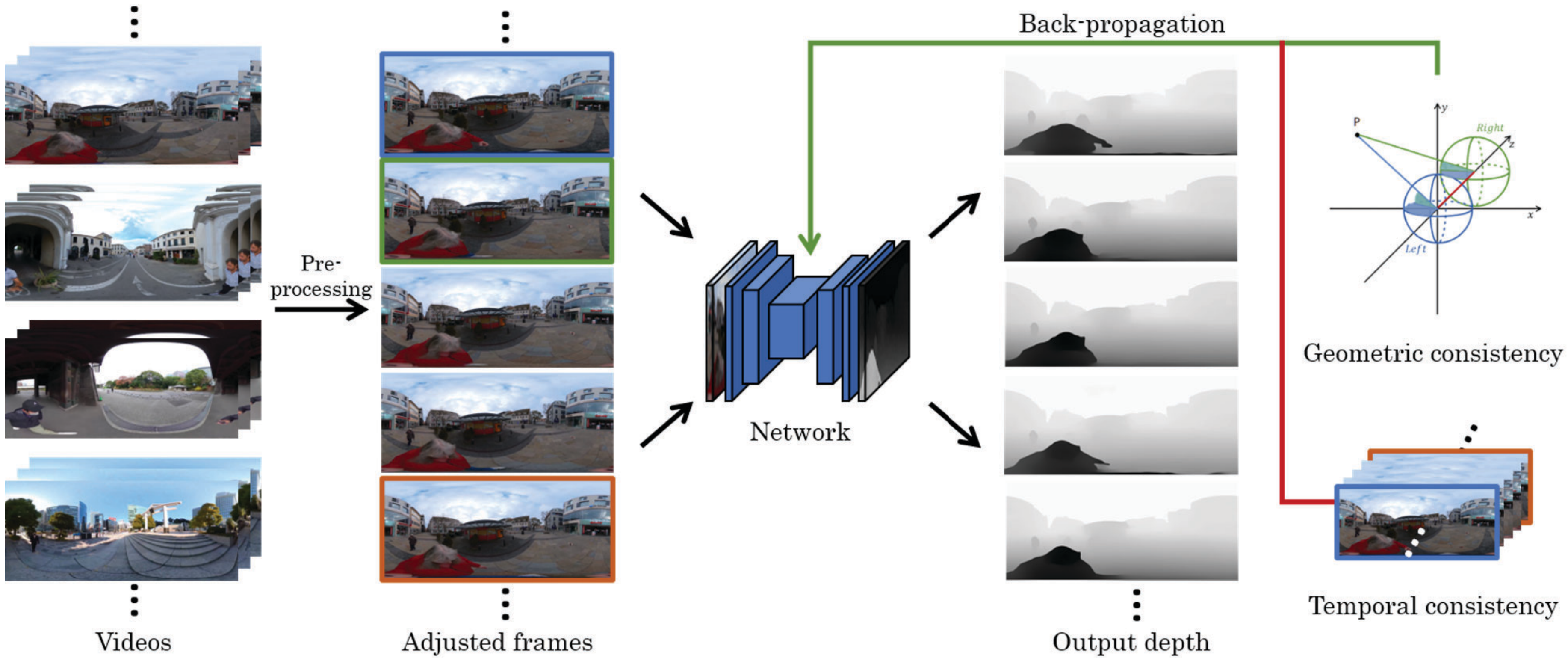


FIGURE 5.2: The overview of our dataset generation method. With monocular videos as input, we sample successive frames from a single sequence and adjust the frames spatially with baselines acquired with SfM and MVS methods. Geometric and temporal constraints of this sequence are then established using a geometric spherical disparity model and a 360°-aware optical flow algorithm. By fine-tuning a learning-based prior with computed losses through back-propagation during test time, we can generate consistent depth output that satisfies the constraints of the corresponding sequence.

videos, we propose a pipeline to utilize rich information in the wild to generate a large-scale dataset.

5.2.2 Monocular Depth Estimation for Perspective Images

Predicting depth from monocular color images is an important task in understanding 3D scene geometries [154]. An accurate estimation can benefit various applications such as autonomous driving [144] and graphics rendering [155]. Traditional methods of monocular depth estimation heavily rely on probabilistic graphical models with hand-crafted local features and constraints (e.g. MRF) [156]. With the advances in deep learning algorithms, recent learning-based approaches [124] [157] [158] show significant improvements in accuracy.

A standard approach to learning an implicit relation between color and depth is to train models with collected RGB images and ground truth depth maps. Eigen et al. [124] propose multi-scale networks to refine coarse depth with local details. This two-scale strategy is further refined to predict high-resolution depth [158]. A fully convolutional architecture with a novel up-projection module proposed by [73] improves the output accuracy. Cao et al. [159] propose to solve depth regression in a classification fashion. Another direction for improving the output quality is to combine graphical models with the use of CNNs, such as incorporating conditional random fields in the form of a loss function into the depth estimation task [72]. However, when directly applying perspective models to 360° images, an inferior performance is observed due to the lack of global consistency and incorrectly modeling the projection's distortion [117].

5.2.3 Monocular Depth Estimation for Omnidirectional Images

As omnidirectional cameras have become more efficient and accessible, the interest in 360° media has surged on the internet owing to novel applications such as virtual reality [160] [13] and mixed reality [161]. For single-view depth estimation, while a large body of research exists for perspective images, scarce work has been done to address this problem for spherical images. The most apparent issue is the distortion introduced when projecting the 3D spherical information onto the 2D plane. Although rotation equivariant CNNs [162] and graph-based learning [163] with spherical cross-correlation directly learn from 3D spherical signals, such equivariant architectures define convolution in the spectral domain and provide a lower network capacity, hindering applicability in generative tasks such as monocular depth estimation. To apply deep learning approaches to omnidirectional content, most approaches are proposed using two projection formats, cubemap, and equirectangular projections.

While cubemap projects spherical signals onto 6 faces of a cube, and thus enables directly feeding non-distorted images into a CNN, the discontinuity along boundaries is problematic when trying to merge results back into a spherical image. A common solution is using cube padding [164] to aid the network in merging estimations for each face into a full omnidirectional output. This method is effective when applied to single-view depth estimation for indoor scenes [74] with a relatively uniform depth distribution and other tasks such as stylization [128] and classification [165]. However, these methods are less effective when each face has wildly changing depth ranges in outdoor scenarios [166]. Since each face only includes very limited information about a local region, dramatically different appearances and the ambiguity of depth scales usually result in distinct estimations, limiting the scalability of such approaches. Recent works using diverse division schemes show improved predictions for indoor samples [167]. However, slice-based methods that exploit relationships of vertical patches [168] also report discontinuities for outdoor cases.

To make the network efficient and directly aware of the distortion in omnidirectional images, work resorted to using equirectangular projection with distorted filters [117] and dilations [169]. However, the effectiveness of these methods is limited. As the layers deepen, non-linearly distributed information across an equirectangular image got lost (e.g. consistency across the sphere). Although this problem is alleviated by a kernel transformer [170] that uses parameterized functions to preserve cross-channel interactions, the model size is still limited. While using equirectangular projection can generate more consistent global prediction due to its wider FOV, small regions with a steep local gradient when regressing the global gradient are harder to learn [114]. Wang et al. [74] and Jiang et al. [171] use a fusion scheme that combines the depth maps estimated with equirectangular and cubemap projections for sharper depth estimation. Although it presents improved accuracy for indoor settings, the disadvantage of limited scalability remains [168]. Instead, we purpose an architecture that fuses a cubemap branch for semantic segmentation with an equirectangular branch for depth estimation. Considering that regressed depth maps for different faces are hard to balance when training with outdoor samples, semantic segmentation can serve to inform the global depth estimation of the local details without the problem of balancing scales between each local view.

5.3 The Depth360 Dataset

We propose the world's first generated large-scale dataset *Depth360* that utilizes 360° videos in the wild to solve the unavailability of a general omnidirectional dataset with dense depth information. We first preprocess a video sequence with an SfM and MVS approach to establish quality frame groups that facilitate computing constraints of the sequences. With a horizontal spherical disparity model, we propose novel temporal and geometric consistencies that are unique to 360° videos. By incorporating constraints into test time training through backpropagation, we generate convincing dense depth maps for the corresponding sequence. The generation process is shown in Figure 5.2, and some examples are shown in Figure 5.3.

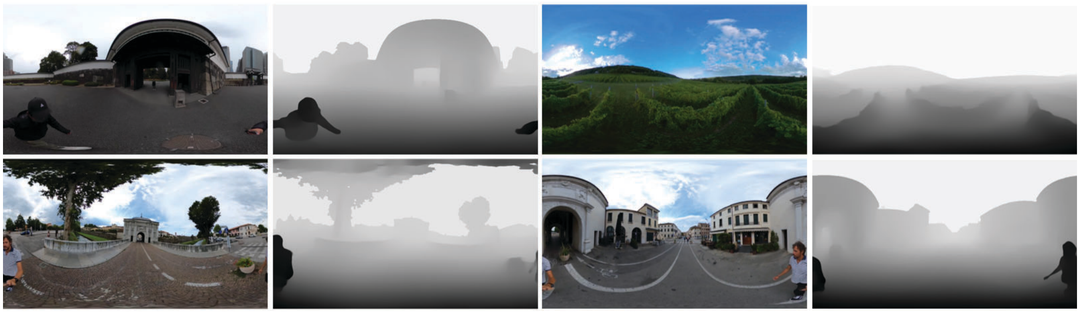


FIGURE 5.3: Examples of generated RGB/depth pairs. The color images are video frames acquired from the internet, and the corresponding depth maps is generated through our test-time training method.

5.3.1 Data generation with Test-time training

We propose a test-time training method that first estimates plausible dense estimations utilizing a learning-based prior, and then iteratively fine-tines the parameters during test time with unique constraints established from a certain 360° sequence to generate accurate depth output. Since 360° videos gathered from the internet usually suffer from unconstrained extrinsic and different intrinsic, existing methods often fail to show satisfying performance for dataset creation. On the one hand, depth produced by reconstruction-based methods is usually sparse and erroneous due to distortions. On the other hand, directly applying learning-based methods for frames independently usually results in inconsistent estimation and sub-par accuracy due to the domain gap between perspective and equirectangular formats. With the proposed test-time training method, we take preprocessed video sequences as input and generate geometrically and temporally consistent dense depth maps for each frame.

We calculate a geometric loss between corresponding frames reprojected from the estimated depth map and stereo pairs' disparity, in addition to a temporal loss that penalizes the error between flow-based and depth-based projections. In each iteration of fine-tuning a pre-trained depth estimation network, we first generate depth maps for multiple frames with the current network. We then update the parameters according to the calculated geometric and temporal losses to ensure its weight can produce a more consistent output for a particular sequence (Figure 5.2).

Preprocessing. We exclude dynamic foreground objects from the frames for better calculating camera extrinsic and establish geometric constraints for the respective sequence. Since people are usually the most common dynamic foreground objects in perspective videos [155], we found this remains true for omnidirectional videos in the wild as well.

We first use OpenVSLAM [172], an open-source visual SLAM framework, to estimate the pose of the camera (r, t) and the distance b between frame pairs. We then use an off-the-shelf SfM pipeline COLMAP [173] to acquire sparse depth maps D^{Recon}. To improve pose estimate for videos with a strong motion, we apply Mask R-CNN [52] to obtain static segmentation for more reliable feature point extraction and matching. During this process, we automatically filter out videos with a static viewpoint and

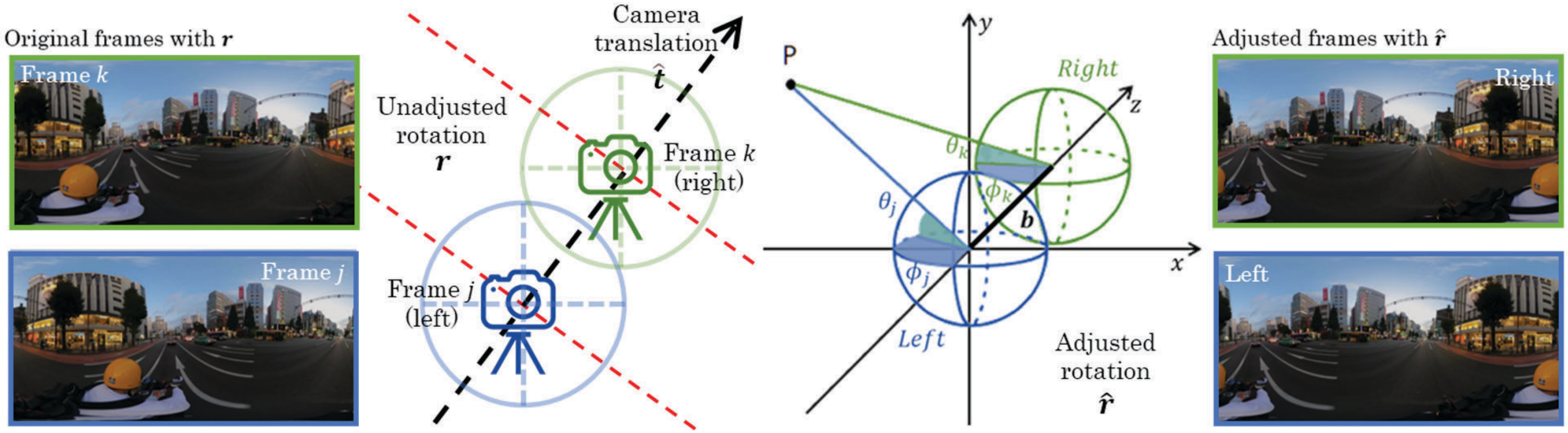

Original frames with $\boldsymbol{r}$
Frame k
Frame j
Unadjusted rotation $\boldsymbol{r}$
Camera translation $\hat{\boldsymbol{t}}$
Frame k (right)
Frame j (left)
P
y
z
x
Right
Left
θ_k
ϕ_k
θ_j
ϕ_j
$\boldsymbol{b}$
Adjusted rotation $\hat{\boldsymbol{r}}$
Adjusted frames with $\hat{\boldsymbol{r}}$
Right
Left

FIGURE 5.4: The spatial adjustment process using geometric horizontal spherical disparity models. The left illustration describes original successive frames that satisfy L-R stereo correspondence using the camera poses. The right illustration describes the model to calculate disparity from two frames with a left-right displacement. The ϕ denotes the longitude while the θ denotes the latitude. The b is the baseline acquired from the previous step, and P is the 3D displacement of a target point. The left examples show unconstrained frames acquired directly from internet videos, and the right examples show spatially adjusted frames that facilitate the calculation of geometric constraints.

vertical motions with estimated poses since they are more challenging in establishing the geometric constraints, and group the remaining videos into consistent short sequences S.

Spatial adjustment. Since learning-based and reconstruction-based methods are independent of each other and both are scale-invariant, we need to first adjust the scale to match the output before establishing geometric constraints. We achieve this by multiplying all estimated camera translations for a single sequence with a scale factor to match the scale of learning-based depth estimations. For sequence S_i with j frames, the scale factor s_i is calculated as:

$$S_i = \sum_j \frac{D_j^{NN}(x)}{D_j^{Recon}(x)} / j | D_j^{Recon}(x) \neq 0 \tag{5.1}$$

where the $D(x)$ is the depth value at pixel x yielded by the learning-based prior before test-time training. The updated camera translation is now $\hat{t_i} = s_i \cdot t_i$.

While it is usually impossible to create aligned stereo pairs from unconstrained perspective videos due to random camera extrinsic, omnidirectional images have the unique feature of rotation-invariance. For short 360° sequences with minimal vertical movements, we can create aligned left-right stereo image pairs by adjusting the rendering camera rotation to $\hat{r}$ so that the trajectory of frame centers stays parallel to the camera translation $\hat{t}$. This process is demonstrated in Figure 5.4.

Geometric loss. To calculate the geometric loss from adjusted left-right image pairs (j, k) with a baseline b, we use a modified spherical disparity model from [145]. For each point p at (x, y, z) in Cartesian coordinate, we use longitude ϕ and latitude θ in spherical polar coordinate to describe the corresponding point (Figure 5.4). In this sense, the radial distance r to a certain point is $\sqrt{x^2 + y^2 + z^2}$, and the horizontal disparity is defined as $\delta = (\phi_j - \phi_k, \theta_j - \theta_k)$. Since the baseline $b = (0, 0, dz)$ is acquired from the previous step, the disparity is now $\delta = (\frac{\partial \phi}{\partial z}, \frac{\partial \theta}{\partial z})$. The transformation between spherical and Cartesian coordinates is omitted to simplify the notations.

To render a target frame $\hat{k}$ from the source frame j, each pixel $p = (\phi, \theta)$ on the equirectangular image is a function of the baseline b and the radial distance r. Since we already have the generated depth map $D_j^{NN}(p)$ for frame j, we can compute the target frame $\hat{k}$ with a function:

$$\hat{k}(p) = \Gamma_{j \to \hat{k}}(D_j^{NN}(p), b_{j \to k}, j(p)) \tag{5.2}$$

Considering the image acquired from online videos are usually not perfect stereo pairs and include dynamic foreground objects, errors often got amplified at certain regions (e.g. top and bottom) on equirectangular projection due to stronger distortion. To alleviate this problem, we further adopt a weight matrix $M(p) = |sin(\phi)||sin(\theta)|$ that assigns different weights for each pixel and aggregates the loss with regard to the

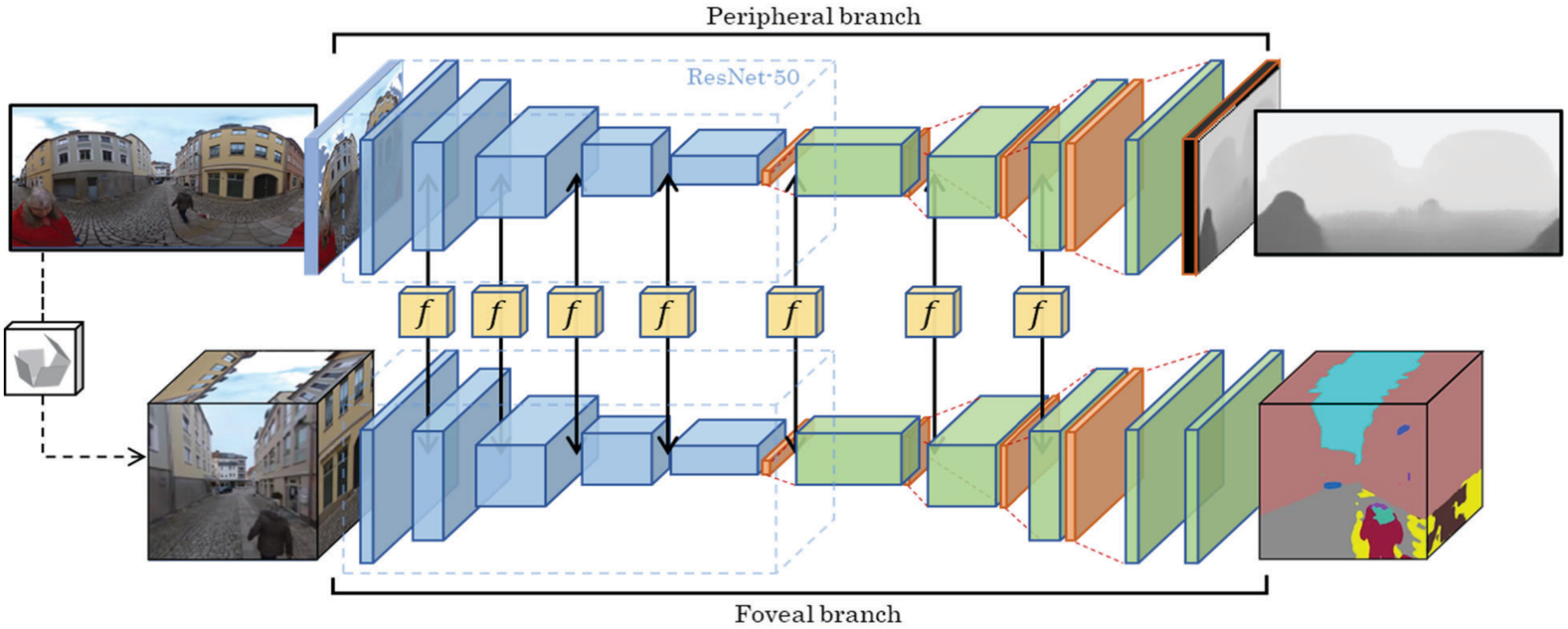


FIGURE 5.5: Overview of the proposed end-to-end two-branch multi-task learning network, SegFuse. Structure-wise, the peripheral branch that uses equirectangular projection is capable of capturing global context while the foveal branch uses cubemap projection produces sharper boundaries for local objects. Objective-wise, semantic segmentation and depth estimation are jointly learned to reveal the scene layout and object shapes, while the peripheral branch enforces more consistent depth estimation with a wider FOV, the foveal branch estimating segmentation is more robust to scale changes which frequently appear in a more general dataset. The fusion modules f further facilitate feature sharing between two branches.

distortion level when calculating the geometric loss:

$$L^{geometric}_{j\to k} = \sum_p ||M\hat{k}p - Mk(p)||_2 \tag{5.3}$$

Temporal loss. Optical flow is a popular option to check short-term consistency in learning-based video processing for its capability of describing the same scene points in successive frames [128]. Since depth-estimation networks estimate depth maps independently, the result for a video is usually unstable and inconsistent. To solve the inconsistency between frames of a 360° video, for all frame pairs (j, k) in sequence S_i, we further calculate a dense optical flow $f_{j\to k}$ to ensure a temporal consistency during test-time training.

It is more suitable to establish short-term and long-term consistency for omnidirectional videos compared to unconstrained perspective videos due to two reasons. First, bad alignment of frames is challenging to cope with for perspective videos while spherical videos can be easily calibrated with simple rotations. Second, while the problem of occlusion remains, objects exiting and re-entering the frame are significantly less prominent in equirectangular videos, making the long-term consistency more reliable. To account for distortions of equirectangular projection, we use a modified version of FlowNet2 [146], OmniFlowNet [174] with a distorted CNN kernel.

For pixel $p = (\phi, \theta)$ on a source equirectangular image j, the corresponding pixel $\widetilde{p}$ on the target frame $\widetilde{k}$ is calculated by:

$$\widetilde{p} = p + f_{j\to k}(p) \tag{5.4}$$

where f denotes the optical flow between two frames. We compute the target frame $\widetilde{k}$ based-on flow with function F:

$$\widetilde{k}(p) = F_{j\to\widetilde{k}}(f_{j\to k}(p), j(p)) \tag{5.5}$$

Similarly, the temporal loss is calculated for each pixel with:

$$L^{temporal}_{j\to k} = \sum_p ||\widetilde{k}p - k(p)||_2 \tag{5.6}$$

Optimization. We then fine-tune the network weights with the combined loss $L_{j\to k}$ between frame pairs through backpropagation for 10 epochs:

$$L_{j\to k} = L^{geometric}_{j\to k}(p) + L^{temporal}_{j\to k}(p) \tag{5.7}$$

The overall loss is a sum of the geometric loss and the temporal loss calculated over all pixels in video frames, and the network parameters are initialized using a pre-trained network [175] trained on the Mix 5 dataset [176]. To reduce the computational cost of computing dense optical flow for image pairs, we calculate the flow between consecutive frames for short-term consistency and left-right pairs for long-term consistency.

5.3.2 Implementation Details

To create the general dataset Depth360, we use the test-time training method to generate convincing depth maps from omnidirectional videos in the wild. We first gathered equirectangular video sequences from the internet that are captured with a hand-held omnidirectional camera. After filtering out samples with strong motion blur, post-editing, and texture-less scenes, we used 30 clips to produce corresponding depth maps. We then fine-tune the weight of the same pre-trained network for each sequence with the geometric and temporal loss using standard backpropagation. By generating consistent depth maps for each sequence with fine-tuned networks after 10 epochs, we create a dataset of paired color images and depth maps with a size of 30,000. Several examples of our generated samples are shown in Figure 5.3.

5.3.3 The Benchmark Dataset

To benchmark the effectiveness of the test-time training method and accuracy of the Depth360 dataset, we propose using rendering-based methods to generate a small-scale synthetic dataset via 3D models and virtual cameras. This additional SynDepth360 dataset is motivated by the challenge to directly acquire the ground truth of the internet videos. While the large-scale Depth360 dataset is advantageous to train end-to-end models for single-view depth estimation, the rendered small-scale outdoor 360° synthetic dataset with diverse settings is helpful for future research, which we will release together with the Depth360.

5.4 SegFuse: A Multi-input Multi-output Learning Network

Combining the advantages of a more consistent global context and sharper local details, we propose an end-to-end two-branch multitask learning network called SegFuse. It estimates depth from a single omnidirectional view by mimicking the human eye, as shown in Figure 5.5. In particular, the upper branch regresses depth maps with equirectangular projection, resembling human's peripheral vision to perceive depth, and the lower branch that estimates semantic segmentation with cubemap projection mimics foveal vision to distinguish between different local objects.

We justify our network design from two aspects. Structure-wise, equirectangular projection is capable of capturing global context but the distortion and a larger FOV restrict its effectiveness against local regions, while cubemap projection produces sharper boundaries for local objects but introduces inconsistency between faces. Objective-wise, since semantic segmentation and depth estimation are two tasks usually jointly learned to reveal the scene layout and object shapes [53] [137] [138], while semantic segmentation is more robust to scale changes, we design our two-branch multitasking network that takes advantage of both global context and local details to learn single-view estimation on a more general omnidirectional dataset.

5.4.1 Network Structure

The peripheral branch. Our peripheral branch regresses a dense global depth estimation from a single view equirectangular image. Its encoder-decoder structure progressively downscales and upscales to the target depth maps. We adopt rectangular filters with changing sizes at the first convolution layer to account for different distortion strengths along the vertical axis of the input equirectangular image. The encoder of this branch shares the same structure of ResNet-50 [133], while the decoder consists of four up-projection blocks [73].

The foveal branch. Our foveal branch receives reprojected cubemap faces of the input equirectangular image as input and generates semantic segmentation as the output. We choose the semantic segmentation task for the cubemap branch for two reasons. First, although directly regressing depth maps for separate cube maps seems to be a more intuitive choice and has shown some improved performance in similar applications [74], the problem of discontinuity at cubemap boundaries is amplified when applied to uncontrolled general samples. We believe that compared to indoor scenes with more uniform and symmetrical structures, our samples generated from online videos are more challenging for the network to learn due to stronger scale ambiguity caused by distinct depth ranges and unsymmetrical depth maps (e.g. sky and ground). This is further verified in our qualitative evaluation. Second, with undistorted cubemap projection, the foveal branch not only facilitates sharing features of local objects, it can also directly utilize traditional perspective-based model weights to accelerate the learning process, improve the model accuracy, and most importantly, circumvent the challenge of acquiring omnidirectional segmentation ground truth.

Structure-wise, to better facilitate feature fusion at each scale, we set up an identical encoder-decoder network with Resnet-50 encoder and four up-projection modules as the decoder. Instead of incorporating a filter at the first layer to account for distortion, we reproject the equirectangular image to cubemap before feeding it to the first layer. We then incorporate a spherical padding process [74] to pass feature maps between layers to connect different cube faces.

The fusion scheme. To encourage feature sharing between the peripheral branch and the foveal branch, we perform a fusion scheme that lets each branch inform the other with respective feature maps to balance both branches during the training process. Unlike [74], we simplify the fusion scheme to improve the training efficiency, and we reduced the number of fused layers to prevent an unstable training process due to different tasks. A more detailed ablation study is presented in the experiment section.

With m_p as the feature map from the peripheral branch and m_f as the feature map from the foveal branch, we first reproject the m_f to $\hat{m_f}$ in equirectangular format and m_p to $\hat{m_p}$ in cubemap projection. We then pass $m_p + C(\hat{m_f})$ to the next layer of the peripheral branch and $m_f + C(\hat{m_p})$ in the foveal branch respectively. The C denotes a convolution layer.

5.4.2 Loss Functions

We use supervised loss constraints for both depth estimation and semantic segmentation tasks. For depth estimation, we use inverse Huber loss defined in [73] as the optimizing objective:

$$L^D(d) = \{ \mid d||d| \geq c\frac{d^2 + c^2}{2c}|d| > c \tag{5.8}$$

where d is the difference between the estimated result and the ground truth for each pixel, $c = \max(d)/5$. The loss function L^S for semantic segmentation is a cross-entropy loss between the estimated segmentation S and the result predicted with a pre-train network S. Combined, the total loss function can be defined as:

$$L^{total} = L^D(D, D) + L^S(S, S) \tag{5.9}$$

During the experiment, segmentation samples are acquired with a pre-trained weight trained on MIT ADE20K dataset [177].

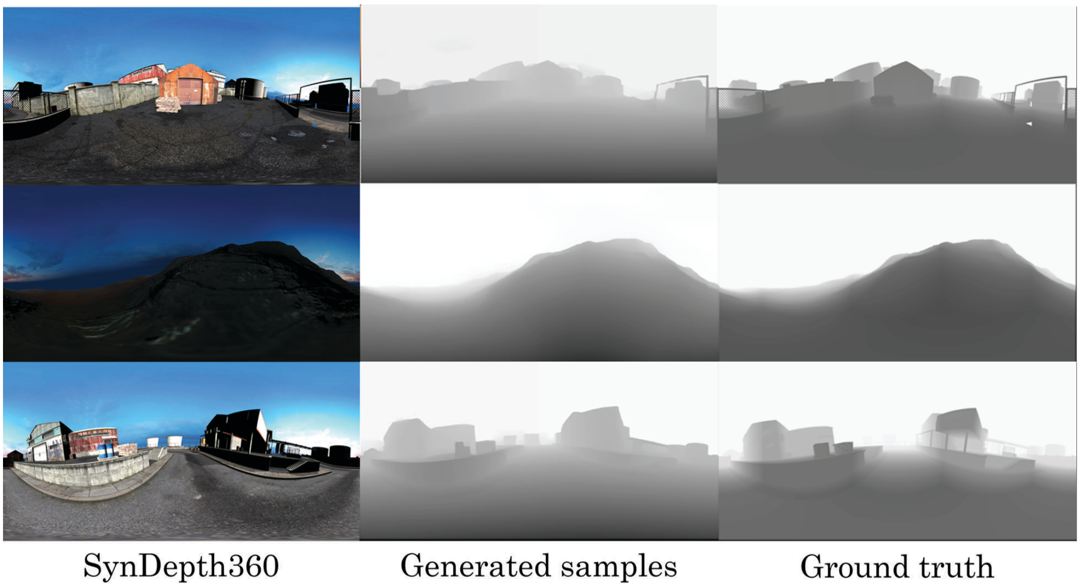


FIGURE 5.6: Evaluating the data generation method with the SynDepth360 dataset. The left column is the synthetic samples generated from 3D models, the middle column is the generated data using the test-time training, and the right column is the rendered ground truth.

5.5 Experimental Evaluation

In this section, we first evaluate the quality of the Depth360 dataset against the synthetic dataset SynDepth360 we generated for benchmark purposes. We then evaluate the proposed method against other state-of-the-art single-view depth estimation methods both qualitatively and quantitatively by training on the Depth360 dataset. We further conduct an ablative study to validate the effectiveness of our network design.

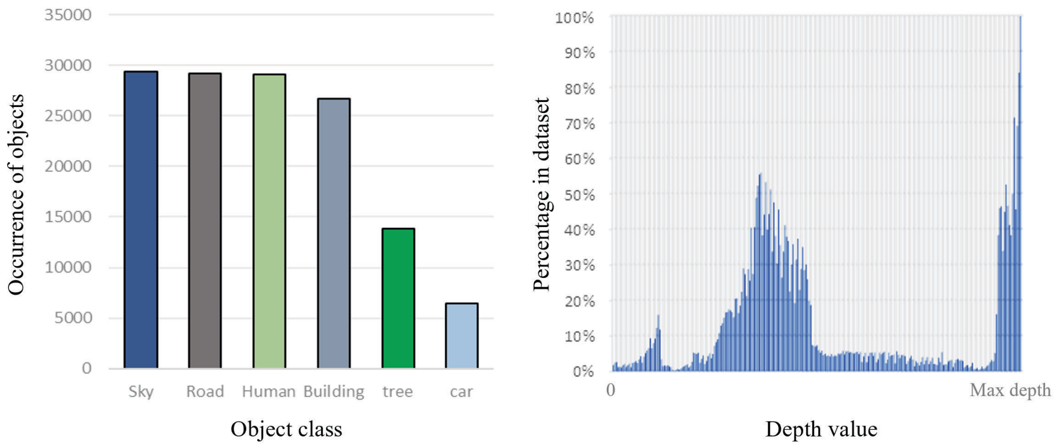


Figure 5.7: Quantitative evaluations of the dataset. The left figure shows occurrence of top objects in the proposed dataset. The power-law shaped distribution indicates the most predominant background 'sky', 'road' and 'building', while the most occurred foreground objects are 'human' and 'tree'. The right figure shows the distribution of depth values. the leftmost and the rightmost peak manifests that internet videos often use a hand-held capturing fashion with outdoor settings.

Both datasets and the source code are available to the community to encourage future research.

5.5.1 Dataset Evaluations

To verify the accuracy of generating depth from internet videos with the proposed test-time training method, we use the synthesized samples from the benchmarking dataset SynDepth360 to evaluate against the ground truth depth maps qualitatively and quantitatively. We further provide a distribution analysis of Depth360 and a quantitative evaluation against existing omnidirectional datasets.

Qualitatively, samples generated from the synthetic sequences are shown in Figure 5.6. As most fine details are faithfully reconstructed, and in-scene objects show clear boundaries, we validate that the generated dataset is useful for further single view estimation training. It is worth noting that since the ground truth is rendered with absolute distances with a range of infinity, a slight depth scale discrepancy is presented. Quantitatively, our method achieves the accuracy of 49.5%, 60.6%, 70.8% for $d1$, $d2$, and $d3$ using the metrics for depth prediction from the literature [124] [75], significantly surpassing naive generation methods. They include omnidirectional models trained with 360° indoor samples, with the highest accuracy of 14.6%, 22.0%, 25.5%, and perspective models trained with mixed samples, with the highest accuracy of 41.7%, 49.3%, 59.2%.

Compared to current state-of-the-art omnidirectional datasets that facilitate single-view depth estimation (Table 5.1), the proposed dataset achieves higher resolution, larger size, and more diverse outdoor settings. Although the size and quality of model-based datasets (e.g. SceneNet [131]) can be improved upon using different rendering

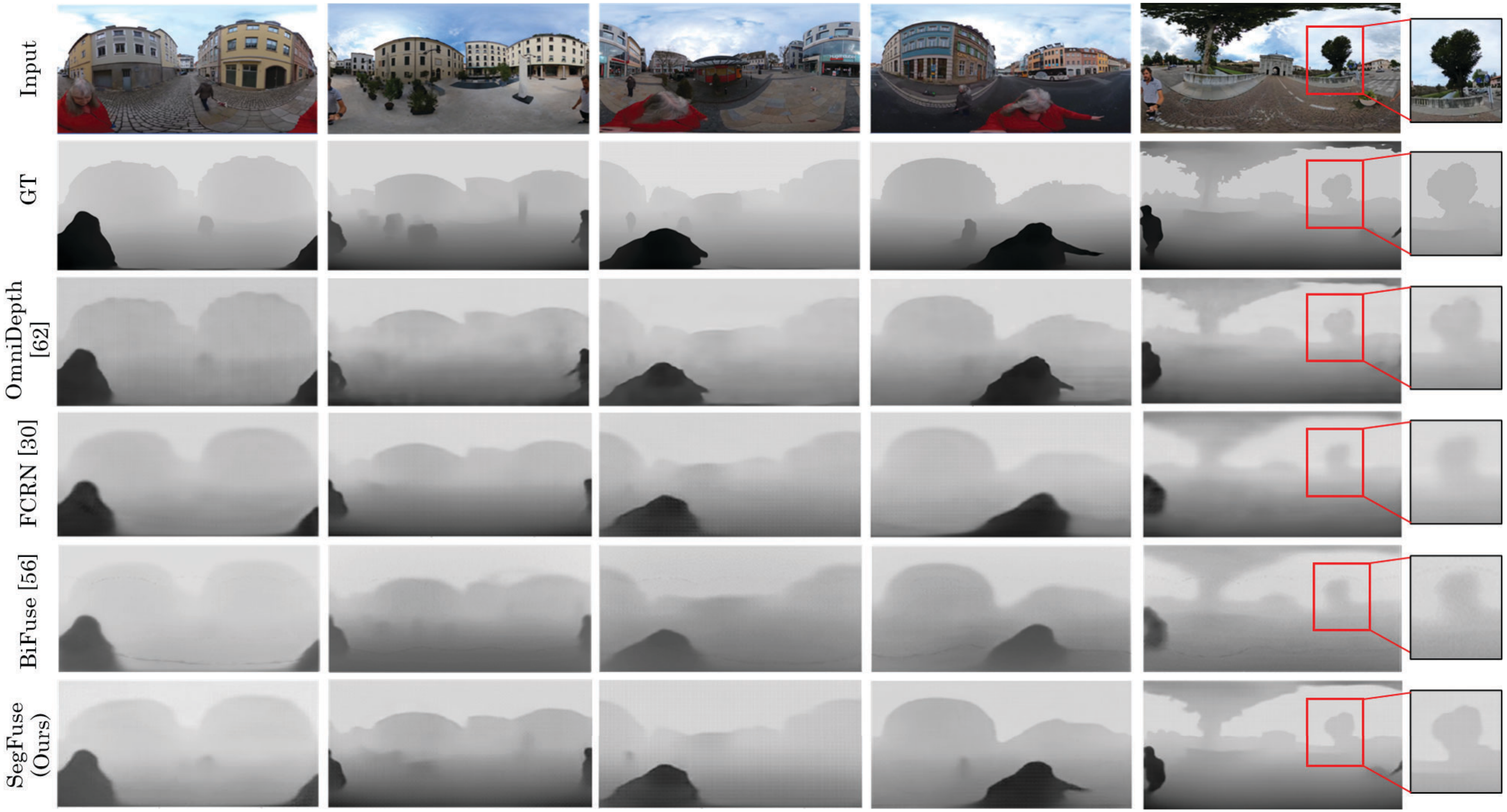


FIGURE 5.8: Qualitative comparison with the state-of-the-art methods. Our method generates globally consistent estimation when compared to [117], and sharper results at local regions compared to the other methods.

methods, our dataset maintains the advantages of varied domains and easy extension with a larger video collection.

TABLE 5.1: Comparison between state-of-the-art 360° datasets.

Name	Type	setting	resolution	# images
PanoSunCG [152]	synthetic	indoor	0.13Mpx	25000
SceneNet [131]	synthetic	indoor	0.13Mpx	25000*
Stanford 2D-3D [120]	real	indoor	0.13Mpx	25000*
Matterport3D [119]	real	indoor	0.13Mpx	25000*
Proposed dataset	**real**	**outdoor**	**1.03Mpx**	**30000**

From the depth value distribution analysis (Figure 5.7, right) of the Depth360 dataset, we can observe three major peaks. The leftmost peak manifests the hand-held 360° camera user, while the rightmost peak shows a common sky background. The normal distribution in the middle presents other objects in the scene such as buildings and trees. This can be validated through the occurrence of top objects (Figure 5.7, left) in the proposed dataset. From the power-law shaped distribution, we can observe that the most predominant background objects are ’sky’, ’road’ and ’building’, and most occurred foreground objects are ’human’.

5.5.2 SegFuse Evaluations

Implementation Details

We implement the SegFuse network with the Pytorch framework [134] 1.4 and train models with a configuration of Nvidia RTX 2080Ti GPU, i7-7800X CPU, and 32GB RAM. We randomly split samples into training and validation datasets from the dataset with a ratio of 90% and 10%. During training, we use Adam optimizer [135], a learning rate of 3e-4 and, a batch size of 1. The peripheral branch uses Xavier initialization [136] while the foveal branch initializes with ImageNet pretrained weights. The same metrics from the literature are used for quantitative evaluation [75]. Our current implementation takes smaller batches due to graphics memory restrictions. We expect a more stable training process with a better hardware configuration, with potential improvements such as batch normalization. Our method costs approximately 100ms with the same configuration to predict a single equirectangular image, favoring interactive frame-rate for applications.

Qualitative Results

We present qualitative results of single-view depth estimation from omnidirectional images with different methods including Omnidepth [117] and FCRN [73]. As we can observe in Figure 5.8, when tested on unseen equirectangular images with challenging outdoor settings, our method generates better sharper estimation while maintaining a smooth global depth map prediction. This can be attributed to the foveal branch

TABLE 5.2: Quantitative results against other single-view omnidirectional depth estimation methods.

Method	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
Real domain (all models are trained with Depth360 dataset)							
OmniDepth [117]	0.3375	0.1967	4.3049	0.7836	80.16%	89.78%	91.93%
BiFuse [74]	0.3596	0.8615	5.0725	0.8316	40.13%	59.17%	67.92%
FCRN [73]	0.2384	0.4057	4.8599	0.7839	80.59%	90.42%	93.88%
SegFuse (Ours)	**0.2275**	**0.1588**	**4.0442**	**0.7777**	**82.26%**	**91.35%**	**94.22%**
Synthetic domain (all models are trained with SynDepth360 dataset)							
OmniDepth [117]	0.1171	0.1753	0.3819	0.0844	91.30%	94.04%	96.35%
BiFuse [74]	0.1473	0.1978	0.4619	0.1012	75.12%	81.77%	84.69%
FCRN [73]	0.1017	0.1525	0.3771	0.0776	93.48%	95.89%	97.79%
SegFuse (Ours)	**0.0973**	**0.1510**	**0.3209**	**0.0734**	**94.74%**	**96.61%**	**98.03%**

that improves local details. More qualitative results with diverse settings are included in the supplementary material.

As we argued that for challenging outdoor cases with wildly changing ranges and unsymmetrical distributions, directly using cubemap projection to regress depth maps for each face and fusing with equirectangular estimation afterward shows sub-optimal performance. Such inconsistencies at face boundaries are presented in Figure 5.10 (BiFuse [74]). We offer a detailed convergence analysis of the proposed method against BiFuse that uses cubemap projection to fuse depth for outdoor samples. The results can be observed in Figure 5.9. We compare the performance via inverse Huber loss when both networks are trained with the Depth360 dataset. We show that our method converges much faster (the blue line) with the help of latent information shared by the pretrained semantic segmentation weight, while the cubemap-based depth regression struggles to effectively merge faces and learn outdoor settings (the orange line). As we can see in the bottom half of Figure 5.10, the middle figure shows the result of regressed depth from the cubemap branch of BiFuse, and the right figure shows the final fused output of BiFuse. Clear boundaries between faces result in deteriorated fused output when compared to a single-branch architecture such as FCRN.

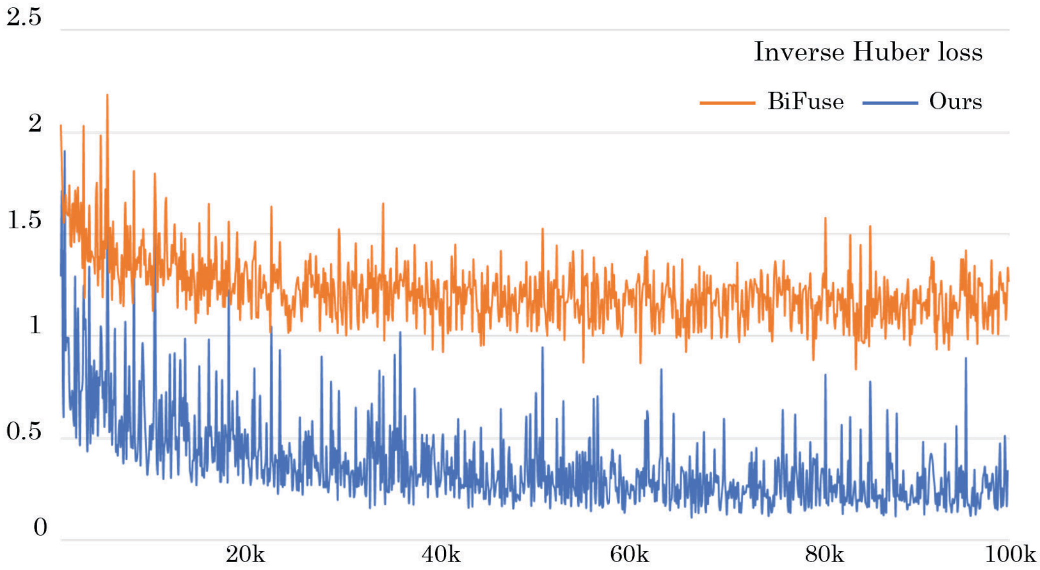


Figure 5.9: To validate our network design, we evaluate against BiFuse [74], a cubemap-based depth fusion method. When trained with outdoor samples, SegFuse converges much faster (the blue line) while the cubemap-based depth regression struggles to effectively merge faces and learn outdoor settings (the orange line).

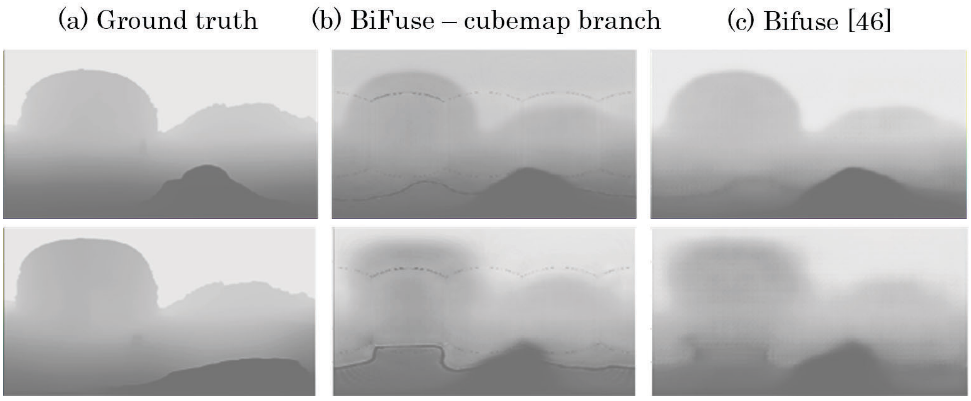


FIGURE 5.10: The result of regressed depth from the cubemap branch of BiFuse and the final fused output of BiFuse.

Quantitative Results

Adopting the metrics for depth prediction from the literature [124] [75], Table 5.2 presents the quantitative evaluation of our method against the state-of-the-art single-view omnidirectional depth estimation methods in both real-world and synthetic domains. We can observe that SegFuse successfully captures the features of the outdoor dataset when compared to other methods. Overall, our method shows favorable results against FCRN, Omnidepth, and BiFuse. We further evaluate the performance in indoor settings by training networks with 3D60 dataset [117], which consists of SunCG [152], SceneNet [131], Stanford2D3D [120], Matterport3D [119]. We benchmark against the ground truth depth with filled-in values for invalid pixels like FCRN [73]. Table 5.3 shows a comparable accuracy of SegFuse with the state-of-the-art designed for indoor predictions [74], and better performance against other omnidirectional methods.

TABLE 5.3: Qualitative results of indoor-only settings.

Method	*RMSE*	*RMSElog*	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
OmniDepth [117]	0.6364	0.1358	77.30%	91.24%	97.21%
BiFuse [74]	**0.5639**	0.1007	**85.12%**	93.38%	**98.16%**
FCRN [73]	0.6429	0.1286	78.08%	92.09%	97.33%
SegFuse (Ours)	0.5729	**0.0986**	84.38%	**94.34%**	98.07%

Ablation Studies

Finally, we perform an ablation analysis between the SegFuse and learning without the foveal branch. We use the same training settings with and without fusing the foveal branch with the peripheral branch, and the quantitative evaluation is shown in Table 5.4. In addition to better accuracy, we also find that the converging speed when training with SegFuse is almost 2x faster at the beginning thanks to the pre-trained segmentation weight. This shows the additional benefit of using a multi-task architecture to solve the depth estimation problem. We then compare the accuracy of the SegFuse network with different numbers of fused layers at the decoder. We find that while connecting three and four layers both achieve close performance, using three fusion blocks usually provides a slightly more stable training process and improved efficiency. A quantitative ablation study is presented in Table 5.5.

Table 5.4: Ablation results of the foveal branch.

Method	$RMSE$	$RMSElog$	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Peripheral only	4.9281	0.8979	57.79%	74.20%	78.11%
SegFuse	**4.0442**	**0.7777**	**82.26%**	**91.35%**	**94.22%**

Table 5.5: Ablation results of connected layers.

	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
1	4.9297	0.8933	70.13%	79.17%	87.92%
2	4.3782	0.8126	77.85%	86.83%	92.50%
3	4.0442	**0.7777**	**82.26%**	91.35%	**94.22%**
4	**4.0168**	0.7994	81.67%	**91.75%**	93.82%

5.5.3 Depth-based Applications

High-quality depth estimation from a single 360° image enables a wide range of interesting applications. We take visual effects as an example to showcase the strength of our method in virtual reality. We first use the proposed method to estimate a high-quality dense depth map from an input omnidirectional RGB image. We then project per-pixel depth values onto a 3D sphere to render a pseudo-reconstructed scene with mesh. This facilitates augmenting the original scenes with effects such as volumetric snowing and flooding. A preview is shown in Figure 5.11, and full video results are included in the supplementary material.

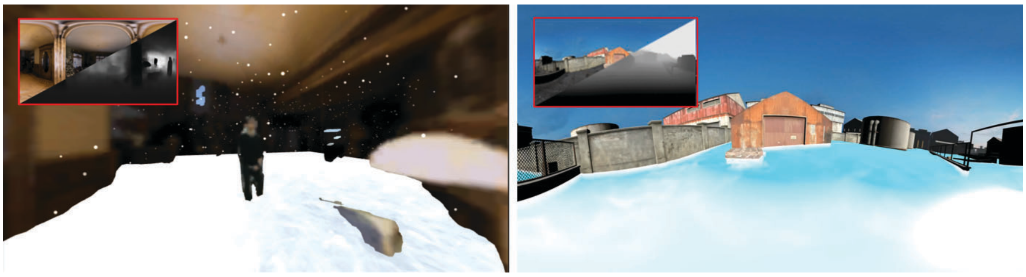


Figure 5.11: An example of depth-enabled applications. By estimating corresponding depth maps from an input 360° image, we add volumetric effects to the scene such as snowing (left) and flooding (right).

5.6 Failure Case and Future Work

Although our data generation method can be applied to larger-scale collections to extend the size of datasets, it shows several limitations. First, online omnidirectional videos present unbalanced distributions, favoring specific scenarios (e.g. urban street views). Second, when establishing the baseline, SfM and MVS methods show suboptimal results when there are texture-less surfaces or reflective materials in the scene. Scenes with excessive dynamic foreground objects or strong motions are problematic for a pseudo-stereo system to acquire accurate geometric consistency. Future work could alleviate these problems by adopting improved SfM algorithms and scaling to a larger variety of input collections. For depth estimation, the current implementation only accepts smaller batch sizes due to hardware limitations. We expect to improve the efficiency of the network and enable more stable training with better normalization methods.

5.7 Conclusion of the Chapter

In this chapter, we first propose to utilize the unlimited source of data, 360° videos from the internet, to overcome the scarcity of a general omnidirectional dataset. We propose geometric and temporal constraints that are unique to 360° videos and use test-time training to generate high-quality depth maps. To fully benefit from our dataset, Depth360, we propose an end-to-end two-branch multitask network, SegFuse, that mimics human vision to estimate depth from a single omnidirectional image. With the peripheral vision to perceive the depth of the scene and foveal vision to distinguish between objects, our network shows favorable results against state-of-the-art methods.

Chapter 6

Employing Scene Understanding in Immersive Mixed Reality

In this chapter, we try to explore practical applications of different scene understanding algorithms in the context of mixed reality in order to further provide users with improved capabilities and immersive experiences. We put forward two different applications: editing foreground objects of interest in pre-captured omnidirectional videos and consistent artistic stylization for pre-captured videos, to demonstrate the benefit and potential of alike studies. The first section concludes with the visual aspect of a previous work that has been published [13] with extended methodologies to further extend its contribution. The second section is rather short in length. Being inspired by the undergraduate work [178], we investigate its feasibility under the mixed reality context by artistically stylizing pre-captured videos. We expect this application-focused chapter can shed more light on more practical employments of newer scene understanding algorithms in the modern virtual/augmented reality era.

6.1 Editing Foreground Objects in Pre-captured 360° Videos

6.1.1 Introduction

With the increasing popularity of mixed reality and high-fidelity commercial 360° cameras being widely accessible, there is a growing number of omnidirectional media being available[116][179]. Thanks to its potential of providing the visual of the entire scene for an immersive user experience, it has been attracting attention in both academic and industrial contexts [24]. However, while the major strength of the 360° format of media is a larger field-of-view, bringing the capability to capture the surroundings at once, it is usually difficult for users to frame the scene as they wanted [13], and put the focus solely on the important objects when compared to traditional perspective formats with limited field-of-view and flexible depth-of-field.

Considering the rising demand for editing algorithms for omnidirectional content, we investigate some basic editing capabilities, namely adding and erasing an undesirable object from pre-captured footage in this chapter. We put forward several non-trivial issues and their respective solutions and propose a complete framework for editing foreground objects in 360° videos.

For traditional perspective videos, there are abundant tools to help the user to edit their pre-captured footage during the post-processing stage, including adding post effects with natural appearances with harmonization [180] or removing unwanted objects with learning-based images inpainting [181]. However, when it comes to 360° contents, the distortion to project spherical information onto 2-dimensional planes during storage usually makes some basic simple editing capabilities difficult to achieve [117], such as adding and erasing an undesirable object from pre-captured footage. Moreover, due to the limited available database to effectively drive the existing learning-based approaches [114], the results are often sub-optimal when directly applying perspective-based methods to omnidirectional contents.

To effectively add and erase foreground objects from pre-captured 360° footage, we present a pipeline that first employs scene understanding to track the object of interest using equirectangular projection, followed by cubemap projections to divide the omnidirectional scene into smaller patches with less distortion. We then realize adding additional objects through frame-wise image harmonization [180] and erasing unwanted foreground objects with image inpainting [181]. We provide a proof-of-concept implementation for editing foreground objects in pre-captured 360° videos. A qualitative evaluation verifies the practicality of the proposed method, and a user study further confirms the improvement of the user's immersive experience of edited footage.

6.1.2 Related Work

Omnidirectional Videos

With the progress in mixed reality hardware and software, pre-captured 360° videos enable users to look around their entire surroundings with 3 degree-of-freedom instead of a fixed perspective during video playback. Over recent years, more accessible commercial 360° cameras (e.g., Insta360) allow normal consumers to conveniently capture 360° footage with high fidelity and little to no post-processing. Online video sharing platforms (e.g. YouTube) further encourage the prosperity of omnidirectional content.

However, since 360° cameras capture the visual information of the entire spherical environment, it requires different compression and projection to store 3-dimensional information onto 2-dimensional planes. Equirectangular projection is the most widely adopted format to store pre-captured 360° footage, which has a bipolar pattern of distortion. Due to stronger distortions towards the polar regions, directly applying perspective-based methods usually lead to incorrect assumptions and severe visual artifacts. Other projections include cubemap projection [114] and equiangular cubemap projection [182]. While the later projections have a consistent density of information across different regions of the frame, inconsistency along boundaries are issue need to be considered.

Image inpainting

Image inpainting is a research topic proposed to visually reconstruct the blank area with convincing appearances based on the global context after removing a certain target from the original image. It can be applied to different areas including video editing [183] and repairing vintage films [184]. Traditional approaches are usually exemplar-based [185], differential equation-based [186] and patch-based [187]. After Pathak et al. [188] proposed to adopt Convolutional Neural Networks into image inpainting tasks, learning-based methods such as using Generative Adversarial Networks [189] have shown great performance to generate plausible output for single image inpainting.

When it comes to video inpainting, image-based methods often yield inconsistent results due to the additional temporal constraints across the frames. Scenes with complex self-motions and dynamic environments severely suffer from flickering and subpar inpainting results. Video-based inpainting that is either patch-based or object-based shows respective weaknesses of being vulnerable to scale changes [190] and strong self-motions [191]. Recent learning-based methods show improved performance by adopting flow information to generate plausible pixels with global features across neighboring frames. However, they are not designed for omnidirectional videos with distortions in mind.

Image Harmonization

Image composition is a basic task for image editing and content creation. After placing the desired object onto a new background, it usually requires extensive labor of the user to manually match the appearance (color, white balance, lighting, etc.) of the composited object with the background to ensure the realism of the final output. To achieve quality image harmonization, traditional methods use hand-crafted features [193] to adjust the statistics of foreground objects and the background without considering the high-level features of both inputs. Recent learning-based methods use global perceptual context to evaluate the realism of the output [194]. Other methods further incorporate semantic segmentation [195] and attention modules [180] to perform end-to-end harmonization with high quality.

However, video harmonization is an under-investigated topic with very limited attention. Considering the frame-by-frame adoption of single image-based harmonization will inevitably result in flickering due to randomized initialization and different regional features [196]. While there are temporal losses proposed in other deep learning-based video methods [197], the various relationship between foreground objects and the background is computationally complex to achieve coherent results. Additionally, the training data of paired background-only and composited results is difficult to acquire, and synthesized databases suffer from domain differences in generating satisfying results. Considering that video harmonization is already limited

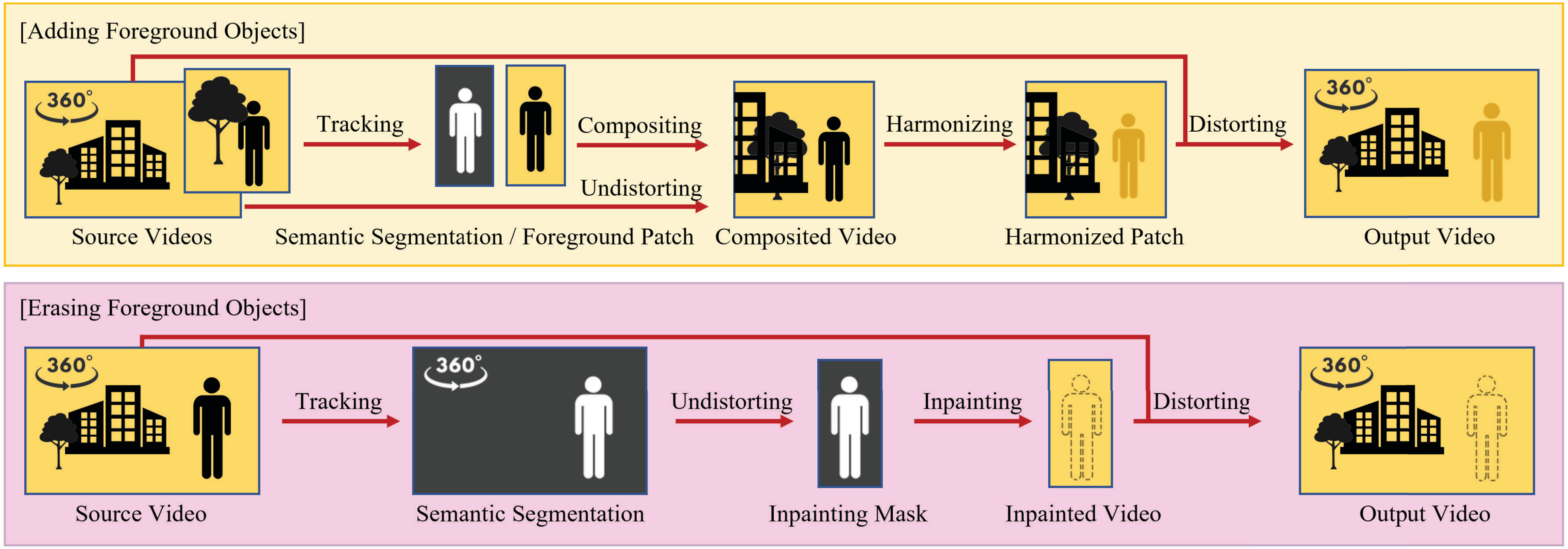


FIGURE 6.1: The workflow of the proposed foreground objects editing framework. For both adding and erasing foreground objects, we first manually specify a target object with a bounding box. We then use SiamMask [192] to extract the semantic segmentation of the target across the frames. Before compositing foregrounds and inpainting the target region, we first reproject the equirectangular input image into cubemap faces to crop out the area of interest. To add the foreground object to the omnidirectional video, we composite the patch to the target cropped face according to the segmentation result. The framework then performs image harmonization inside the mask of the foreground objects. To erase the foreground object, we perform the video inpainting within the semantic segmentation. Finally, we composite the cube faces together and reproject the output into the equirectangular format.

for traditional perspective inputs, it is more challenging to add foreground objects to 360° videos.

6.1.3 The Foreground Editing Framework

We describe the proposed foreground objects editing framework for pre-captured omnidirectional videos. For adding new objects from another pre-captured footage into the target omnidirectional video, the input consists of a source perspective footage and a target omnidirectional footage. For erasing existing foreground objects from pre-captured omnidirectional footage, the input consisted of the footage and user-specified (through a drag-and-release bounding box) foreground objects. With an accurate semantic segmentation algorithm that is designed for omnidirectional images, we obtain the desirable foreground objects across the frames after solving the distortion. We then apply the state-of-the-art video inpainting [181] and image harmonization [180] methods to achieve natural edited results. The workflow is demonstrated in Figure 6.1.

Erasing Foreground Objects

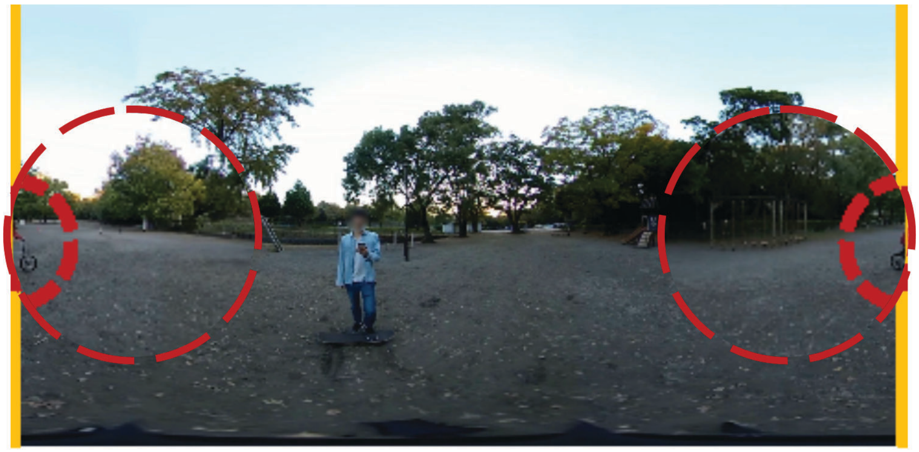


FIGURE 6.2: An example of discontinuity when an object is moving across the boundary when using a 2-dimensional projection.

For erasing foreground objects from pre-captured omnidirectional videos, we first determine the object of interest through a user-specified bounding box followed by re-projecting the patch to a 2-dimensional plane. Considering that directly applying the perspective-based image inpainting methods frame by frame would cause two problems: discontinuity and distortion, we explain how the proposed method tackles the challenges respectively. For spherical input, there are no distinct boundaries anywhere on the surface. However, when we try to adopt a traditional neural network that is designed for 2-dimensional images, we inherently project 3-dimensional information to a 2-dimensional plane. This inevitably leads to two boundaries either at the top and the bottom edges or at the left and the right edges. In case a target object is moving horizontally or vertically across the frame, it will be “exiting” and “re-entering” disconnected boundaries (an example is shown in Figure 6.2). This problem of discontinuity needs to be solved to apply existing inpainting approaches.

In this work, instead of tracking the target object across the frames, we fix the object at the center of the frame by rotating the entire equirectangular projection to solve the discontinuity issue. After the object of interest is specified with a bounding box, we determine the semantic segmentation of the object with SiamMask [192], and then calculate the center of mass. The process is shown in Figure 6.3.

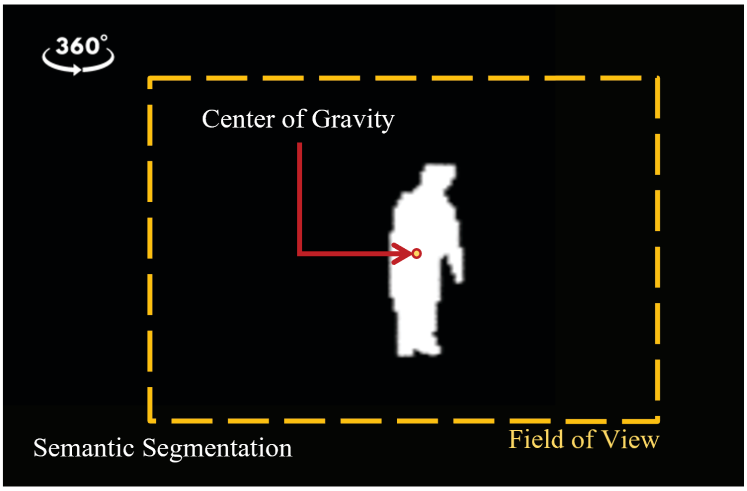


FIGURE 6.3: An example of calculating the center of gravity using the predicted semantic segmentation.

The proposed framework rotates subsequent frames in a incremental and iterative manner to predict semantic segmentation so that the discontinuity issue will not occur in the successive frame. We rotate the frame F_t at the timestamp t according to the following equation:

$$F_{t+1} = \mathbf{R}(F_t, \Delta_t). \tag{6.1}$$

To determine the amount of rotation,

$$\Delta_t = \frac{H}{2} - \tilde{c}_t, \tag{6.2}$$

where

$$c_t = -\sum_{t'=0}^{t} \Delta_{t'}. \tag{6.3}$$

$\tilde{c}_t$ is the rotated center of gravity of the semantic segmentation for the frame F_t, while c_t is the center of gravity for non-rotated frames. H and V are dimensions of the input equirectangular frame.

We then solve the problem of distortion by projecting input equirectangular frames into 2-dimensional images with reduced field-of-view. With adjusted frames and respective semantic segmentation of all frames, we determine the area to crop according to the following equation:

$$\theta = \frac{h}{H} \times 2\pi, \tag{6.4}$$

$$\phi = \frac{v}{V} \times \pi, \tag{6.5}$$

where θ (elevation) and ϕ (azimuth) determine the field of view of the reprojected patches, h and v are the maximum values of the semantic segmentation dimensions for all frames. We process each patch through deep video inpainting [181] and acquire the inpainted frames. Finally, we reproject the results back into the equirectangular format in the reversed order.

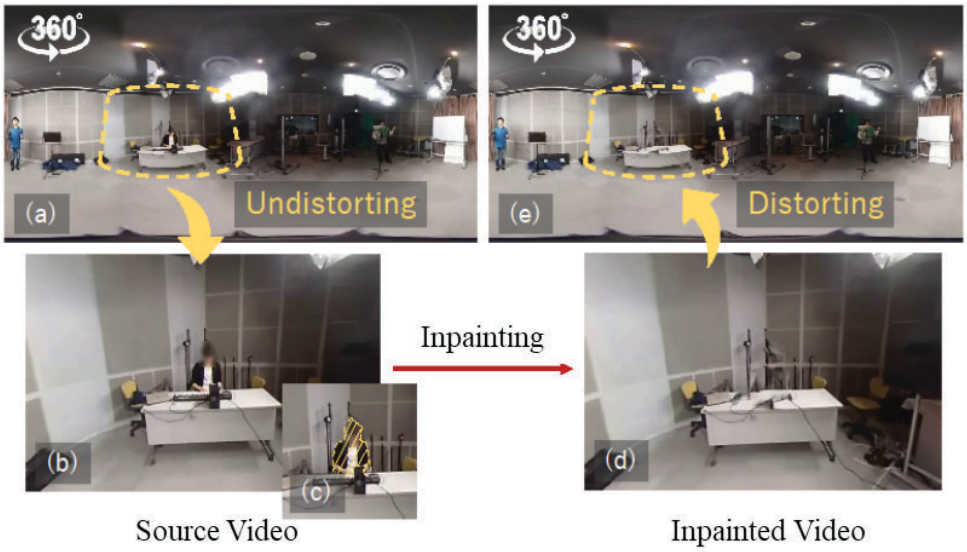


FIGURE 6.4: The process of inpainting the target area with distortion by reprojecting between the equirectangular and cubemap projection.

Adding Foreground Objects

In this section, we explain how we add existing foreground objects from a pre-captured omnidirectional or perspective video to the target omnidirectional video sequence.

We first acquire the semantic segmentation of the target object through SiamMask [192] in a similar fashion to the erasing process. We then determine the area of interest by asking the user to manually assign a coordinate that functions as the center of gravity to further augment the foreground object. During this stage, we repurpose the semantic segmentation as the harmonization mask to facilitate the harmonization process. To prevent distortion, we reproject the equirectangular input source video to cubemap projection in a similar fashion to the erasing process:

$$\theta = \frac{m}{H} \times 2\pi, \phi = \frac{n}{V} \times \pi, \tag{6.6}$$

It is worth mentioning that when we directly projecting equirectangular and cubemap representations back and forth using the following transformation:

$$\theta_f = arctan(\frac{q_x}{q_z)}, \tag{6.7}$$

$$\phi_f = arctan(\frac{q_y}{q}), \tag{6.8}$$

$$q = \mathbf{R}_f \cdot p, \tag{6.9}$$

where pixel p with the coordinate (x, y, z) on the 2-dimensional plane f base on the assumption that $0 \leq x, y \leq w - 1$ and $z = w/2$, there will be pixels with unknown value that cannot be directly one-to-one mapped from integer coordinates. Therefore, we use inverse mapping to approximate the value of the unknown pixel on the cube faces based on its corresponding pixel on the equirectangular image.

6.1.4 Experimental Evaluation

In this section, we report the experiment and the user study that we conducted to verify the effectiveness of the proposed foreground object editing framework for omnidirectional videos. We captured multiple omnidirectional video sequences with different conditions to qualitatively evaluate our method. For the erasing process, the conditions include two outdoor scenes and one indoor scene. For the adding process, the conditions include one outdoor scene and one indoor scene.

During the experiment, all sequences are captured with a commercial hand-held 360-degree camera (i.e., Ricoh Theta V). To process the pre-captured videos with the proposed framework, we conducted the experiment on a desktop computer. The hardware configuration is listed below, and the qualitative results can be observed in Figure 6.5

TABLE 6.1: Experimental configuration of the hardware

Component	Details
CPU	Intel i9-9900K 5.0GHz
GPU	Nvidia GeForce GTX 2080Ti
Omnidirectional camera	Ricoh Theta V (30fps@2160 × 1080)
Playback device	HTC Vive Pro (90fps@1440 × 1600 per eye, 110° FOV)

In the user study, we try to verify if the processed footage of the proposed foreground object editing framework is satisfactory and can bring an immersive experience to users. 17 subjects (with average ages of 23.1 +- 1.5) participated in the study. 5 participants have no virtual reality/augmented reality experience, 10 participants have a moderate amount of experience with mixed reality, and 2 participants are familiar with video playback in mixed reality. During the study, users are assigned with

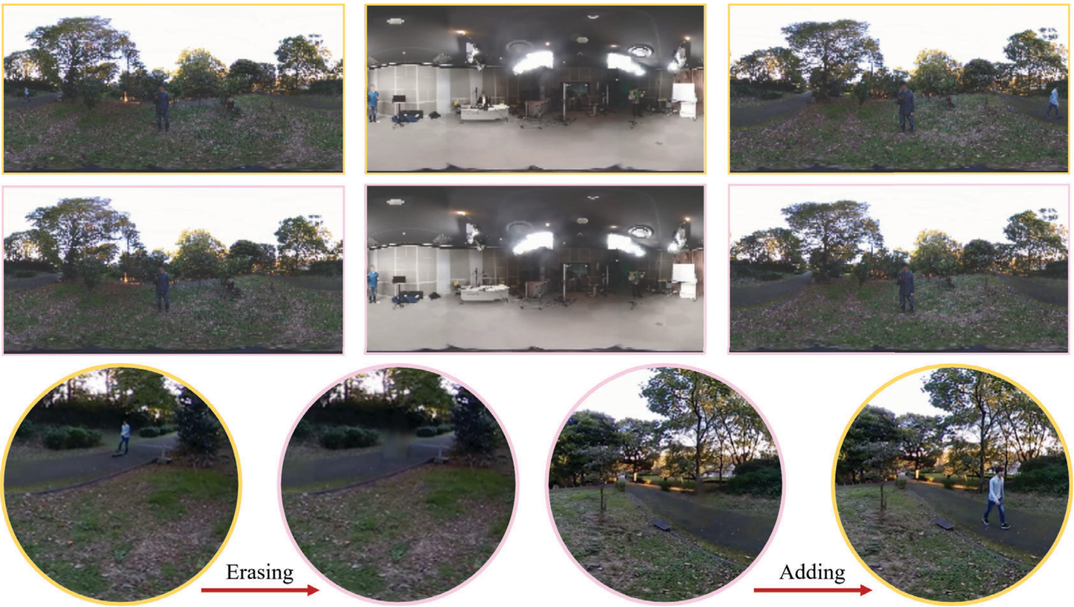


FIGURE 6.5: The result of the proposed foreground object editing framework. The first row is input and the second row is the result. The last row is a zoomed in view for better visualization.

randomized order of processed videos to prevent bias. For each scene, the user follows instructions to start the playback or rewatch the result for arbitrary times until they have made an established decision regarding the current scene. According to the feedback of the questionnaire in addition to the free comments in the subsequent semi-structured discussions, we confirmed a positive experience with the implemented framework. This can be observed in Figure 6.6

For question 1, whether the target object was visually erased from the original scene, users generally had positive feedback that the visuals were pleasing, especially for outdoor footage. This can be attributed to a smaller portion of the target object when compared to the entire scene, and also more complex textures are harder for users to notice the artifacts that are introduced during the editing phase. For question 2, whether the edited footage is natural, we observed a lower score when compared to question 3, whether the users were satisfied with the final output. According to the comments during the discussion, this is mainly due to that while the object is visually erased, the ambiance of the target object still remains, breaking the immersive experience to a certain degree. While the importance of multi-modal editing is already confirmed [13], considering that auditory processing is beyond the scope of this thesis, we will omit the detailed discussions. At the current stage, while users are relatively satisfied with the editing result, this is a result of a lower-than-perfect resolution for both capturing process and the capacity of the neural networks used for inpainting and harmonization. Considering that the resolution of commercial mixed reality devices is far higher than mainstream neural network capacity (usually up to full HD), efficient networks to facilitate high-fidelity applications are both important and challenging in the near future.

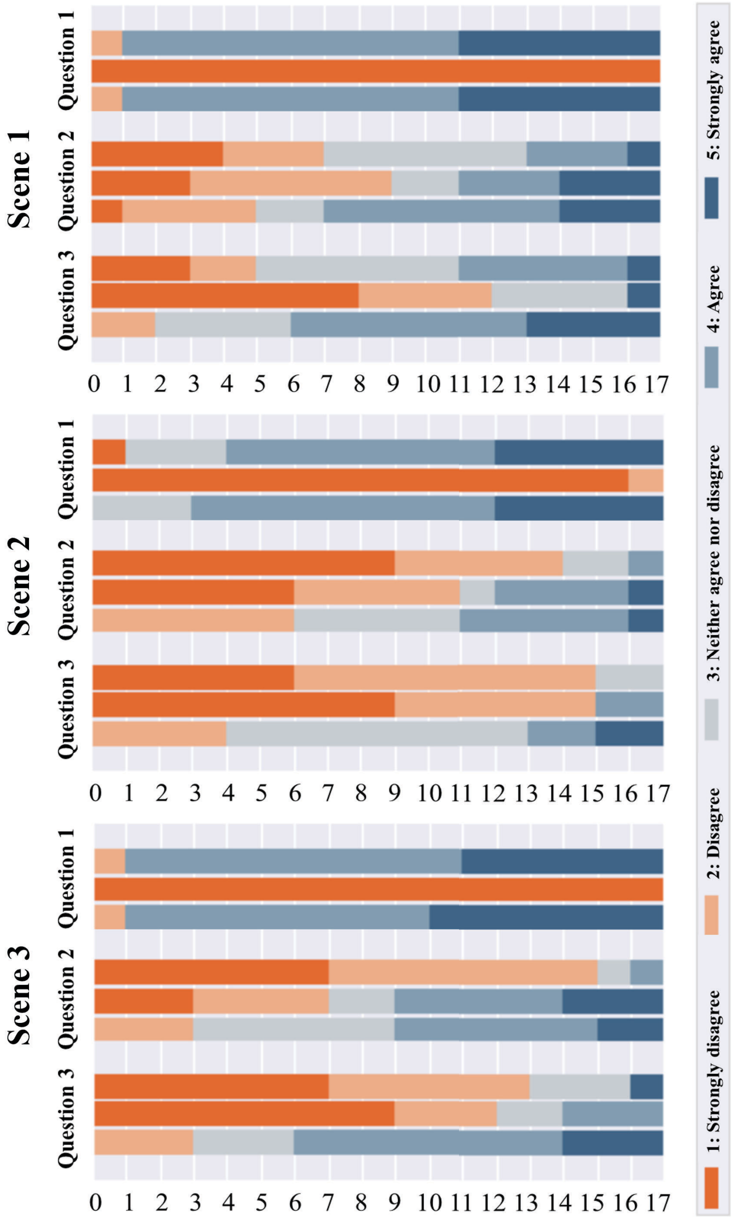


FIGURE 6.6: The result of the user study. For each scene, the users answer a Likert-scale survey for three questions regarding the editing result.

6.2 Consistent and Foreground-aware Neural Style Transfer for 360° Videos

6.2.1 Introduction

Neural style transfer is a research field that studies using deep learning algorithms to artistically stylize photo-realistic images with painting-like aesthetics. Although art creation was believed to be a complex and challenging task for the machine to understand and accomplish, Gatys et al. [198] decompose the process into statistically solvable components of "style" and "content", which can be effectively learned by neural networks individually. This leads to a whole new direction of research and encourages abundant following work. While the initial idea is to iteratively optimize the input of randomized noise to match the features of the target style image by calculating the Gram matrix, later work [42] greatly improves the capabilities of the original method, including real-time performance [42], cross-frame consistency [199] [128], guided transfer with user input [200], content-aware transfer [201]. In the context of mixed reality, previous methods [128] propose a naïve method to directly transfer individual components of cubemap projection for omnidirectional input.

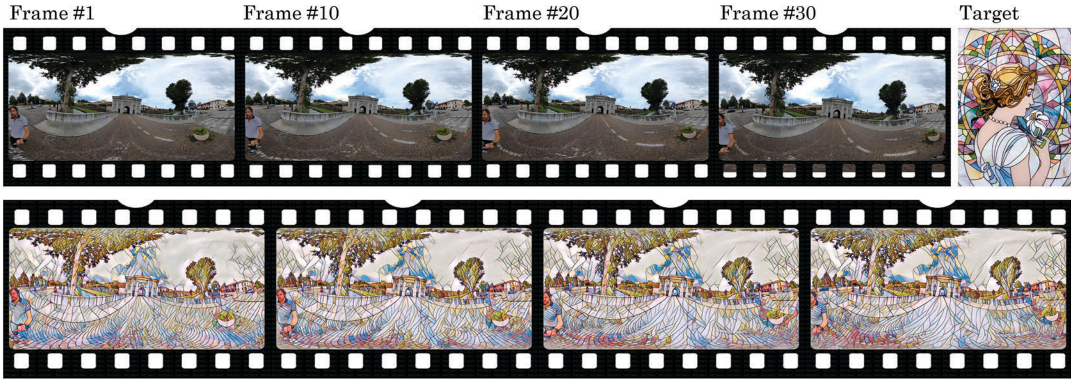


FIGURE 6.7: Consistent and foreground-aware stylized results using the proposed method.

In this work, I present a method that can achieve results that are both spatially and temporally consistent for omnidirectional videos. Given the input of an omnidirectional video and a target style, the system calculates the omnidirectional-aware optical flow between consecutive frames. Because each frame is stylized individually from different noises, this enforces the temporal consistency by calculating the distance between reprojected frames and minimizing the deviation. The system further achieves spatially consistency by utilizing semantic segmentation for omnidirectional videos to understand the context of the input. After separating the foreground and the background, it stylizes the inpainted background and the foreground patch respectively. Finally, by adopting a padded cubemap projection mentioned in the research of Chapter 5, the output of the method achieves visually satisfying results for omnidirectional videos. A preview of the experimental result is shown in Figure 6.7. It shows the great potential of scene understanding in practical mixed reality applications.

6.2.2 Related Work

Neural style transfer for images. The ability to transfer the artistic style from a reference image to an input image is an interesting idea to achieve non-photorealistic rendering. Before learning-based approaches became popular, patch-based and non-parametric methods were usually widely used to achieve texture transfer for non-photorealistic rendering. Image quilting [202] and analogies [203] are used to learn the mapping between the texture of two images and synthesize the result that resembles the texture of the style image. Edge orientation is later incorporated to ensure a natural gradient [204]. To accelerate the process, Markov random field [205] is used to improve the efficiency of searching for candidate pixels.

Inspired by the initial work of Gatys et al. [206], artistic style transfer has become a popular deep learning topic for its capability to achieve art creation, which was believed difficult for the machine to accomplish. By iteratively optimizing the similarity of high-level features in artistic style and image content, it has shown impressive results for artistic style transfer tasks. To reduce the lengthy optimization process, a feed-forward network structure is proposed [42] along with a perceptual loss to facilitate real-time applications. Using instance normalization instead of batch normalization in convolutional networks [207] demonstrates a greatly improved separation between high-level features. Conditional instance normalization [208] [209] further brings the capability of learning several art styles simultaneously. Generative adversarial networks are also a popular choice for image-to-image translation [210].

Neural style transfer for videos. For video artistic style transfer, because each frame is usually initialized from different noise, the appearance of output can be drastically different from each other, causing inconsistency across frames and leading to unsatisfying stylization. Anderson et al. [199] propose to initialize per-frame optimization based on the previous frame to solve the flicker issue and improve the consistency of stylized results. Likewise, a long-term temporal loss is introduced by Ruder et al. [211] to solve false stylization before and after occlusions. Later, Ruder et al. [128] further incorporate feed-forward networks to achieve a faster stylization speed and extend the capability of style transfer to 360-degree input by directly solving each face of a cubemap projection independently. Although it shows the potential of artistic style transfer in the context of mixed reality, the performance of the method is quite limited with its trivial addition over its perspective counterpart.

6.2.3 Consistent Artistic Style Transfer

Overview. In this section, I will describe the proposed neural style transfer method for omnidirectional videos which is temporally consistent and foreground aware. To achieve a better understanding of context, we stylize the foreground object and the background independently through semantic segmentation and an image inpainting module. To ensure a globally smooth result for the background, we use the image inpainting technique explained in the previous section, Section 6.1, combined with a cube padding approach for consistent boundaries. To ensure temporal consistency across frames, we use an optical flow algorithm to calculate the deviation between reprojected frames. Finally, we composite the background and the foreground to output finished frames. The overall framework is shown in Figure 6.8.

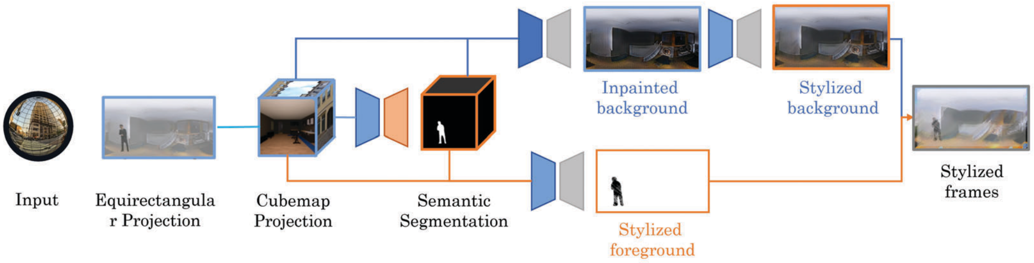


FIGURE 6.8: The framework of the proposed temporally consistent and foreground-aware video style transfer method.

One of the critical components of current neural style transfer methods is perceptual loss, which can be used to preserve the high-level spatial structure of the input image and leave out the detailed color and texture. This can lead to perceptually close results to the original input, instead of forcing an exact mapping between the two. To achieve this, the perceptual loss is calculated with additional neural networks with fixed weights, and most of the time, they are pretrained with popular perspective datasets for image classification, such as ImageNet [111] and Microsoft COCO dataset [212]. As a result, directly optimizing output from equirectangular formats will result in sub-optimal performance.

Cube padding. To effectively take advantage of the perceptual loss and learned knowledge of object semantics, the proposed system first reprojects the equirectangular format into cubamap projection. When compared to the equirectangular projection, each face of the cubemap projection has a very similar field-of-view to traditional perspective images, and thus can effectively utilize the underlying knowledge. Considering that independently stylizing each face of the cubemap and stitching them afterward will result in inconsistency along boundaries, a cubemap padding method is used to prevent this.

Instead of directly extending the neighboring pixel along a single edge [128], we use a larger field-of-view when rendering each cube face. This process is illustrated in Figure 6.9. During the experiment, we extend the reprojection field-of-view for each face σ from 90° to 110°. This can effectively overlap each face with no missing regions and enforce continuity along boundaries. By compositing the overlapping region with linear blending:

$$g(x) = (1 - \alpha)f_1(x) + \alpha f_2(x), \tag{6.10}$$

where α is $0 \rightarrow 1$. The projection model between equirectangular and cubemap is the same as we used in Chapter 5.

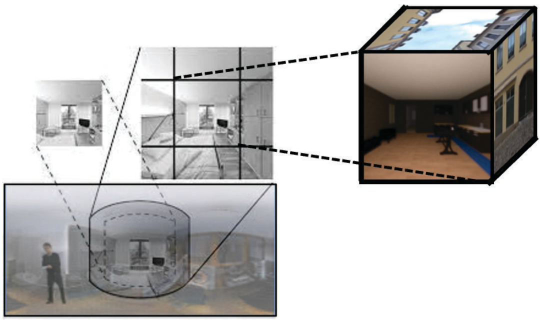


FIGURE 6.9: Instead of stitching neighboring pixels along the boundaries, we use a larger field-of-view when projecting each cube face.

Foreground-background separation. To separate the foreground object from the background of the pre-captured omnidirectional videos, we adopt a pretrained semantic segmentation network, mask R-CNN [52], to obtain the mask of the foreground object. However, considering that it is highly likely that the foreground object will move out of cube faces for a longer range of frames, a different appearance is likely to cause failure for the semantic segmentation task. Therefore, we pre-process the footage by rotating the spherical surface in a similar fashion described in Section 6.1.3. To be specific, we rotate the consecutive frames $R(F_t, \Delta_t)$ based on the mask of foreground objects in the previous frame F_{t-1}. By fixing the center of gravity of the semantic segmentation in the same place and adjusting the global environment, we can prevent the foreground moves across cube boundaries. After the entire sequence is predicted, we reverse the process of rotation with$\tilde{R}(F_t, \Delta_t)$. Afterward, we process the background patches with foreground objects removed through a state-of-the-art deep video inpainting algorithm [181].

Style transfer for a single frame. Previously, neural style transfer methods decomposed the problem into two components, style loss and content loss. For an input image with content $\mathbf{c}$, the target image style $\mathbf{s}$, and output image $\mathbf{x}$, we denote the feature maps of each one respectively as $\mathbf{C}$, $\mathbf{S}$, and $\mathbf{F}$. Then the style loss can be defined as

$$L_{style}(\mathbf{x}, \mathbf{s}) = \sum \frac{1}{M^2 N^2} \sum (G_{\mathbf{F}} - G_{\mathbf{S}}), \tag{6.11}$$

where M, N are dimensions of the input and G stands for the Gram matrix [206]. Similarly, the content loss is

$$L_{style}(\mathbf{x}, \mathbf{c}) = \sum \frac{1}{M^2 N^2} \sum (G_{\mathbf{F}} - G_{\mathbf{C}}). \tag{6.12}$$

Therefore, the total loss is the sum of both content loss and style loss with respective weight w:

$$L_{total}(\mathbf{x}, \mathbf{c}, \mathbf{s}) = w_i L_{style}(\mathbf{x}, \mathbf{c}) + w_j L_{style}(\mathbf{x}, \mathbf{c}) \tag{6.13}$$

Instead of iteratively optimize the output $\mathbf{x}$, feed-forward network suggest calculating a perceptual loss [42] against a pretrained network ϕ:

$$L_{perceptual} = \frac{1}{MN} \|\phi(\mathbf{c}) - \mathbf{x}\|_2 \tag{6.14}$$

During the runtime, we stylize the foreground patch and the background patch independently and composite the results together afterwards with regard to the semantic segmentation results.

Temporal consistency. For video input, even for visually identical frames, it is very likely that two consecutive frames that are initialized differently will result in stylized output that has drastically diverse appearances. To prevent flickering and instability across frames and provide a visually satisfying stylization result, we further incorporate a temporal consistency to improve the performance of videos. This is enforced through optical flow with a forward-backward warping. Given the optical flow between frames F_i and F_j in the forward direction,

$$\tilde{f}(F_i, F_j) = f((F_i, F_j) + f(F_j, F_i)) \tag{6.15}$$

Given pixel $p = (\phi, \theta)$, the warped pixel $\widetilde{p}$ on the target frame $\widetilde{k}$ is then:

$$\widetilde{p} = p + \tilde{f}_{j \to k}(p), \tag{6.16}$$

$$\widetilde{k}(p) = \tilde{F}_{j \to \widetilde{k}}(f_{j \to k}(p), j(p)) \tag{6.17}$$

The temporal consistency between two frames is then:

$$L^{temporal}_{j \to k} = \sum_{p} ||\widetilde{k}p - k(p)||_2 \tag{6.18}$$

6.2.4 Experimental Evaluation

Implementation details. In this section, we present the implementation details and experimental results of the proposed method. We tested the method on omnidirectional videos we collected from the internet, which is described in Chapter 5, to showcase a good generalization. All video frames were resized to 1440 x 720 to ensure a good visual fidelity when viewed in mixed reality applications. For each style image, we train the feed-forward network [42] for 20,000 iterations. During the experiment, the batch size was limited to 2 due to the higher resolution of omnidirectional images. The learning rate was set to 10^{-3} with Adam optimizer. Hardware details are given in the following table.

Experimental results. We present the qualitative result of the proposed method in Figure 6.10. By showcasing the stylized results of frame #1, frame #10, frame #20, and frame #30 (for a 30 frame-per-second video), we can observe temporally consistent results even for more distant frames. When focusing on the foreground target, in this case, the driver, we can see a clear boundary for the target, and the stylized results

TABLE 6.2: Experimental details of the hardware

Component	Details
CPU	Intel i7-7800X 4.0GHz
GPU	Nvidia GeForce GTX 2080Ti
Memory	32GB
Input Resolution	1440 × 720
Output Resolution	1440 × 720

(color, stroke, etc.) stay consistent and within the semantic segmentation of the target across the frames. For stylizing every frame, the proposed method takes approximately 500ms.

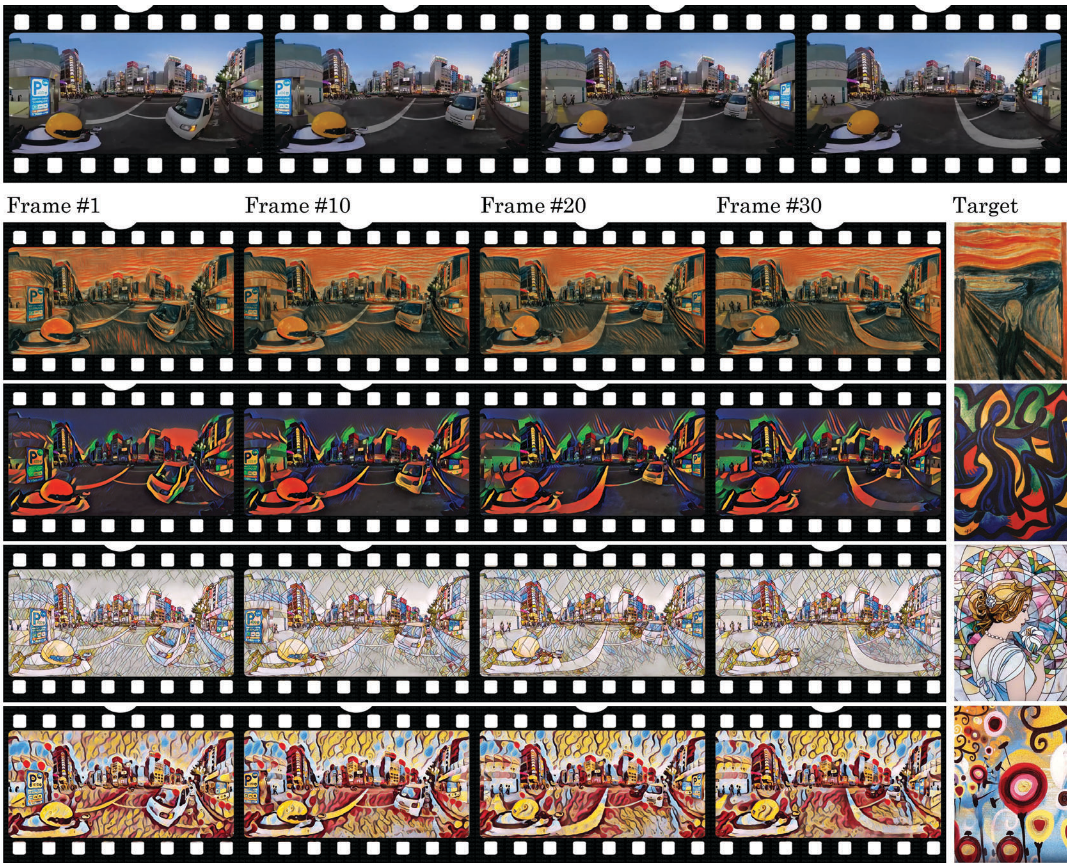


FIGURE 6.10: Qualitative results of the proposed methods.

6.3 Conclusion of the Chapter

Mixed reality as rising technology, is a result of advancements in computer vision. This chapter offers some degree of insight into how to practically take advantage of better scene understanding algorithms that can solve multiple challenging tasks effectively and efficiently.

We first propose a video editing framework that erases and add foreground objects with natural appearances for pre-captured omnidirectional videos. With an extensive user study, we successfully verify the effectiveness of our proof-of-concept implementation of an omnidirectional video editing framework. This method is possible to be applied to other applications in mixed reality, such as assisting users by highlighting important objects, or privacy protection by erasing sensitive information.

In the second half, we propose a video stylization method that artistically transfers omnidirectional videos into a target-style image with distinct foreground representations and temporal consistency across the frames. By designing a framework that can successfully take advantage of the perceptual knowledge from perspective training data, the results show good generalization and visually pleasant results even for videos randomly collected from the internet.

In the future, we envision proposing more efficient neural networks that are capable of processing high-fidelity input while maintaining a real-time performance for visually editing and stylizing omnidirectional videos.

Chapter 7

Conclusion

7.1 Summary

In this dissertation, we investigated how to use scene understanding and data augmentation to improve immersive mixed reality from multiple spatial scales and studied employing computer vision algorithms for practical mixed reality applications. After we review the backgrounds of related research fields, we present four research work in four respective chapters to tackle existing challenges in mixed reality and improve the immersive user experience.

In Chapter 3, we presented a real-time method to handle the hand-object occlusions in mixed reality. We propose a photo-realistic RGBD hand-object database with precise hand postures and semantic segmentation annotations to facilitate our occlusion-aware joint learning system. With a novel real-time optimization pipeline, we utilize the jointly predicted postures and segmentation to calculate occlusion masks and render objects with correct occlusions. The experimental results show better quantitative and qualitative performance than previous literature, and a user study verifies a more realistic mixed reality experience of hand-object interactions. The implementation shows good accuracy, robustness, and speed with the potential to be further adapted to other applications.

Currently, we are using an existing framework to handle the object augmentation during the usage, and hence there is room to improve and resolve the issue of misalignment when localizing the optimized hand model. In addition, there is no reliable occlusion-aware object tracking in the current implementation and this leads to losing augmentation of the object during experiments due to strong occlusions. Joint tracking of hand and object is a promising direction for a more consistent and sophisticated mixed reality experience.

In Chapter 4, We presented a data augmentation method to generate high-quality equirectangular databases with paired color and ground-truth depth annotations by repurposing abundant and easily obtainable 2-dimensional RGBD databases. With this database, we first introduced and implemented an auxiliary network that calculates local depth loss to resolve an issue that small regions of interest are frequently smoothed out during optimizing global gradients. We take humans, a crucial subject in 360° contents, as an example to show the efficacy of our approach. We showed improved accuracy of our approach compared to the state-of-the-art technique.

To further improve the accuracy of the foreground-aware scene understanding, we proposed a foreground-aware bi-projection-based design. The proposed architecture produces consistent global depth prediction with the equirectangular projection, while enforcing local detailed features through the cubemap projection. An additional foreground loss acquired through a multitask learning approach of semantic segmentation complementarily provides sharper boundaries of predicted foreground objects. With quantitative and qualitative evaluation, we successfully verified the effectiveness of the proposed method. We believe the ability to accurately predict depth information for omnidirectional images can facilitate a wide range of applications such as 3-dimensional reconstruction and mixed reality.

At present, our data augmentation method is based on the premise that both 2-dimensional and 360° data are captured with similar extrinsic parameters (e.g. cameras are aligned horizontally, positioned at average eye-level height) and lighting conditions, while it is true for most data captured in lab conditions, its application for in-the-wild images is limited. Furthermore, our approach works for both indoor and outdoor settings. Nevertheless, for outdoor settings, a higher dynamic range of luminosity and sunlight's ambient IR will render capturing RGB and depth information inherently difficult. While self-supervised methods or multi-view-based generation methods to further reduce the need to acquire expensive ground truth data are extensively studied in our subsequent works, exploring generating samples with different lighting conditions with GANs to improve the robustness of depth estimation seems to be effective and promising future research.

In Chapter 5, we first proposed to utilize the unlimited source of data, 360° videos from the internet, to overcome the scarcity of a general omnidirectional dataset. We propose geometric and temporal constraints that are unique to 360° videos and use test-time training to generate high-quality depth maps. To fully benefit from our dataset, Depth360, we propose an end-to-end two-branch multitask network, SegFuse, that mimics human vision to estimate depth from a single omnidirectional image. With the peripheral vision to perceive the depth of the scene and foveal vision to distinguish between objects, our network shows favorable results against state-of-the-art methods. With the ability to estimate high-quality depth information of the global context of omnidirectional images, we implement an application to showcase how scene understanding can help improve the immersive experience in mixed reality.

Although our data generation method can be applied to larger-scale collections to extend the size of datasets, it shows several limitations. First, online omnidirectional videos present unbalanced distributions, favoring specific scenarios (e.g. urban street views). Second, when establishing the baseline, SfM and MVS methods show sub-optimal results when there are texture-less surfaces or reflective materials in the scene. Scenes with excessive dynamic foreground objects or strong motions are problematic for a pseudo-stereo system to acquire accurate geometric consistency. Future work could alleviate these problems by adopting improved SfM algorithms and scaling to a larger variety of input collections. For depth estimation, the current implementation only accepts smaller batch sizes due to hardware limitations. We expect to improve the efficiency of the network and enable more stable training with better normalization methods.

In Chapter 6, we explored practical applications of different scene understanding algorithms in the context of mixed reality to further provide users with improved capabilities and immersive experiences. We put forward two different applications: editing foreground objects of interest in pre-captured omnidirectional videos and consistent artistic stylization for pre-captured videos, to demonstrate the benefit and potential of alike studies. We successfully verified the feasibility to employ existing scene understanding algorithms to solve practical problems in the mixed reality environment through our proof-of-concept implementations. In the future, we envision proposing more efficient neural networks that are capable of processing high-fidelity input. Another stream of future work for visually editing omnidirectional videos would be taking temporal consistency into consideration for yielding more stable outputs.

Throughout the thesis, we followed the order of different spatial scopes of scene understanding and studied how each scale improves the immersive mixed reality. We started from a smaller scale of understanding local hand-object interactions, followed by observing the foreground objects of interest in mixed reality scenes, and finally, we studied the entirety of the scene and try to comprehend the global context. We presented multiple practical employment of scene understanding and data augmentation algorithms in mixed reality applications and validated their effectiveness through user studies.

7.2 Future Direction

To strive for better immersive mixed reality, based on our research on scene understanding and data augmentation with their respective limitations, we try to discuss some potential directions for future research that utilizes scene understanding to improve mixed reality experience. Throughout the process of designing the hardware to present convincing visualizations in front of the users' eyes, there are multiple important steps that we introduced at the beginning of the thesis rely on a robust, efficient, and effective understanding of both the virtual environment and the physical world that surrounds the user.

On a global level, the current generation of fully or partially immersive mixed reality devices still rely on depth and infra-red sensors to sense the environment and establish the mapping for subsequent augmentation processes, and the result is that outdoor mixed reality is quite limited and under-investigated when compared to the indoor environment. With sophisticated image-based scene understanding algorithms, it is possible to use predicted information with high accuracy to circumvent the challenge. At the same time, Lidar-based augmented reality is drawing steadily increasing attention from mobile developers and researchers. Considering that cameras are almost universally accessible for hand-held devices, multi-modal research topics that take advantage of high-performance image-based algorithms together with robust light detection and ranging information are highly encouraged.

On a regional level, understanding the focus and the foreground objects is of high importance as well. Although limited computing performance and power consumption of existing hardware are important terms in the whole equation, mixed reality requires a high-level understanding of the focus and foreground objects to provide an immersive experience such as foveated rendering. We believe that contextual awareness is helpful for user-centered research including human-computer interactions and remote collaboration.

On a local level, to achieve a convincing visualization for users to perceive the augmented information as a natural part of the environment, it is essential to seamlessly provide correct occlusions and lighting to the virtual objects. The sensitive human eye can instantly recognize unnatural behaviors and inconsistencies caused by the algorithm, therefore, a faithful understanding of the dimensions and properties of both physical and virtual objects, as well as interactions are essential to maintain an immersive experience. A necessary and promising direction would be real-time strategies of scene understanding (e. g. object recognition, semantic segmentation, reconstruction) with high accuracy that is robust to occlusions and different lighting conditions to facilitate future high-quality mixed reality applications.

Finally, it is necessary to actively evaluate the existing methodologies and receive feedback from users during exploration. Being an emerging technology and a new format of tool that assists tasks in daily life, mixed reality applications are complicated to design and evaluate. Therefore, we believe it is beneficial to study the practicality and compatibility for users in parallel to designing and implementing new scene understanding algorithms.

List of Figures

List of Tables

Bibliography

[1] Charles E Hughes et al. “Mixed reality in education, entertainment, and training”. In: *IEEE computer graphics and applications* 25.6 (2005), pp. 24–30.

[2] Zachary Walker et al. “Beyond Pokémon: Augmented reality is a universal design for learning tool”. In: *Sage Open* 7.4 (2017), p. 2158244017737815.

[3] Wolfgang Hoenig et al. “Mixed reality for robotics”. In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2015, pp. 5382–5387.

[4] Asier Marzo, Benoît Bossavit, and Martin Hachet. “Combining multi-touch input and device movement for 3D manipulations in mobile augmented reality environments”. In: *Proceedings of the 2nd ACM symposium on Spatial user interaction*. 2014, pp. 13–16.

[5] Paul Milgram and Fumio Kishino. “A taxonomy of mixed reality visual displays”. In: *IEICE TRANSACTIONS on Information and Systems* 77.12 (1994), pp. 1321–1329.

[6] Rafal Wojciechowski et al. “Building virtual and augmented reality museum exhibitions”. In: *Proceedings of the ninth international conference on 3D Web technology*. 2004, pp. 135–144.

[7] Long Chen et al. “Context-aware mixed reality: A framework for ubiquitous interaction”. In: *arXiv preprint arXiv:1803.05541* (2018).

[8] Qi Feng, Hubert PH Shum, and Shigeo Morishima. “Resolving occlusion for 3D object manipulation with hands in mixed reality”. In: *Proceedings of the 24th ACM Symposium on Virtual Reality Software and Technology*. 2018, pp. 1–2.

[9] Philipp A Rauschnabel, Alexander Rossmann, and M Claudia tom Dieck. “An adoption framework for mobile augmented reality games: The case of Pokémon Go”. In: *Computers in Human Behavior* 76 (2017), pp. 276–286.

[10] Waraporn Viyanon et al. “AR furniture: Integrating augmented reality technology to enhance interior design using marker and markerless tracking”. In: *Proceedings of the 2nd International Conference on Intelligent Information Processing*. 2017, pp. 1–7.

[11] Andrew C Boud et al. “Virtual reality and augmented reality as a training tool for assembly tasks”. In: *1999 IEEE International Conference on Information Visualization (Cat. No. PR00210)*. IEEE. 1999, pp. 32–36.

[12] Victor Fragoso et al. “TranslatAR: A mobile augmented reality translator”. In: *2011 IEEE workshop on applications of computer vision (WACV)*. IEEE. 2011, pp. 497–502.

[13] Ryo Shimamura et al. “Audio–visual object removal in 360-degree videos”. In: *The Visual Computer* 36.10 (2020), pp. 2117–2128.

[14] Michael W Boyce et al. “Characterizing the cognitive impact of tangible augmented reality”. In: *International conference on human-computer interaction*. Springer. 2019, pp. 416–427.

[15] Avinash Gupta et al. “A Virtual Reality Enhanced Cyber-Human Framework for Orthopedic Surgical Training”. In: *IEEE Sys. J.* 13.3 (2019), pp. 3501–3512.

[16] Google LLC. *Google Trend Data of Mixed Reality Related Topics*. 2022. URL: https://trends.google.com/trends/ (visited on 06/10/2022).

[17] Elsevier. *Search Results of Mixed Reality Related Topics*. 2022. URL: https://www.scopus.com (visited on 06/10/2022).

[18] Leonardo Rodriguez et al. “Developing a mixed reality assistance system based on projection mapping technology for manual operations at assembly workstations”. In: *Procedia computer science* 75 (2015), pp. 327–333.

[19] Jonathan Steuer. “Defining virtual reality: Dimensions determining telepresence”. In: *Journal of communication* 42.4 (1992), pp. 73–93.

[20] Takuji Narumi et al. “Augmented reality flavors: gustatory display based on edible marker and cross-modal interaction”. In: *Proceedings of the SIGCHI conference on human factors in computing systems*. 2011, pp. 93–102.

[21] Woon-Sung Lee, Jung-Ha Kim, and Jun-Hee Cho. “A driving simulator as a virtual reality tool”. In: *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No. 98CH36146)*. Vol. 1. IEEE. 1998, pp. 71–76.

[22] Benoit Bideau et al. “Using virtual reality to analyze sports performance”. In: *IEEE Computer Graphics and Applications* 30.2 (2009), pp. 14–21.

[23] Vladislav Angelov et al. “Modern virtual reality headsets”. In: *2020 International congress on human-computer interaction, optimization and robotic applications (HORA)*. IEEE. 2020, pp. 1–5.

[24] Qi Feng, Hubert PH Shum, and Shigeo Morishima. “360 Depth Estimation in the Wild–The Depth360 Dataset and the SegFuse Network”. In: *arXiv preprint arXiv:2202.08010* (2022).

[25] PC Thomas and WM David. “Augmented reality: An application of heads-up display technology to manual manufacturing processes”. In: *Hawaii international conference on system sciences*. Vol. 2. ACM SIGCHI Bulletin. 1992.

[26] Ivan E Sutherland. “A head-mounted three dimensional display”. In: *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*. 1968, pp. 757–764.

[27] Hao Zhou et al. “Deep single-image portrait relighting”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 7194–7202.

[28] Mark Fiala. “Designing highly reliable fiducial markers”. In: *IEEE Transactions on Pattern analysis and machine intelligence* 32.7 (2009), pp. 1317–1324.

[29] Andrew J Davison. “Real-time simultaneous localisation and mapping with a single camera”. In: *Computer Vision, IEEE International Conference on*. Vol. 3. IEEE Computer Society. 2003, pp. 1403–1403.

[30] Andrew J Davison et al. “MonoSLAM: Real-time single camera SLAM”. In: *IEEE transactions on pattern analysis and machine intelligence* 29.6 (2007), pp. 1052–1067.

[31] Georg Klein and David Murray. “Parallel tracking and mapping for small AR workspaces”. In: *2007 6th IEEE and ACM international symposium on mixed and augmented reality*. IEEE. 2007, pp. 225–234.

[32] Richard A Newcombe and Andrew J Davison. “Live dense reconstruction with a single moving camera”. In: *2010 IEEE computer society conference on computer vision and pattern recognition*. IEEE Computer Society. 2010, pp. 1498–1505.

[33] Wei Tan et al. “Robust monocular SLAM in dynamic environments”. In: *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE. 2013, pp. 209–218.

[34] Aparna Lakshmi Ratan, W Eric L Grimson, and William M Wells. “Object detection and localization by dynamic template warping”. In: *International Journal of Computer Vision* 36.2 (2000), pp. 131–147.

[35] Zhiwei Zhu et al. “Real-time global localization with a pre-built visual landmark database”. In: *2008 ieee conference on computer vision and pattern recognition*. IEEE. 2008, pp. 1–8.

[36] Richard A Newcombe et al. “KinectFusion: Real-time dense surface mapping and tracking”. In: *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*. IEEE. 2011, pp. 127–136.

[37] Kunihiko Fukushima and Sei Miyake. “Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition”. In: *Competition and cooperation in neural nets*. Springer, 1982, pp. 267–285.

[38] Karen Simonyan and Andrew Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556* (2014).

[39] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems* 25 (2012).

[40] Gao Huang et al. “Densely connected convolutional networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2017, pp. 4700–4708.

[41] Christian Szegedy et al. “Going deeper with convolutions”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2015, pp. 1–9.

[42] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. “Perceptual losses for real-time style transfer and super-resolution”. In: *European conference on computer vision.* Springer. 2016, pp. 694–711.

[43] Kyunghyun Cho et al. “Learning phrase representations using RNN encoder-decoder for statistical machine translation”. In: *arXiv preprint arXiv:1406.1078* (2014).

[44] Ian Goodfellow et al. “Generative adversarial nets”. In: *Advances in neural information processing systems* 27 (2014).

[45] Mehdi Mirza and Simon Osindero. “Conditional generative adversarial nets”. In: *arXiv preprint arXiv:1411.1784* (2014).

[46] Martin Arjovsky, Soumith Chintala, and Léon Bottou. “Wasserstein generative adversarial networks”. In: *International conference on machine learning.* PMLR. 2017, pp. 214–223.

[47] Alec Radford, Luke Metz, and Soumith Chintala. “Unsupervised representation learning with deep convolutional generative adversarial networks”. In: *arXiv preprint arXiv:1511.06434* (2015).

[48] Zhengwei Wang, Qi She, and Tomas E Ward. “Generative adversarial networks in computer vision: A survey and taxonomy”. In: *ACM Computing Surveys (CSUR)* 54.2 (2021), pp. 1–38.

[49] David G Lowe. “Object recognition from local scale-invariant features”. In: *Proceedings of the seventh IEEE international conference on computer vision.* Vol. 2. Ieee. 1999, pp. 1150–1157.

[50] Navneet Dalal and Bill Triggs. “Histograms of oriented gradients for human detection”. In: *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR’05).* Vol. 1. Ieee. 2005, pp. 886–893.

[51] Ross Girshick et al. “Region-based convolutional networks for accurate object detection and segmentation”. In: *IEEE transactions on pattern analysis and machine intelligence* 38.1 (2015), pp. 142–158.

[52] Kaiming He et al. “Mask r-cnn”. In: *Computer Vision (ICCV), 2017 IEEE International Conference on.* IEEE. 2017, pp. 2980–2988.

[53] Ross Girshick. "Fast r-cnn". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1440–1448.

[54] Tsung-Yi Lin et al. "Feature pyramid networks for object detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 2117–2125.

[55] Jonathan Long, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 3431–3440.

[56] Shaoqing Ren et al. "Faster r-cnn: Towards real-time object detection with region proposal networks". In: *Advances in neural information processing systems* 28 (2015).

[57] Donggeun Yoo et al. "Multi-scale pyramid pooling for deep convolutional representation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2015, pp. 71–80.

[58] Wei Liu et al. "Ssd: Single shot multibox detector". In: *European conference on computer vision*. Springer. 2016, pp. 21–37.

[59] Joseph Redmon et al. "You only look once: Unified, real-time object detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 779–788.

[60] Cheng-Yang Fu et al. "Dssd: Deconvolutional single shot detector". In: *arXiv preprint arXiv:1701.06659* (2017).

[61] Xiang Li et al. "Object detection in the context of mobile augmented reality". In: *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE. 2020, pp. 156–163.

[62] Andreas Geiger, Philip Lenz, and Raquel Urtasun. "Are we ready for autonomous driving? the kitti vision benchmark suite". In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE. 2012, pp. 3354–3361.

[63] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation". In: *International Conference on Medical image computing and computer-assisted intervention*. Springer. 2015, pp. 234–241.

[64] Zoltan Kato and Ting-Chuen Pong. "A Markov random field image segmentation model for color textured images". In: *Image and Vision Computing* 24.10 (2006), pp. 1103–1114.

[65] Florian Schroff, Antonio Criminisi, and Andrew Zisserman. "Object Class Segmentation using Random Forests." In: *BMVC*. 2008, pp. 1–10.

[66] Xiang-Yang Wang, Ting Wang, and Juan Bu. "Color image segmentation using pixel wise support vector machine classification". In: *Pattern Recognition* 44.4 (2011), pp. 777–787.

[67] Falong Shen et al. “Semantic segmentation via structured patch prediction, context crf and guidance crf”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 1953–1961.

[68] Sachin Mehta et al. “Espnet: Efficient spatial pyramid of dilated convolutions for semantic segmentation”. In: *Proceedings of the european conference on computer vision (ECCV)*. 2018, pp. 552–568.

[69] Brett R Jones et al. “IllumiRoom: peripheral projected illusions for interactive experiences”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2013, pp. 869–878.

[70] Ashutosh Saxena, Min Sun, and Andrew Y Ng. “Make3d: Learning 3d scene structure from a single still image”. In: *IEEE transactions on pattern analysis and machine intelligence* 31.5 (2008), pp. 824–840.

[71] Kevin Karsch, Ce Liu, and Sing Bing Kang. “Depth transfer: Depth extraction from video using non-parametric sampling”. In: *IEEE transactions on pattern analysis and machine intelligence* 36.11 (2014), pp. 2144–2158.

[72] Fayao Liu, Chunhua Shen, and Guosheng Lin. “Deep convolutional neural fields for depth estimation from a single image”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 5162–5170.

[73] Iro Laina et al. “Deeper depth prediction with fully convolutional residual networks”. In: *2016 Fourth international conference on 3D vision (3DV)*. IEEE. 2016, pp. 239–248.

[74] Fu-En Wang et al. “Bifuse: Monocular 360 depth estimation via bi-projection fusion”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 462–471.

[75] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. “Unsupervised monocular depth estimation with left-right consistency”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 270–279.

[76] Zhichao Yin and Jianping Shi. “Geonet: Unsupervised learning of dense depth, optical flow and camera pose”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 1983–1992.

[77] Tinghui Zhou et al. “Unsupervised learning of depth and ego-motion from video”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1851–1858.

[78] Chih-Kuan Yeh et al. “Learning deep latent space for multi-label classification”. In: *Thirty-first AAAI conference on artificial intelligence*. 2017.

[79] Qi Feng, Hubert PH Shum, and Shigeo Morishima. “Bi-projection-based Foreground-aware Omnidirectional Depth Prediction”. In: *Visual Computing + VC Communications*. Tokyo, Japan: DU, 2021, pp. 1–6.

[80] Kamalika Chaudhuri et al. "Multi-view clustering via canonical correlation analysis". In: *Proceedings of the 26th annual international conference on machine learning*. 2009, pp. 129–136.

[81] Micah Hodosh, Peter Young, and Julia Hockenmaier. "Framing image description as a ranking task: Data, models and evaluation metrics". In: *Journal of Artificial Intelligence Research* 47 (2013), pp. 853–899.

[82] Shikun Liu, Edward Johns, and Andrew J Davison. "End-to-end multi-task learning with attention". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 1871–1880.

[83] Alex Kendall, Yarin Gal, and Roberto Cipolla. "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 7482–7491.

[84] Zhao Chen et al. "Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks". In: *International conference on machine learning*. PMLR. 2018, pp. 794–803.

[85] Nathan Silberman et al. "Indoor segmentation and support inference from rgbd images". In: *European Conference on Computer Vision*. Springer. 2012, pp. 746–760.

[86] Ziwei Liu et al. "Large-scale celebfaces attributes (celeba) dataset". In: *Retrieved August* 15.2018 (2018), p. 11.

[87] Marius Cordts et al. "The cityscapes dataset for semantic urban scene understanding". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 3213–3223.

[88] Franziska Mueller et al. "Ganerated hands for real-time 3d hand tracking from monocular RGB". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 49–59.

[89] Feng Qi. "Resolving Hand-objection Occlusion for Egocentric Mixed Reality with CNNs from RGB-D". Master's Thesis. Waseda University, 2019.

[90] Qi Feng, Hubert PH Shum, and Shigeo Morishima. "Resolving hand-object occlusion for mixed reality with joint deep learning and model optimization". In: *Computer Animation and Virtual Worlds* 31.4-5 (2020), e1956.

[91] Denis Kalkofen, Erick Mendez, and Dieter Schmalstieg. "Interactive focus and context visualization for augmented reality". In: *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*. IEEE Computer Society. 2007, pp. 1–10.

[92] Edward Zhang et al. "Emptying, refurnishing, and relighting indoor spaces". In: *ACM Trans. on Graph. (TOG)* 35.6 (2016), pp. 1–14.

[93] Takahiro Tsuda et al. “Visualization methods for outdoor see-through vision”. In: *IEICE transactions on information and systems* 89.6 (2006), pp. 1781–1789.

[94] Max Krichenbauer et al. “Augmented Reality versus Virtual Reality for 3D Object Manipulation”. In: *IEEE transactions on visualization and computer graphics* 24.2 (2018), pp. 1038–1048.

[95] Aleksander Holynski and Johannes Kopf. “Fast depth densification for occlusion-aware augmented reality”. In: *SIGGRAPH Asia 2018 Technical Papers*. ACM. 2018, p. 194.

[96] Yuan Tian, Tao Guan, and Cheng Wang. “Real-time occlusion handling in augmented reality based on an object tracking approach”. In: *Sensors* 10.4 (2010), pp. 2885–2900.

[97] Chao Du et al. “Edge snapping-based depth enhancement for dynamic occlusion handling in augmented reality”. In: *Mixed and Augmented Reality (ISMAR), 2016 IEEE International Symposium on*. IEEE. 2016, pp. 54–62.

[98] Ronald T. Azuma. “Making Augmented Reality a Reality”. In: *Imaging and Applied Optics 2017*. Optical Society of America, 2017, JTu1F.1. URL: http://www.osapublishing.org/abstract.cfm?URI=AIO-2017-JTu1F.1.

[99] Vincent Lepetit and M-O Berger. “Handling occlusion in augmented reality systems: a semi-automatic method”. In: *Augmented Reality, 2000.(ISAR 2000). Proceedings. IEEE and ACM International Symposium on*. IEEE. 2000, pp. 137–146.

[100] Yuan Tian et al. “Handling occlusions in augmented reality based on 3D reconstruction method”. In: *Neurocomputing* 156 (2015), pp. 96–104.

[101] Anjul Patney et al. “Perceptually-based foveated virtual reality”. In: *ACM SIGGRAPH 2016 Emerging Technologies*. ACM. 2016, p. 17.

[102] David R Walton and Anthony Steed. “Accurate real-time occlusion for mixed reality”. In: *Proceedings of the 23rd ACM Symposium on Virtual Reality Software and Technology*. ACM. 2017, p. 11.

[103] Nikolaos Kyriazis et al. “A generative approach to tracking hands and their interaction with objects”. In: *Man–Machine Interactions 4*. Springer, 2016, pp. 19–28.

[104] Andrea Tagliasacchi et al. “Robust articulated-ICP for real-time hand tracking”. In: *Comp. Graph. Forum*. Vol. 34. 5. 2015, pp. 101–114.

[105] Chen Qian et al. “Realtime and robust hand tracking from depth”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014, pp. 1106–1113.

[106] Franziska Mueller et al. "Real-time Hand Tracking under Occlusion from an Egocentric RGB-D Sensor". In: *Proceedings of International Conference on Computer Vision (ICCV)*. 2017. URL: http://handtracker.mpi-inf.mpg.de/projects/OccludedHands/.

[107] Javier Romero et al. "Embodied Hands: Modeling and Capturing Hands and Bodies Together". In: *ACM Trans. on Graph.* 245:1–245:17 36.6 (Nov. 2017), 245:1–245:17.

[108] Franziska Mueller et al. "Real-time pose and shape reconstruction of two interacting hands with a single depth camera". In: *ACM Trans. on Graph. (TOG)* 38.4 (2019), p. 49.

[109] Bugra Tekin et al. "H+ O: Unified egocentric recognition of 3D hand-object poses and interactions". In: *Proc. IEEE conf. comp. vis. pat. recog.* 2019, pp. 4511–4520.

[110] Tomas Simon et al. "Hand keypoint detection in single images using multiview bootstrapping". In: *Proc. IEEE conf. comp. vis. pat. recog.* 2017, pp. 1145–1153.

[111] Jia Deng et al. "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.

[112] Jiawei Zhang et al. "3d hand pose tracking and estimation using stereo matching". In: *arXiv preprint arXiv:1610.07214* (2016).

[113] Hui Liang et al. "Ar in hand: Egocentric palm pose tracking and gesture recognition for augmented reality applications". In: *Proceedings of the 23rd ACM international conference on Multimedia*. ACM. 2015, pp. 743–744.

[114] Qi Feng et al. "Foreground-aware Dense Depth Estimation for 360 Images". In: *Journal of WSCG* 28 (2020), pp. 79–88. DOI: 10.24132/JWSCG.2020.28.10.

[115] José Gaspar, Niall Winters, and José Santos-Victor. "Vision-based navigation and environmental representations with an omnidirectional camera". In: *IEEE Transactions on robotics and automation* 16.6 (2000), pp. 890–898.

[116] Jingwei Huang et al. "6-DOF VR videos with a single 360-camera". In: *2017 IEEE Virtual Reality (VR)*. IEEE. 2017, pp. 37–44.

[117] Nikolaos Zioulis et al. "Omnidepth: Dense depth estimation for indoors spherical panoramas". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 448–465.

[118] Taco Cohen et al. "Spherical cnns". In: *International Conference on Learning Representations (ICLR)* (2018).

[119] Angel Chang et al. "Matterport3d: Learning from rgb-d data in indoor environments". In: *International Conference on 3D Vision (3DV)* (2017).

[120] Iro Armeni et al. "Joint 2d-3d-semantic data for indoor scene understanding". In: *arXiv preprint arXiv:1702.01105* (2017).

[121] Shuran Song et al. "Semantic scene completion from a single depth image". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 2017, pp. 1746–1754.

[122] Amir Atapour-Abarghouei and Toby P Breckon. "Real-time monocular depth estimation using synthetic data with domain adaptation via image style transfer". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 2018, pp. 2800–2810.

[123] Derek Hoiem, Alexei A Efros, and Martial Hebert. "Geometric context from a single image". In: *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1.* Vol. 1. IEEE. 2005, pp. 654–661.

[124] David Eigen and Rob Fergus. "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture". In: *Proceedings of the IEEE international conference on computer vision.* 2015, pp. 2650–2658.

[125] Pierre Sermanet et al. "Overfeat: Integrated recognition, localization and detection using convolutional networks". In: *arXiv preprint arXiv:1312.6229* (2013).

[126] Ravi Garg et al. "Unsupervised cnn for single view depth estimation: Geometry to the rescue". In: *European Conference on Computer Vision.* Springer. 2016, pp. 740–756.

[127] Junyuan Xie, Ross Girshick, and Ali Farhadi. "Deep3d: Fully automatic 2d-to-3d video conversion with deep convolutional neural networks". In: *European Conference on Computer Vision.* Springer. 2016, pp. 842–857.

[128] Manuel Ruder, Alexey Dosovitskiy, and Thomas Brox. "Artistic style transfer for videos and spherical images". In: *International Journal of Computer Vision* 126.11 (2018), pp. 1199–1219.

[129] Yu-Chuan Su and Kristen Grauman. "Learning spherical convolution for fast features from 360 imagery". In: *Advances in Neural Information Processing Systems.* 2017, pp. 529–539.

[130] Kevin Matzen et al. "Low-cost 360 stereo photography and video capture". In: *ACM Transactions on Graphics (TOG)* 36.4 (2017), p. 148.

[131] Ankur Handa et al. "Scenenet: An annotated model generator for indoor scene understanding". In: *2016 IEEE International Conference on Robotics and Automation (ICRA).* IEEE. 2016, pp. 5737–5743.

[132] Chunhui Liu et al. "PKU-MMD: A large scale benchmark for continuous multimodal human action understanding". In: *arXiv preprint arXiv:1703.07475* (2017).

[133] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2016, pp. 770–778.

[134] Adam Paszke et al. "Automatic differentiation in PyTorch". In: (2017).

[135] Diederik P Kingma and Jimmy Ba. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).

[136] Xavier Glorot and Yoshua Bengio. “Understanding the difficulty of training deep feedforward neural networks”. In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. 2010, pp. 249–256.

[137] Seungryong Kim et al. “Unified depth prediction and intrinsic image decomposition from a single image via joint convolutional neural fields”. In: *European conference on computer vision*. Springer. 2016, pp. 143–159.

[138] Ishan Misra et al. “Cross-stitch networks for multi-task learning”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 3994–4003.

[139] Benjamin Ummenhofer et al. “Demon: Depth and motion network for learning monocular stereo”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 5038–5047.

[140] Ana Serrano et al. “Motion parallax for 360 RGBD video”. In: *IEEE Transactions on Visualization and Computer Graphics* 25.5 (2019), pp. 1817–1827.

[141] Yinda Zhang et al. “Physically-based rendering for indoor scene understanding using convolutional neural networks”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 5287–5295.

[142] Yongjie Zhu et al. “Spatially-Varying Outdoor Lighting Estimation from Intrinsics”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 12834–12842.

[143] Fangchang Ma, Guilherme Venturelli Cavalheiro, and Sertac Karaman. “Self-supervised sparse-to-dense: Self-supervised depth completion from lidar and monocular camera”. In: *2019 International Conference on Robotics and Automation (ICRA)*. IEEE. 2019, pp. 3288–3295.

[144] Yiping Chen et al. “Lidar-video driving dataset: Learning driving policies effectively”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 5870–5878.

[145] Nikolaos Zioulis et al. “Spherical view synthesis for self-supervised 360 depth estimation”. In: *2019 International Conference on 3D Vision (3DV)*. IEEE. 2019, pp. 690–699.

[146] Eddy Ilg et al. “Flownet 2.0: Evolution of optical flow estimation with deep networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 2462–2470.

[147] Yue Luo et al. “Single view stereo matching”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 155–163.

[148] Nikolaus Mayer et al. “A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 4040–4048.

[149] Zhengqi Li and Noah Snavely. “Megadepth: Learning single-view depth prediction from internet photos”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 2041–2050.

[150] Yasamin Jafarian and Hyun Soo Park. “Learning High Fidelity Depths of Dressed Humans by Watching Social Media Dance Videos”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 12753–12762.

[151] Christian Richardt et al. “Megastereo: Constructing high-resolution stereo panoramas”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2013, pp. 1256–1263.

[152] Fu-En Wang et al. “Self-supervised Learning of Depth and Camera Motion from 360 Videos”. In: *Asian Conference on Computer Vision*. Springer. 2018, pp. 53–68.

[153] Georgios Albanis et al. “Pano3D: A Holistic Benchmark and a Solid Baseline for 360deg Depth Estimation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 3727–3737.

[154] Shang-Ta Yang et al. “Dula-net: A dual-projection network for estimating room layouts from a single rgb panorama”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 3363–3372.

[155] Xuan Luo et al. “Consistent video depth estimation”. In: *ACM Transactions on Graphics (TOG)* 39.4 (2020), pp. 71–1.

[156] Ashutosh Saxena, Sung H Chung, Andrew Y Ng, et al. “Learning depth from single monocular images”. In: *NIPS*. Vol. 18. 2005, pp. 1–8.

[157] Clément Godard et al. “Digging into self-supervised monocular depth estimation”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 3828–3838.

[158] S Mahdi H Miangoleh et al. “Boosting Monocular Depth Estimation Models to High-Resolution via Content-Adaptive Multi-Resolution Merging”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 9685–9694.

[159] Yuanzhouhan Cao, Zifeng Wu, and Chunhua Shen. “Estimating depth from monocular images as classification using deep fully convolutional residual networks”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 28.11 (2017), pp. 3174–3182.

[160] Tobias Bertel et al. “OmniPhotos: casual 360° VR photography”. In: *ACM Transactions on Graphics (TOG)* 39.6 (2020), pp. 1–12.

[161] Taehyun Rhee et al. “Mr360: Mixed reality rendering for 360 panoramic videos”. In: *IEEE transactions on visualization and computer graphics* 23.4 (2017), pp. 1379–1388.

[162] Taco S Cohen et al. “Spherical CNNs”. In: *International Conference on Learning Representations*. 2018.

[163] Renata Khasanova and Pascal Frossard. “Graph-based classification of omnidirectional images”. In: *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 2017, pp. 869–878.

[164] Hsien-Tzu Cheng et al. “Cube padding for weakly-supervised saliency prediction in 360 videos”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 1420–1429.

[165] Benjamin Coors, Alexandru Paul Condurache, and Andreas Geiger. “Spherenet: Learning spherical representations for detection and classification in omnidirectional images”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 518–533.

[166] Igor Vasiljevic et al. “Diode: A dense indoor and outdoor depth dataset”. In: *arXiv preprint arXiv:1908.00463* (2019).

[167] Cheng Sun, Min Sun, and Hwann-Tzong Chen. “Hohonet: 360 indoor holistic understanding with latent horizontal features”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 2573–2582.

[168] Giovanni Pintore et al. “SliceNet: deep dense depth estimation from a single indoor panorama using a slice-based representation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 11536–11545.

[169] Clara Fernandez-Labrador et al. “Corners for layout: End-to-end layout recovery from 360 images”. In: *IEEE Robotics and Automation Letters* 5.2 (2020), pp. 1255–1262.

[170] Yu-Chuan Su and Kristen Grauman. “Kernel transformer networks for compact spherical convolution”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 9442–9451.

[171] Hualie Jiang et al. “Unifuse: Unidirectional fusion for 360 panorama depth estimation”. In: *IEEE Robotics and Automation Letters* 6.2 (2021), pp. 1519–1526.

[172] Shinya Sumikura, Mikiya Shibuya, and Ken Sakurada. “OpenVSLAM: A versatile visual SLAM framework”. In: *Proceedings of the 27th ACM International Conference on Multimedia*. 2019, pp. 2292–2295.

[173] Johannes L Schonberger and Jan-Michael Frahm. “Structure-from-motion revisited”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 4104–4113.

[174] Charles-Olivier Artizzu et al. “OmniFlowNet: a Perspective Neural Network Adaptation for Optical Flow Estimation in Omnidirectional Images”. In: *2020 25th International Conference on Pattern Recognition (ICPR).* IEEE. 2021, pp. 2657–2662.

[175] Zhengqi Li et al. “Learning the depths of moving people by watching frozen people”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2019, pp. 4521–4530.

[176] René Ranftl et al. “Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer”. In: *arXiv preprint arXiv:1907.01341* (2019).

[177] Bolei Zhou et al. “Scene parsing through ade20k dataset”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2017, pp. 633–641.

[178] Feng Qi. “Fast Neural Style Transfer for Video”. Bachelor’s Thesis. Waseda University, 2017.

[179] Michael Zink, Ramesh Sitaraman, and Klara Nahrstedt. “Scalable 360 video stream delivery: Challenges, solutions, and opportunities”. In: *Proceedings of the IEEE* 107.4 (2019), pp. 639–650.

[180] Wenyan Cong et al. “Dovenet: Deep image harmonization via domain verification”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2020, pp. 8394–8403.

[181] Dahun Kim et al. “Deep Video Inpainting”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 2019, pp. 5792–5801.

[182] Jian-Liang Lin et al. “Efficient projection and coding tools for 360 video”. In: *IEEE Journal on Emerging and Selected Topics in Circuits and Systems* 9.1 (2019), pp. 84–97.

[183] Miguel Granados et al. “Background inpainting for videos with dynamic objects and a free-moving camera”. In: *European Conference on Computer Vision.* Springer. 2012, pp. 682–695.

[184] Nick C Tang et al. “Video inpainting on digitized vintage films via maintaining spatiotemporal continuity”. In: *IEEE Transactions on Multimedia* 13.4 (2011), pp. 602–614.

[185] Olivier Le Meur, Josselin Gautier, and Christine Guillemot. “Examplar-based inpainting based on local geometry”. In: *2011 18th IEEE international conference on image processing.* IEEE. 2011, pp. 3401–3404.

[186] Marcelo Bertalmío et al. “Pde-based image and surface inpainting”. In: *Handbook of mathematical models in computer vision.* Springer, 2006, pp. 33–61.

[187] Tijana Ružić and Aleksandra Pižurica. “Context-aware patch-based image inpainting using Markov random field modeling”. In: *IEEE transactions on image processing* 24.1 (2014), pp. 444–456.

[188] Deepak Pathak et al. “Context encoders: Feature learning by inpainting”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2536–2544.

[189] Yanhong Zeng et al. “Learning pyramid-context encoder network for high-quality image inpainting”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 1486–1494.

[190] Miguel Granados et al. “How not to be seen—object removal from videos of crowded scenes”. In: *Computer Graphics Forum*. Vol. 31. 2pt1. Wiley Online Library. 2012, pp. 219–228.

[191] Kedar A Patwardhan, Guillermo Sapiro, and Marcelo Bertalmío. “Video inpainting under constrained camera motion”. In: *IEEE Transactions on Image Processing* 16.2 (2007), pp. 545–553.

[192] Qiang Wang et al. “Fast Online Object Tracking and Segmentation: A Unifying Approach”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2019).

[193] Daniel Cohen-Or et al. “Color harmonization”. In: *ACM SIGGRAPH 2006 Papers*. 2006, pp. 624–630.

[194] Jun-Yan Zhu et al. “Learning a discriminative model for the perception of realism in composite images”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015, pp. 3943–3951.

[195] Yi-Hsuan Tsai et al. “Deep image harmonization”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 3789–3797.

[196] Haozhi Huang et al. “Temporally coherent video harmonization using adversarial networks”. In: *arXiv preprint arXiv:1809.01372* (2018).

[197] Weijiang Feng et al. “Audio visual speech recognition with multimodal recurrent neural networks”. In: *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2017, pp. 681–688.

[198] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. “A Neural Algorithm of Artistic Style”. In: *CoRR* abs/1508.06576 (2015). URL: http://arxiv.org/abs/1508.06576.

[199] Alexander G. Anderson et al. “DeepMovie: Using Optical Flow and Deep Neural Networks to Stylize Movies”. In: *CoRR* abs/1605.08153 (2016). URL: http://arxiv.org/abs/1605.08153.

[200] Alex J. Champandard. “Semantic Style Transfer and Turning Two-Bit Doodles into Fine Artworks”. In: *CoRR* abs/1603.01768 (2016). URL: http://arxiv.org/abs/1603.01768.

[201] Rujie Yin. “Content aware neural style transfer”. In: *arXiv preprint arXiv:1601.04568* (2016).

[202] Alexei A Efros and William T Freeman. “Image quilting for texture synthesis and transfer”. In: *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. 2001, pp. 341–346.

[203] Aaron Hertzmann et al. “Image analogies”. In: *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM. 2001, pp. 327–340.

[204] Hochang Lee et al. “Directional texture transfer”. In: *Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering*. 2010, pp. 43–48.

[205] Chuan Li and Michael Wand. “Combining markov random fields and convolutional neural networks for image synthesis”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2479–2486.

[206] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. “Texture synthesis and the controlled generation of natural stimuli using convolutional neural networks”. In: *arXiv preprint arXiv:1505.07376* 12 (2015).

[207] Dmitry Ulyanov et al. “Texture Networks: Feed-forward Synthesis of Textures and Stylized Images”. In: *CoRR* abs/1603.03417 (2016). URL: http://arxiv.org/abs/1603.03417.

[208] Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. “A learned representation for artistic style”. In: *arXiv preprint arXiv:1610.07629* (2016).

[209] Xun Huang and Serge J. Belongie. “Arbitrary Style Transfer in Real-time with Adaptive Instance Normalization”. In: *CoRR* abs/1703.06868 (2017). URL: http://arxiv.org/abs/1703.06868.

[210] Jun-Yan Zhu et al. “Unpaired image-to-image translation using cycle-consistent adversarial networks”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2223–2232.

[211] Manuel Ruder, Alexey Dosovitskiy, and Thomas Brox. “Artistic style transfer for videos”. In: *CoRR* abs/1604.08610 (2016). URL: http://arxiv.org/abs/1604.08610.

[212] Tsung-Yi Lin et al. “Microsoft coco: Common objects in context”. In: *European conference on computer vision*. Springer. 2014, pp. 740–755.

Appendix A

List of Publications

Journal

1. Nozawa Naoki, Shum P. H. Hubert, Feng Qi, Ho S. L. Edmond, Morishima Shigeo, "3D car shape reconstruction from a contour sketch using GAN and lazy learning", The Visual Computer, 1-14, April 2021.

2. Feng Qi, Hubert P. H. Shum, Shigeo Morishima, "Resolving hand-object occlusion for mixed reality with joint deep learning and model optimization", Computer Animation and Virtual Worlds, 31(4-5), e1956, September 2020.

3. Shimamura Ryo, Feng Qi, Koyama Yuki, Nakatsuka Takayuki, Fukayama Satoru, Hamasaki Masahiro, Goto Masataka, Morishima Shigeo, "Audio–visual object removal in 360-degree videos", The Visual Computer, 36(10), 2117-2128, October 2020.

4. Feng Qi, Hubert P. H. Shum, Shigeo Morishima, "Foreground-aware Dense Depth Estimation for 360 Images", Journal of WSCG, 28(1-2), 79-88, June 2020.

Conference

1. Feng Qi, Hubert P. H. Shum, Shigeo Morishima, "360 Depth Estimation in the Wild - The Depth360 Dataset and the SegFuse Network", IEEE conference on virtual reality and 3D user interfaces (VR), Pages 664-673, New Zealand (online), March 2022.

2. Feng Qi, Hubert P. H. Shum, Shigeo Morishima, "Bi-projection-based Foreground-aware Omnidirectional Depth Prediction", Visual Computing + VC Communications, Pages 1-6, Tokyo (online), September 2021.

3. Feng Qi, Hubert P. H. Shum, Shigeo Morishima, "Foreground-aware Dense Depth Estimation for 360 Images", International Conference in Central Europe on

Computer Graphics, Visualization and Computer Vision, Pages 79-88, The Czech Republic (online), May 2020.

4. Shimamura Ryo, Feng Qi, Koyama Yuki, Nakatsuka Takayuki, Fukayama Satoru, Hamasaki Masahiro, Goto Masataka, Morishima Shigeo, "Audio–visual object removal in 360-degree videos", Computer Graphics International 2020, Pages 1-8, Geneva (online), October 2020.

5. Feng Qi, Hubert P. H. Shum, Shigeo Morishima, "Resolving occlusion for 3D object manipulation with hands in mixed reality", Proceedings of the 24th ACM Symposium on Virtual Reality Software and Technology, Pages 1-2, Tokyo, November 2018.

6. Feng Qi, Hubert P. H. Shum, Shigeo Morishima, "Occlusion for 3D Object Manipulation with Hands in Augmented Reality", In Proceedings of The 21st Meeting on Image Recognition and Understanding, Pages 1-4, Sapporo, August 2018.

Acknowledgements

First and foremost, I would like to show my deep and sincere gratitude to my research supervisor, Professor Shigeo Morishima for his patient instructions and kind support throughout my entire graduate. The completion of my projects and this thesis will not be possible without his unwavering guidance. Without all the valuable opportunities provided by him along with my graduate life, it will be extremely difficult to extend my professional knowledge, acquire useful skills, and meet different people. All of them will be the greatest treasure in my life.

A debt of gratitude is also owed to my vice supervisor, Hubert P. H. Shum from Durham University, for his invaluable guidance throughout my academic career. His insightful suggestions and tremendous effort have had a profound impact on my ability to carry out research in a professional manner. I very much appreciate his help as a referee for this dissertation. I would also like to extend my genuine appreciation to Professor Hideyuki Sawada, for his constructive advice for my dissertation and keen help as a referee for my dissertation defense.

I would like to show my appreciation to my collaborators from the National Institute of Advanced Industrial Science and Technology, Dr. Yuki Koyama, Dr. Satoru Fukayama, and Prof. Masahiro Hamasaki. The helpful discussions with them have inspired and motivated my research.

I had the great pleasure of working with all student members from the Morishima laboratory. As a student studying abroad, it has been a wonderful experience to exchange ideas from different cultures and customs. My deep appreciation is owed to Dr. Takayuki Nakatsuka, Dr. Naoki Nozawa, Dr. Shintaro Yamamoto, and Dr. Shugo Yamaguchi for your kind discussion with me. I am also grateful to the current and graduated MIC group members. It was my privilege to have the opportunity to meet and work with them.

Last but not least, I would like to express my appreciation to my family and all my friends who provide me with persistent encouragement and great care throughout my entire life.