Perceiver – A Five-view Stereo System for High-quality Disparity Generation and its Application in Video Post-Production

by

Chang An Zhu

A thesis submitted to the Faculty of Graduate and Postdoctoral Affairs in partial fulfillment of the requirements for the degree of

Master of Digital Media

in

Digital Media

Carleton University
Ottawa, Ontario

Abstract

Element extraction from videos has always been a time-consuming process in the entertainment industry. In this research, we explored the possibility of simplifying the video object extraction technique with corresponding depth sequences. Based on post-production quality requirements, we developed our disparity enhancing system by integrating our two-axis-multi-view-stereo method that perceives an environment from five different perspectives on both x and y axes. Our research results have shown that the disparity quality of our approach is both visually and quantitatively more accurate than the traditional one-stereo-pair method, and its object extraction (i.e., matting) quality is comparable with existing mature matting technique to a certain extent. This research output can be applied in video object cut-out, visual effects composition, video's 2D to 3D conversion, and image post-processing. With further improvement, our system might be applicable in AR, VR, machine vision, and auto-pilot areas.

Acknowledgements

The journey of doing this research was quite an unforgettable experience. Although there are still many imperfections in this research, I'm proud of what I have learnt during this process and would love to continuously update this work to present a better outcome.

This project would not have been finished without my supervisor, prof. Chris Joslin, I would like to sincerely thank him for his guidance and advice in the numerous studies and a variety of research methods. I also greatly appreciate his support on the equipment and laboratory that I won't be able to finish my 3D printing right before the shutdown of COVID-19 without his help.

I would like to thank prof. Vincent Nozick from Université Gustave Eiffel, who kindly updated his multi-view rectification program to support my research.

I also want to thank my friend Alf who helped me get hands-on the 3D printer so efficiently, it was such a delightful moment for us to finally fit the models into the 3D printer. My friends Archika, Rana, Jinay, Ayoub, Theo, Abdihakim, Thankgod, and Leroy, who always exchange ideas about studying with me and have given me so much encouragement, they are a big part of my beautiful memories during the master study.

Finally, I am grateful for my family's support and encouragement during my master's study. It might sound a bit funny but my parents and my husband have spent so much their precious time cooking and watering the lawn while letting me focus on my researches. I love them with all my heart.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction & Background

1.1 Elements Extraction in Entertainment Industry

In modern entertainment industry, video element extraction is one of the most widely used techniques in post-production, the idea is to isolate the target elements from existed video clips to either composite them in another scene or to insert new elements between or behind them.

In terms of the entertainment media genres such as live action movie, television series, advertisement and video game, the industry has various quality requirements on element extraction, where live action movies have the highest standards. An excellent element extraction result shall have decent accuracy relative to the original video resolution and outstanding consistency on element boundaries and details that enables a natural and realistic composited result.

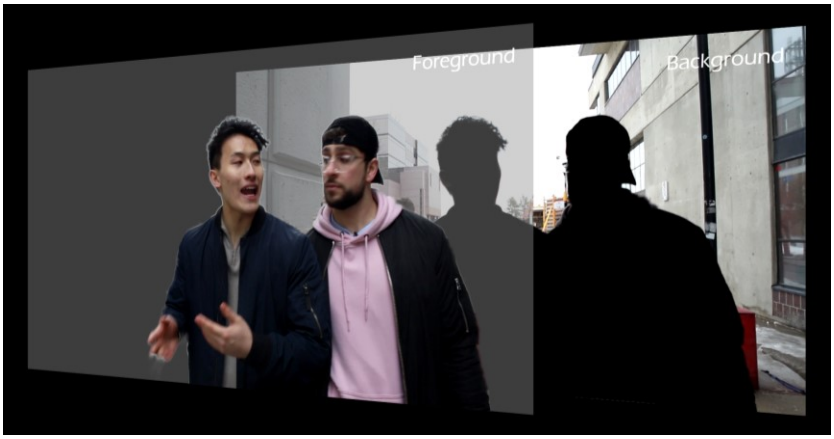


Figure 1.1 - The illustration of video element extraction

1.1.1 Chroma Keying

To achieve this goal, the visual effects industry has made endeavors on countless technologies and methods. Chroma keying, also known as color keying, is one of the

most effortless and accurate approaches. By filming the target element against a constant color background (usually green or blue since they differ most distinctly from human skin tones), the key color is selected to define transparency, and post-editors can replace the background with other environments or visual effects.

Figure 1.2 - The scenic setup used in The Hobbits [1]

Chroma keying is not only useful in the film industry but also widely applied in many real-time scenarios such as live newscast and weather forecast. However, it takes unnecessary costs on the lighting and scenic setup when a shot only needs to be filmed in a real-world environment.

The Keylight by The Framestore and Foundry is a good example. It is an industry-proven color difference keyer, which is compatible with most of current high-end compositing software and is almost an automatic progress. Its presence significantly reduced VFX artists' burden and is loved by the industry for decades, yet issues such as

color-spill and unexpected shadow due to improper lighting are influencing its quality. Besides, in area that is not covered by green or blue screen, it still requires manual matting for the target element.

1.1.2 Roto-scoping

Roto-scoping is another popular technique for element extraction, it is a process of tracing object silhouette frame by frame from the video or film footage, which allows to cut an element out from the footage and to insert effects between foreground and background. Moreover, it is also used to paint animated effects or remove unexpected objects.

The routines of roto-scoping differ in terms of the functions it serves, e.g., for element extraction, artists create vector shapes (i.e., splines) to manually cut an element out of its background and reposition these shapes on various keyframes (with the help of frame interpolation) to get the dynamic result, by breaking down big elements into multiple roto splines (see in Figure 1.3), a software can interpolate more accurately; for some simple effects and matte generation, artists directly paint 2D animation or effects on each keyframe, mattes can be created with this process as well; as for object or wire removal, spatial cloning allows artists to clone a pixel from a frame to another position of the frame; and temporal cloning paints a pixel from one frame to another.

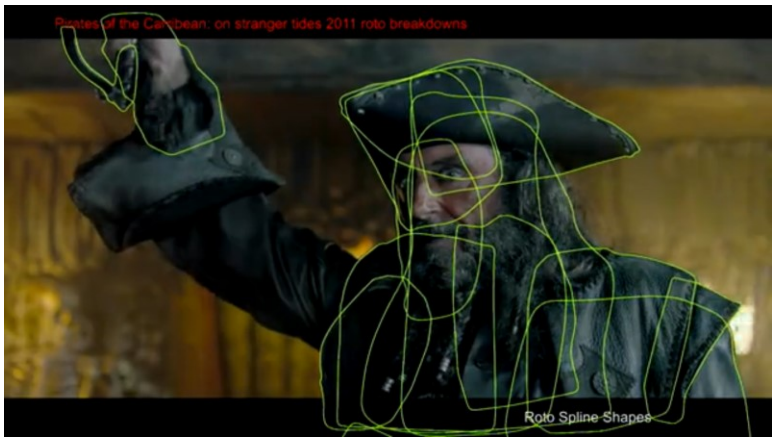


Figure 1.3 - Roto-splining in Pirates of the Caribbean [2]

In the early years, rotoscoping was used in animation pipeline for artists to trace over motion pictures to produce realistic action. It was applied in many famous animation movies such as Snow White, Alice's Adventure in Wonderland and The Little Mermaid. At the same time, some live-action movies integrated rotoscoping in shots to create visual effects (e.g., The Birds (1963) by Alfred Hitchcock, Mary Poppins (1964) by Robert Stevenson).

Figure 1.4 - The Little Mermaid rotoscoping before and after [3]

In modern post-production, rotoscoping has progressed into a much more standardized workflow and has been extended to a broader usage like 2D to 3D conversion (Figure 1.5). Although roto-scoping has multiple branches like matte creation, cloning, and painting, element extraction is still the main role of it.

At the same time, new tools have been developed to reduce the workload of it. Software such as Nuke, After Effects, Photoshop and Fusion all provide various solutions for simplifying rotoscoping (e.g., keyframe interpolation in roto-splining, auto-painting), yet roto-scoping is still having disadvantages on consistency and details and involves many manual adjustments that are tedious and time-consuming.

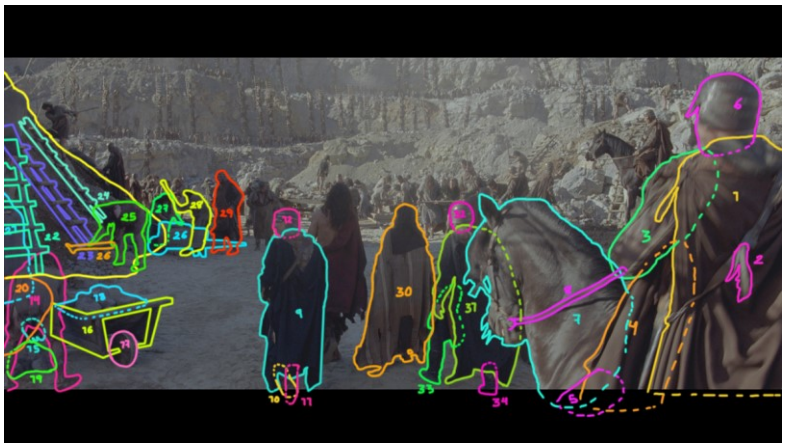


Figure 1.5 - Rotoscoping in Stereo Conversion [4]

1.1.3 Motion Tracking

Motion tracking is a strong tool that is playing a significant role in post-production, its performance in element extraction is also outstanding. In a 2D tracking, the program analyzes a pixel or subpixel in a video and follows the pixel or subpixel to find out its exact coordinate on each frame. In this case, the computer can automatically track the contour of a certain object or stabilize the camera shake, meanwhile the post-production artists can use this information to extract an element or composite elements into the clip. In a 3D tracking, i.e., match moving, computer program extracts feature points from 2D video footage and recreates a virtual camera that could precisely render a virtual object with the exact same movement of the real camera. Besides, it enables to

composite the virtual object into live-action shots with correct scale, position, orientation, and motion.

Figure 1.6 - 2D motion tracking

Figure 1.7 - Match moving in Boujou

Motion tracking is a helpful tool to produce convincing rotoscoping result and it stands out in spatial accuracy in the area of stereo conversion, there are many software dedicate to motion tracking such as Mocha, Flame, and Silhouette.

On the other hand, motion tracking is limited under several situations: The contour tends to jitter when the target is blocked by other objects (e.g. a passenger passes

by the main character) and motion blur or over exposure could cause shifting of the trackers, which means a certain amount of manual assistance in this process is still unavoidable.

1.2 Problem Description

As is discussed in Section 1.1, the requirement of element extraction exists in almost every shot that involves visual effects.

Current element extraction solutions in visual effects industry are either constrained by the environment or high cost. If under ideal situation (e.g., clear background, a small amount of motion blur, big screen size, foreground color distinct from the background significantly), a rotoscoping process or a motion tracking might take less time to edit; a chromakey method does not take as much work as rotoscoping, but requires more preparation in the preliminary setup of the background screen and the lighting of the scene. When under most real-life video shooting environments (e.g., complex background, shadows on the green screen, limited screen size, foreground color similar to the background color), the processing time of rotoscoping might significantly increase and the accuracy of motion tracking might drop. At the same time, the editing time of chromakeying will also be influenced by the color spill and shadows on the background, requiring further editing like garbage matte and might cause some inevitable detail loss.

As we aim to simplify the element extraction process through disparity sequences, we shall create a high-quality disparity sequence with accurate depth information since the quality of element extraction is a decisive factor that determines whether a method is usable in post-production. Existing active depth acquisition methods such as Time-of-

Flight and LiDar are constrained by environment and spatial density, while passive method like correspondence stereo presents limited accuracy due to the discontinuities in stereo pair such as specular and occluded areas (See more details in Section 2.2). Although researches have made endeavors on enhancing disparity accuracy, few of the results can satisfy the quality requirements and be applicable in the post-production pipeline as object extraction tools.

Thus, our objective is to develop a solution that could automatically generate high-accuracy disparity sequences by reducing discontinuities in stereo pairs; and use the results to separate the target elements from a video without any additional tools or setup. The output might hugely improve film post-production efficiency, free many VFX artists from tedious manual operations, and lower the cost of visual effects.

In this research, we also want to argue that whether the computational cost is the priority factor in a video element extraction scenario. From our understanding, the critical point of automating this procedure is to free the VFX artists from repeated work to save time for other tasks that require more human intelligence. Similar to rendering, a high-quality element extraction approach might take some time to process due to the limitation of current computing capability, however, under the condition of constraining the processing time within a specific range, giving a high-quality output that is usable in a high standard scenario is our primary goal.

1.3 Proposed Solution

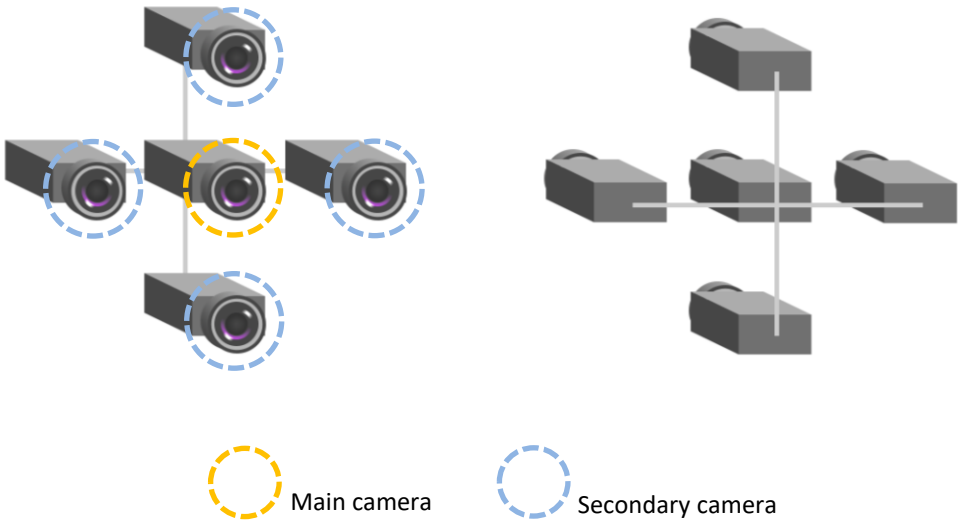


Figure 1.8 - Illustration of our five-camera setup

We propose a system that integrated a variant of multi-baseline stereo to perceive depth with five high-resolution cameras. By always considering the center camera as the reference, we acquire four stereo pairs from horizontal and vertical directions that centripetally align the disparity results. In this case, we can interpolate the missing information (i.e., occluded area) from one axis to another.

From a post-production point of view, disparities acquired from this system might be an efficient assistive tool for the element extraction process.

1.4 Contribution

Our contribution mainly consists of the following parts:

- We first suggested a depth acquisition procedure that is fully compatible with two non-overlapped baseline axes.

- We designed and built a five-camera system that is applicable in real-life scenario and can capture five synchronized videos from both horizontal and vertical directions.
- We first proposed a two-axis image rectification solution.
- We designed an evaluation approach that helps quantify the foreground extraction quality that can be easily transplanted into similar researches.
- We first contributed a stereo vision dataset that involves stereo pairs from five views and two axes, which are:
    - the CG stereo pairs generated under a relatively ideal circumstance with the distinguishable color difference among objects and no spatial deviation among camera positions;
    - the CG pairs from semi-simulated real-life scenes with hard shadow and similar colors in objects;
    - the CG pairs from fully simulated real-life scenes that have hard shadows, low-light areas, similar-color objects, deviated camera position;
    - the corresponding depth and matte ground truth;
    - three sets of live-action videos captured by our five-camera system.
- We also provided a potential approach that might considerably reduce an element extraction workload and a depth composition option for live-action objects.

1.5 Thesis Organization

The following content structures as follows:

- Chapter two reviewed previous literature on element extraction and presented our pre-experiment design and result.
- Chapter three described our research methodology, which includes the overview of our five-camera system, the general breakdown of our algorithm, and our programming environment.
- Chapter four presented our design to quantify the quality of element extraction and the experiment based on it to compare our method with other methods.
- Chapter five discussed the result and findings of our research; chapter six summarized this thesis and its contribution.

Chapter 2: Related Works

With the progression of photography resolution and the rising of people's consciousness of picture quality, the entertainment industry is pushing media resolution to its limit, which requires the supplementary techniques to match this standard in every aspect. Unfortunately, some processes that previously take many labor works such as rotoscoping are getting even more complicated and time-consuming due to this growing demand for quality. This has led companies and studios to look for optimizing solutions or potentially alternative techniques. Depth information, however, directly shows the order of elements in a scene by indicating the distance between the elements and the camera, which could possibly become the most potent assistive tool to help to separate video stream into ordered layers. Thus, using depth information as an assistive tool to speed up the current video element extraction process has been explored by researchers such as Kanade et al. [5], Gvili et al. [6], and Wang et al. [7], and have had some promising result.

Figure 2.1 - An ideal disparity map from 3d environment rendering [8]

On the other hand, the depth reconstruction techniques are updated rapidly due to its broad applicability, some methods to enhance the reconstruction results are also well-

discussed by researchers like Park et al. [9], He et al. [10], and Hosni et al. [11]. In the first section of this chapter, we introduced some representative element extraction approaches in academia; in the second section, current depth acquisition methods are introduced while the emphasis locates on the passive stereo and recent correspondence algorithms; the third section described current depth enhancement methods; section four discussed the advantages and disadvantages of exiting element extraction methods, depth acquisition methods, and depth enhancement method.

2.1 Current Element Extraction Approaches in Academia

In our research, element extraction is a general idea that indicates the process of pulling target elements out from a video stream, which includes many widely known concepts in academia such as Video Matting, Object Extraction, and Segmentation. Current element extraction solutions in academia fall in the categories of color keying methods, alpha matting methods, and other innovative methods like depth and defocus.

According to the most relevant literature we found, the idea of using depth information as an element extraction tool was presented early. In 1996, T. Kanade [5] and his team had explored the application of their stereo machine for video-rate depth mapping, which first addressed the possibility of using depth to isolate elements from a video clip automatically. In 2003, Givili et al. [6] aroused the concept of “depth keying”, which proposed to separate foreground objects from the background using their relative distance from the camera. After this, video matting and object extraction seemed to be more accessible, and various approaches were presented such as the combination of camera arrays and image variance that could even handle alpha and work in real-time

[12], and the application of new technology that uses time-of-flight camera to extract foreground [7].

2.1.1 Color Keying Methods

The color keying element extraction has the most extended history in both academia and industry. Dating back to 1940, Larry Buttler first suggested the Traveling Matte process, which splits the three primary colors into three films and utilizes blue as the key color. The target elements are shot against the blue screen while intensify the blue channel to finally generate the mask [13]. In 1959, Petro Vlahos applied a similar green screen technique after he accepted the challenge from MGM to create a solution for the movie *Ben-Hur* [14].

With the progressing of digital composition, more approaches were presented for element extraction that contains alpha information instead of a simple silhouette. The algorithms developed in these approaches are now branched into color difference key and color vector key.

Color Difference Key: As is mentioned above, color difference keying is the very first method of element extraction. Further developed by Petro Vlahos [15], color difference based chroma key was used in the first studio-level digital color difference keyer – Ultimatte. The latest version of Ultimatte can work in real-time with outstanding edge handling, color separation, and spill suppression even in a dark shadow area or through transparent objects.

The Keylight developed by Framestore and Foundry also adapted the idea of color difference keying, which is to create transparency by splitting an image into matte A and B. Matte A bases transparency on regions that do not contain a second different color and

matte B uses the specified key color as an accordance of the transparency. By combining the two mattes into an alpha matte, the Color Difference Key could provide more accurate transparency values [16].

Color Difference Key provides high-quality matting result for objects that are evenly lit-up and filmed against a blue or greenscreen, and especially stands out when working with images containing transparent or translucent substances such as smoke, shadows, and glass [16].

Color Vector Key: The idea of color vector key is to consider the RGB components of each pixel as a coordinate(e.g., x, y, z value) of a point in 3D space, take some sample pixels that is known to be the key color (e.g., pixels near the edges) and average them to create a reference point or plane that represents the average chroma key color. Any pixel within a specific range from the reference can be considered part of the background, while pixels further away will be considered to be part of the foreground.

In 1994, Y. Mishima presented the Hexoctahedral Color Space Model [17]; it suggests obtaining the improved foreground by separating colors into four regions: absolute background colors, absolute foreground colors, blended foreground and background colors around the boundaries, blended foreground colors with spilled background colors, which provided better isolation of various color vector and helped to preserve the colors in the foreground [18].

Undoubtedly, the color keying solution is one of the most practical element extraction methods so far, which meets the requirements of entertainment media at any level in terms of quality and efficiency. Nevertheless, just like prior research mentioned

that on color keying element extraction, this method is constrained within a specific environment that requires particular background and other setups.

2.1.2 Alpha Matting Methods

Comparing the alpha matting methods with the color keying method, the alpha matting is more often discussed in academia, which mostly located in the area of computer vision.

According to a survey from Wang and Cohen, there are two main categories of alpha matting element extraction: pixel sampling and pixel affinity [19]. Pixel sampling methods estimate the alpha values by sampling within sets of certain foreground and background colors, while pixel affinity analyze the spatial neighborhood of selected pixels to get the alpha [19]. In this section, we will introduce the presentative algorithms in both pixel sampling and pixel affinity area as well as pointing out the advantages and limitations of them.

Pixel Sampling: In 2000, Ruzon and Tomasi [20] proposed the parametric sampling algorithm, which estimates alpha value along a manifold connecting the frontiers of each object's color distribution. Their approach assumes the unknown region to be a narrow band, the selection of anchor points and the creation of non-overlapping local windows are ad hoc, where the colors are modeled by non-oriented Gaussians with diagonal covariance matrices, which may generate significant fitting errors for textured regions [19]. Since the unknown pixels' alpha values are computed independently, it could also lead to discontinuities in the matting result.

Another representative pixel-sampling-based approach is Bayesian matting. In 2001, Chuang et al. [21] presented Bayesian matting, which shows how complex mattes

can be generated. Their method modeled colors as mixtures of Gaussians, but used a continuously sliding window to define the neighborhood, where the foreground and background Gaussians can get influenced by all the pixels around the neighborhood. As a result, it generates accurate mattes when the given trimap is well-specified. However, the result can be noisy when the provided trimap is rough or when the input image contains highly textured areas.

Pixel Affinities: Pixel affinity method has an advantage over the sampling-based method in the following aspects: since affinities are defined in a relatively small neighborhood, the correlation of pixels is higher, making the smoothness assumption work even for the intricate image. When encounter rough input trimap, a sampling-based method has to collect samples that are far from the target pixel, thus the samples may become useless [19]. In contrast, in such situations, the affinity-based method tends to regularize the result to be locally smooth, which helped avoid the discontinuity issue that might happen in the sampling-based method.

One representative algorithm of affinity-based matting is Poisson Matting [22], a process that estimates the gradient of matte from the image, then reconstructs the matte by solving Poisson equations. As it directly operates on the matte gradient, it reduces the error from the misclassified color samples in a high-texture shot.

In the research of Sun et al. [22], they assume that intensity change in the foreground and background is smooth. Thus, they proposed a global and a local Poisson matting, when the global method fails to produce high-quality matte due to a complex background, the local method would manipulate a continuous gradient filed in a local

region, which brings in human interaction [22]. Typically, the approximate matte gradient is obtained by taking partial derivatives on the matting equation [22]:

$$\nabla I_z = (F_z - B_z)\nabla\alpha_z + \alpha_z\nabla F_z + (1 - \alpha_z)\nabla B_z \quad (1)$$

Where $\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}\right)$ is the gradient operator. When foreground F and background B are smooth, an approximate matte gradient $\nabla\alpha_z$ can be defined as [22]:

$$\nabla\alpha_z = \frac{\nabla I_z}{F_z - B_z} \quad (2)$$

Hence, $F_z - B_z$ shall be known to get the absolute gradient value. In the work of Sun et al., F_z and B_z are selected as the foreground and background colors for unknown pixels [19], which limited the accuracy of the matte. To eliminate this issue, they introduced the user-interaction to manually refine the matte, which ended up with good result but could involve too much extra labor work.

Later, Levin et al. [23] presented a closed-form solution in 2007, where they use symmetric Laplacian matrix and minimize a quadratic function to propagate the frontier constrains. Although the closed-form approach stands out from most affinity-based approaches due to its explicit derivation of cost function from local smoothness assumptions on the foreground and background colors [23], it is limited in propagation. In this case, the isolated unknown regions that are only surrounded by foreground or background regions may get incorrect alpha value while unknown regions might get propagation errors. Similar to Poisson matting, this approach requires manual adjustment.

2.1.3 Depth-Based Methods

Depth-based element extraction is relatively new by comparing it with the two methods above. As the depth-related technologies are getting mature, researchers started

making attempts to extract element with the aid of depth map or depth camera. The current most advanced system of depth-based element extraction is the zKey by zLense. zKey is a screenless 3D keying system aims at augmented reality application, which could output the matte and alpha channel with its the depth camera attachment in their system.

Figure 2.2 − The real-time screenless keying from zKey, the character occludes with the virtual earth figures based on the depth changing [24]

The zLense does not disclose their technology behind zKey, while Takeo Kanade from CMU, who is one of the forerunners of depth-based keying, has raised the concept of Z-Key in 1995. According to Kanade [5], the basic idea of Z-Keying is to use the pixel-by-pixel depth information in the form of a depth map as a switch; the Z-Key compares each pixel's depth value to the virtual objects' depth value, extracts pixels that are closer to the camera, merge them with the virtual image and output. As a result, they can create a merged image where real and virtual elements occlude correctly.

Although exciting application has been created and wide applicable areas could be imagined, depth-based element extraction is still under research, in most cases, people

only use depth information as an assistive tool for better segmentation or matting result. In the following section, we will discuss a few representative depth-based element extraction solutions currently available.

Defocus: Researchers have started their exploration on depth from defocus since the 1960s. Depth from Defocus (DFD) is an optimized idea of Depth from Focus (DFF), which is based on the fact that in a convex lens optical system, the elements in a scene will be blurred if they are not at the same distance as the focused element.

According to the lens equation (Equation 3), we can get the distance ***u*** between optical center and the target element, by knowing the focal length ***f*** and the distance ***v*** between lens center and the focused image (Figure 2.3). To find the ***u***, focal length f and the distance ***s*** between image detector ***ID*** and lens center must be known, thus, the DFF method needs an adjustment of ***f*** and ***s*** until getting the clearest target element image. In this case, the DFF requires 10 to 12 images to get a result while the scene must remain the same during the process of taking these pictures.

$$\frac{1}{f} = \frac{1}{u} + \frac{1}{v} \tag{3}$$

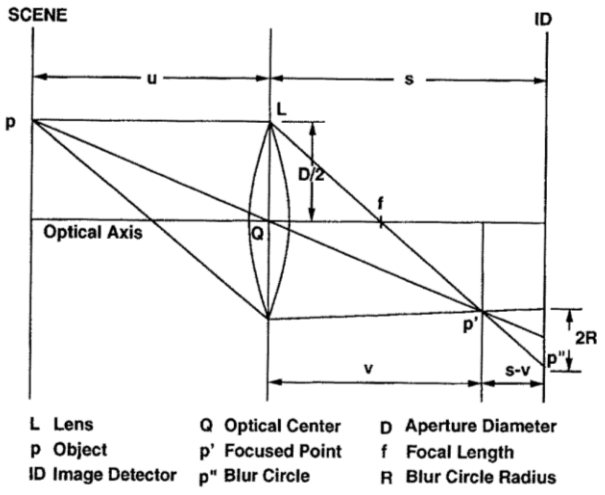


Figure 2.3 - The convex lens [25]

To improve this situation, researchers Subbarao et al. [25] presented a method named Depth from Defocus (DFD), which determines depth from image defocus and rapid autofocusing. Since it is based on one-dimensional Fourier coefficients and does not restrict camera parameters, the DFD is computationally advanced from DFF, which only requires 2-3 images to get the depth information. DFD has been implemented on Stonybrook Passive Autofocusing and Ranging Camera System SPARCS, which reported a result with an accuracy of 3.7% RMS error in autofocusing, 4% RMS error at 0.6m distance and changed linearly to about 30% RMS error at 5.0m distance in ranging application [25].

In 2005, McGuire et al. [26] investigated the potential of integrating DFD into video matting that is capable of automatically pulling out a matte with multiple synchronized video streams that share same perspective but different focus. The solution is to extract a trimap from the varying focus videos, constrain optimization and post-process to remove noise and get the most probable matte.

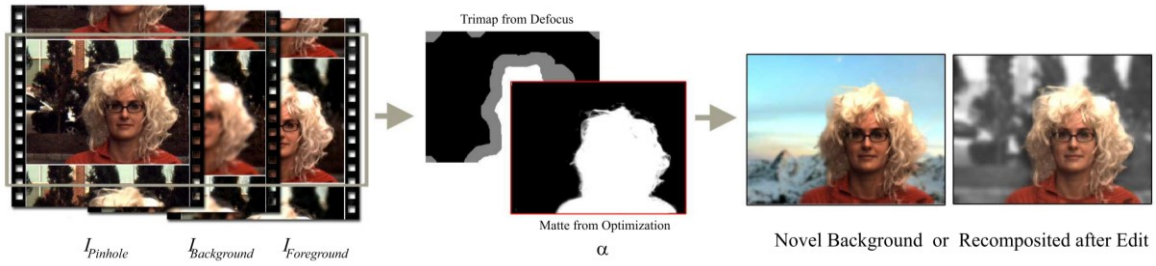


Figure 2.4 - The process of defocus matting [26]

However, the Defocus-based video matting is limited under several situations. Since the defocus approach is based on depth discontinuity, two objects that are too close to each other are unable to be distinguished; it is hard to accurately process an over-exposed area or separate elements with high level of motion blur from the background.

Its accuracy is also constrained if the foreground and background of a scene are visually hard to distinguish.

Time-of-flight: Time-of-flight (TOF) is a relatively new technology that measures distance based on the duration of a signal's round trip between the sensor and the objects. Current TOF cameras are available with signals of infrared laser and LED, which can create high resolution depth map relative to prior technologies.

In 2007, Wang et al. [27] suggested adding depth information acquired by time-of-flight camera to enhance natural video matting results, they claim that the additional depth information reduced the ambiguities arise from elements with similar color. In their work, they first auto-generated an accurate trimap through an unsampling, thresholding, and dilating process; then use the high-resolution depth information to disambiguate regions that are prone to error when using standard natural matting such as Bayesian Matting and Poisson Matting [27].

Similar example of integrating depth information into standard matting process can also be found in the work of Wang et al. in 2010 [28]. They fused color and depth cues in a unified framework while an adaptive weighting scheme is used to control the color and depth [28].

A limitation of these approaches is that they require a dividing value of distance to separate the foreground and background, which means the target element has to move along a certain plane paralleling to the camera, besides, objects or elements that are closer to the camera than the target will also be counted as the foreground.

2.1.4 Other Approaches

Contour Tracking: Looking into some commercial post-production software such as the built-in tracking tools in After Effects, they are only capable of feature points tracking, which are far from enough in an element extraction context. However, researchers from computer vision and image processing areas have provides abundant researches on contour tracking.

One of the most representative methods in contour tracking is *Snake*, proposed by Kass et al. [29] in 1988, it is an energy-minimizing spline guided by external constrains and influenced by image forces that pulls it towards lines and edges. The method allows dynamic energy function minimization, where user can pre-define the contour and the contour will be automatically refined to fit the edges. However, this method only tracks forward, which made the frame that starts having tracking error very ambiguous while user correction does not prevent the eventual tracking error.

In 2004, Agarwala et al. [30] improved the contour tracking by combining the energy function and user interaction, where the users can adjust the tracking curves and set the adjusted result as a keyframe while the algorithm executes later tracking according to the prior keyframe.

Later, researches using contour tracking as assistance for better matting results are also presented. Li et al. [31] in 2005 presented a system that can cut out video objects and paste them in another video. Their algorithm obtains the binary segmentation of video objects by: 1) a 3D graph-cut-based segmentation and a tracking-based local refinement; 2) using coherent matting to produce the alpha matte. According to Li et al. [31], their method generates better alpha matte than Bayesian Matting since it fully exploits the information in the binary segmentation with a regularization term for the alpha matte. In

2010, Chuang and Chen [32] presented a video segmentation method with the aid of Markov random field (MRF)-based contour tracking, where they used MRF contour tracking to propagate the shape of target object and graph-cut to refine the shape and improve accuracy.

The limitations of tracking-based element extraction are obvious that most of the tracking-based approaches require user interaction, discontinuities can be introduced due to the feature points lying on or beyond the image boundaries.

Pixel Variance: The video matting using camera array is an innovative method presented by Joshi et al. [12], vary from the defocus method mentioned above, their system uses several cameras at different perspectives to focus on the same target element. Thus, mattes are generated by creating a synthetic aperture image that is focused on the foreground, which reduces the variance of pixels re-projected from the foreground while increasing the variance of pixels from the background [12].

Based on the standard matting equation, a composited image *I(x, y)* can be blended from a background *B(x, y)* and a foreground *F(x, y)* with its alpha matte *α(x, y)* by the matting equation [33]:

$$\mathrm{I} = \alpha\mathrm{F} + (1 - \alpha)\mathrm{B} \tag{4}$$

Whereas in chroma keying, the *B* is known, in natural video matting, all variables α, F and B need to be estimated [22], which constrained the quality of natural matting or manual assistance has to be involved. Nevertheless, Joshi et al. project the color values from each camera to the depth of the foreground object, and use the mean and variance of from these values to automatically generate a trimap, *α* ,and *F* [12]. Their updated matte equation by avoiding the main constrains of natural video matting that has

to compute the background depth, which brings benefits to computational efficiency and is able to run in almost real-time [12].

$$\alpha = \begin{cases} 0 & \mathbf{var}(\mathbf{I}) > \max(\mathbf{var}(\mathbf{B}), \mathbf{var}(\mathbf{F})); \\ \frac{\mathbf{var}(\mathbf{B}) + \sqrt{\Delta}}{\mathbf{var}(\mathbf{B}) + \mathbf{var}(\mathbf{F})} & \mathbf{var}(\mathbf{B}) < \mathbf{var}(\mathbf{I}) \leq \mathbf{var}(\mathbf{F}); \\ \frac{\mathbf{var}(\mathbf{B}) - \sqrt{\Delta}}{\mathbf{var}(\mathbf{B}) + \mathbf{var}(\mathbf{F})} & \mathbf{var}(\mathbf{F}) < \mathbf{var}(\mathbf{I}) \leq \mathbf{var}(\mathbf{B}); \\ \frac{\mathbf{var}(\mathbf{B})}{\mathbf{var}(\mathbf{B}) + \mathbf{var}(\mathbf{F})} & \frac{\mathbf{var}(\mathbf{B})\mathbf{var}(\mathbf{F})}{\mathbf{var}(\mathbf{B}) + \mathbf{var}(\mathbf{F})} \leq \mathbf{var}(\mathbf{I}) \leq \min(\mathbf{var}(\mathbf{F}), \mathbf{var}(\mathbf{B})); \\ 1 & \mathbf{var}(\mathbf{I}) < \frac{\mathbf{var}(\mathbf{B})\mathbf{var}(\mathbf{F})}{\mathbf{var}(\mathbf{B}) + \mathbf{var}(\mathbf{F})}. \end{cases} \tag{5}$$

$$\alpha F = \langle I \rangle - (1 - \alpha)\langle B \rangle \tag{6}$$

They computed α and αF as above, where ⟨I⟩ is the mean of the corresponding pixel value in all images, α is recovered from Equation 5, and ⟨B⟩ is the mean of the background pixel values [12].

Figure 2.5 - The process of pixel variance matting [12]

On the other hand, this method is constrained in several aspects: The quality of the output is constrained by the aliasing in the light field, thus is can not be applied in outdoor scenarios; the alpha matte in this system are generated based on the assumption that it is fixed instead of view-dependent, while it against the rule of some self-occluded elements. Finally, they assume the background variance is far greater than the foreground

variance, which may not be true especially when foreground and background share the similar color and material.

2.2 Current Depth Acquisition Methods

Since our approach proposes to separate elements from video based on depth information, we also studied current depth acquisition methods. Depth acquisition is tightly connecting with the area of computer vision. The most popular way of classifying depth acquisition approaches is to categorize them into active and passive methods. The active methods typically employ active sensors to estimate the distance with the traveling time that light bounces back from an element or with the distortion of lights after projecting them on elements. In contrast, the passive method refers to techniques that attempt to find the corresponding points from a pair of images that share slightly different viewpoints of a scene. Moreover, approaches fused the advantages of both active and passive techniques are also presented. We will discuss researchers' endeavors on depth quality optimization in the third section.

2.2.1 Active Methods

Depth from Time-of-flight (TOF): As is introduced in Section 2.1.2, Time-of-Flight is a method that computes the depth map based on the time that light or laser (mostly infrared light) spent on its round trip from the emission point or camera center to the target element. Although the resolution of the disparity map from the TOF sensor cannot match the resolution of a standard RGB camera, there are plenty of ways presented [9] [34] that can efficiently enhance its quality.

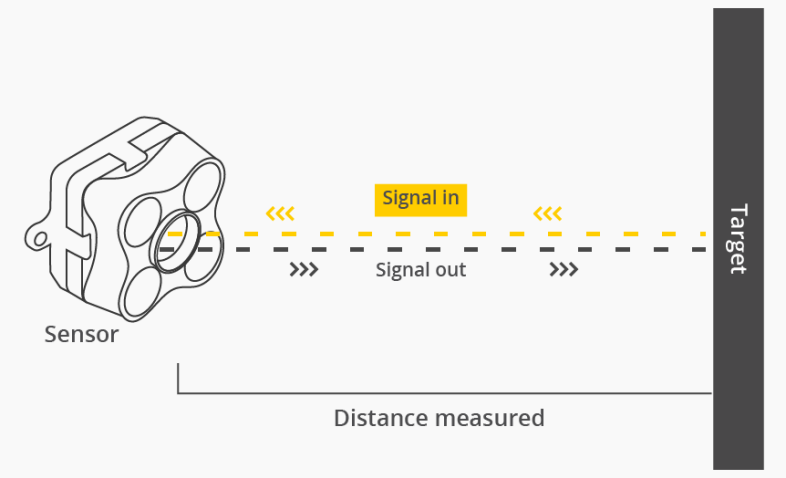


Figure 2.6 - The illustration of TOF sensor [35]

Benefit from its simple mechanism and straight-forward output, TOF-based depth acquisition is also suffering from many limitations. e.g., since TOF mostly uses infrared light as the emission signal, it can interfere with sunlight as sunlight also contains infrared light, which would create colossal noise or inaccuracy in the resulting map used in an outdoor environment. Besides, it is easy to be influenced by non-systematic deviation, such as strong reflection and scattering. The valid working distance of TOF devices is also limited; they can only perform satisfactorily up to a maximum distance of approximately 5-7 meters [36]. Due to these limitations, stereo-vision-based method (i.e., passive method) is more reliable for an outdoor environment [36].

Depth from LiDar: Sharing similar mechanism with Time-of-Flight, LiDar uses near-infrared laser to scan and reconstruct objects, which is more used for geographical and archeological purpose and there are also researches integrating LiDar into disparity generating process. In 2008, Strecha et al. [37] provided a dataset to investigate the possibility of replacing LIDAR with current stereo techniques. In 2018, Park et al. [38] fused LiDar and correspondence stereo (see in Section 2.2.2) with CNN, their result shows significantly higher accuracy than the baseline methods.

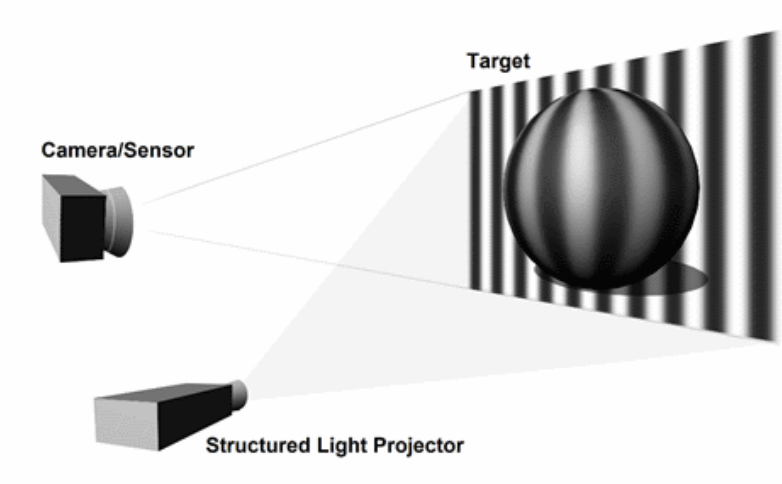


Figure 2.7 - A striped pattern is projected onto the ball. The rounded surface of the ball distorts the stripes, the distorted image is then captured by a camera for analysis and object reconstruction [39]

Depth from Structured Light: Vuylsteke and Oosterlinck [39] are of the forerunners of depth from structured light. In 1990, they presented a coding scheme on the basis of a single fixed binary encoded illumination pattern that contains all required identifiable information for individual strikes visible in a camera image [39].

With the developing of structured-light system, corresponding problems have arisen, e.g., what are the projection patterns that give most accurate result in a given total number of projection patterns and noise level? Horn and Kiryati [40] gave a solution to this problem in 1999. In their work, they referred the signal design of digital communication for the projecting patterns design, the structure light methods that seem to be unrelated like the intensity ratio technique and gray code scheme, are unified in their framework and have led to some promising results.

Instead of using only one camera, Scharstein and Szeliski [41] in 2003 presented a technique that uses two cameras and a light pattern projector to produce pairs of real-world images of complex scenes, where each pixel is labeled with its correspondence in the other image, which can be used to test the accuracy of stereo algorithms. In their

system, each camera determines a unique label for each pixel by using the light patterns. In this case, correspondence can be found by searching the pixel pairs with same label. One advantage of their work is that when a point/pixel is only visible in one image, its corresponding point/pixel can be estimated accurately, thus the output using their system is close to the ground truth. In the same year, Keller et al. [42] patented their methods and system for real-time structured light depth extraction. In 2011, Silberman and Fergus [43] implemented the indoor scene segmentation using structured light, which revealed the possibility of using structure light for scene understanding. A set of indoor scene data were introduced with accurate depth maps and dense label coverage, which shows that combination of depth and intensity images gives much higher performance than only using intensity images.

On the other side, structured light-based depth acquisition is suffering from its limitation on light interference, similar to the TOF method, a structured light-based system requires a strict control over lights that has to remove the ambient illumination, which does not fit any outdoor environment.

2.2.2 Passive Methods

Passive stereo is an early concept that inspired by the human visual system; it is an idea to recover the 3D information such as distance/depth from two slightly different viewpoints, which simulated the process that human eyes perceive the depth/distance by finding corresponding points in a scene. In stereo vision, the ideal camera setup to achieve stereo is parallel cameras setting (Figure 2.8), which makes the two cameras' image planes parallel; while in real-life, the cameras mostly follow a converged camera setting due to the positional deviation, where the two cameras' image planes angle toward

each other. The main challenge in this procedure is to find the proper method of estimating the difference between the two images and to map the correspondence of the environment.

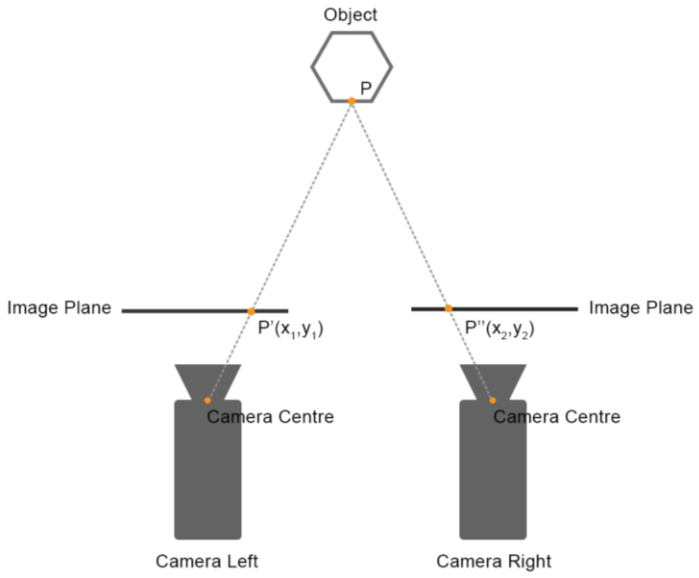


Figure 2.8 - Illustration of a stereo camera pair

According to Scharstein and Szeliski [44], current stereo matching algorithms mostly follow these four steps: matching cost computation, cost aggregation, disparity selection, and disparity refinement. Generally speaking, we can categorize the algorithms into local and global approaches [36].

The local method, also known as window-based method, computes a pixel value only by referring its intensity within a certain area, which only consider local information, thus, it has a lower computation cost. It obtains the disparity map through winner takes all (WTA) optimization, each given pixel is assigned with the corresponding disparity value that has the minimum cost [36].

The global method, on the contrary, considers disparity assignment as a process of minimizing a global energy function for all disparity values [36]. Disparity map from the

global method are created by assigning similar depth value to neighbour pixels, which gives better results but has higher computational cost.

Depth from Correspondence: The passive stereo has become a hot topic since the 1980s. In 1993, Fua [45] provided a correlation algorithm to compute reliable dense depth maps by using parallel techniques. His algorithm performs correlation over every image pixel and retain only matches that shown valid [45]. Comparing to prior correspondence algorithms that only attempt finding interest points on images, his approach gives higher reliability and accuracy of generated depth map.

In 1996, Kanade et al. [5] integrated the multi-baseline stereo algorithm in their multi-camera stereo machine, which consists of three steps: first, filtering the input images with the Laplacian of Gaussian (LOG); second, compute the SSD of all image pairs to get the SSSD function; finally, identify and localize the minimum of the SSSD function to get the inverse depth [5]. Based on this, they extended the algorithm for parallel, low-cost machine implementation, the three main changes are: using small integers to represent the image data; using absolute values instead of squares in the SSD; giving the algorithm the ability of rectifying geometry compensation [5].

Recently, depth from stereo vision is applied widely in areas such as machine vision (e.g. autopilot), 3D reconstruction, and image/video post-production (e.g. image de-haze, auto depth-of-filed). Hane et al. [46] proposed a method to reconstruct indoor environment from image pairs for robot navigation in 2011. In 2015, Jeon et al. [47] introduced an algorithm that can accurately estimate depth maps using light-field camera and the multi-view stereo correspondence.

Depth from Defocus: As is mentioned in Section 2.1.3, DFD is a passive depth acquisition method derived from DFF which eliminated the inefficiency and lacking flexibility in a DFF process while only need two to three images. In 1994, Subbarao and Surya [48] proposed a method called S-Transform Method (STM) based on their previous work [25], which did the computation in spatial domain and was local in nature. Implemented in their SPARCS system, STM method yielded an RMS error of 2.3% in autofocusing in a distance estimation between 0-0.6m, and 20% at 5m.

Recently, the number of researches on the implementation of DFD and more efficient approach for DFD based depth acquisition is getting bigger. In 2011, Zhou et al. [49] used two coded apertures to complement each other in the scene frequencies they preserve, which led to the depth with higher fidelity and the capability of obtaining high quality all-focused image. The development of light-field camera also made the result of defocus-based depth more accurate as it provides the possibility of re-focus after taking an image.

The DFD is having advantages on occlusion problem when comparing to correspondence approaches, however, the measurement range of DFD is limited by the dept-of-field; the accuracy of DFD depth recovery might get unreliable when features in the window are weak or when image is noisy [50].

Depth from Correspondence and Defocus: In recent years, some researches showed outstanding performances on depth estimation by utilizing the advantages of both correspondence and defocus methods.

In 2013, Tao et al. [51] presented a principled algorithm (CADC) that computes dense depth with both defocus and correspondence techniques. In their approach, defocus

depth was used for spatial variance while correspondence depth was used for angular variance. Combining these two cues, their system is able to handle computer vision applications such as matting, depth-of-field, and surface reconstruction [51]. In 2015, Tao et al. [52] continued their research on this area and arose an improved technique for local shape estimation from defocus and correspondence, this work contributed on: formulated new depth measurements and shading estimation constrains, a new local depth algorithm for correspondence and defocus using angular coherence, a new shading constraint, and a framework for depth refinement.

Although reasonable results from these works have been presented, these works are suffering from the limitation brought by the algorithm. The CADC method tends to be less reliable when encountering backgrounds without texture or when noisy, holes and discontinue disparities are showed in homogeneous regions [51].

2.3 Depth Enhancement Techniques

Researchers in recent years have contributed many depth enhancement techniques, some combines two or more depth acquisition techniques for less noise and new calibration approach, some integrated filters for enhancing certain features that can be used in daily contexts, and some use one or multiple RGB images as up-sampling references.

2.3.1 Image Fusion Enhancement

An Image fusion up-sampling is a well-discussed topic in academia, which refers to the techniques that up-sample depth map by combining low-resolution depth map and high-resolution images. These techniques are especially useful for upsampling the low-

resolution depth generated by stereo methods such as ToF or correspondence stereo, thus many researches fall in this genre.

Yang et al. [34] in 2007, introduced the depth enhancing method that uses color images as reference. By refining the low-resolution input depth map iteratively in terms of spatial resolution and depth precision, their method displays up to 100× resolution enhancement with the evaluation using Middlebury benchmark [34].

Inspired by the works using nonlocal means filtering to regularize depth map, Park et al. [9] in 2011 presented an up-sampling method for the depth from ToF camera by extending the regularization with additional high-resolution RGB input as it assumes depth discontinuities and image edges have joint regions of homogenous color, which have similar 3D geometry. This work, however, requires users' correction for most of the occluded or fuzzy area.

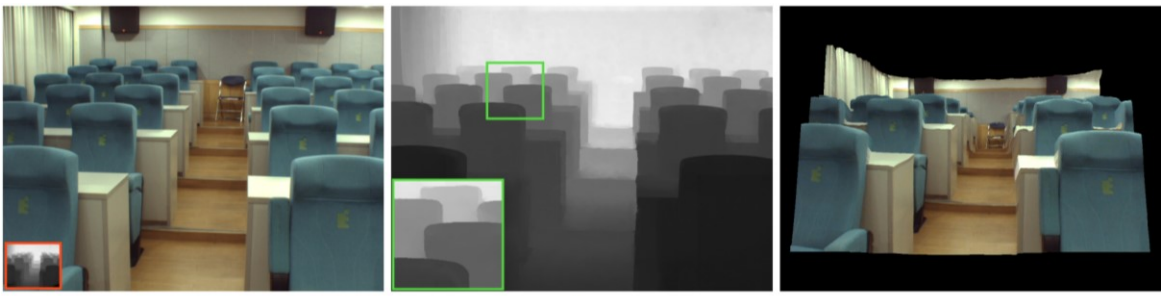


Figure 2.9 - (a) Their input, the low-resolution depth maps are shown on the lower left corner. (b) Their up-sampling results. (c) Novel view rendering of the result. [9]

In 2013, Ferstl et al. [53] derived a numerical algorithm for enhancing low resolution depth map generated by ToF sensors, which is based on a primal-dual formulation that is efficiently parallelized and runs at multiple frames per second. Moreover, their work also introduced novel datasets with highly accurate groundtruth, which first enabled the use of real sensor data to benchmark depth upsampling methods.

Later in 2015, Barron et al. [54] developed an algorithm that can process a 4-megapixel stereo pair within one second and has applied it in an auto-defocus context. They integrated bilateral filtering in the algorithm, which avoided the problems from pixel space that are not edge aware and slow to compute.

2.3.2 Super Resolution and Multi-Baseline Enhancement

A super-resolution up-sampling merges multiple low-resolution depth maps for a higher resolution; a multi-baseline enhancement uses parallel multi-camera setup that detects depth by using camera pairs with different baseline distances. Zhu et al. [55] fused time-of-flight sensors and passive stereo to get high accuracy depth map. They contributed a method for using data from both time-of-flight and stereo methods to produce enhanced depth estimates and a calibration method that leads to more efficient working of TOF sensor.

Honegger et al. [56] in 2017, designed a multi-baseline system that perceive depth by three parallel camera pairs with various baselines. According to their result, their four-camera-multi-baseline system helps bringing in more than 95% correct matches within a five-pixel error band while the two-camera result only has 20-35% valid matches.

2.4 Summary

According to the three sections above, the element extraction methods in academia are various, and all have their own strengths and weakness. The keying-based methods are currently most accurate and efficient but are limited by the environment, preliminary cost, and technical issues such as color spill and shadows. The matting solutions and contour tracking share similar features, which are less expensive on the setup but absent on the continuity and robustness on the fuzzy area. The depth-based

approaches are not mature enough, but we can see some exciting results and functions derived from depth-based strategies that are unachievable by other methods in the element extraction context.

Current depth acquisition methods are rather abundant, the active methods win from their straight-forward mechanism but suffer from the limitations in distance and outdoor environment and the mismatch of resolution between depth map and color image is another issue. Although the passive stereo methods show less advantage in a dark environment, they are more adoptable in an outdoor environment. Besides, with the aid of current depth enhancement techniques, the passive-stereo-based depth is widely welcomed by the smartphone market as it does not require extra equipment and big aperture size.

Chapter 3: Methodology

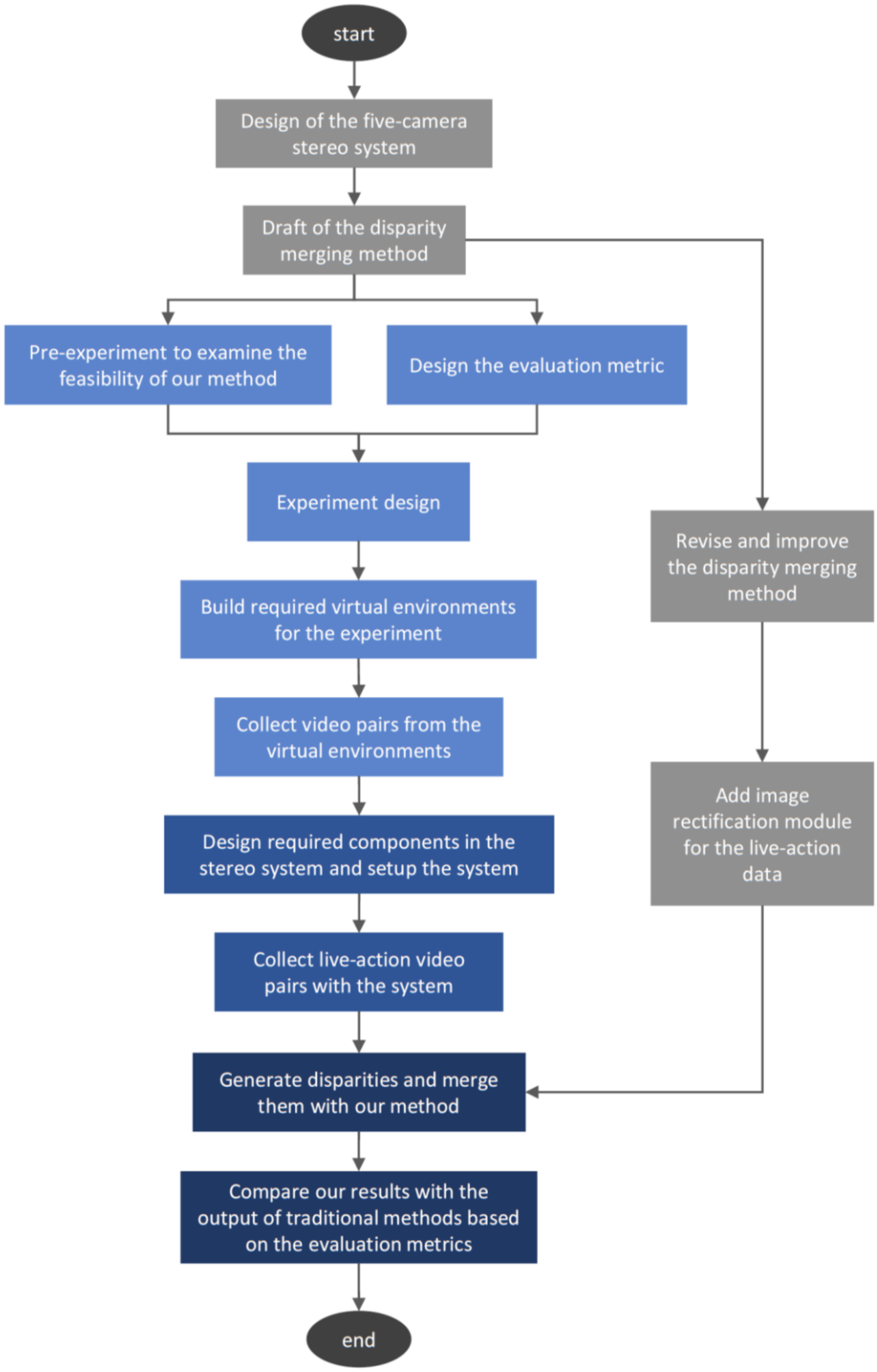


Figure 3.1 - Methodology overview

Our methodology consists of 5 main parts:

1. Pre-validate stage, to preliminarily validate our idea under a relatively ideal environment, to better understand the target results and data that we should aim to, and to find a proper data analysis approach for our research.
2. Design and set up the hardware components for real-life data collection.
3. Modify our method based on the problems that we found during the pre-validate stage, add required modules to the procedure for better performance.
4. Design the formal experiment and collect data.
5. Refine the evaluation metrics and analyze the data.

As is mentioned in Section 1.2, our objective, in general, is to improve disparity quality by reducing the blind spots in a scene; explore the potential of its application in an auto element extraction scenario with the more precise depth information, but what kind of accuracy should we pay extra attention to?

In an element extraction scenario, smooth and accurate matte information is essential; the transparency of the matte shall follow the solidness of target objects and have convincing edge information.

In traditional stereo vision, algorithms focus more on producing accurate depth estimates than well-localizing the depth boundaries. The missing information from occluded regions causes one reason for this limitation. If the output is applied in a machine vision scenario to detect obstacle distance roughly, accurate distance estimates weighted more. However, when it comes to image processing context (e.g., dehaze, defocus), such methods fail to create a satisfying result. In existing commercial object extracting solutions, the boundaries of objects are better preserved. However, the matte

quality is very dependent on the scenic setup, while the depth of the specific object can not be acquired from these solutions. A consistent disparity sequence with relatively accurate boundary information and a smooth depth gradient is the goal of our output.

3.1 Pre-experiment and Evaluation Metrics

3.1.1 Pre-experiment

Since the element extraction is a more industry-based topic, our research aims to develop a usable system in the entertainment industry, which possesses the potential to reach a high level of quality. Therefore, we consider current industrial element extraction solutions as part of the related works and have designed a pre-experiment to validate our method's feasibility preliminarily. This pre-experiment also served as a path to find the proper evaluation criteria for our experiment.

For the pre-experiment, we built two scenes, both with a version of a green screen setup and a version with our multi-camera setup in Maya 2019 [57], where it is easy to create virtual cameras, 3D objects, generating ground truth disparities and mattes for quantitative comparison.

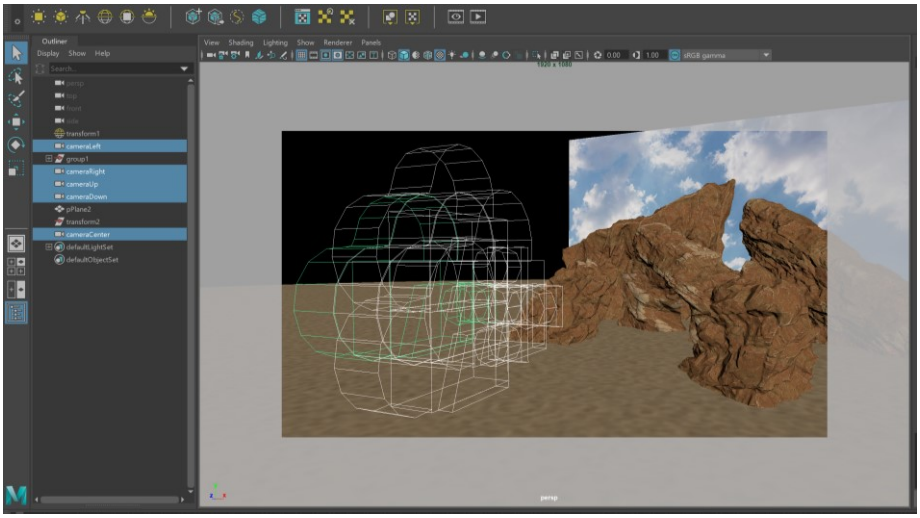
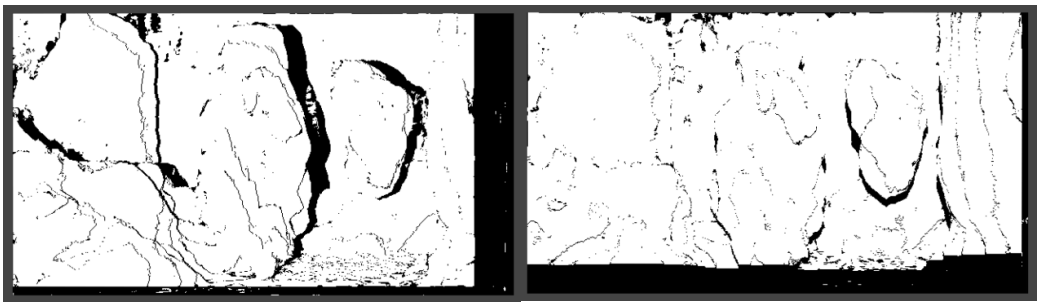
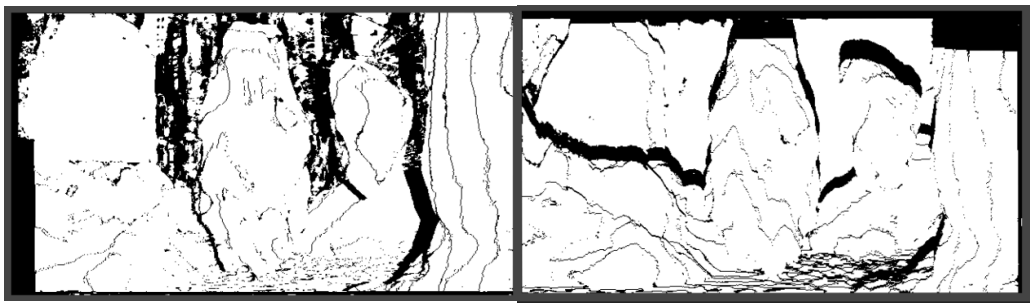


Figure 3.2 – The virtual environment that we build in Maya 2019

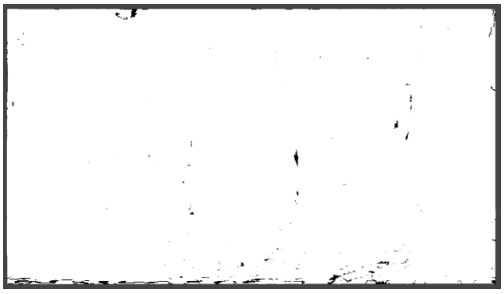
(a)



(b) (c)



(d) (e)



(f)

Figure 3.3 - (a) rendered color image (b) occlusion map from the left and center camera (c) occlusion map from the up and center camera (d) occlusion map from the right and center camera (e) occlusion map from the down and center camera (f) occlusion map by adding the (b), (c), (d), (e) four occlusion maps together. The black regions are the occluded area, the occluded area in (f) is obviously reduced.

With the rendered images from the 3d scenes, we can use existing stereo algorithms to generate depth maps and corresponding occlusion maps from the four camera pairs. Here we used an implementation [58] of Fast Cost-Volume Filtering [11] that uses the Winner Takes All (WTA) approach to get the most appropriate disparity for each pixel and Guided Filter [10] for smoothing the matching cost.

Considering occluded area as a shadow cast by a foreground object, a disparity map from only one pair of stereo images has a certain amount of shadows that cannot be perceived by the cameras, such that objects with smaller distance to the camera center have a larger shadow, which gives the matching algorithms and filters more challenges to fill up the unknown regions. However, with cameras recording the scene from three more angles, the scene is lit up by a shadow-less lamp, the shadows (i.e., occluded areas) are considerably reduced (see in Figure 3.3).

This result has given us a definite hint that our method shall bring more certainties to a correspondence matching process. To further validate the idea, we developed a simple program to merge the disparities generated from the four stereo pairs.

Since the secondary cameras all have a certain amount of translations relative to the primary camera, the different cameras pairs missed information from various directions. This phenomenon leads to the effect that the most prominent occluded regions appear at the left, upper, right, and lower boundaries of each disparity. Thus, we started the merging procedure by cropping out the unknown regions, creating a confidence map based on this pattern, and picking the pixel values from the most reliable disparities to create a merged result.

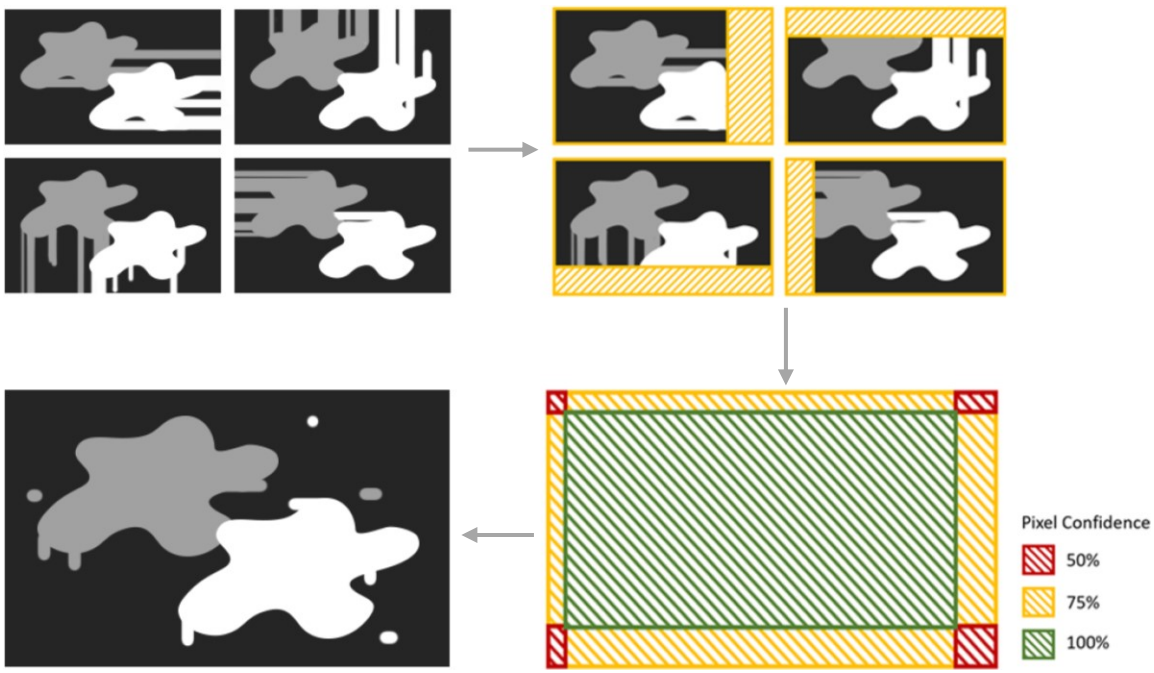


Figure 3.4 - The procedure of the mask merging. Step1: generate four disparities from the four camera pairs. Step2: mask out the uncertain regions and create a confidence map accordingly. Step 3: Select most reliable pixel value from the four disparities according to the confidence map. Step 4: store the result into a merged disparity.

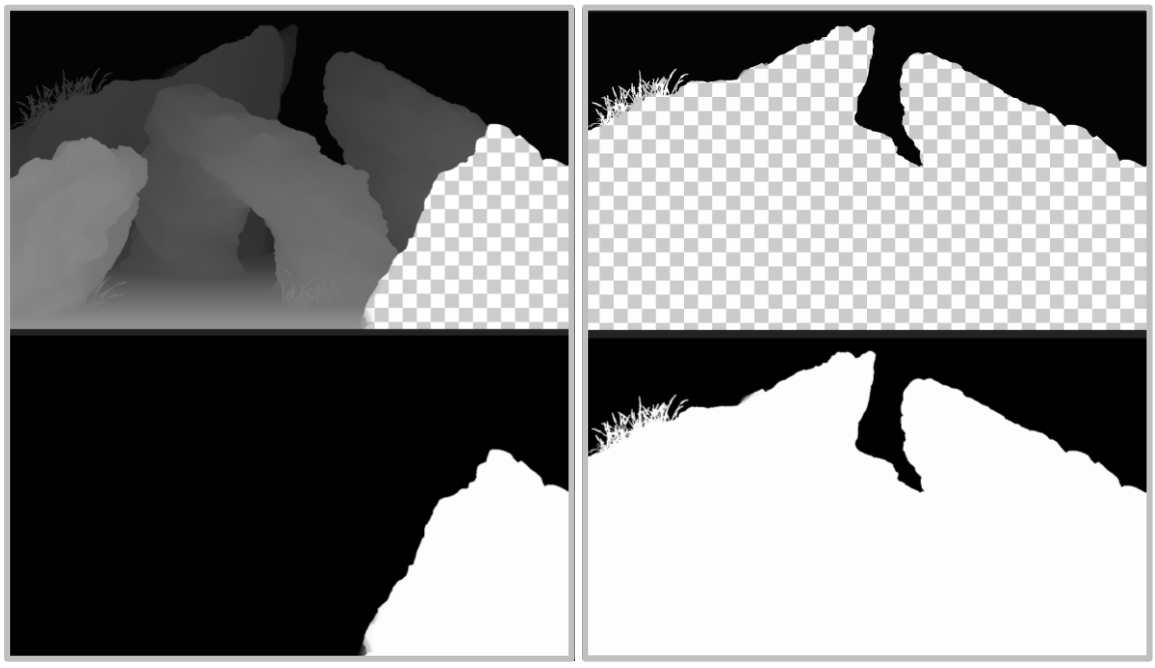
This approach seemed efficient with images rendered from virtual environments since the virtual cameras created for our pre-experiment are perfectly aligned on the same plane and have no unexpected rotation or translation between each other. Although the result of this method showed us the possibility to patch the uncertain areas of one disparity from another. This method however, did not sufficiently use the occlusion maps and have ignored the occluded objects in the center of the scene. The challenges that we might encounter in a live-action scene, such as a non-colinear camera arrangement, the fake edges from a hard shadow, the similar object colors, are not yet considered. Therefore, we designed a more comprehensive experiment to validate our concept as well as improved our merging algorithm to fully utilize the occlusion information for more complicated scenarios (see more details in Section 3.3 and Section 3.4).

3.1.2 Evaluation Metrics

Although the pre-experiment was conducted under an ideal situation and has not taken many real-life challenges into account, it showed us the right direction to continue our research and has provided useful resources to the evaluation stage.

Our research is looking for a higher disparity accuracy and a new object extraction solution. In this case, comparing the depth and matting quality of our output with traditional methods would be the main path to evaluate and validate our approach.

Ideally, our output shall be a depth sequence with clear object boundary and smooth depth gradient, with which, we can select an arbitrary range of depth value to spatially mask out objects within this range (see in Figure 3.5).



(a) (b)

Figure 3.5 - (a) Select a small depth range in a ground truth disparity map and create a matte from it. (b) Select a bigger depth range in a ground truth disparity map and create a matte from it.

The renderer Arnold in Maya 2019 is able to calculate the depth and alpha information of a scene, the depth map is calculated through the distance between vertices

of objects and the center of a virtual camera, the alpha reflects the solidness of objects. With some simple processing, these two channels can provide the ground truth of both disparities and matte.

With these ground truth data, we can get an overall quantitative evaluation of our disparity and matting accuracy by calculating the Structural Similarity Index (SSIM) and Peak Signal-to-noise Ratio (PSNR) of our results. The reason for choosing SSIM as our main evaluation metric is that researches [59] [60] have revealed that SSIM is more understandable than PSNR and MSE from the perspective of the human visual system.

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{7}$$

In our comparison, μ_x is the mean of the ground truth frame, μ_y is the mean of our disparity or matte frame, σ_x and σ_y are respectively the variance of the ground truth and our frames, σ_{xy} is the covariance of the ground truth frame and our frame, and C_1, C_2 are constants for stabilizing the division with weak denominator.

On the other hand, since SSIM is less sensitive to the changes in brightness and contrast [60], which might be important factors that reflect the quality of a disparity map, we adopted PSNR as an assistive metric in the disparity accuracy part to ensure the reliability of our evaluation result.

$$PSNR = 10 \cdot log_{10}\left(\frac{MAX_I^2}{MSE}\right) \tag{8}$$

In our comparison, MSE is the Mean Squared Error between the ground truth frame and our disparity frame, MAX_I is the maximum valid pixel value, which is 255 in our case.

3.2 System Overview

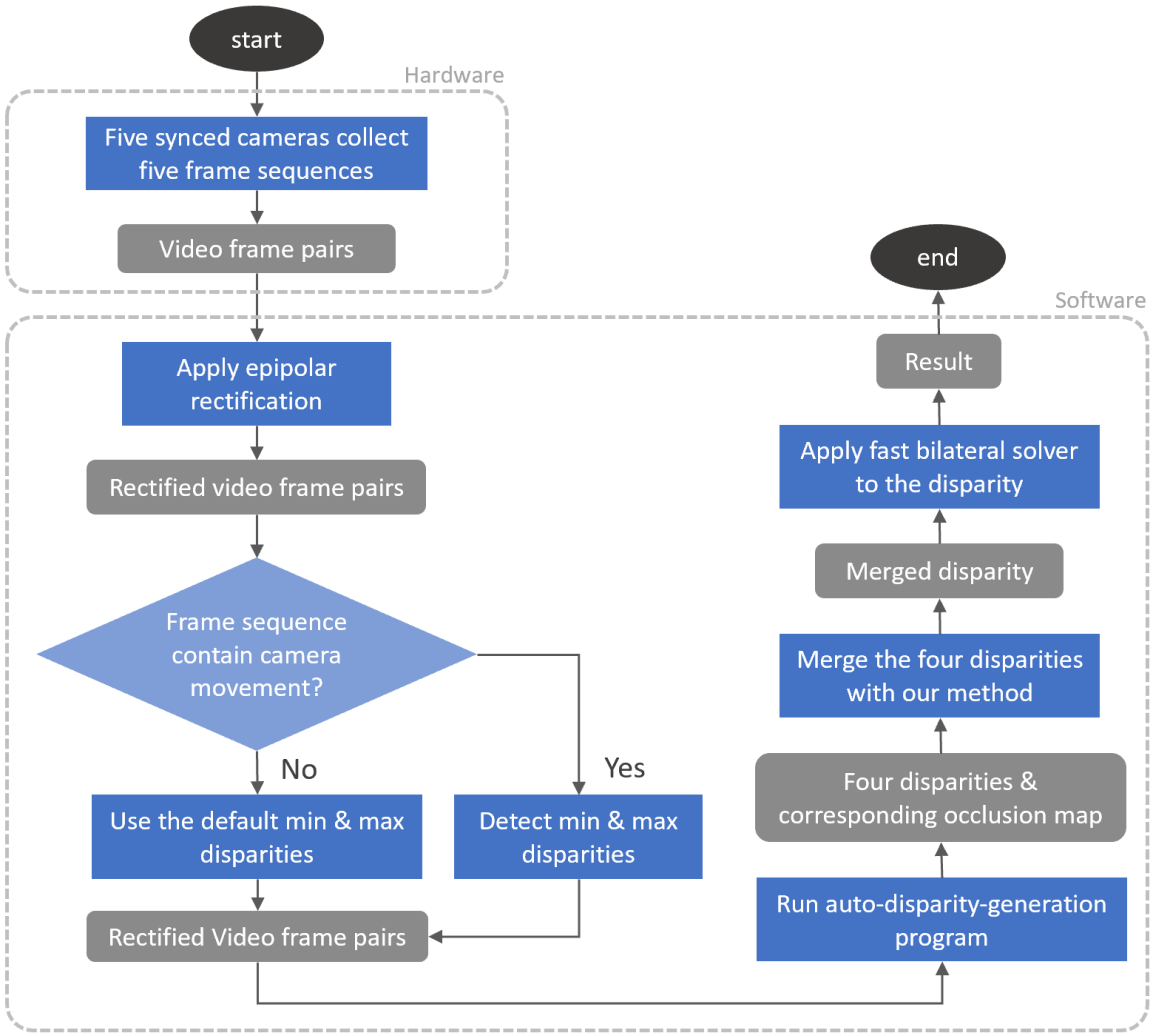


Figure 3.6 - The overall workflow of our system

Our system contains a five-camera installation and a framework that we developed to process all collected videos and to generate high-accuracy disparity sequences. The five-camera installation has four secondary cameras that are triggered by the primary camera; and the frame work consists of an optional min/max disparities detector, an image rectification module, a disparity merging module, and a bilateral-blur-based optimizer.

3.2.1 Hardware

In our five-camera stereo system, we used FLIR Grasshopper3 cameras and RICOH 12.5mm IRIS lenses. The Grasshopper3 cameras are programmable and can be synchronized through the applications provided by FLIR. The five cameras are connected with GPIO cables (see details in Section 3.2.2), and the power is delivered to the cameras through the USB3 ports, which also provided up to 5GB/s bandwidth to transfer the images or videos.

Figure 3.7 - FLIR Grasshopper3 with GPIO and USB3 interfaces [61]

To arrange the five computer vision cameras according to our designed pattern, we designed and 3D printed a five-camera mount that is able to fix the cameras to certain position; and a lens holder that could reduce unexpected rotation among cameras so to keep them as close to a co-planar status as possible.

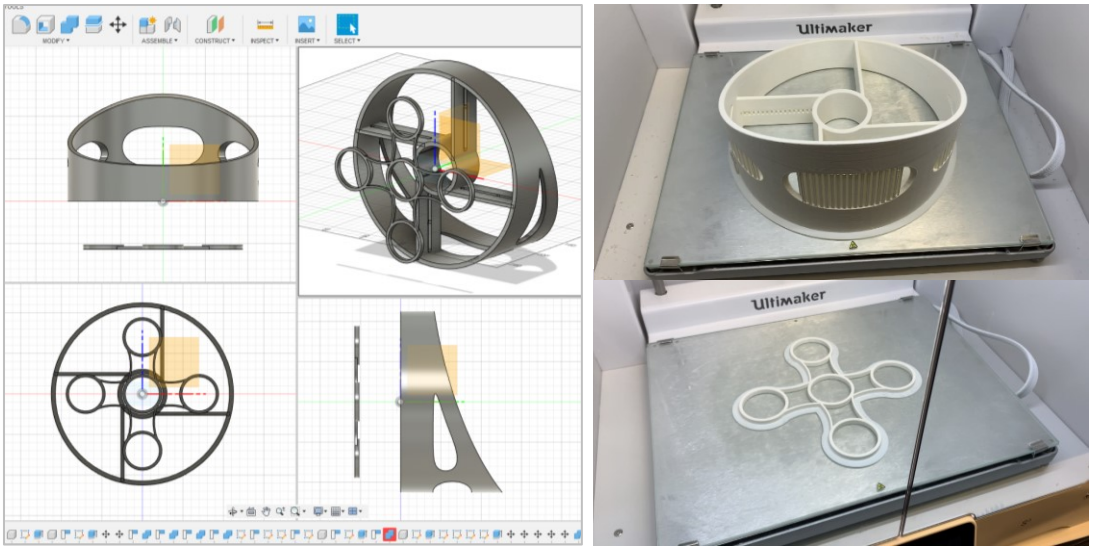


Figure 3.8 - Our model design in Fusion 360 [62] (left), and our 3D-printed models with Ultimaker S5 [63] (right)

In the first draft of the mount design, we created adjustable camera rails to allow an adjustment of the distance between the main camera and the secondary cameras. However, due to the weight of the lenses and the limitation of the screws, the cameras can easily rotate during a recording. Thus, we added a lens holder to constrain the camera directions.

Figure 3.9 - The final setup of the five-camera stereo system

The final hardware setup consists of: five programmable computer vision cameras, a 3D-printed five-camera mount, a lens holder, a computer to deliver power and receive data.

3.2.2 Software Used in the System

To make sure the five cameras capture images or video frames at the same time, we used FlyCapture from FLIR to manage the basic imaging parameters (e.g. brightness, exposure, white balance, gamma), trigger signal, GPIO input output, and the data format during collection.

As is shown in Figure 3.10, through a physical connection from GPIO cables, the output channel of the primary camera connects with the input channels of the secondary cameras. In this case, the recording signal will be sent to the secondary cameras when we trigger the primary camera, so the five cameras can capture videos or images with synchronized image parameters and framerate.

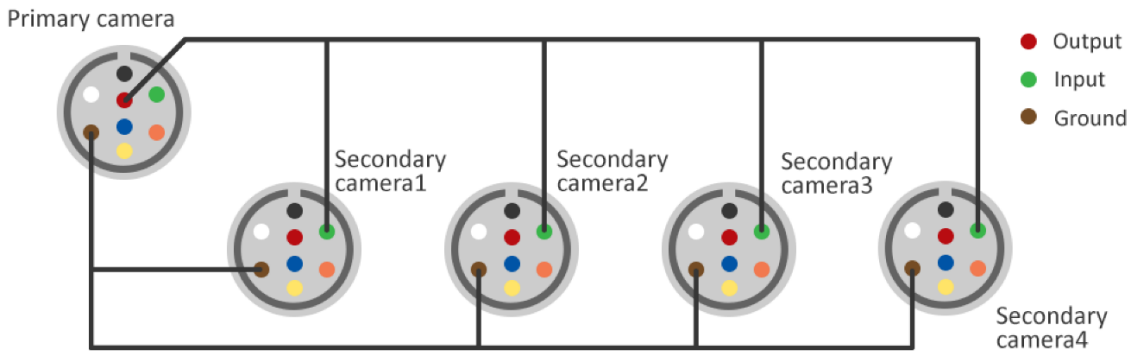


Figure 3.10 - Illustration of how we connected the GPIO channels for the primary (center) camera and the secondary cameras

We set the trigger signal to positive polarity with no time delay and the output format of collected videos as RAW8 image sequences to remain the unprocessed information for each frame.

Another big part of the software is a framework developed by ourselves, it contains some implementation of current depth generation methods, epipolar rectification, our minimum and maximum disparities detector, and our disparities merging methods, which will be explained in Section 3.3.

3.3 Development

Since the method we proposed has plenty of unknown points, we partially referred the Spiral Development Model [64] to ensure our developing direction is on the right track. Our developing process contains two iterations; the first one, as is mentioned in Section 3.1.1, is the disparity merging method for ideal virtual environments. It preliminarily validated our concept but only utilized the still images and did not consider the challenges that we might encounter in the real-life scenarios. The problem of our first iteration quickly got exposed when we were trying to add complexity into the input data. Thus, we planned and started our second iteration of development with a more comprehensive design, which are explained in Section 3.3.1 to 3.3.5.

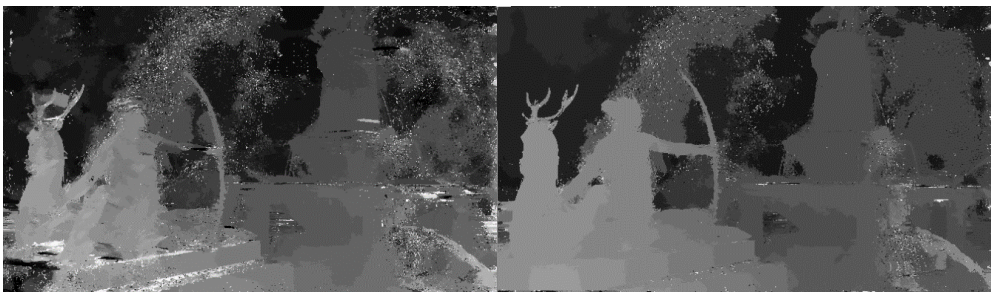
3.3.1 Development Environment and Platform

In the development, we used python 3.7 for the overall program, OpenCV library for most of the image processing tasks (e.g. feature detection, image perspective warping, image quality examination.), NumPy and SciPy for manipulating the matrices and the calculation relating to linear algebra, and MatPlotLib for visualizing our output data.

3.3.2 Camera Calibration and Image Rectification

In a live-action stereo vision context, stereo pair usually runs into two steps before the correspondence matching, which are respectively camera calibration and epipolar rectification. The necessity of these two steps is tightly related to the features of the correspondence algorithms. Most of the correspondence algorithms select a point from the reference image and search for correspondence point on a horizontal line in the query image, this helps lowering the computational costs and speeding up the process, but if the two cameras for data capturing are not co-planar, which is most of the cases in a real-life scenario, the captured stereo pair will have vertical parallax regarding the corresponding points. In this case, the correspondence algorithms might fail finding the correspondence point from the query image, hence output low-quality disparities (see in Figure 3.11).

(a)



(b) (c)

Figure 3.11 - (a) A stereo pair. (b) Disparity from unrectified stereo pair. (c) Disparity from rectified stereo pair. [65]

This requires us to adjust the stereo pair to minimize the vertical parallax among correspondence points. As can be observed from Figure 3.12, a non-co-planar camera

pair receive the image of a scene from two different image planes (the gray planes). Our goal is to find out the amount of relative rotation and translation between the two cameras, with this information, we are able to project the images to a common-image-plane (the yellow planes) that is parallel to the baseline (the line between the two camera centers) so to eliminate the vertical parallax and continue the matching process.

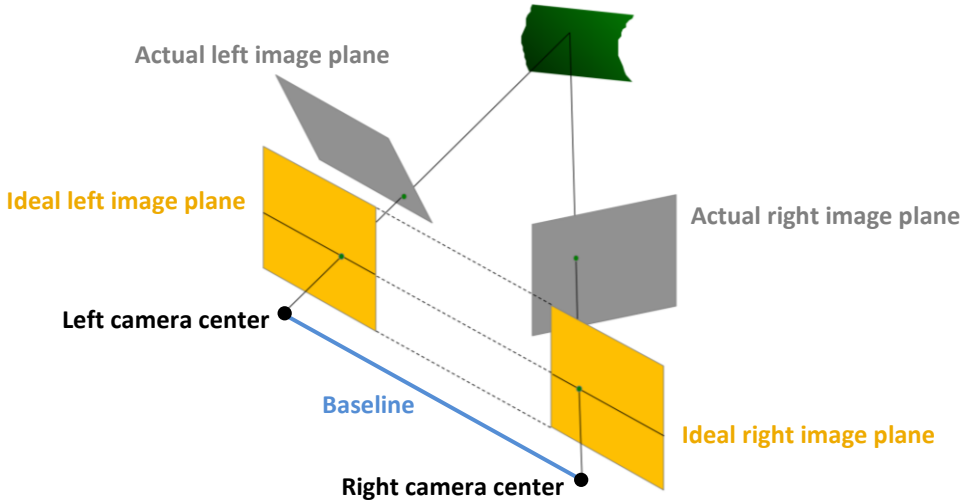


Figure 3.12 - Illustration of realistic image planes and ideal image plane [66]

The camera calibration process, which estimates the intrinsic and extrinsic parameters of a camera with certain patterns that describe a camera's focal length, distortion, image center as well as its position and orientation in the world coordinate [67].

The epipolar rectification, right after the calibration, uses this information to re-project image planes onto a common plane that is parallel to the baseline between camera centers [66]. According to epipolar geometry, the correspondence points in a stereo pair can be described as:

$$x'^{T}Fx = 0 \quad (7)$$

Where x and x′ are the corresponding points in the two images, F is the fundamental matrix that contains rotation and translation between the two cameras and intrinsic information about the cameras, which describes the location of the second camera relative to the first in pixel coordinates [68]. An epipole is the projection of one camera center in the view of another, it is described as:

$$Fe = 0 \qquad F^T e' = 0 \tag{9}$$

Where e and e′ are the epipoles of the left and right images. And a complete epipolar rectification process can be described as follows [66] [69]:

1. Estimate F using the 8-point algorithm (SVD)
2. Solve $Fe = 0$ to get the epipole e
3. Build R_{rect} from e
4. Decompose F into R and T
5. Set $R1 = R_{rect}$ and $R2 = RR_{rect}$
6. Rotate each left camera point $x' \sim Hx$ where $H = KR_1$
7. Repeat 6 and 7 for right camera points using R_2

However, when it comes to epipolar-geometry-based multi-view rectification, due to the fact that an essential matrix or a fundamental matrix are computed basing on the relative position of the camera pair, it is theoretically impossible to rectify multiple stereo pairs on the same time with baselines on different directions. The camera arrangement in researches on multi-baseline stereo or multi-view rectification are mostly one-direction [70] [71] [72] [73]. Research on multi-baseline stereo with different baseline direction also addressed that “It is impossible to rectify a vertical stereo pair of images with the horizontal stereo rectification approach” [74].

3.3.3 Two-Axis Image Rectification

In our research, the camera arrangement is on two directions as we are aiming to reduce occlusions from vertical and horizontal directions. This has brought a big challenge to our stereo pair rectification process.

As is demonstrated in Section 3.3.2, an epipolar based rectification method is unlikely to work for our setup. Therefore, we looked into some non-epipolar methods, such as correspondence-point-based rectification [75]. In the research of Nozick [75], he considers the image rectification process as a rotation around the optical center and an update of the focal length, and simplified the problem to finding the relations between original image and corrected image, which is represented as homography matrix H_i, and it shall satisfy the equation:

$$(H_i x_i^k)_y - y_k = 0 \tag{10}$$

Where y_k is the average y-coordinate of the k^{th} rectified correspondence points, x_i^k is the correspondence points so the points are horizontally aligned to the same y-coordinate [75]. However, this also shows that this method cannot be integrated into our system as we conduct horizontal correspondence matching towards the image pairs from vertical direction by rotating the stereo pair. Putting which in the context of this research, we are minimizing the parallax in x-coordinate for the stereo pairs in vertical direction. In another word, the stereo pairs from horizontal and vertical directions are rectified based on different y_k, as a result, we cannot rectify the images from both directions at the same time, otherwise, the rectified stereo pairs will generate different center disparity and the consistency of the disparity sequence might get influenced.

Finally, to ensure the final disparity output is always aligning to the center as well as to preserve the consistency of the disparity sequence, we solved the problem in a simple but reversed way.

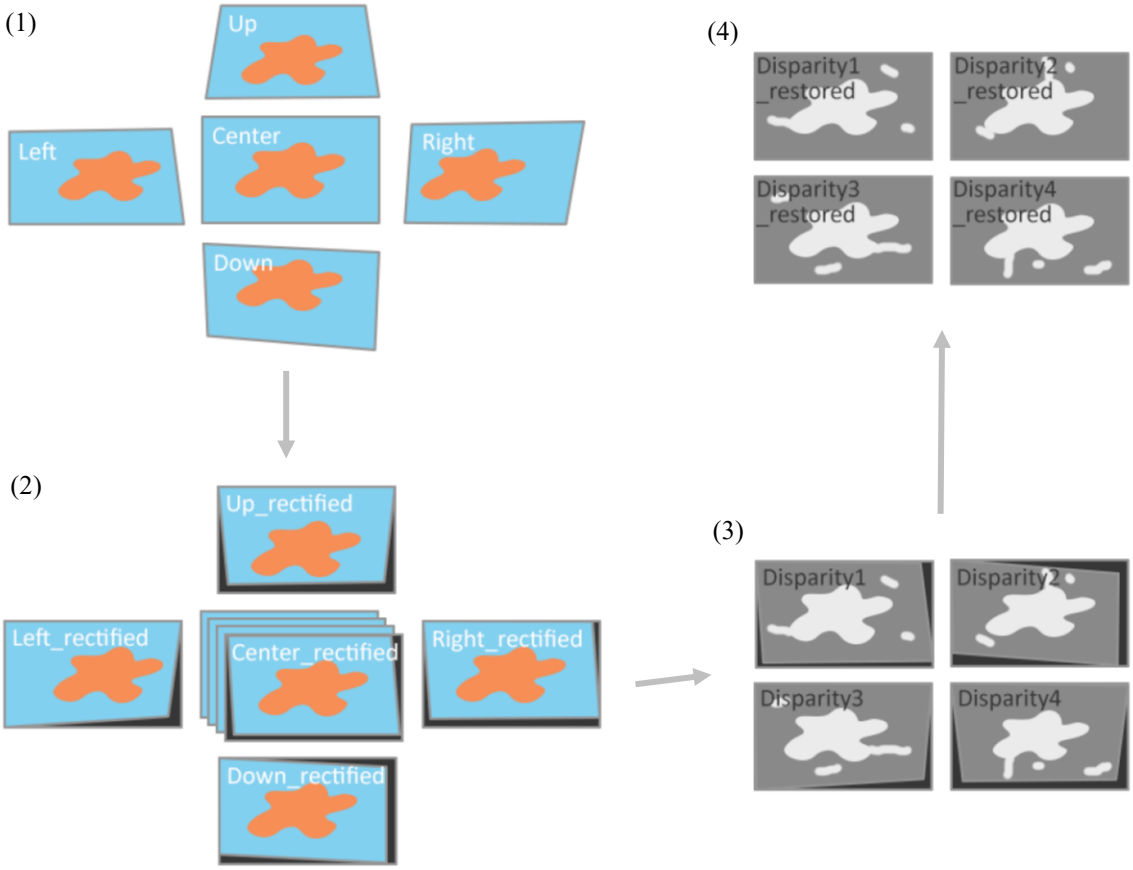


Figure 3.13 - Our image rectification procedure. (1) Images captured by our system; images are on different image planes. (2) Rectified image pairs; each image pair has their own common image plane; four versions of rectified center image and corresponding homogrphies are generated. (3) Using the four rectified stereo pairs to get four center-aligned disparities; disparities have noises from different directions. (4) Multiply the reverse-homogrphies with the disparity and get the restored disparities that are ready to be merged.

To get a relatively high-accuracy rectified result for better matching quality, we inherited the epipolar rectification method for each stereo pair, here we used a combined implementation [76] of Quasi-Euclidean Uncalibrated Epipolar Rectification [77] and

Automatic Homographic Registration of a Pair of Images with A Contrario Elimination of Outliers [78]. In this implementation, the features and matches are detected by SIFT, the outliers are removed by ORSA, and it also gives the homographies that applied in the rectified image pairs. So far, we have got four rectified stereo pairs that with four different center images, with which we can generate four disparities for the center image but with various homographies applied on them. Finally, we calculate the reversed homography matrices for each disparity, multiply them with the disparities to restore the disparities to the original center camera plane.

3.3.4 Minimum and Maximum Disparities Auto-detector

In a correspondence matching procedure, the process would run more efficiently with known minimum and maximum disparities since they constrain the searching distance and reduce computational cost. The minimum and maximum disparities remain unchanged if the camera is stationary; however, in a video shooting scenario, the shots are not always still. Thus, a moving camera shot would require dynamic min and max disparities information to run the matching procedure better.

To fit our system in such scenarios, we developed a simple min-max-disparities detector based on SURF [79] and FLANN [80] matching, where we used SURF to compute the feature points from a stereo pair, FLANN to filter the reliable matches by using Euclidean distances among the descriptors of features. However, since FLANN matching is quick in the speed but not always robust on matching, we enter an upper disparity bound and a lower disparity bound for the whole video sequence (which is easy to determine with human eyes and quick to find with software like photoshop) in the main program, and we compute the distance between matching points and find the

relatively accurate min/max disparities with the help from both the lower and upper bounds and the image size.

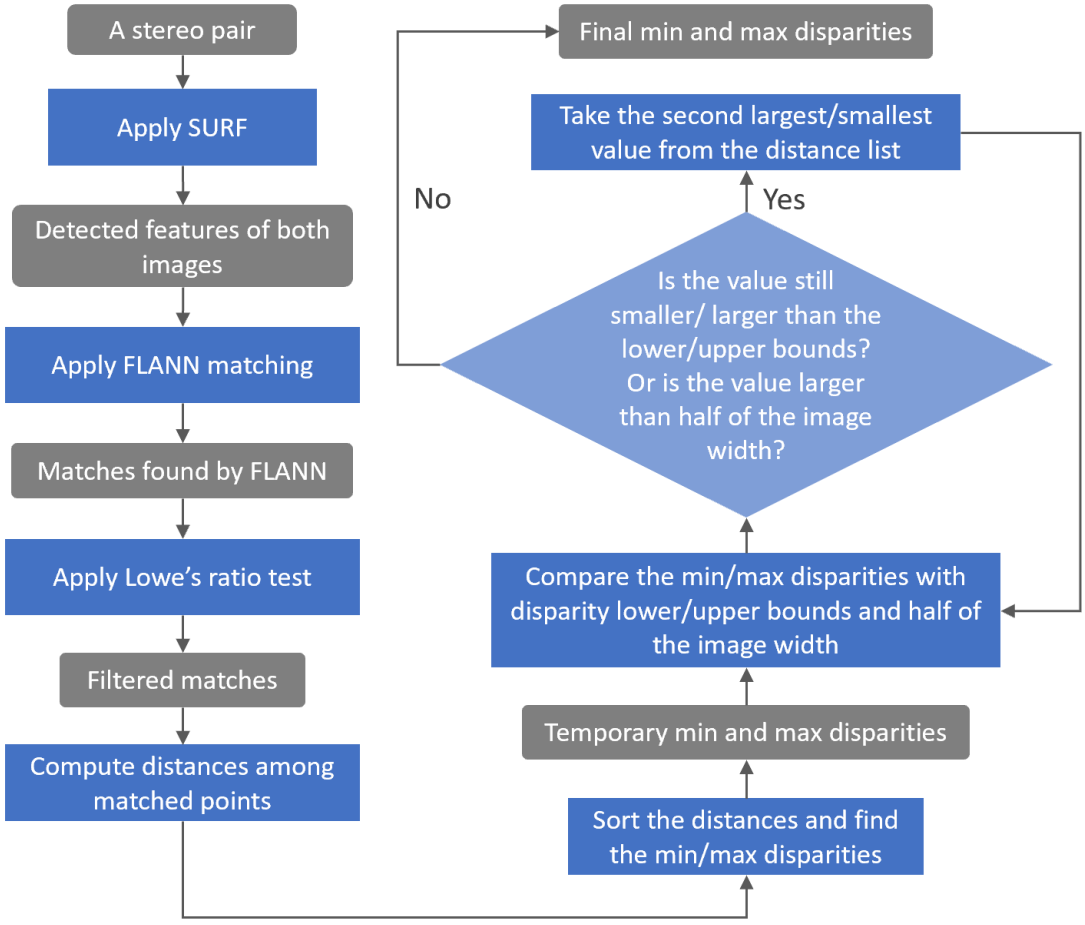


Figure 3.14 – Procedure of our min-max-disparities detector

3.3.5 Occlusion-minimized Disparity Merging

With the restored disparities and occlusion maps, we are able to merge the disparities as they are re-projected to the original center image plane and the objects boundaries are aligned.

In our occlusion-minimized merging method, we introduced a threshold that is inspired by the ratio test in SIFT algorithm from D. Lowe [81]. The detail procedure of our method follows the steps below:

1. Read in the four disparities and corresponding occlusion maps.

2. Check the occlusion maps to see what areas in certain disparity are occluded.
3. If a pixel is not occluded in all four disparities or it is all occluded in the four disparities, we check whether one of the pixels is (1 + threshold) times larger or (1 - threshold) times smaller than the other three pixels. If so, we dispose this value and take the mean of the other three, otherwise, we take the mean of the all four pixels.
4. If a pixel is occluded in three disparities, we take the pixel value from the disparity that is not occluded.
5. If a pixel is occluded in two disparities, we take the mean of the pixel values from the two disparities that are not occluded.
6. If a pixel is only occluded in one disparity, we check whether one of the pixels is (1 + threshold) times larger or (1 - threshold) times smaller than the other two pixels. If so, we dispose this value and take the mean of the other two, otherwise, we take the mean of the three pixels.

We believe this merging method can more sufficiently use the known regions from the disparities from four directions since the first iteration of this approach by simply cropping out unknown borders and patching them with the known area have showed advantages over the traditional method. We then evaluated the output of this approach in the experiment stage.

3.4 Experimental Design and Data Collection

Using the pre-experiment as reference, we refined our formal experiment design and divided it into four scenarios, which are respectively an ideal virtual environment, a semi-realistic virtual environment, a fully simulated realistic virtual environment and a

live-action environment, where the first three environments all have two versions, the normal version for both disparity accuracy numeric comparison, and a chromakey version for the matting quality numeric comparison; the live-action scene, however, due to the fact that we can hardly get a ground truth from an outdoor environment, we use it as a quality reference.

Figure 3.15 - The Red Rock scene (left), the Forest Rock scene (right).

Table 3.1 - The environment setup for disparity accuracy comparison

	One-stereo-pair method	**Our method**
Ideal (Virtual Environment)	• No camera rotation/translation • Obvious color difference among objects • Evenly lit up	
Semi-realistic (Virtual Environment)	• No camera rotation/translation • Objects with similar color and texture • Low light, strong shadow	
Realistic (Virtual Environment)	• Have camera rotation/translation • Objects with similar color and texture • Low light, strong shadow	

Table 3.2 – The environment setup for matting quality comparison

	Chromakey matting	**Our disparity matting**
Ideal (Virtual Environment)	• No color-spill • Consistent screen color (no folds or hard shadow) • No green objects in the foreground	• No relative camera rotation/translation • Obvious color difference among objects • Evenly lit up

Semi-realistic (Virtual Environment)	• No color-spill • Inconsistent screen color (folds and hard shadow) • With green objects in the foreground	• No relative camera rotation/translation • Objects with similar color and texture • Low light, strong shadow
Realistic (Virtual Environment)	• Color-spill • Inconsistent screen color (folds and hard shadow) • With green objects in the foreground	• Have relative camera rotation/translation • Objects with similar color and texture • Low light, strong shadow

3.4.1 The Ideal Virtual Environment

For the ideal virtual environment for disparity generation, we built two scenes with Maya 2019, one is a red rock scene with some irregular object edges (e.g., the moving field grass), another is a forest rock scene that the edges of the objects are clearer, but the background is more complicated that has trees and branches. Both of the scenes are lit up from the front side with light shadow and have distinctive color difference among objects, the relative positions of the virtual cameras are perfectly aligned such that the five cameras share the same image plane.

Figure 3.16 - A stereo pair rendered from the ideal Forest Rock scene, objects are evenly lit up and have distinguishable color difference, no vertical parallax among the correspondence points

As showing in Figure 3.16, the correspondence points (the green dots) on the stereo pair when under an ideal environment have no vertical parallax, the scene is evenly lit-up.

For the ideal chromakey version, we set a green screen behind the objects that we want to cut-out, the green screen is flat and has no hard shadow projected on it while the objects are set to not receiving any color spill from the green screen (Figure 3.17).

Figure 3.17 - A frame rendered from the green screen version of the ideal Forest Rock scene

The data collection procedure of the virtual scenes is rather straight forward. For the regular version, we merge the Z depth into the center frames, so to make sure the rendered EXR sequences include both color image and the depth information; the depth maps, and color images are from the unified perspective, thus, it is ready to play the role of a disparity ground truth. The frames from the secondary cameras only need the color information, so we rendered them directly with the Arnold renderer.

For the green screen version, we only need to render the view from the center camera. The alpha channel is also rendered from this scene since we can easily remove the green screen and get a ground truth matte for the foreground. The data collection workflows in Section 3.4.2 and Section3.4.3 both follow the same routine.

3.4.2 The Simulated Semi-Realistic Environment

In the semi-realistic disparity environment, we brought some challenges into the scenes, we gave very similar colors and textures to the objects and the lighting is not from the front anymore. As a result, more hard shadows are introduced into the rendered

videos, and the obscureness on object edges are increased, yet the virtual cameras are still perfectly aligned (Figure 3.18).

Figure 3.18 - A stereo pair rendered from the simulated semi-realistic Forest Rock scene. Objects have similar color and textures; stronger shadows are introduced. No vertical parallax among correspondence points.

For the semi-realistic chromakey version, we added a wind field in the scene to create some folds and shadows on the green screen, which follows some of the challenges that chromakey technique might encounter in a real-life scenario (Figure 3.19). However, the objects are not yet influenced by color spill as what is normally happening in a realistic environment.

Figure 3.19 - A frame rendered from the green screen version of the semi-realistic Forest Rock scene

3.4.3 The Fully Simulated Realistic Environment

Based on the semi-realistic setup, we added positional deviation to the virtual cameras in the fully simulated realistic scene, some cameras have rotation, some have

vertical offset relative to the baseline. Correspondingly, the vertical parallax between correspondence points start to show in the stereo pairs (Figure 3.20).

Figure 3.20 - A stereo pair rendered from the fully simulated realistic Forest Rock scene. Obvious vertical parallax can be found among the correspondence points, the environment is low-light and has strong shadow.

In the realistic chromakey version, not only did we add wind field for the folds and shadows on the green screen, objects are also receiving color spill from the green screen, as is shown in Figure 3.21, the objects have some green tint in the highlight and shadow area.

Figure 3.21 - A frame rendered from the green screen version of the realistic Red Rock scene

3.4.4 The Live-action Environment

The live-action scene is not manually setup, which reduced the bias of intentionally separate objects with strong color contrast. On contrary, we randomly picked an outdoor environment that has complex object boundaries as well as hard

shadow on some of the area, which might create a lot of noise in a traditional stereo matching process (Figure 3.22).

Figure 3.22 - Some of the scenes that we selected for the live-action capturing

The data collection procedure for the live-action scene used our five-camera stereo system, which follows the steps below:

1. Setup the system and test the cameras to make sure the framerate and the camera parameters are well synchronized.
2. Test the trigger strobe, making sure it triggers the five cameras at the same time, and the camera configuration application FlyCapture can successfully save the frame sequences.
3. Trigger the primary camera and start the recording.
4. Save the frame sequences with the timestamp and sequence number.
5. Import the frame sequences into After Effects, check if there is a corrupted frame or skipped frame, and export the sequences if the data is complete.

To reduce the artifacts from the skipped frame or corrupted frame due to the memory limit, we record all sequences with 24 frames per second, and each of them lasts for 100 frames.

Chapter 4: Results

In this chapter, we present both quantitative and qualitive results from our experiment, the results are also analyzed at the end of each section.

4.1 Quantitative Results

To provide a comparison of disparity accuracy between our method and traditional stereo method as well as the difference of matting accuracy between our disparity matting accuracy and current matting method accuracy, we used Structural Similarity Index (SSIM) and Peak Signal-to-Noise Ratio (PSNR) to obtain the pixel and quality difference between our experimental output and the ground truth.

As is illustrated in Section 3.1.2, SSIM is considered an efficient tool to measure image difference from a human visual perspective [59] and our research objective is to apply our method to post-editing, where the quality is mainly evaluated through human eyes; on the other hand, although studies have shown that SSIM is more referable than PSNR in perception and saliency-based errors, it shows less sensitivity to brightness and contrast change [60], which might be an significant factor in the accuracy of a disparity map, thus, we added PSNR as an assistive tool to help examining the overall performance of our method.

For both disparity and matte accuracy parts, we did not only use one frame for comparison since that might end up with picking the frame with the best accuracy and represent the result of our research in a biased way. To illustrate our results in a relatively fair context, we animated both scenes with movements applied on the objects (Forest Rock scene) or cameras (Red Rock scene), each sequence is 100 frames long and our

comparison took the mean accuracy of the 100 frames instead the accuracy of only one frame.

4.1.1 Disparity Accuracy

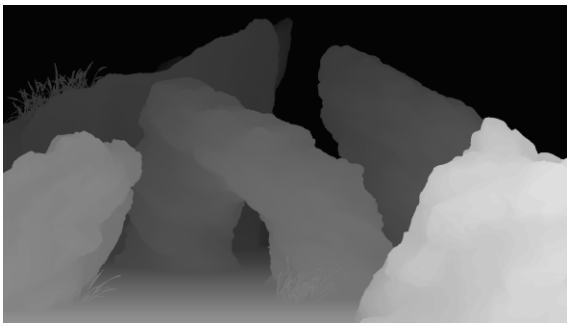
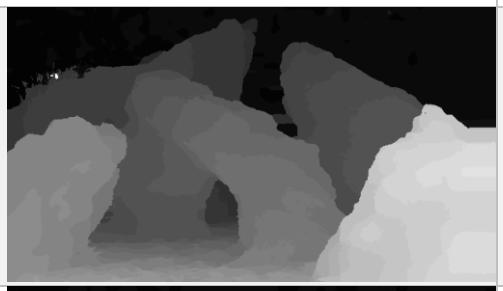
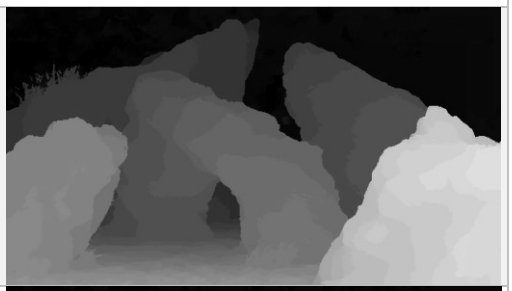
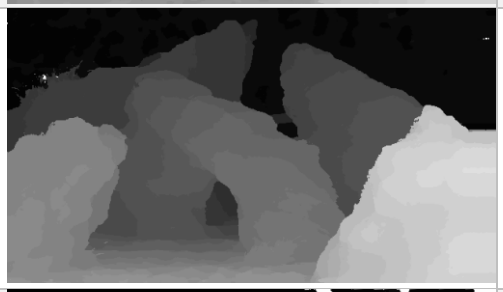
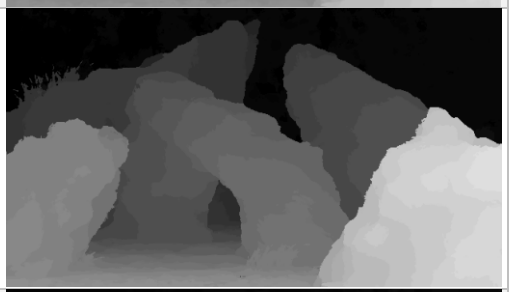
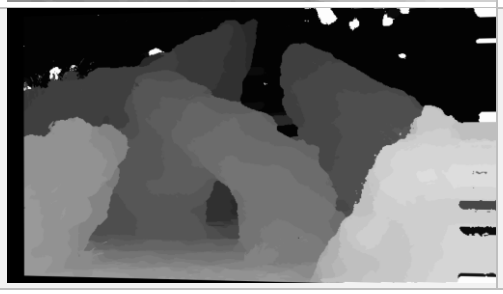
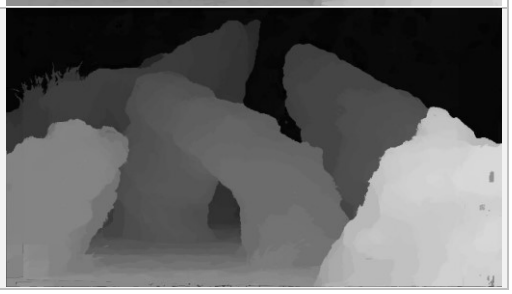


Figure 4.1 - Ground truth disparity of the Red Rock scene

Table 4.1 - Disparities of the Red Rock scene

	Disparity from one-stereo-pair method	Disparity from our method
Ideal		
Semi-Realistic		
Realistic		

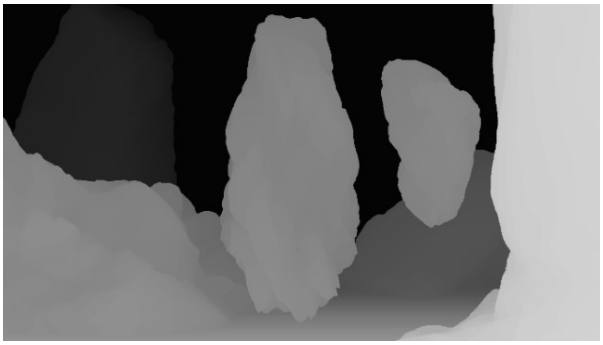
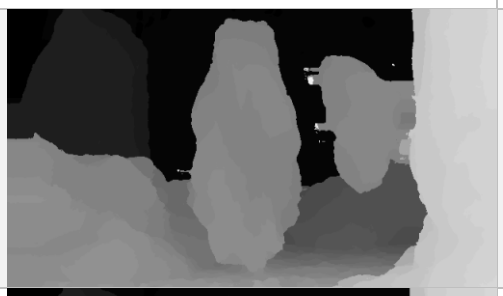
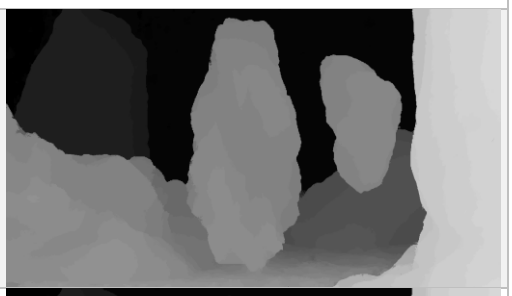
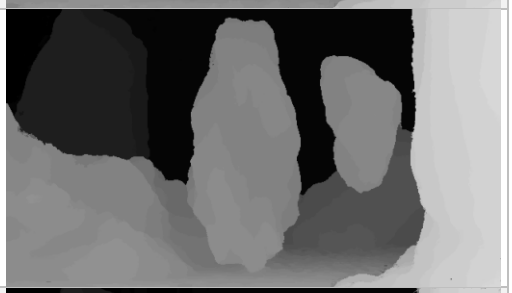
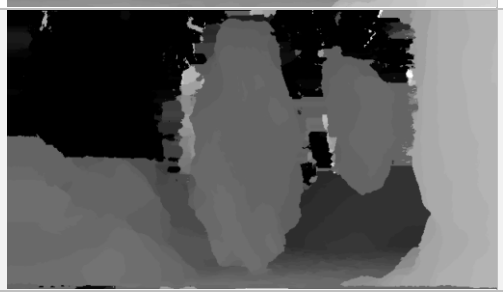
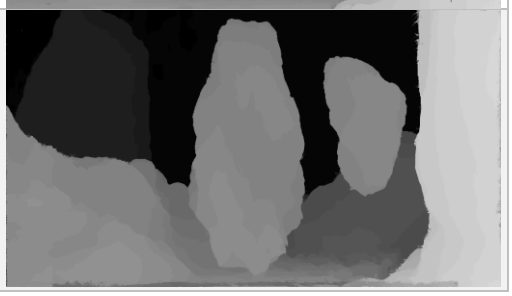


Figure 4.2 - Ground truth disparity of the Forest Rock scene

Table 4.2 - Disparities of the Forest Rock scene

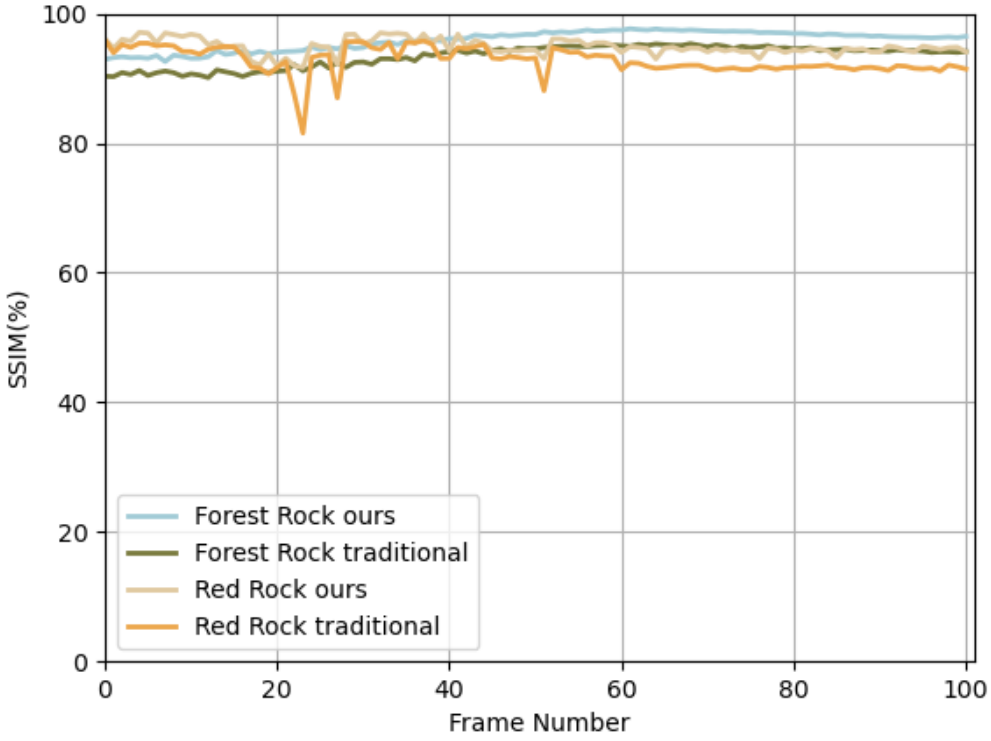
	Disparity from one-stereo-pair method	Disparity from our method
Ideal		
Semi-Realistic		
Realistic		

In the disparity comparison part, we rendered the ground truth disparity sequences for both scenes and have compared the disparities from traditional one-stereo pair method and our method under ideal, semi-realistic, and realistic environments as is defined in

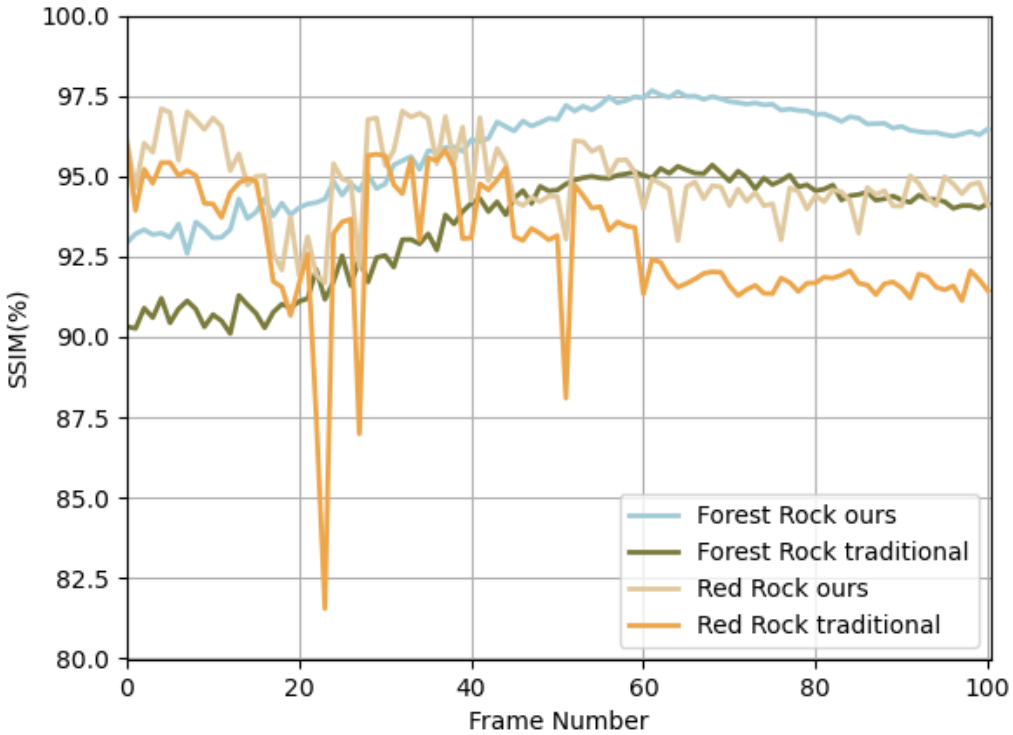
Table 3.1. The one-stereo pair method and our method both adapted the same correspondence algorithm [10] to constrain the variables of our experiment.

As is shown in Table 4.1 and Table 4.2, the depth quality (from visual perception perspective) of one-stereo pair method drastically decrease when the video shooting environments go down from ideal to realistic. The one-stereo pair results under realistic environment have a lot more errors in texture-less regions (Red Rock) or occluded area (Forest Rock) as well as noticeable information loss brought by epipolar rectification; while our method provides much more stable results except for a small amount of ambiguities on the edge-details. Our disparity-merging method takes good advantage of the occlusion maps and has patched the occluded regions with the depth information from other disparities.

In ideal and semi-realistic environments, since the very influential factor (camera relative rotation and translation) has not been introduced, the SSIM value difference between traditional method and our method is not very significant (see in Figure 4.3 and Figure 4.5) while PSNR has shown that our method has noticeable better performance over the traditional method.



(a) SSIM result (original version)



(b) SSIM result (scaled version)

Figure 4.3 - SSIM result of the one-stereo-pair disparity sequence accuracy and our disparity sequence accuracy in ideal environment

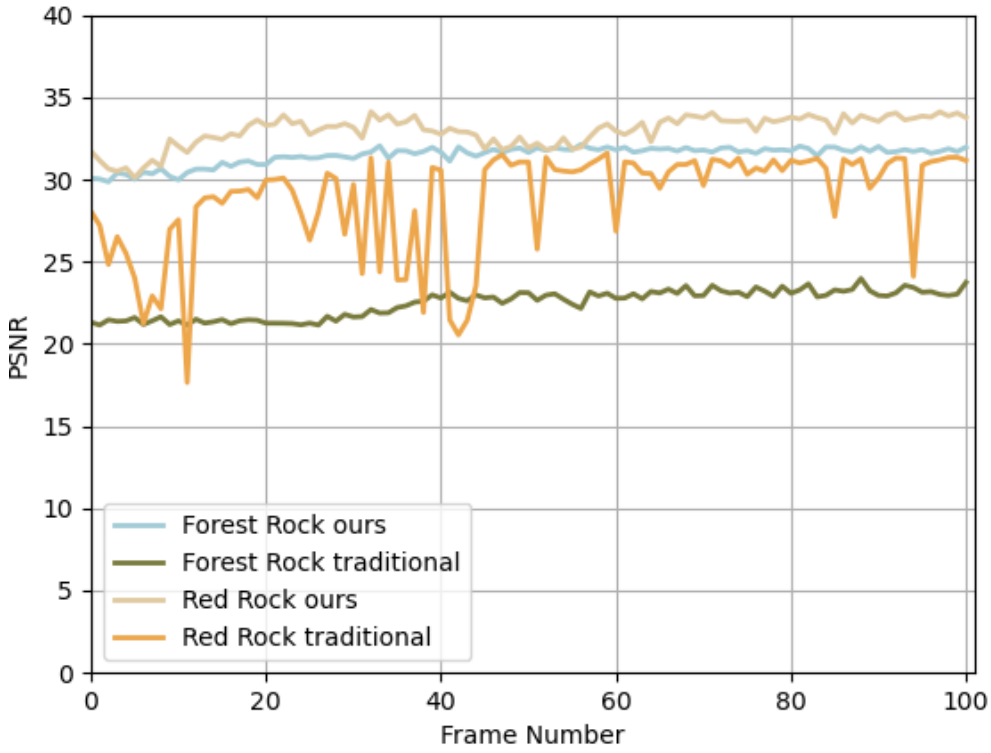
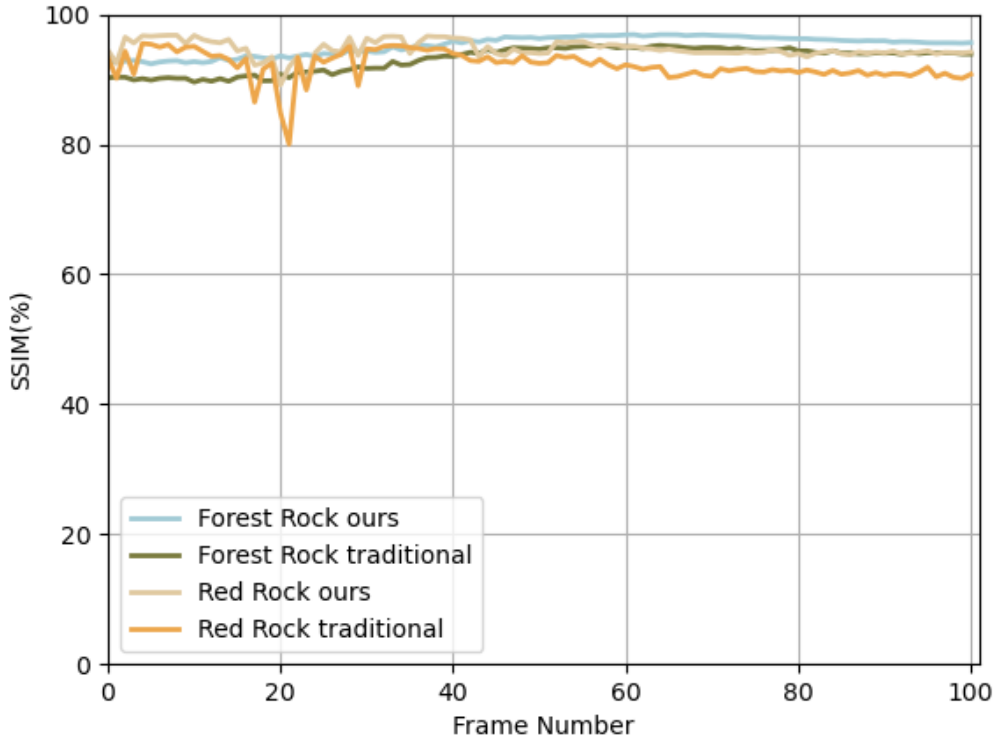
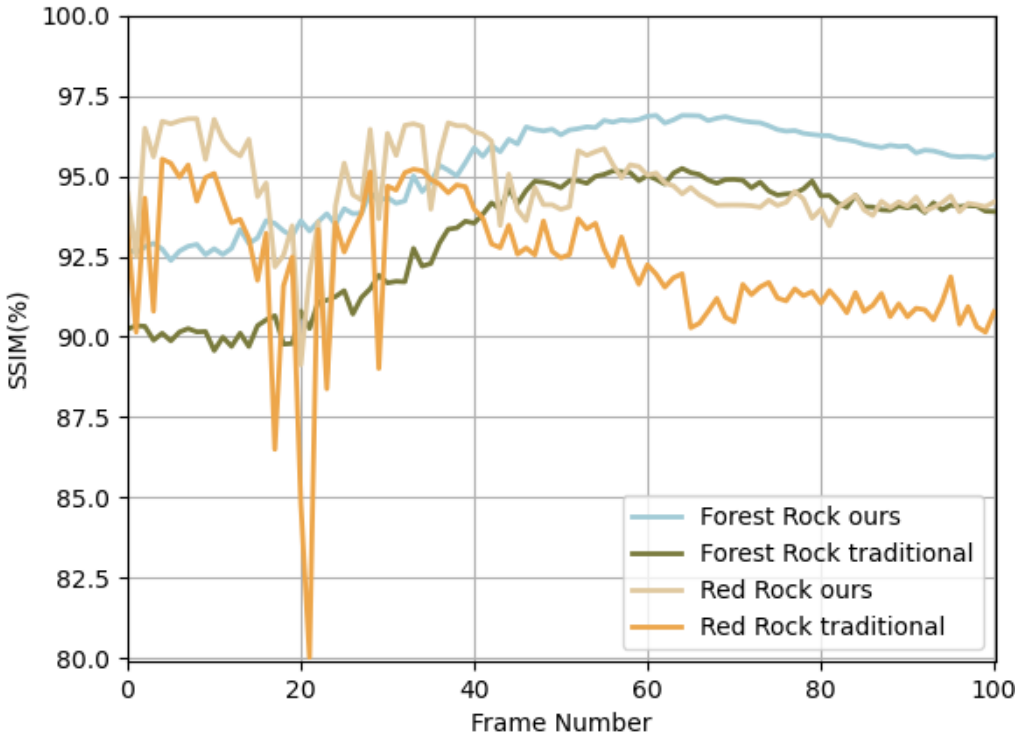


Figure 4.4 - PSNR result of the one-stereo-pair disparity sequence accuracy and our disparity sequence accuracy in ideal environment



(a) SSIM result (original version)



(b) SSIM result (scaled version)

Figure 4.5 - SSIM result of the one-stereo-pair disparity sequence accuracy and our disparity sequence accuracy in semi-realistic environment

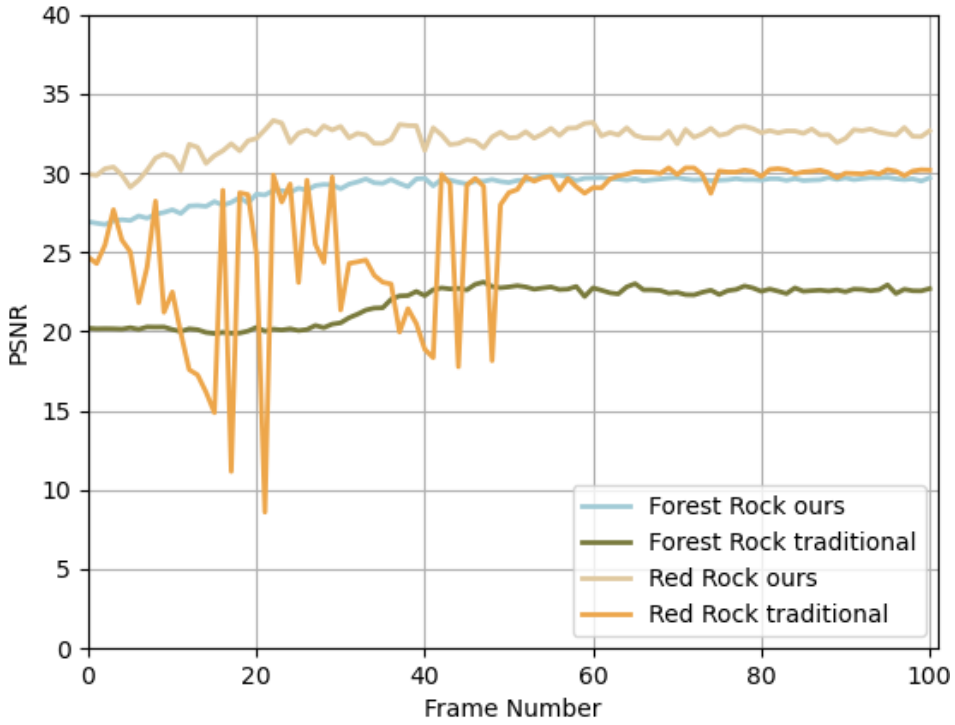
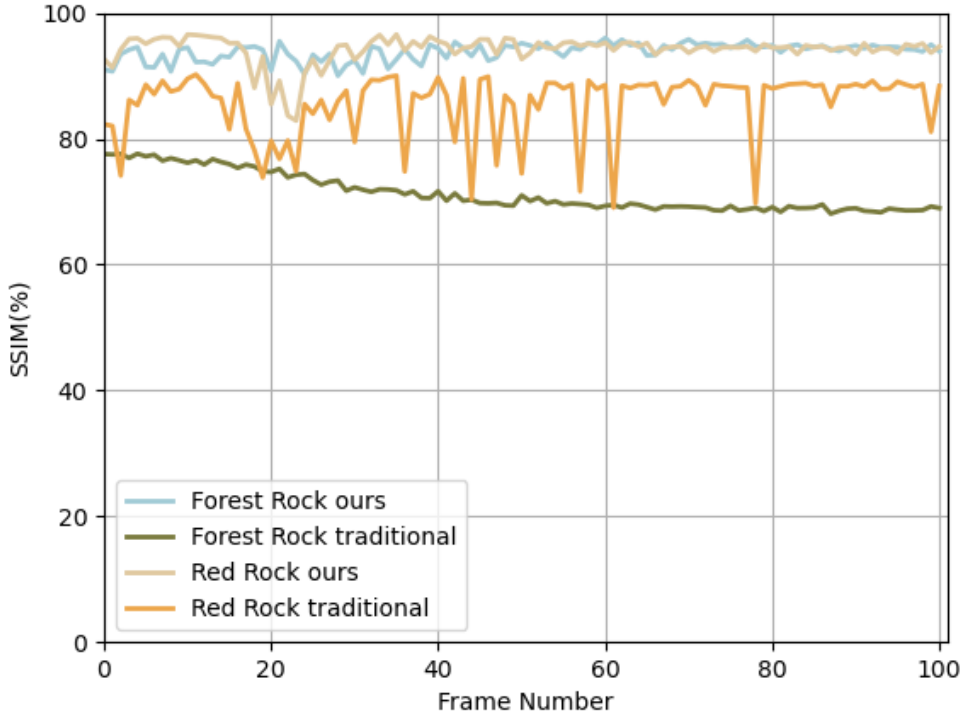
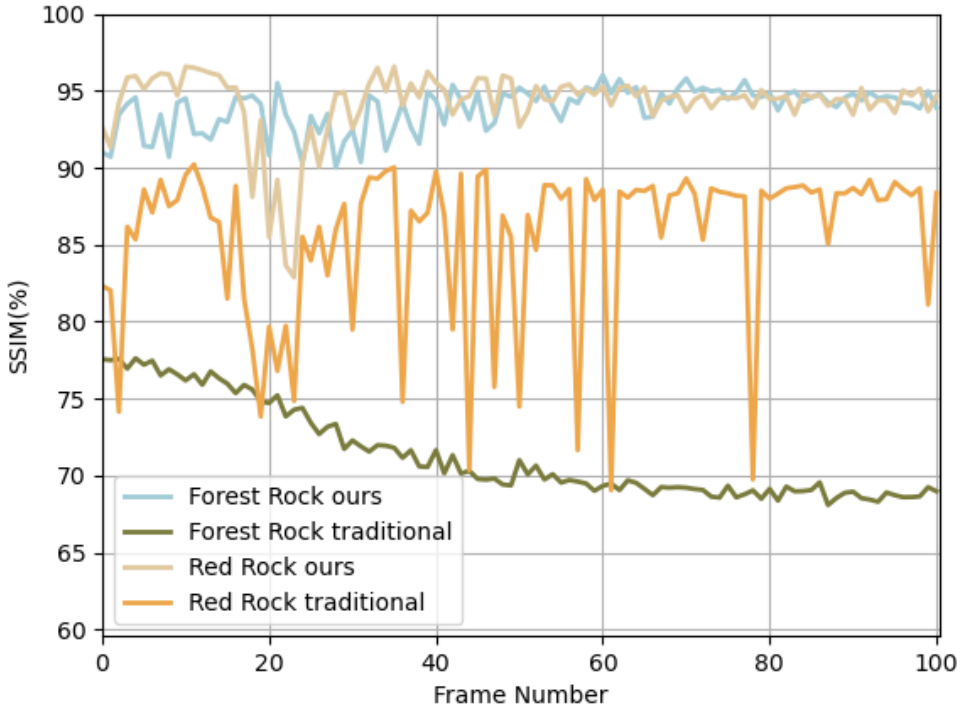


Figure 4.6 - PSNR result of the one-stereo-pair disparity sequence accuracy and our disparity sequence accuracy in semi-realistic environment

When it comes to the realistic environment, although we have carefully rectified all the stereo pairs, the vertical parallax between stereo pairs cannot be eliminated. Thus, the simulated camera position deviation can still bring some mismatching to the correspondence algorithm and the output. Adding the errors from existing occlusion and texture-less areas, the mean depth accuracy of the traditional one-stereo-pair curves possesses a wider accuracy gap relative to the curves from our method.



(a) SSIM result (original version)



(b) SSIM result (scaled version)

Figure 4.7 - SSIM result of the one-stereo-pair disparity sequence accuracy and our disparity sequence accuracy in realistic environment

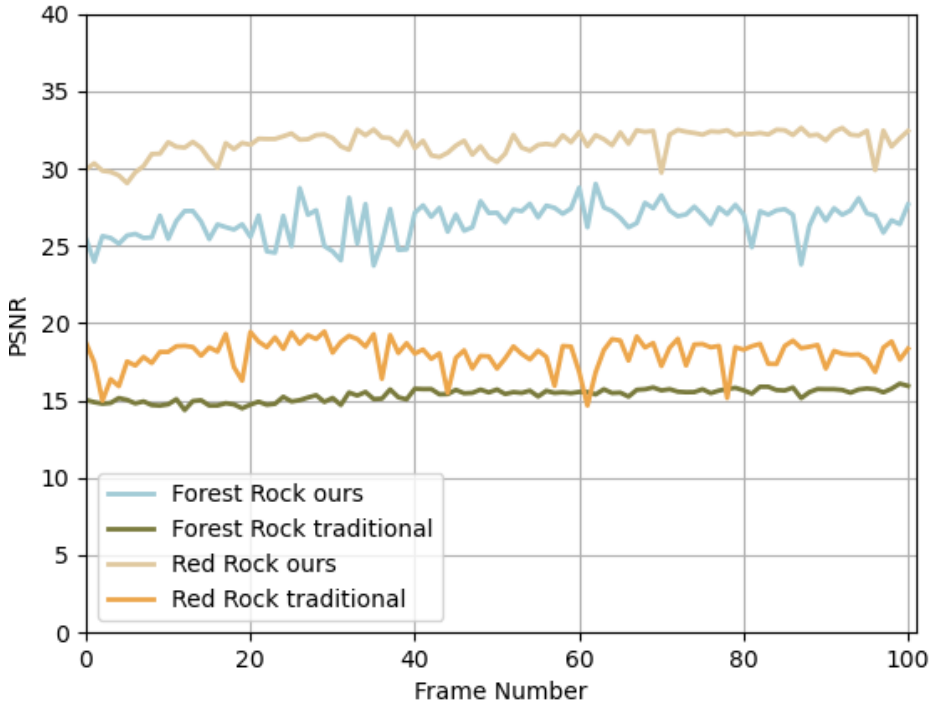


Figure 4.8 - PSNR (c) result of the one-stereo-pair disparity sequence accuracy and our disparity sequence accuracy in realistic environment

In the 100-frame depth accuracy comparison, the Red Rock curve seems bumpier due to the moving camera setup and the texture-less background, the Forest Rock curve is much smoother as its camera setup is stationary. One unignorable fact is that our method smoothed out many of the accuracy valley of the traditional method in both SSIM and PSNR results, which can prove that our method generates a more stable and consistent disparity sequences than the traditional one-stereo pair method.

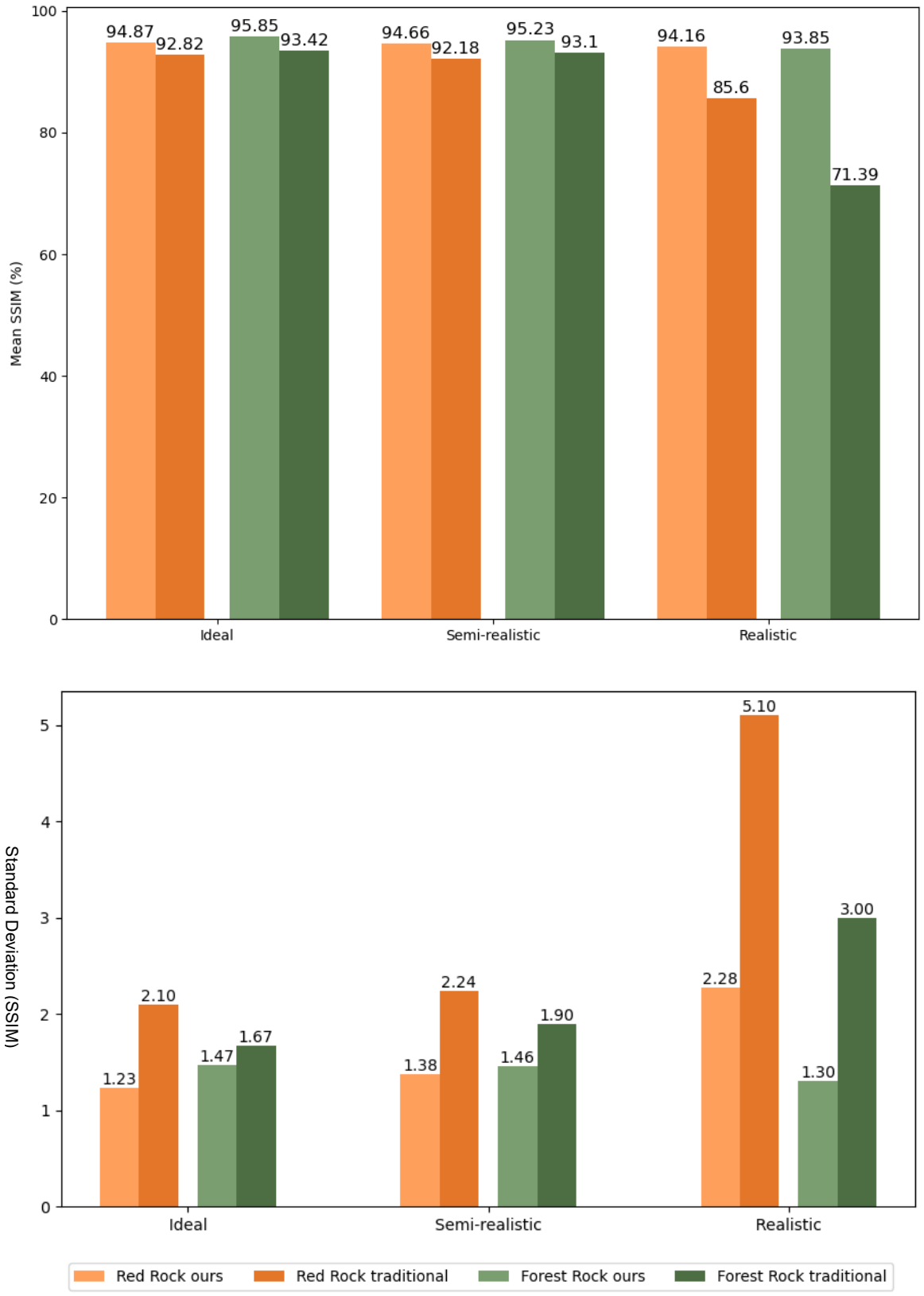


Figure 4.9 - The SSIM result (up) and the standard deviation (down) of the disparity accuracy under three simulated environments

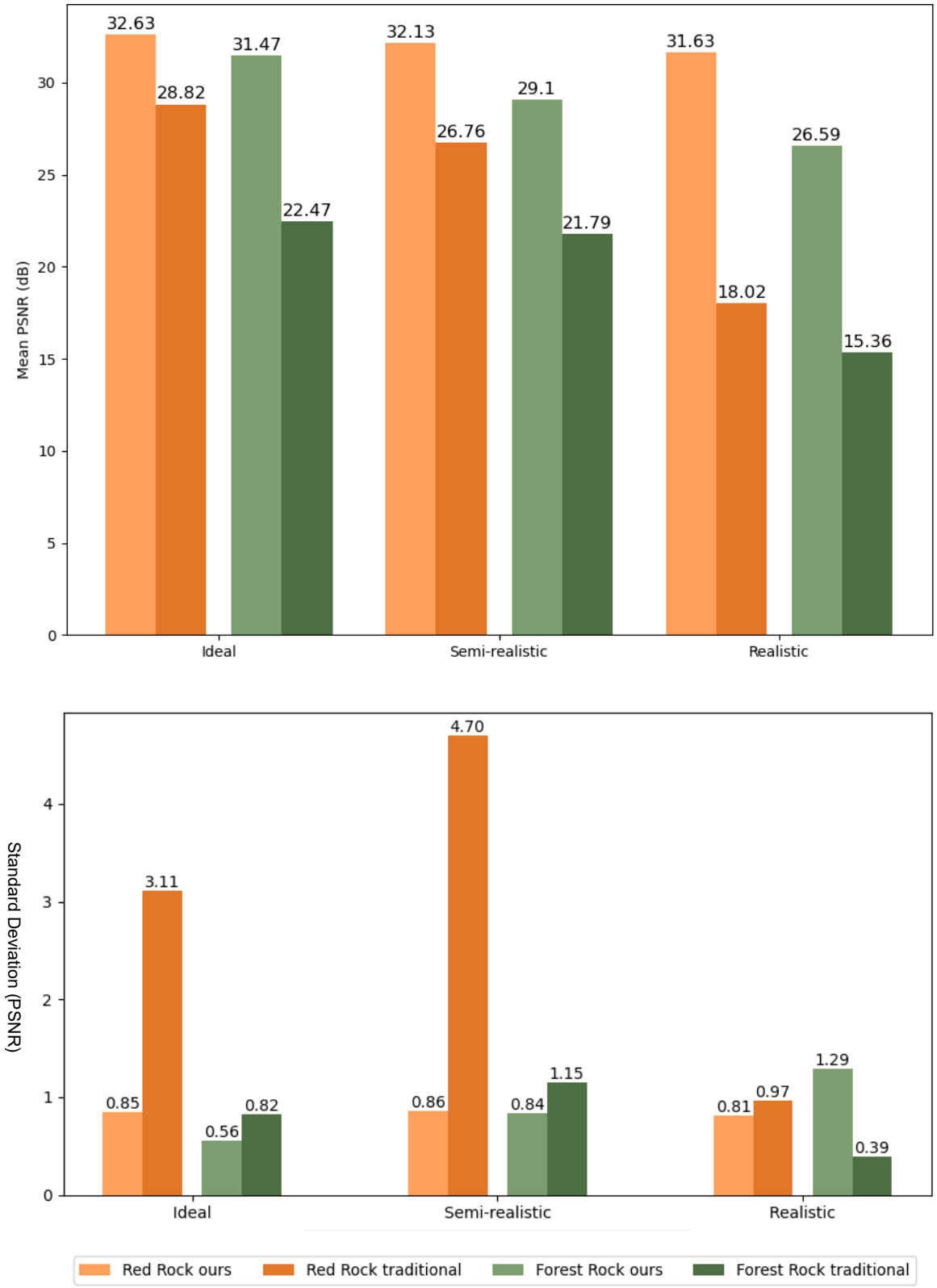


Figure 4.10 - The PSNR result (up) and the standard deviation (down) of the of the disparity accuracy under three simulated environments

Figure 4.9 and Figure 4.10 is a better illustration of the overall performance of our method, the standard errors are shown in separated charts since some of them are too small to be illustrated as error bars. In Figure 4.9, our method shows higher SSIM value in ideal and semi-realistic environments and the value remain stable in a realistic environment while the value of traditional method drops significantly. In Figure 4.10, the PSNR value follows the similar pattern of SSIM that the accuracy gaps between our method and traditional method are relatively small in ideal and semi-realistic environments but widened when it comes to realistic environment. In both the SSIM and PSNR results, our method's standard errors are lower than the traditional method in most of the cases.

4.1.2 Matting Accuracy

In the matting accuracy part, we compared the foreground extraction matte accuracy of chromakey and our disparity matting method. The chromakey plugin that we used in the experiment is Keylight 1.2, an industry-proven keyer with outstanding keying results and abundant adjusting parameters that can give the user-friendly control over the matting quality.

The matting accuracy is also evaluated under ideal, semi-realistic, and realistic environments. The Keylight matting results are separated into raw and adjusted versions for a more comprehensive comparison since Keylight might reach a high matting accuracy with some parameter changes. The raw data is simply extracting the green screen color from the video without any parameter adjustment, and the adjusted version is the best Keylight matting accuracy we can get by changing the screen balance, despill

bias, and screen matte parameters. The adjustment does not involve any garbage matte or editing aside from Keylight.

However, in this section, we did not adapt PSNR as part of the evaluation metric since most of the matte data are very binary-alike images, while PSNR might generate misleading results in such circumstances [82]. In Figure 4.11, the four distorted images share the same PSNR value when comparing them with the reference image, but their quality from human visual perspective obviously vary [82].

(a) (b)

Figure 4.11 – An example of image distortion (a) reference image, (b) distorted images [82]

Figure 4.12 - Ground truth matte of the Red Rock scene

Table 4.3 - Matting result of the Red Rock scene

	Raw matting result from Keylight 1.2	Adjusted matting result from Keylight1.2	Matting result from our method
Ideal			

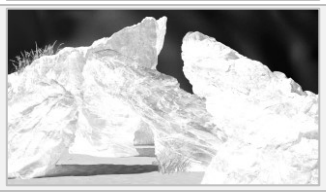
Semi-Realistic			
Realistic			

Figure 4.13 - Ground truth matte of the Forest Rock scene

Table 4.4 - Matting result of the Forest Rock scene

	Raw matting result from Keylight1.2	Adjusted matting result from Keylight1.2	Matting result from our method
Ideal			
Semi-Realistic			
Realistic			

As can be observed from Table 4.3 and Table 4.4, in an ideal or semi-realistic situation, the adjusted Keylight results and our results show similar matting quality. The adjusted Keylight results even show higher accuracy on the object edges when there are irregular boundaries and fine details.

However, when it comes to realistic environments, the color spill and the shadow on the screen caused many matting errors. We can hardly remove the errors from the color spill and shadows even with manual adjustment, where it might require a frame by frame garbage matte that is a time-consuming process similar to the rotoscoping technique.

Looking into the matting accuracy as per frame, it is not surprising to see that adjusted chromakey results are sometimes more superior than ours in ideal and semi-realistic environment, chromakey is a mature technique that has been developed for decades and surely has its advantages over many methods. However, one thing we should not ignore is that the unadjusted chromakey results are always less accurate than ours, which means it always has to involve manual adjustment for a decent output while our method performs well by running automatically.

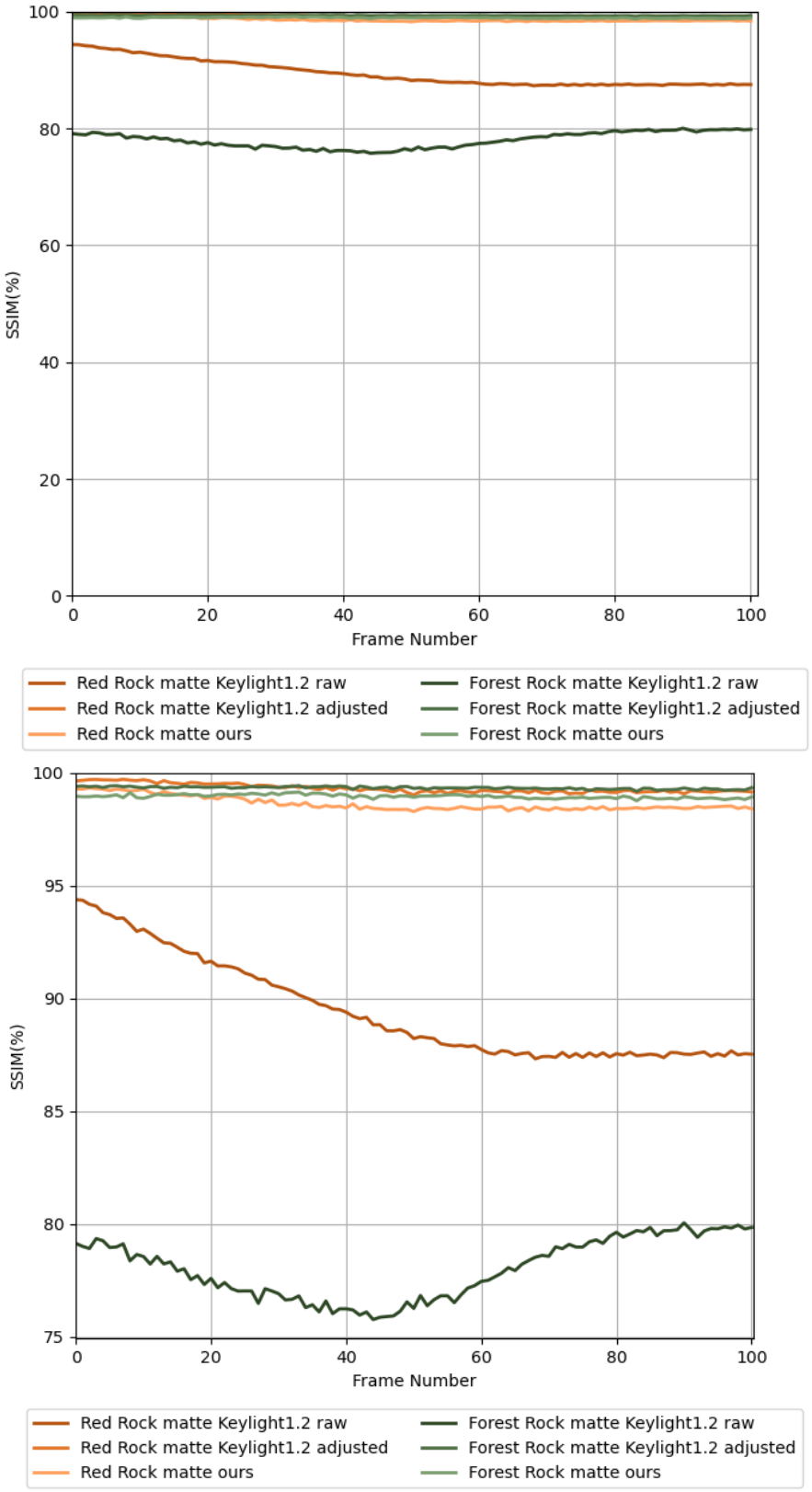


Figure 4.14 –SSIM result of the chromakey matting accuracy and our disparity matting accuracy in ideal environment (up: original version; down: scaled version)

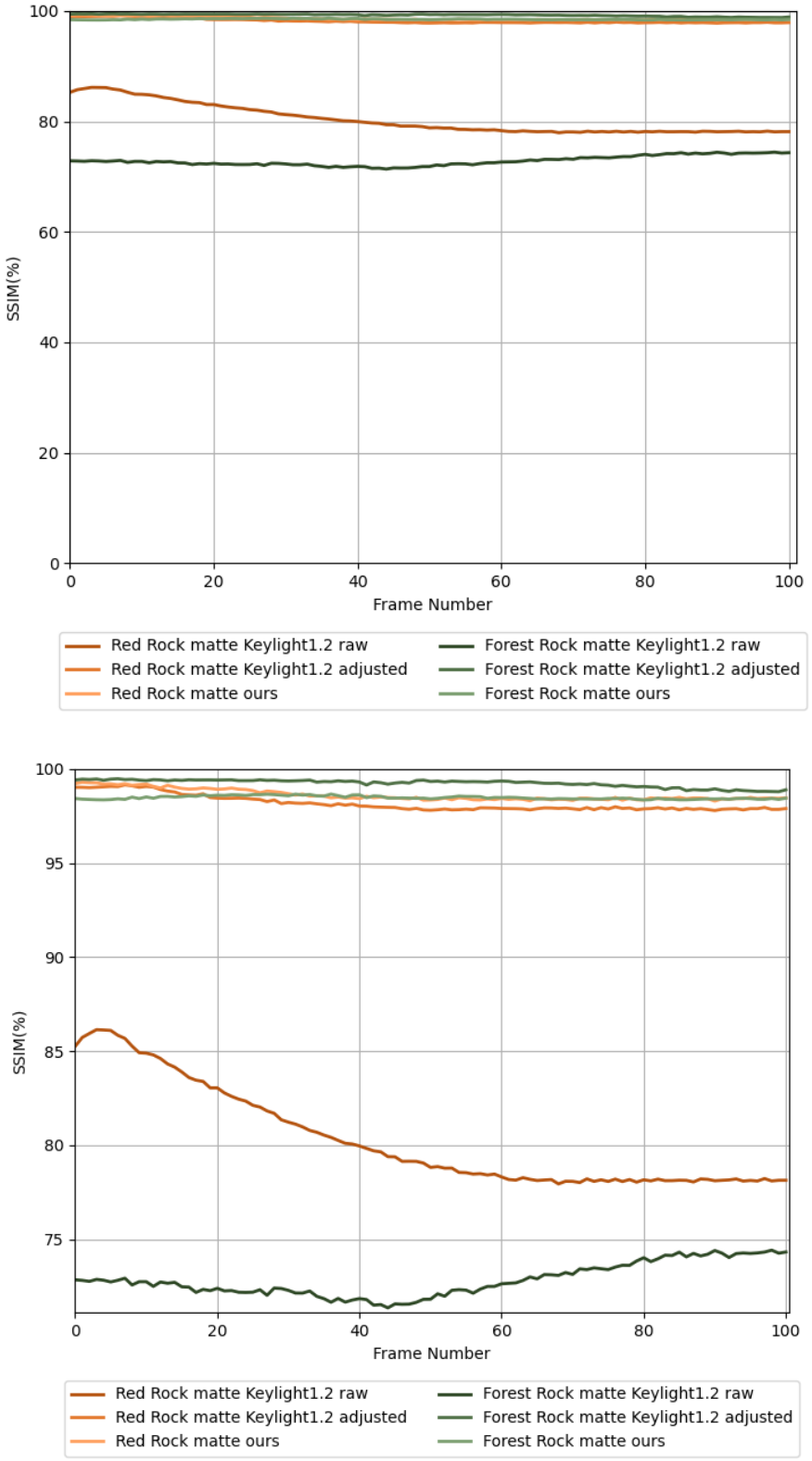


Figure 4.15 - SSIM result of the chromakey matting accuracy and our disparity matting accuracy in semi-realistic environment (up: original version; down: scaled version)

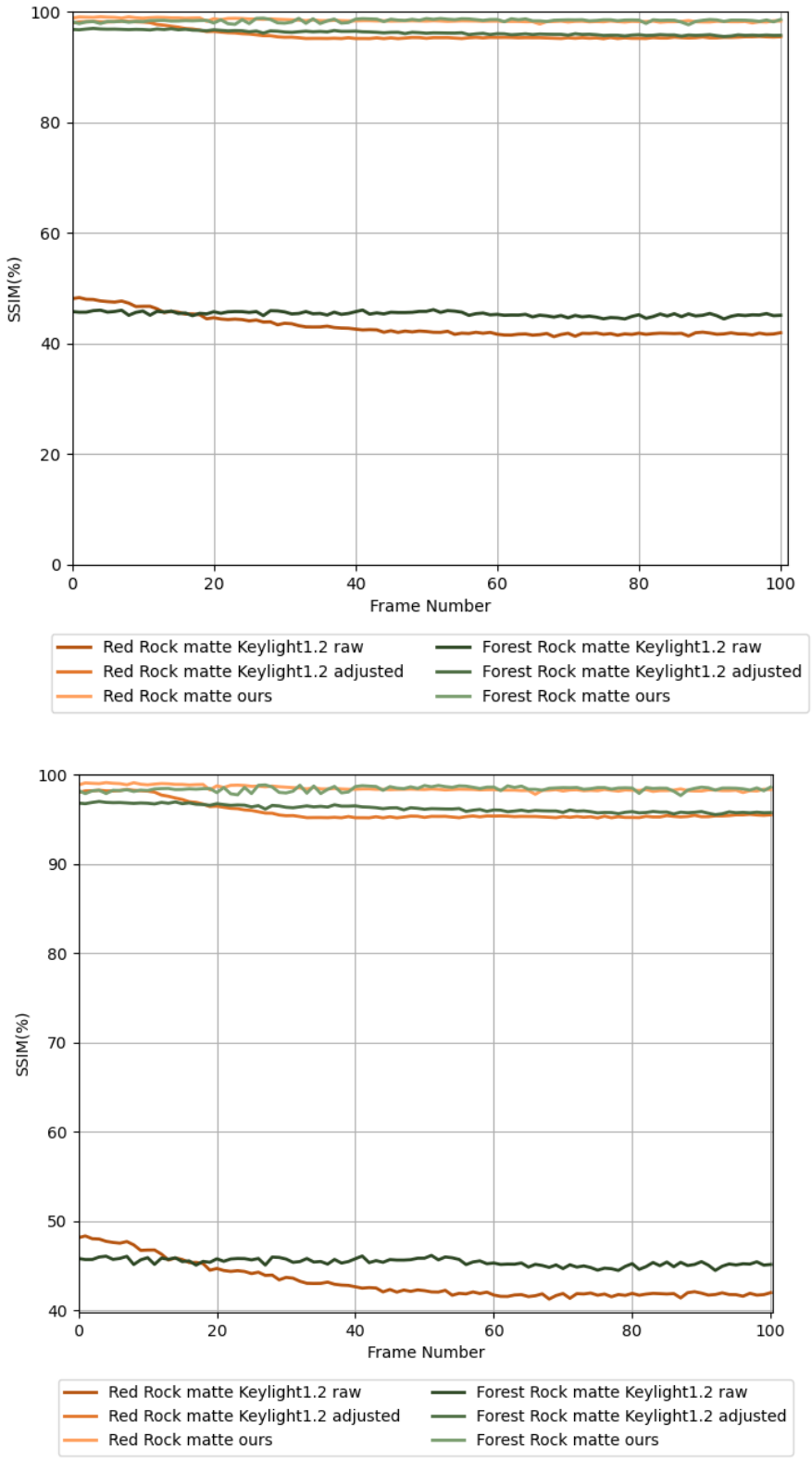


Figure 4.16 - SSIM result of the chromakey matting accuracy and our disparity matting accuracy in realistic environment (up: original version; down: scaled version)

When it turns into realistic environment, the unadjusted chromakey results drop a lot, the adjusted results however, become less accurate than our method due to the color spill and shadows.

About the big fluctuations of the raw chromakey accuracy curves, since both scenes have animated objects or cameras, the movements might cause various reflection color or color spill amount on the foreground objects, hence the accuracy of matting are influenced accordingly.

Figure 4.17 shows the overall matting accuracy under three environments, the standard error is shown in a separated chart since some of them are too small to be shown as error bars. Different from the disparity accuracy, the accuracy difference of matting accuracy between our method and adjusted Keylight matte is not very big though the raw Keylight matte has relatively low accuracy. Our method is still more superior than unprocessed Keylight results in all the cases and is also better than the processed Keylight results in realistic setups. Besides, our standard errors are also lower than Keylight in most of the environments.

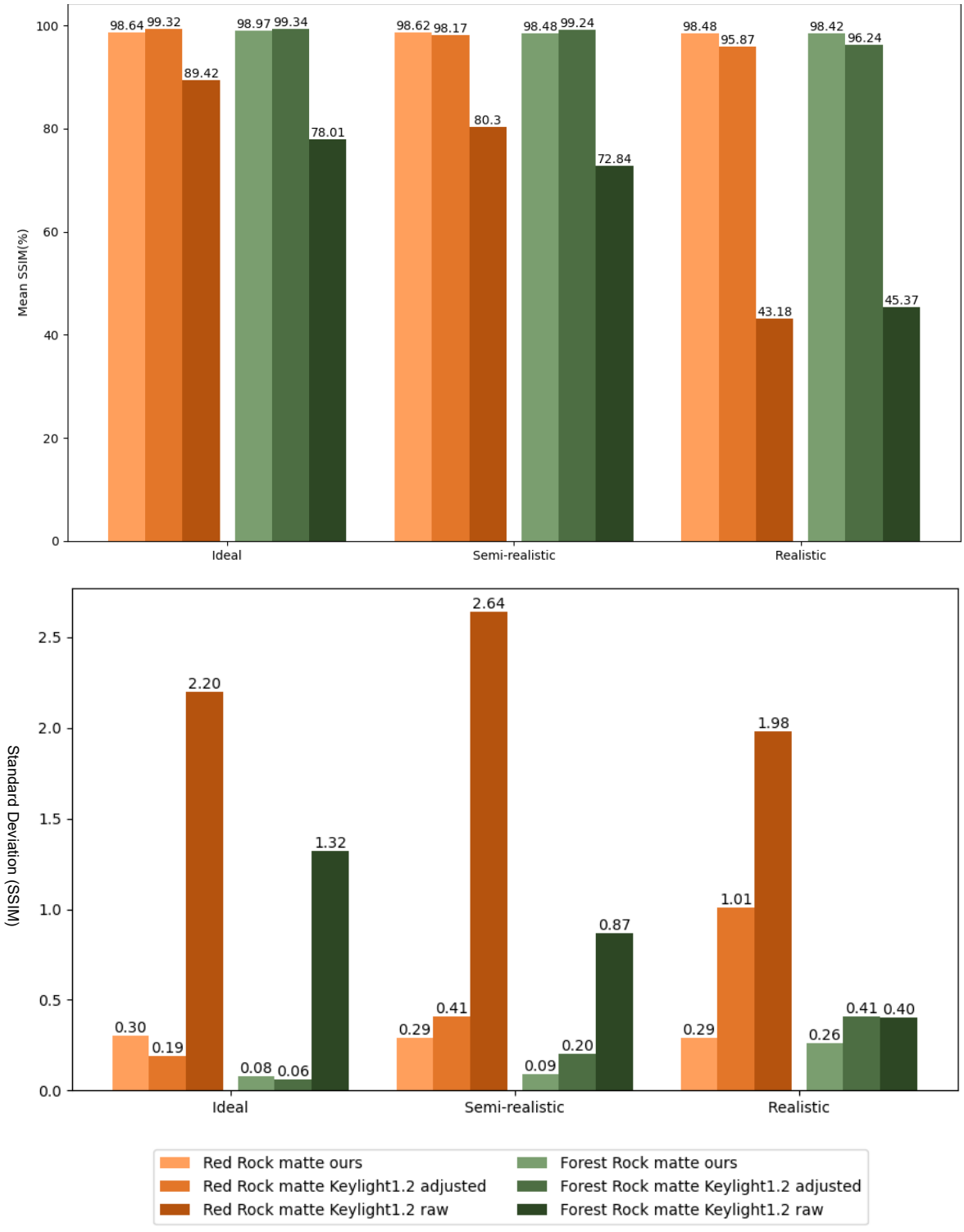


Figure 4.17 - The SSIM result (up) and the standard deviation (down) of the matting accuracy under three simulated environments

4.1.3 Summary

Our experiment's quantitative results have shown that our method has the most significant advantage in the realistic environment, which is the decisive scenario to validate our hypothesis. Our method visually and numerically provides better disparity accuracy than the traditional one-stereo-pair method and a satisfactory performance when comparing its matting accuracy with a mature matting tool – Keylight1.2. Besides, our method is automated and does not require further manual adjustments.

4.2 Quality Results

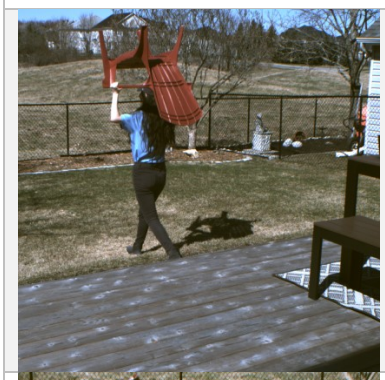
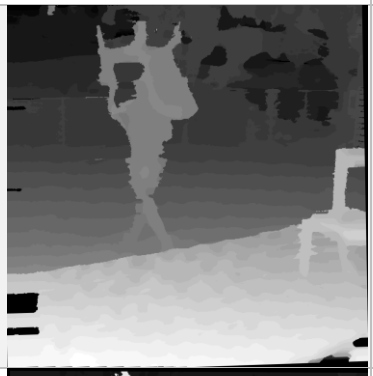
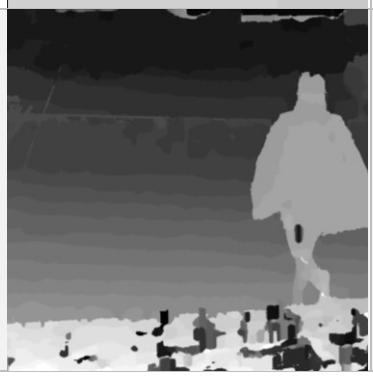
Since it requires precision equipment to replicate an object's movement in a live-action scenario and current depth estimation equipment are either constrained by sunlight or distance, it is hard to acquire various groups of data with only our goal-controlled variables well as getting a ground truth. As a result, we consider the live-action environment more as quality evaluation.

For the live-action environment, we recorded videos from eight scenes and selected three that are most well-synchronized with no corrupted or skipped frames.

4.2.1 Disparity Quality

In the disparity quality part, we generated depth sequences with both the traditional one-stereo-pair method and our method. When comparing their results, our output showed cleaner edges, smoother depth gradient, and has filled many depth information gaps and fixed many disparity errors brought by occlusion, mismatching, or ambiguous boundaries.

Table 4.5 - Disparities of the live-action scenes

Original frame	Disparity from one-stereo-pair method	Disparity from our method
		
		
		

As is shown in Table 4.5, our method also works for monochrome videos though the object edge will show slightly more fuzziness since the original sequences do not possess any color information. This leads to a limited optimization when we use color-aware approaches to optimize the final output while the traditional one-stereo-pair method creates much more errors on regions with repeated patterns.

4.2.2 Matting Quality

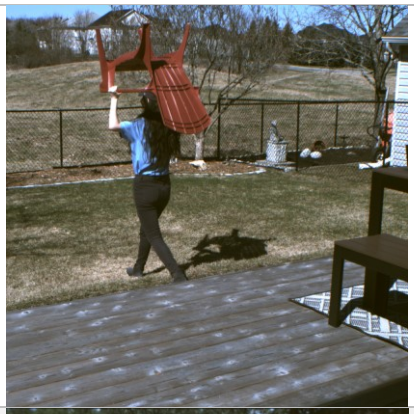
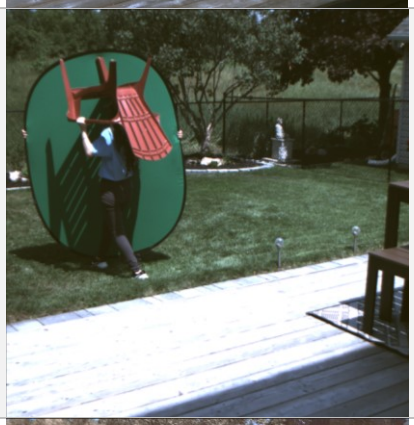
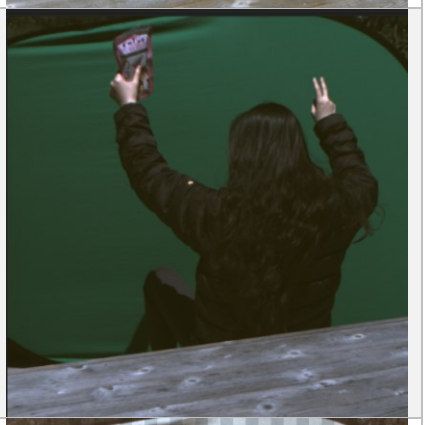
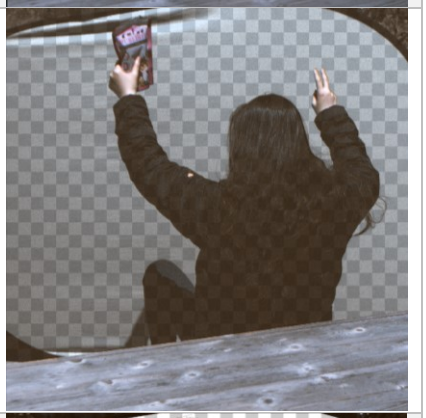
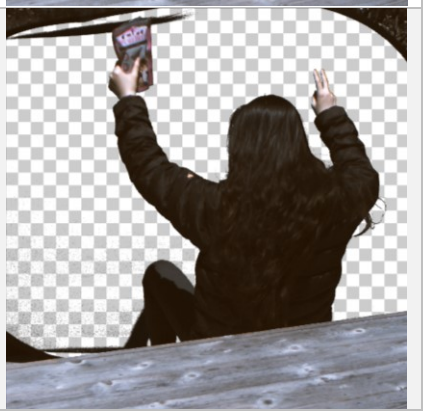
To evaluate the matting quality of the live-action environment, we followed the idea in disparity accuracy evaluation part that we put the matting results from Keylight and our method into comparison.

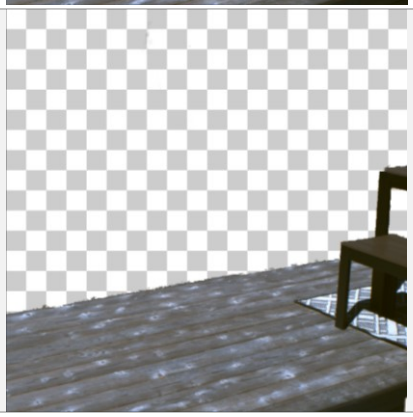
For the disparity sequences, we used our system to capture screen-free scenes from five perspectives; for the chromakey sequences, we tried to replicate the movements from the screen-free scenes and have filmed corresponding videos.

Our results showed that if the green screen size is not big enough, we will need to garbage matte many areas from the scene. Even on the green screen covered regions, it is hard to acquire a clean and noise-free matte if there is hard shadow projected on the screen or any object with green tint in the shots. For example, the unadjusted Keylight matting results look terrible in both video sequences (see in Table 4.6). The adjusted version, by looking closely, we can still find many noises on the shadow area and some random hollows on target objects, and the no-screen regions are waiting to be manually matted.

On the contrary, our method does not require any extra setup; we only need to generate the disparity sequence for the shot, select the range of depth that we want to create the matte from, and easily remove the background and maintain the solidness of the foreground objects. This also leads to another advantage of our method: we can select an arbitrary range of depth from a generated disparity sequence to mask out any part of the scene instead of only extracting the object with a green screen behind.

Table 4.6 – Matting comparison between chromakey and our method

Stereo frame		
Chromakey frame		
Matting result from Chromakey raw		
Matting result from Chromakey processed		

Matting result from our method (big matting range)		
Matting result from our method (small matting range)		

Asides from the advantages that we mentioned above, we still need to admit that our matte quality has its drawback on edges and fine details such that the armrest of the chair and the hair of the character are not entirely cut out, and the edges of the target objects have some slight jittering due to the homogenous colors in the foreground and background.

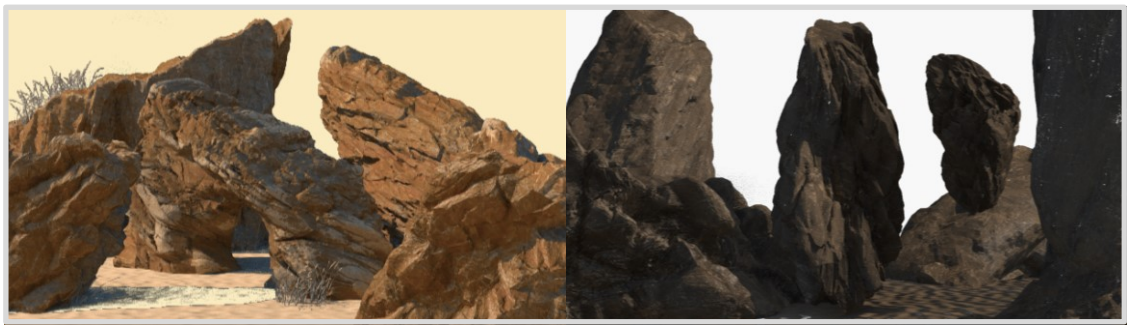
4.2.3 Summary

The quality results of our experiment have shown that the matte created from our results has its superiority on simple setup, arbitrary matting range, and reliable matting accuracy within the objects; the chromakey method takes more efforts to set up and has stricter requirements to the environment as well as limited matting accuracy if there is

key color spill on target objects or sharp shadows on the screen while it preserves better edges or detail information.

4.3 Composition Test

To further demonstrate the matting capability of our method, we composite some visual effects into the videos and compared it with the composition result from chromakey.



(a) with Chromakey (Keylight1.2)

(b) with our disparity sequence (small depth range)

(c) with our disparity sequence (big depth range)

Figure 4.18 – Insert effects into the realistic environment with Chromakey (Keylight1.2) and our method

(a) with Chromakey (Keylight1.2) **(b) with our disparity sequence**

Figure 4.19 – Replace the background and composite effects into the realistic environment with Chromakey (Keylight1.2) and our method

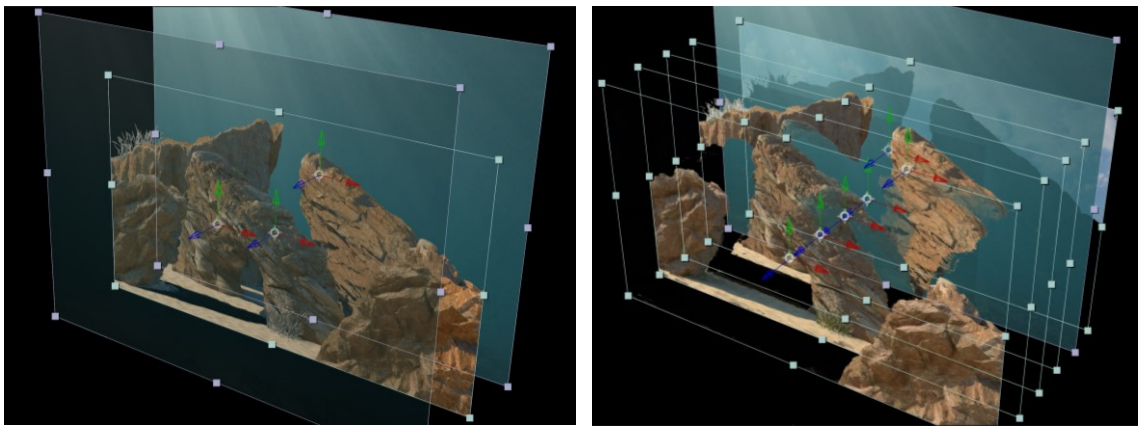


Figure 4.20 – Difference between Chromakey composition(left) and our method's composition(right)

As can be observed from Figure 4.18, since Chromakey only provides a two-layer matte that separates the foreground and background as two planes, we can only put the sandstorm effects in behind the rocks. However, with disparity sequence, the scenes are sliced into multiple layers (Figure 4.20), where we can select a specific range to insert the effects. When compositing effects such as storm, fog, or flood, objects that are farther

away from the cameras can get more influence from the effects while the objects closer to the camera can get less, which follows the way that human eyes perceive an environment and provides more sense of depth to the viewers.

In Figure 4.19, we replaced the background of the videos and add some translucent effects into the scene. It turns out that with our method, the effects are better blended into the videos while chromakey only receive the effects as a layer on top.

(a) with Chromakey (Keylight1.2), no garbage matte

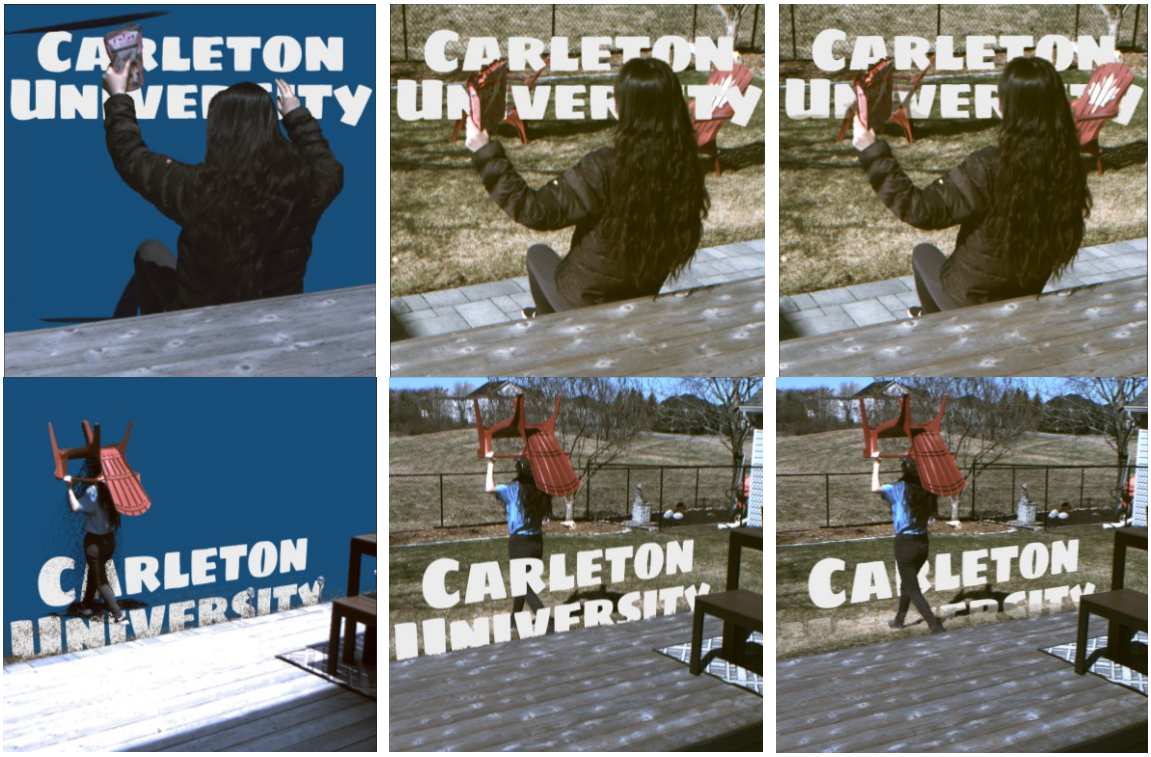
(b) with Chromakey (Keylight1.2), garbage matted

(c) with our method

Figure 4.21 -Compositing fog into the live-action videos

The composition test with the live-action videos also shows the drawback of Chromakey technique that: the background has to be completely replaced so we will have to refilm the background if we want to reserve it; the areas that are not covered by green screen have to be garbage matted; any hard shadow on the screen can cause big errors to the matting result; and the effects with depth can hardly be composited. On the other

hand, our method can be used to flexibly reserve or replace the background, and the effects with depth can be naturally composited into the scene.



(a) with Chromakey (Keylight1.2), garbage matted

(b) with our method, small depth range matting

(c) with our method, big depth range matting

Figure 4.22 – Compositing text into the live-action videos

Chapter 5: Conclusions

In this research, we present a new multi-baseline stereo idea that perceives a scene with our five-camera system, where cameras from different directions filled the blind spots that other cameras might not be able to capture due to occlusion or other noises. We also present a two-axis image rectification solution that efficiently reduced the difficulty of rectifying stereo pairs from various directions, and a SURF-based minimum and maximum disparities auto detector, with which our system can be compatible with moving camera shots. Besides, we explored the potential of applying our method into a post-production pipeline that provides a new solution of auto-object-extraction.

5.1 Findings

According to the quantitative and qualitative results from our experiment, our five-camera system does help improve disparity quality by reducing occluded areas and patching them with information acquired from other directions. By comparing it with the traditional one-stereo-pair disparity acquisition method, our disparities have better performance in texture-less and low-light regions. It shows significant advantages both visually and numerically when we use them for simulated real-life challenges.

Comparing the matte accuracy between our disparity-based-matting method and the commercial color-key-matting method, our results have less accuracy in an edge-detail-preserving perspective but provide outstanding matte accuracy within objects though after adding highly influential noises such as camera positional deviations and color-inconsistency. Besides, our results smoothly reflect the depth gradient of objects, which provides visual effects artists another option to composite big scale effects like fog, explosion, storm.

5.2 Limitations

Asides from the contributions and the exciting results we have acquired, our research has its limitations in some aspects.

Limitation in comparison: In this research, we compared the disparity quality of our method and traditional one-stereo pair method. However, we haven't compared our two-axis method with other one-axis multi-baseline stereo methods. We believe by adding this part, our comparison could be more comprehensive and convincing.

Limitation in disparity quality: The algorithm used for disparity generation is basing on a localized correspondence matching algorithm with guided filter, which is independent from window size. Therefore, it is fast and efficient. On the other hand, just like many other local methods, this algorithm is limited in object boundary details and smoothness, while global methods that perform better in this part are most time-consuming. This has given us limited space to improve the disparity quality, and we believe the absence of our own correspondence algorithm is the crux.

Limitation in transparent objects: Our research has not yet explored the disparity matting for transparent or semi-transparent objects while it might be a frequently encountered task in real-life video filming scenarios, we shall explore the possibility of applying our method in such circumstances and replenish our system for more complicated situations.

5.3 Future Works

One positive influence that might be brought by our system is that it provides a potential new matting or say object extraction approach for film post-production. However, our system still has a big room to improve, whether in hardware deployment or

software design. We can update the devices in the hardware to minimize the camera positional and optical deviation, which would be a constructive improvement to the final output quality. In the software part, we shall design our own correspondence algorithm and integrate the multiple occlusion maps into the matching procedure. In this case, we may acquire a better-unified result instead of externally enhancing the disparities after they are generated. In the application, if the system can be optimized to real-time, we believe our system can also provide a more interactive VR or AR experience by adding the depth information; and it is not hard to imagine applying such technique in auto-pilot or better image post-processing tasks.

References

[1] J. Nėjė, "18 Revealing Before-And-After VFX Shots From Your Favorite Movies And TV Series," BoredPanda, 2014. [Online]. Available: https://www.boredpanda.com/before-after-vfx-movies-tv-hollywood-magic/. [Accessed August 2019].

[2] RotoSpline, "Roto Shape Reels," RotoSpline, 2011. [Online]. Available: https://vimeo.com/channels/franciswong1982/24652326. [Accessed August 2019].

[3] E. M. O'Connell, "Amazing Photos That Reveal The Real Footage Behind Disney Films," 2016. [Online]. Available: https://guff.com/amazing-photos-that-reveal-the-real-footage-behind-disney-films. [Accessed August 2019].

[4] M. Seymour, "The Art of Roto: 2011," fxguide, 10 October 2011. [Online]. Available: https://www.fxguide.com/fxfeatured/the-art-of-roto-2011/attachment/pr_033_0300_v07_rotoref_a/. [Accessed August 2019].

[5] T. Kanade, A. Yoshida, K. Oda, H. Kano and M. Tanaka, "A stereo machine for video-rate dense depth mapping and its new applications," in *CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, USA, 1996.

[6] R. Gvili, A. Kaplan, E. Ofek and G. Yahav, "Depth Keying," *Stereoscopic Displays and Virtual Reality Systems X,* vol. 5006, 2003.

[7] L. Wang, M. Gong, C. Zhang, R. Yang, C. Zhang and Y.-H. Yang, "Automatic real-time video matting using time-of-flight camera and multichannel poisson equations," *International Journal of Computer Vision,* pp. 1-18, 2011.

[8] F. Devernay, "How do you get precise 3D data for accurate and intelligently automated real-time interaction?," Framos, 2018.

[9] J. Park, H. Kim, Y.-W. Tai, M. S. Brown and I. Kweon, "High Quality Depth Map Upsampling for 3D-TOF Cameras," in *2011 International Conference on Computer Vision*, Barcelona, 2011.

[10] K. He, J. Sun and X. Tang, "Guided Image Filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 35, no. 6, pp. 1397-1409, 2013.

[11] A. Hosni, C. Rhemann, M. Bleyer and C. R. a. M. Gelautz, "Fast Cost-Volume Filtering for Visual Correspondence and Beyond," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 35, no. 2, pp. 504-511, 2013.

[12] N. Joshi, W. Matusik and S. Avidan, "Natural video matting using camera arrays," *ACM Transactions on Graphics,* vol. 25, no. 3, pp. 779-786, 2006.

[13] J. A. Ball, "The Technicolor Process of Three-Color Cinematography," *Journal of the Society of Motion Picture Engineers,* vol. 25, no. 2, pp. 127-138, 1935.

[14] L. Kelion, "Blue and green-screen effects pioneer Petro Vlahos dies," BBC News, 14 February 2013. [Online]. Available: https://www.bbc.com/news/technology-21463817. [Accessed 10 August 2020].

[15] P. Vlahos, A. Dadourian and G. Sauve, "Method and apparatus for adjusting parameters used by compositing devices". USA Patent 5907315, May 1999.

[16] Adobe, "Adobe After Effects User Guide," Adobe, [Online]. Available: https://helpx.adobe.com/ca/after-effects/using/keying-effects.html#color_difference_key_effect. [Accessed 19 8 2019].

[17] Y. Mishima, "A software chromakeyer using polyhedric slice," in *NICOGRAPH 1992*, Tokyo, Japan, 1992.

[18] L. Yin, *Automatic Stereoscopic 3D Chroma-Key Matting Using Perceptual Analysis and Prediction,* Ottawa: School of Electrical Engineering and Computer Science Faculty of Engineering University of Ottawa, 2014.

[19] J. Wang and M. F. Cohen, "Image and video matting: A survey," *Foundations and Trends in Computer Graphics and Vision,* vol. 3, no. 2, pp. 97-175, 2007.

[20] M. A. Ruzon and C. Tomasi, "Alpha estimation in natural images," in *2000 IEEE Conference on Computer Vision and Pattern Recognition*, Hilton Head Island, 2000.

[21] Y. Chuang, A. Agarwala, B. Curless, D. Salesin and R. Szeliski, "Video Matting of Complex Scenes," in *29th annual conference on Computer graphics and interactive techniques*, San Antonio, 2002.

[22] J. Sun, J. Jia, C.-K. Tang and H.-Y. Shum, "Poisson matting," *ACM Transactions on Graphics,* vol. 23, no. 3, pp. 315-321, 2004.

[23] A. Levin, D. Lischinski and Y. Weiss, "A Closed-Form Solution to Natural Image Matting," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 30, no. 2, pp. 228-242, 2008.

[24] zLense, "zKey Greenless Keying," zLense, April 2017. [Online]. Available: http://zlense.com/zkey-greenles-keying/. [Accessed 2019-2020].

[25] M. Subbarao and T. Wei, "Depth from defocus and rapid autofocusing: a practical approach," in *1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Los Alamitos, CA, USA, 1992.

[26] M. McGuire, W. Matusik, H. Pfister, J. F. Hughes and F. Durand, "Defocus video matting," in *ACM Transactions on Graphics*, New York, 2005.

[27] O. Wang, J. Finger, Q. Yang, J. Davis and R. Yang, "Automatic Natural Video Matting with Depth," in *15th Pacific Conference on Computer Graphics and Applications*, Maui, HI, USA, 2007.

[28] L. Wang, C. Zhang, R. Yang and C. Zhang, "Tofcut: Towards robust real-time foreground extraction using time-of-flight camera," in *3DPVT 2010*, Paris, France, 2010.

[29] M. Kass, A. Witkin and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision,* pp. 321-331, 1988.

[30] A. Agarwala, A. Hertzmann, S. Seitz and D. Salesin, "Keyframe-based tracking for rotoscoping and animation," in *SIGGRAPH 2004*, Los Angeles, CA, US, 2004.

[31] Y. Li, J. Sun and H.-Y. Shum, "Video object cut and paste," in *ACM SIGGRAPH 2005*, Los Angeles, CA, US, 2005.

[32] C.-Y. Chung and H. H. Chen, "Video Object Extraction via MRF-Based Contour Tracking," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 20, no. 1, pp. 149-155, 2010.

[33] T. Porter and T. Duff, "Compositing digital images," in *11th annual conference on Computer graphics and interactive techniques*, Minneapolis, MN, USA, 1984.

[34] Q. Yang, R. Yang, J. Davis and D. Nister, "Spatial-Depth Super Resolution for Range Images," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, USA, 2007.

[35] TeraBee, "Technology insights - Time of Flight principle," [Online]. Available: https://www.terabee.com/time-of-flight-principle/. [Accessed 2019-2020].

[36] R. A. Hamzah and H. Ibrahim, "Literature Survey on Stereo Vision Disparity Map Algorithms," *Journal of Sensors,* 2015.

[37] C. Strecha, W. v. Hansen, L. V. Gool, P. Fua and U. Thoennessen, "On benchmarking camera calibration and multi-view stereo for high resolution imagery," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, AK, 2008.

[38] K. Park, S. Kim and K. Sohn, "High-Precision Depth Estimation with the 3D LiDAR and Stereo Fusion," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, 2018.

[39] P. Vuylsteke and A. Oosterlinck, "Range image acquisition with a single binary-encoded light pattern," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 12, no. 2, pp. 148-164, 1990.

[40] E. Horn and N. Kiryati, "Toward optimal structured light patterns," *Image and Vision Computing,* vol. 17, no. 2, pp. 87-97, 1999.

[41] D. Scharstein and R. Szeliski, "High-accuracy stereo depth maps using structured light," in *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Madison, WI, USA, 2003.

[42] K. P. Keller, J. D. Ackerman, M. H. Rosenthal, H. Fuchs and A. State, "Methods and systems for real-time structured light depth extraction and endoscope using real-time structured light depth extraction". USA Patent US6503195B1, 24 5 1999.

[43] N. Silberman and R. Fergus, "Indoor scene segmentation using a structured light sensor," in *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, Barcelona, 2011.

[44] D. Scharstein and R. Szeliski, "A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms," *International Journal of Computer Vision,* vol. 47, pp. 7-42, 2002.

[45] P. Fua, "A parallel stereo algorithm that produces dense depth maps and preserves image features," *Machine Vision and Applications,* vol. 6, pp. 35-49, 1993.

[46] C. Häne, C. Zach, J. Lim, A. Ranganathan and M. Pollefeys, "Stereo depth map fusion for robot navigation," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Francisco, CA, 2011.

[47] H.-G. Jeon, J. Park, G. Choe, J. Park, Y. Bok, Y.-W. Tai and I. S. Kweon, "Accurate Depth Map Estimation From a Lenslet Light Field Camera," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, 2015.

[48] M. Subbarao and G. Surya, "Depth from Defocus: A Spatial Domain Approach," *International Journal of Computer Vision,* vol. 13, no. 3, pp. 271-294, 1994.

[49] C. Zhou, S. Lin and S. Nayar, "Coded aperture pairs for depth from defocus," in *2009 IEEE 12th International Conference on Computer Vision*, Kyoto, 2009.

[50] A. Horii, Depth from defocusing, Stockholm: Computational Vision and Active Perception Laboratory, 1992.

[51] M. W. Tao, S. Hadap, J. Malik and R. Ramamoorthi, "Depth from Combining Defocus and Correspondence Using Light-Field Cameras," in *IEEE International Conference on Computer Vision*, Sydney, 2013.

[52] M. W. Tao, P. P. Srinivasan, J. Malik, S. Rusinkiewicz and R. Ramamoorthi, "Depth From Shading, Defocus, and Correspondence Using Light-Field Angular Coherence," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, 2015.

[53] D. Ferstl, C. Reinbacher, R. Ranftl, M. Ruether and H. Bischof, "Image Guided Depth Upsampling using Anisotropic Total Generalized Variation," in *2013 IEEE International Conference on Computer Vision (ICCV)*, Sydney, 2013.

[54] J. T. Barron, A. Adams, Y. Shih and C. Hernandez, "Fast Bilateral-Space Stereo for Synthetic Defocus," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, 2015.

[55] J. Zhu, L. Wang, R. Yang and J. Davis, "Fusion of time-of-flight depth and stereo for high accuracy depth maps," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, 2008.

[56] D. Honegger, T. Sattler and M. Pollefeys, "Embedded real-time multi-baseline stereo," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, Singapore, 2017.

[57] Autodesk, "Maya - 3D computer animation, modeling, simulation, and rendering software," Autodesk, 2019. [Online]. Available: https://www.autodesk.ca/en/products/maya/overview?plc=MAYA&term=1-YEAR&support=ADVANCED&quantity=1. [Accessed 2019-2020].

[58] U. Capeto, "3D Stereoscopic Photography - Depth Map Automatic Generator 5 (DMAG5)," 24 5 2014. [Online]. Available: http://3dstereophoto.blogspot.com/p/software.html. [Accessed 2019-2020].

[59] U. Sara, M. Akter and M. S. Uddin, "Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study," *Journal of Computer and Communications,* vol. 7, no. 3, pp. 8-18, 2019.

[60] Z. Kotevski and P. Mitrevski, "Experimental Comparison of PSNR and SSIM Metrics for Video Quality Estimation," in *International Conference on ICT Innovations (ICT Innovations 2009)*, Berlin, Heidelberg, 2009.

[61] FLIR, "Grasshopper3 USB3," FLIR, [Online]. Available: https://www.flir.com/products/grasshopper3-usb3/. [Accessed April 2020].

[62] Autodesk, "Fusion 360," Autodesk, 2020. [Online]. Available: https://www.autodesk.ca/en/products/fusion-360/overview?mktvar002=3529781|SEM|1618310782|99147174467|kwd-330308867034&gclsrc=aw.ds&&ef_id=EAIaIQobChMI1Lbw-

P716gIVDfDACh2mCgVcEAAYASAAEgKEWPD_BwE:G:s&s_kwcid=AL!11172!3!420419553004!e!!g!!fusion%20360!161831. [Accessed April 2020].

[63] Ultimaker, "Ultimaker S5," Ultimaker, [Online]. Available: https://ultimaker.com/3d-printers/ultimaker-s5. [Accessed April 2020].

[64] B. W. Boehm, "A spiral model of software development and enhancement," *Computer,* vol. 21, no. 5, pp. 61-72, 1988.

[65] U. Capeto, "3D Stereoscopic Photography - Epipolar Rectification 9 (ER9)," 5 12 2015. [Online]. Available: http://3dstereophoto.blogspot.com/2015/12/epipolar-rectification-9-er9.html. [Accessed 2019-2020].

[66] K. Kitani, "Stereo Vision," Carnegie Mellon University, Pittsburgh.

[67] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 22, no. 11, pp. 1330-1334, 2000.

[68] A. Mordvintsev and K. Abid, "OpenCV-Python Tutorials - Camera Calibration and 3D Reconstruction," 2013. [Online]. Available: https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_calib3d/py_epipolar_geometry/py_epipolar_geometry.html. [Accessed 2019-2020].

[69] C. Loop and Z. Zhang, "Computing Rectifying Homographies for Stereo Vision," in *1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Fort Collins, CO, USA, 1999.

[70] D. Honegger, T. Sattler and M. Pollefeys, "Embedded real-time multi-baseline stereo," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, Singapore, 2017.

[71] A. Milella, G. Reina and M. M. Foglia, "A multi-baseline stereo system for scene segmentation in natural environments," in *2013 IEEE Conference on Technologies for Practical Robot Applications (TePRA)*, Woburn, MA, 2013.

[72] Y. Kang, C. Lee and Y. Ho, "An Efficient Rectification Algorithm for Multi-View Images in Parallel Camera Array," in *2008 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video*, Istanbul, 2008.

[73] S. L. Hill, *Scalable multi-view stereo camera array for real world realtime image capture and three-dimensional displays,* Cambridge, MA: Mass. Inst. Technol.,, 2004.

[74] F. Kangni and R. Laganiere, "Projective Rectification Of Image Triplets From The Fundamental Matrix," in *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, Toulouse, 2006.

[75] V. Nozick, "Multiple view image rectification," in *2011 1st International Symposium on Access Spaces (ISAS)*, Yokohama, 2011.

[76] U. Capeto, "3D Stereoscopic Photography - Epipolar Rectification 9b (ER9b)," 6 2 2016. [Online]. Available: http://3dstereophoto.blogspot.com/2016/02/epipolar-rectification-9b-er9b.html. [Accessed 2019-2020].

[77] A. Fusiello and L. Irsara, "Quasi-Euclidean uncalibrated epipolar rectification," in *2008 19th International Conference on Pattern Recognition*, Tampa, FL, 2008.

[78] L. Moisan, P. Moulon and P. Monasse, "Automatic Homographic Registration of a Pair of Images with A Contrario Elimination of Outliers," *Image Processing On Line,* vol. 2, pp. 56-73, 2012.

[79] H. Bay, T. Tuytelaars and L. V. Gool, "Surf: Speeded up robust features," in *9th European Conference on Computer Vision*, Graz, Austria, 2006.

[80] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," in *International Conference on Computer Vision Theory and Applications*, 2009.

[81] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision,* vol. 60, no. 2, pp. 91-110, 2004.

[82] H. Lu, J. Wang, A. C. Kot and Y. Q. Shi, "An objective distortion measure for binary document images based on human visual perception," in *Object recognition supported by user interaction for service robots*, Quebec City, 2002.