

NAIST-IS-DD0761012

Doctoral Dissertation

Multi-dimensional Image Completion by Minimizing Energy Based on Pattern Similarity

Norihiko Kawai

March 17, 2010

Department of Information Systems
Graduate School of Information Science
Nara Institute of Science and Technology

A Doctoral Dissertation
submitted to Graduate School of Information Science,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Doctor of ENGINEERING

Norihiko Kawai

Thesis Committee:

| | |
|---------------------------------------|-----------------|
| Professor Naokazu Yokoya | (Supervisor) |
| Professor Kunihiro Chihara | (Co-supervisor) |
| Associate Professor Kazumasa Yamazawa | (Co-supervisor) |
| Assistant Professor Tomokazu Sato | (Co-supervisor) |

Multi-dimensional Image Completion by Minimizing Energy Based on Pattern Similarity*

Norihiko Kawai

Abstract

With the spread of digital measurement devices for real scenes and objects such as digital cameras and rangefinders, it is becoming popular to use digital multi-dimensional images (still images, videos and range images) of real environments for various purposes. However, it is sometimes difficult to use original data because these images may contain undesirable objects or missing regions caused by occlusions between a measurement device and a target object. In order to increase the utility value of these data, this thesis proposes novel methods for image, video and 3D surface completion which remove undesirable objects and fill in missing regions in multi-dimensional images based on a unified framework of energy minimization using pattern similarity measures.

As for image and video completion, energy minimization methods based on pattern similarity have already been proposed and have obtained some good results. However, these conventional methods still have some problems. The existing image completion methods easily generate unnatural textures in missing regions due to two factors: (1) available samples in an image are quite limited; and (2) pattern similarity is one of the necessary conditions but is not sufficient for reproducing natural textures. This study extends the conventional energy function in order to solve such problems by considering brightness changes in sample textures to obtain effective samples (for (1)) and introducing spatial locality of texture patterns as a constraint that is usually satisfied in many of real

* Doctoral Dissertation, Department of Information Systems, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-DD0761012, March 17, 2010.

scenes (for (2)). The conventional video completion methods fill in missing regions based on the assumption that the appearance of objects does not change largely between different frames. It is difficult for them to plausibly fill in missing regions in omnidirectional video caused by a blind side of an omnidirectional camera. The proposed video completion method successfully completes missing regions in omnidirectional video by compensating for changes in the appearance of textures by considering the shape around the missing regions and extrinsic camera parameters.

As for 3D surface completion, there have been no existing methods which have treated the completion task as a global optimization problem. The conventional methods, which successively copy similar local shapes to missing regions, often generate inconsistent shapes in missing regions. In this study, missing regions are filled in by minimizing an energy function that is defined based on the similarity of local shapes. As a result, the proposed method can generate complex and consistent shapes in missing regions.

This thesis is organized as follows. Chapter 1 describes the introduction of the thesis and related works. Chapter 2 presents an image completion method for still images. Chapter 3 presents a video completion method for omnidirectional videos. Chapter 4 describes a 3D surface completion method. Finally, Chapter 5 summarizes the present study.

Keywords:

multi-dimensional image completion, inpainting, pattern similarity, energy minimization

パターン類似度に基づくエネルギー最小化による 多次元画像の欠損修復*

河合 紀彦

内容梗概

デジタルカメラやレンジファインダなどの計測機器の普及により、計測された現実環境の多次元画像を利用することが一般的に行われている。しかし、どのような計測機器を用いた場合においても、計測対象と計測機器の間の遮蔽物や計測機器の構造に起因して、得られるデータに不要な物体や欠損が存在するため、そのままの利用が難しい場合がある。本論文では、このような不要な物体や欠損を持つデータの利用価値を高めることを目的として、静止画像・動画像・三次元モデルにおける欠損領域を、パターン類似度を用いたエネルギー関数の最小化による統一した枠組みを用いて違和感なく修復する新たな手法を提案する。

静止画像、動画像の欠損修復に関しては、従来、パターン類似度に基づくエネルギー最小化を用いた手法は既に提案されている。しかし、従来手法には未だ問題が残されている。従来の静止画像に対する欠損修復手法は、以下に挙げる2つの問題により不自然なテクスチャが生じやすい。(1) データ領域におけるテクスチャパターンの種類に限りがあるため、違和感のない修復に最低限必要なパターンがデータ領域に存在しないことが多い。(2) 欠損領域とデータ領域間のパターン類似度のみによる指標は自然なテクスチャの再現条件としては不十分である。本研究では従来のエネルギー関数を拡張することでこれらの問題を解決する。具体的には、(1)の問題に対して、テクスチャの明度変化を許容したパターン類似度を用いることで表現できるテクスチャパターンの拡張を行う。(2)の問題に対して、類似したテクスチャは近傍に存在する確率が高いという性質(テクスチャの局所性)を表すコスト関数を用いることで不自然なテクスチャの生成を抑制する。従来の動画像に対する欠損修復手法は、フレーム間で同一物体の見え方が大

* 奈良先端科学技術大学院大学 情報科学研究科 情報システム学専攻 博士論文, NAIST-IS-DD0761012, 2010年3月17日.

きく変化しないという仮定のもと修復を行っている。従って、全方位カメラの死角により生じる全方位動画像の欠損を違和感なく修復することは困難である。本研究では、欠損領域の周辺の形状とカメラの位置・姿勢を考慮することでテクスチャの見え方の変化を補償し、全方位動画像の欠損領域を違和感なく修復する。

三次元モデルの欠損修復に関しては、未だ修復を全体最適化問題として扱う手法が存在しない。類似した局所形状を欠損領域に逐次的にコピーする従来手法では、欠損領域内において不連続な形状が生じやすい問題がある。本研究では、局所形状の類似度に基づくエネルギー関数を最小化することで欠損領域を修復する。これにより、複雑で不連続のない形状が欠損領域内に生成できる。

キーワード

多次元画像の欠損修復, パターン類似度, エネルギー最小化

Contents

| | | |
|----------|---|-----------|
| 1 | Introduction | 1 |
| 1.1. | Related Works in Multi-dimensional Image Completion | 2 |
| 1.1.1 | Image Completion | 2 |
| 1.1.2 | Video Completion | 11 |
| 1.1.3 | 3D Surface Completion | 15 |
| 1.2. | Positioning of this Study | 18 |
| 1.3. | Organization of this Thesis | 21 |
| 2 | Image Completion Considering Brightness Changes and Spatial Locality of Textures | 22 |
| 2.1. | Introduction | 22 |
| 2.2. | Overview of the Method | 23 |
| 2.3. | Definition of Energy Function Based on Pattern Similarity | 23 |
| 2.4. | Extension of Energy Function Considering Brightness Changes and Spatial Locality | 25 |
| 2.4.1 | Pattern Similarity Considering Brightness Changes | 26 |
| 2.4.2 | Cost Term for Spatial Locality | 27 |
| 2.5. | Energy Minimization | 28 |
| 2.5.1 | Searching for Similar Texture Pattern | 28 |
| 2.5.2 | Parallel Updating of Pixel Values | 30 |
| 2.5.3 | Coarse-to-fine Approach | 32 |
| 2.6. | Order of Computational Cost | 34 |
| 2.7. | Experiments | 36 |
| 2.7.1 | Qualitative Evaluation | 41 |
| 2.7.2 | Discussion of Resultant Images | 61 |

| | | |
|----------|---|------------|
| 2.7.3 | Quantitative Evaluation | 70 |
| 2.7.4 | Discussion about Computational Cost | 70 |
| 2.7.5 | Discussion about Local Minima Problem | 71 |
| 2.7.6 | Discussion about Evaluation Methods | 77 |
| 2.8. | Conclusion | 80 |
| 3 | Video Completion for Generating Omnidirectional Video without Invisible Areas | 81 |
| 3.1. | Introduction | 81 |
| 3.2. | Overview of the Method | 83 |
| 3.3. | Estimation of Extrinsic Camera Parameters and Positions of Natural Feature Points | 83 |
| 3.4. | Generation of Images Projected on Planes | 85 |
| 3.5. | Video Completion by Minimizing Energy Function | 88 |
| 3.5.1 | Definition of Energy Function Based on Pattern Similarity | 88 |
| 3.5.2 | Determination of Data Region | 90 |
| 3.5.3 | Energy Minimization | 93 |
| 3.5.4 | Coarse-to-fine Approach | 95 |
| 3.6. | Re-projection of Images Projected on Planes to Panoramic Images | 95 |
| 3.7. | Experiments | 96 |
| 3.7.1 | Acquisition of Input Information | 96 |
| 3.7.2 | Completion for Images Projected on Planes | 98 |
| 3.7.3 | Omnidirectional Telepresence without Invisible Areas . . . | 111 |
| 3.8. | Conclusion | 118 |
| 4 | Surface Completion By Minimizing Energy Function Based on Similarity of Local Shape | 119 |
| 4.1. | Introduction | 119 |
| 4.2. | Overview of the Method | 120 |
| 4.3. | Definition of Energy Function Based on Similarity of Local Shapes | 120 |
| 4.3.1 | Definition of Energy Function | 121 |
| 4.3.2 | Similarity of Local Shape | 122 |
| 4.4. | Energy Minimization | 125 |
| 4.4.1 | Searching for Similar Local Shape | 125 |

| | | |
|----------|--|------------|
| 4.4.2 | Parallel Updating of Positions of Vertices | 126 |
| 4.4.3 | Adding and Integrating Vertices | 128 |
| 4.4.4 | Coarse-to-fine Approach | 129 |
| 4.5. | Experiments | 131 |
| 4.5.1 | Qualitative Evaluation | 131 |
| 4.5.2 | Discussion about Influence of Parameters | 139 |
| 4.6. | Conclusion | 142 |
| 5 | Conclusion | 143 |
| 5.1. | Summary | 143 |
| 5.2. | Future work | 144 |
| | Acknowledgements | 146 |
| | References | 147 |
| | List of Publications | 155 |

List of Figures

| | | |
|------|---|----|
| 1.1 | Examples of target data in this thesis. | 3 |
| 1.2 | Example of image completion by Bertalmio et al. [BSCB00]. | 5 |
| 1.3 | Example of image completion by Amano et al. [AS07]. | 6 |
| 1.4 | Example of image completion by Criminisi et al. [CPT04]. | 8 |
| 1.5 | Example of image completion by Wexler et al. [WSI07]. | 10 |
| 1.6 | Example of video completion by Matsushita et al. [MOG ⁺ 06]. | 12 |
| 1.7 | Example of video completion by Yamashita et al. [YFKM08]. | 13 |
| 1.8 | Example of video completion by Jia et al. [JTWT06]. | 14 |
| 1.9 | Example of surface completion by Verdera et al. [VCBS03]. | 16 |
| 1.10 | Example of surface completion by Sagawa et al. [SI08] | 17 |
| 1.11 | Example of surface completion by Breckon et al. [BF08]. | 19 |
| | | |
| 2.1 | Flow diagram of the proposed image completion method. | 24 |
| 2.2 | Missing and data regions in an image. | 24 |
| 2.3 | Sigmoid function for cost of spatial locality. | 27 |
| 2.4 | Flow of searching for the most similar pattern and generating list. | 29 |
| 2.5 | Relationship between pixels in energy calculation. | 31 |
| 2.6 | Coarse-to-fine Approach. | 33 |
| 2.7 | Input images with missing regions | 38 |
| 2.8 | Web page for questionnaire evaluation. | 42 |
| 2.9 | Resultant images and scores | 44 |
| 2.10 | Experiment for Image 09. | 63 |
| 2.11 | Experiment for Image 62. | 64 |
| 2.12 | Experiment for Image 78. | 65 |
| 2.13 | Finally determined correspondences of pixels in Image 62. | 66 |
| 2.14 | Finally determined correspondences of pixels in Image 78. | 66 |

| | | |
|------|---|-----|
| 2.15 | Images for which Criminisi’s method obtained the best score. . . . | 69 |
| 2.16 | Images for which Wexler’s method obtained the best score. . . . | 69 |
| 2.17 | Change of energy and results of Image 04 with respect to different initial values. | 73 |
| 2.18 | Change of energy and results of Image 09 with respect to different initial values. | 74 |
| 2.19 | Change of energy and results of Image 62 with respect to different initial values. | 75 |
| 2.20 | Change of energy and results of Image 78 with respect to different initial values. | 76 |
| 2.21 | Relationships between ARE and the number of images and subjects. | 78 |
| 2.22 | Correlation coefficient between RMSE and subjective score. . . . | 79 |
| 2.23 | Images with inverse correlation between RMSE and subjective evaluations. | 80 |
| 3.1 | Omnidirectional panoramic image with a missing region (black region) caused by a blind side. | 82 |
| 3.2 | Flow diagram of the proposed video completion method for an omnidirectional video. | 84 |
| 3.3 | Selection of feature points around missing region. | 85 |
| 3.4 | Generation of an image projected on a plane. | 86 |
| 3.5 | Missing and data regions in projected images for completion process. | 89 |
| 3.6 | Projection to the other frame. | 92 |
| 3.7 | Projection to the other frame. | 93 |
| 3.8 | Omnidirectional multi-camera system ”Ladybug”. | 96 |
| 3.9 | 1st frame of input image sequence captured by six camera units (scene (1)). | 99 |
| 3.10 | 1st frame of input image sequence captured by six camera units (scene (2)). | 99 |
| 3.11 | Estimated positions and postures of an OMS and positions of natural feature points (scene (1)). | 100 |
| 3.12 | Estimated positions and postures of an OMS and positions of natural feature points (scene (2)). | 100 |
| 3.13 | Images projected on planes (scene (1)). | 102 |

| | | |
|------|---|-----|
| 3.14 | Images projected on planes (scene (2)). | 103 |
| 3.15 | Comparison of results by projection using a plane and proposed method. | 104 |
| 3.16 | Comparison of results by projection using a plane and proposed method. | 105 |
| 3.17 | Successive frames with a missing region (scene (1)). | 106 |
| 3.18 | Successive completed projected images (scene (1)). | 107 |
| 3.19 | Successive frames with a missing region (scene (2)). | 108 |
| 3.20 | Successive completed projected images (scene (2)). | 109 |
| 3.21 | Example of failure results. From 298th frame to 301st frame. | 110 |
| 3.22 | Filled panoramic image of 11th frame of scene (1). | 112 |
| 3.23 | Filled panoramic image of 71st frame of scene (2). | 113 |
| 3.24 | Example of looking around using omnidirectional video (11th frame of scene (1)). | 114 |
| 3.25 | Example of looking around using omnidirectional video (101st frame of scene (1)). | 115 |
| 3.26 | Example of looking around using omnidirectional video (71st frame of scene (2)). | 116 |
| 3.27 | Example of looking around using omnidirectional video (101st frame of scene (2)). | 117 |
| 4.1 | Flow diagram of the proposed method. | 121 |
| 4.2 | Missing and data regions in a 3D model. | 122 |
| 4.3 | Alignment of vertex clouds and surface. | 123 |
| 4.4 | Relationship between vertices in energy calculation. | 127 |
| 4.5 | Conversion of parameters. | 128 |
| 4.6 | Addition and integration of vertices. | 129 |
| 4.7 | Surface completion for Model (I). | 134 |
| 4.8 | Surface completion for Model (II). | 135 |
| 4.9 | Changes in shape with iterations for Model (II). | 136 |
| 4.10 | Surface completion for Model (III). | 137 |
| 4.11 | Changes in energy with respect to iterations for each model. | 138 |
| 4.12 | Results of Model (I) with different values of parameter s in weight w | 140 |

| | |
|---|-----|
| 4.13 Results of Model (II) with different values of parameter s in weight | |
| w . | 141 |

List of Tables

| | | |
|-----|---|-----|
| 2.1 | Parameters in image completion experiments. | 37 |
| 2.2 | Average score from 100 images. | 43 |
| 2.3 | The number of times the method obtained the highest score. . . . | 43 |
| 2.4 | T-value produced by the t-test between a pair of methods with a 5% significant level. (If significant difference is observed, the t-value is underlined.) | 43 |
| 2.5 | Average RMSE from 29 images. | 70 |
| 2.6 | Comparison with respect to computational cost. | 71 |
| 3.1 | Parameters in experiments. | 97 |
| 4.1 | Parameters in the proposed method in experiments (l_{ave} indicates average length between vertices in data region). | 132 |
| 4.2 | Processing time with different values of parameter s for each model. | 139 |

Chapter 1

Introduction

With the spread of digital measurement devices for real scenes and objects such as digital cameras and rangefinders, it is becoming popular to use digital multi-dimensional images (still images, videos and range images) of real environments in daily life. For example, digital images and video captured with an ordinary monocular camera are used in web sites and magazines for entertainment and information. In addition, since omnidirectional cameras, which can simultaneously capture almost all directions, have been developed recently, omnidirectional images and video are also used in web sites such as Google Street View [Goo] and amusement attractions for telepresence [Min80], which allow users to feel as if they exist in a remote site by making them able to look around in a remote scene. In addition, 3D models generated from range images captured with a rangefinder are used for digital museums and 3D maps.

However, it is sometimes difficult to use the original data captured with digital measurement devices because it may contain undesirable objects or missing regions caused by occlusions between the measurement devices and target objects. In addition, especially as for omnidirectional images and video, since ordinary omnidirectional cameras cannot capture all directions due to the structure of the camera, missing regions appear in omnidirectional images and video in the blind side of the camera. Even if there is a camera which can capture completely all directions entirely, missing regions appear due to occlusions by the persons or vehicles on which the camera is mounted. In order to increase the utility value of these data, this thesis proposes novel methods for image, video and 3D sur-

face completion which remove undesirable objects and fill in missing regions in multi-dimensional images based on a unified framework of energy minimization using pattern similarity measures. In this thesis, as target data, we pay attention to three kinds of data: still images captured with a monocular camera, omnidirectional video captured with an omnidirectional camera, and 3D mesh models composed of vertices and faces as shown in Figure 1.1.

Until now, various methods for image, video and 3D surface completion have been proposed to plausibly fill in missing regions in images, video and 3D mesh models. In this chapter, we first review conventional methods for image, video and 3D surface completion, respectively. Then, we describe the positioning of this study against the conventional methods. Finally, we give the organization of this thesis.

1.1. Related Works in Multi-dimensional Image Completion

1.1.1 Image Completion

First, this section describes image completion methods for still images. Existing image completion methods can be classified into two categories:

- Method using information around missing regions
- Method using the similarity of textures

Although these methods have been investigated by many researchers, they still present some problems. In the following, we describe the characteristics and problems of the methods in sequence.

Image completion method using information around missing regions

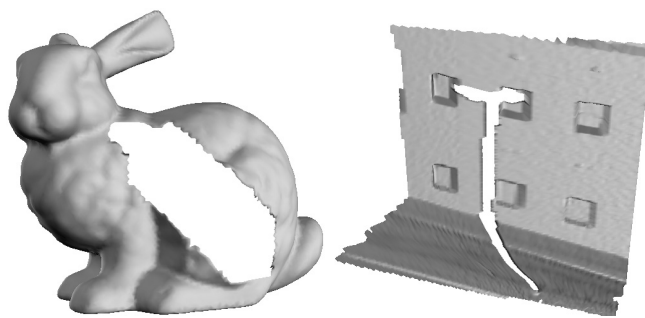
The methods using the information around missing regions complete the missing regions by calculating pixel values considering the continuity of pixel intensity from the boundary of missing regions assuming that neighbor pixels have similar pixel values [MTO86, MM98, BSCB00, BBS01, BBSV01, BCV⁺01, BBMN04,



Still image with undesirable objects captured by a monocular camera



Omnidirectional video with a blind side captured by an omnidirectional camera



3D mesh model with holes composed of vertices and faces

Figure 1.1. Examples of target data in this thesis.

Tsc06, CS01a, CS01b, CKSO02, ES03, LZW03, VABF04, YOT05, OHK05] . In addition, these methods pay attention to preserving edges when missing regions are filled in.

Maeda et al. [MTO86] have proposed a method by which an edge is extended into a missing region preserving the direction of it when the edge exists on the boundary of the missing region, otherwise a target pixel value is made the average of pixel values around the target pixel. In this method, edges may not connect in the middle of the missing region when the missing region is large because the pixel values are successively decided from the boundary of the missing region. Masnou et al. [MM98] linearly connect edges in a missing region by determining beforehand pairs of edges which reach the boundary. It is difficult for the method to make pairs of edges when the edges of a boundary are complex. In addition, rounded edges cannot be generated even when edges around the missing regions curve.

For these methods, the following measures have been used in order to connect edges smoothly:

- Partial Differential Equation [BSCB00, BBS01, BBSV01, BCV⁺01, BBMN04, Tsc06]
- Total Variation [CS01a, CS01b]
- Euler's elastica [CKSO02]
- Mumford-Shah-Euler [ES03]
- Markov Random Field [YOT05, OHK05]
- Probabilistic model [LZW03, VABF04]

These methods can produce the continuous intensity of pixel values and smooth edges and are effective for small image gaps like scratches in a photograph as shown in Figure 1.2. However, the resultant images easily become unclear when the missing regions are large because the methods cannot generate complex textures in principle.

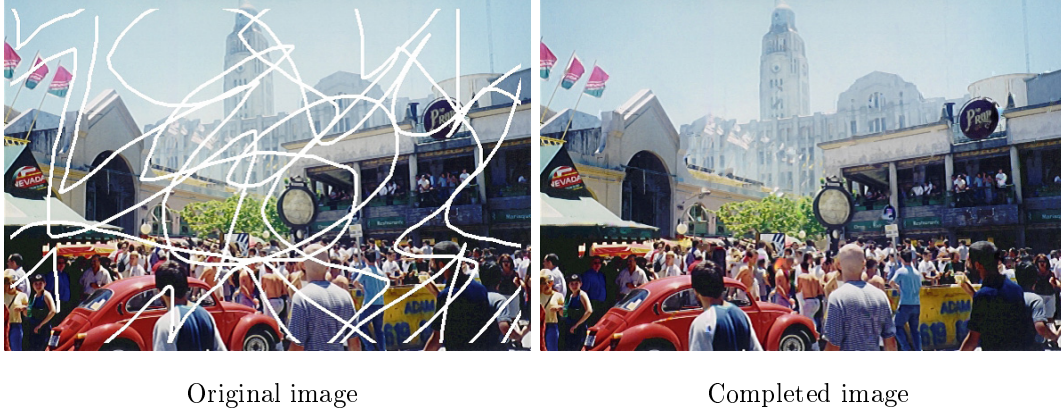


Figure 1.2. Example of image completion by Bertalmio et al. [BSCB00].

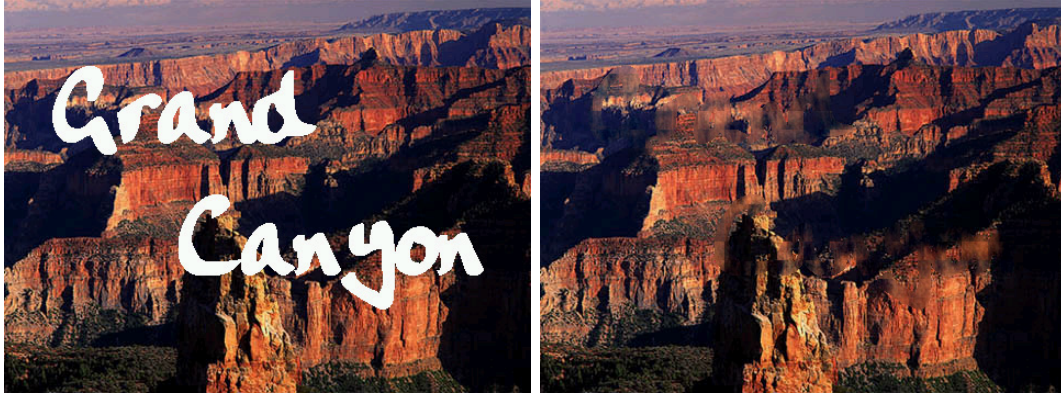
Image completion method using the similarity of textures

Image completion methods using the similarity between missing regions and the rest of the image are classified into two types with respect to the similarity measure. One method uses feature spaces transformed from textures in an image and the other uses original textures in an image.

- **Image completion method using feature space**

Image completion methods using feature space calculate features in a certain range of area including the missing region and fill in the missing region using feature vectors generated from the rest of the image, which are similar to the calculated features around the missing region. As feature space, Fourier space, wavelet domain and eigenspace have been used.

As for methods using Fourier space, Shoji [Sho88] has used Fourier vectors in a data region based on the assumption that the position invariance of Fourier amplitude spectrum in textured images. Hirani et al. [HT96] have proposed a method which completes missing regions using both frequency domain and spatial domain. Rane et al. [RRS96] have reconstructed a missing region in the wavelet-domain using the correlation between the missing region and its neighboring areas. These methods are effective for images with cyclical texture patterns. However, it is difficult for them



Original image

Completed image

Figure 1.3. Example of image completion by Amano et al. [AS07].

to obtain good results for images with non-cyclical texture patterns. In addition, a user is required to manually specify regions used for completion in method [HT96].

Against these methods, eigenvectors have been used from the idea that features specific to each image are more effective for image completion than general frequency domain. Amano et al. [AS07] have proposed a BPLP (Back Projection for Lost Pixels) method using eigenvectors. This method interpolates missing regions by combining eigenvectors which are extracted from a data region as learning samples. This method can generate complex textures in missing regions for images with an autocorrelation property as shown in Figure 1.3. The effectiveness of this is demonstrated by comparing results using eigenvectors and Fourier vector in [AS07]. Izoë et al. [IKK02] have proposed a method using eigenvectors by paying attention to fractal and autocorrelation properties. This method also can produce complex textures because the eigenvectors are made from textures in the data region.

As described earlier, the methods using feature space can generate complex textures in missing regions. However, these methods basically require the strong constraints, such as the property of autocorrelation for images. In addition, the size of missing regions to which the methods can be applied is

quite limited because a patch has to include both a missing region and textures around it when features around the missing region are calculated. As a result, it is difficult for these methods to complete images with relatively large missing regions.

- **Image completion method using original textures as exemplars**

The methods using textures as exemplars complete the missing regions in an image based on the assumption that textures appropriate for missing regions are similar to those in a data region. Such an approach has been intensively developed because it can synthesize complex textures in missing regions. The methods in this approach can be classified into two categories. One is based on successive texture copy and the other on global optimization. These methods are hereafter described in more detail.

- **Method based on successive texture copy**

The method based on successive texture copy searches a data region for textures similar to those on the boundary of the missing region and copies the most similar textures successively. This method can generate detailed textures in a large missing region as shown in Figure 1.4. This approach is based on a texture synthesis technique, which generates large texture fields from a small patch of texture. The application of the texture synthesis technique to image completion was originated by Efros et al. [EL99]. Bertalmio et al. [BVSO03] have proposed a method combining the method considering the continuity of intensity [BSCB00] and texture synthesis [EL99]. In these methods, a patch of texture is successively copied to the boundary of the missing region and the pixel value is fixed once it is copied. From this characteristic, the processing cost is small but discontinuous texture is easily generated because the quality of a completed image largely depends on the order of copy. For this problem, in order to make more plausible textures, the order of texture copy has been determined according to the following criteria:

- * Similarity of texture [Har01]
- * Inverse matte [DCOY03]

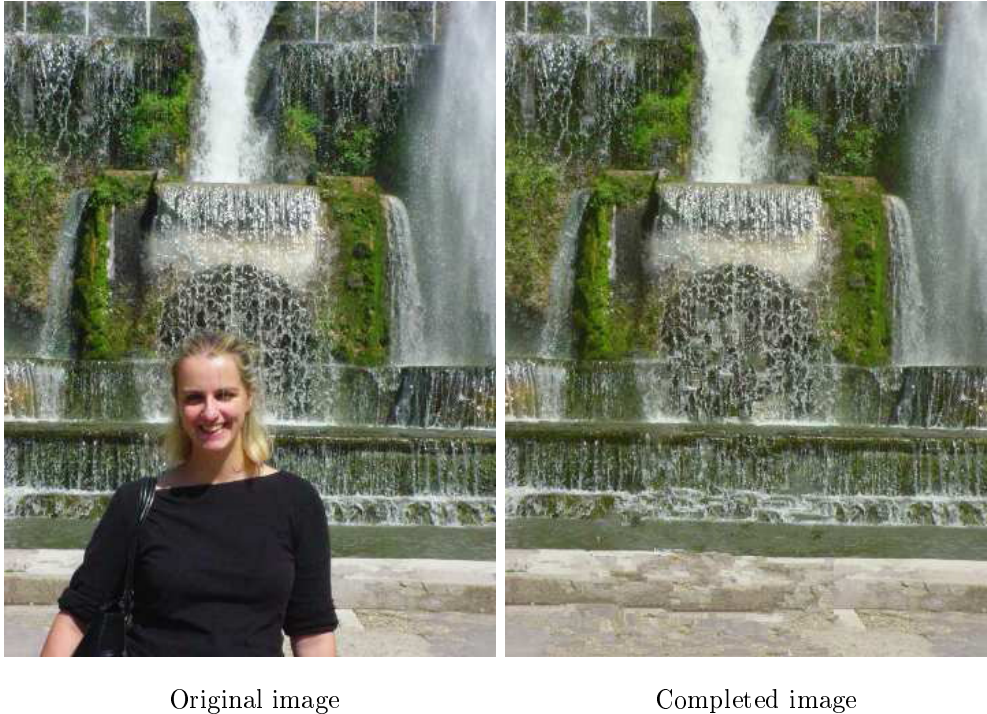


Figure 1.4. Example of image completion by Criminisi et al. [CPT04].

- * Number of fixed pixels in a patch [BLLC02]
- * Number of fixed pixels in a patch and strength of an edge [CPT04]
- * Number of fixed pixels in a patch, strength of an edge and similarity of texture [LQS05]

Nevertheless, these methods still have the problem that a discontinuous texture tends to be generated for images which have complex texture patterns around missing regions.

As another approach to decide the order of texture copy, the area of edges are specified beforehand manually or automatically and textures are preferentially copied to the specified area [SYJS05, JT03]. However, it is difficult to correctly specify effective edges when textures around missing regions are complex.

– **Method based on global optimization**

The method based on successive texture copy has the problem that dis-

continuous textures are easily generated. In order to settle this, methods which complete missing regions by optimizing an objective function for whole the missing region have been proposed [KT06, AP06, WSI07]. Komodakis et al. [KT06] have proposed a method which determines the appropriate order of texture synthesis by optimizing an objective function based on Priority-BP extended from Belief Propagation and labeling pixels in missing regions. Allene et al. [AP06] have completed missing regions by splitting the missing regions and determining the optimum combination of texture patches in a data region using a graph cut approach. Wexler et al. [WSI07] have proposed a method which completes missing regions by defining an objective function based on pattern similarity between the missing and data regions and optimizing the function. This method is usually used for video completion and also can be used for image completion by not considering the temporal similarity in the energy function. These methods based on global optimization do not depend on the order of copy and tend to generate plausible textures for whole the missing region as shown in Figure 1.5. Although the methods based on global optimization have obtained better results for many images than the methods based on successive copy, unnatural images are still generated due to two factors. One is that available samples in a data region are quite limited. A discontinuous change in brightness often appears in missing regions because textures with the same patterns but different brightnesses cannot be used for completion. The other is that the similarity of all the textures in an image are treated under the same conditions. Due to these factors textures are often blurred because of inappropriate correspondence between textures in the missing region and data region.

As described earlier, image completion methods using textures as exemplars can generate detailed textures for large missing regions and obtain good results for some images. However, there are still some problems with this because available texture patterns are limited and the conditions of similarity are not sufficient.



Original image



Completed image

Figure 1.5. Example of image completion by Wexler et al. [WSI07].

1.1.2 Video Completion

In this section, video completion methods are reviewed. For video completion, the image completion methods for still images described in Section 1.1.1 can be applied to video because video consists of sequential still images. However, textures may discontinuously change between successive frames because the methods use only spatial information in each frame.

Therefore, video completion methods usually use not only spatial information but also temporal information. Video completion methods are classified into two categories:

- Method using information around missing regions in spatio-temporal video images
- Method using the similarity of textures

These conventional completion methods have basically been proposed for video captured with an monocular camera. In the following, we describe the characteristics and problems of the methods in sequence.

Video completion method using information around missing regions

The methods using the information around missing regions complete the missing regions by calculating pixel values considering the spatial and temporal continuity of pixel intensity from the boundary of missing regions. These methods can be classified into two categories. One method does not employ the motion information [OM04] and the other method uses the motion of objects or a camera [LKK03, MOG⁺06, SLCF06, YFKM08].

The first method [OM04] fills in missing regions using partial differential equation (PDE) to generate smooth intensity among the spatial and temporal domain. This method cannot generate complex textures in missing regions in principle. On the other hand, methods that use motion information complete missing regions based on the correspondences between pixels in a target frame and neighbor frames. Pixel correspondences are determined based on the estimated motion of objects or a camera. These methods can be classified into two types. One uses



Input image sequence



Completed image sequence

Figure 1.6. Example of video completion by Matsushita et al. [MOG⁺06].

the motion of objects on an image plane [MOG⁺06, LKK03] and the other uses the posture of a camera [SLCF06, YFKM08].

As for methods using the motion of objects on an image plane, Litvin et al. [LKK03] and Matsushita et al. [MOG⁺06] have proposed methods that estimate the motion of textures in missing regions using optical flows for the whole image and pixel values are copied to the missing regions from different frames based on the estimated motion as shown in Figure 1.6. However, it is difficult for these methods to determine appropriate optical flows in an omnidirectional video because the appearance of textures in the omnidirectional video largely changes between successive frames and the motion of pixels in the large missing regions cannot be accurately estimated by 2D interpolation.

As for methods using the motion of a camera, Shen et al. [SLCF06] and Yamashita et al. [YFKM08] have proposed methods which complete missing regions using a fixed-viewpoint pan-tilt camera. Because this method can calculate the angle of pan-tilt, the missing regions are successfully completed using the same direction of lighting in neighbor frames as shown in Figure 1.7. However, these methods cannot be applied to video captured with a freely-moving camera



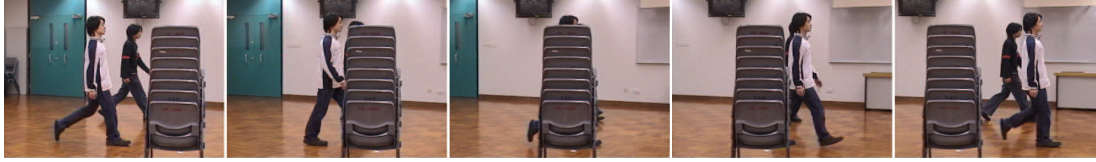
Figure 1.7. Example of video completion by Yamashita et al. [YFKM08].

since the fixed viewpoint is the indispensable condition to make correspondence between pixels in different frames.

Video completion method using the similarity of textures

The methods using the similarity between missing regions and the rest of the video complete missing regions by copying the similar textures or optimizing an objective function based on the similarity measure. These methods are also classified into two categories. One method does not use the motion information [CFJ08, WSI07] and the other method uses the motion of objects [ZXS05, JTWT06, PSB07].

These methods [CFJ08, WSI07] define the similarity of local volumes between the missing regions and the rest of the video and fill in the missing regions by



Input image sequence



Completed image sequence

Figure 1.8. Example of video completion by Jia et al. [JTWT06].

copying the similar local volumes or optimizing a similarity-based objective function. These methods can generate continuous textures through successive frames. However, real scenes behind occlusions are not always used as exemplars even if the scene behind the occlusions appears in different frames due to the movement of objects and the camera since any local volumes in a whole video may be used as exemplars. As a result, the inappropriate textures which actually do not exist behind occlusions sometimes appear in the missing regions. In addition, the methods do not consider the change in the appearance of textures caused by camera motion. Because the appearance of texture appropriate for a missing region in a frame changes in successive frames of a moving omnidirectional camera, it is difficult for them to successfully complete missing regions in an omnidirectional video.

On the other hand, methods that use the motion model of objects have been proposed [ZXS05, JTWT06, PSB07]. In these methods, a given video is divided into foreground (moving objects) and background layers. The background layer is completed using 2D optical flow in the similar way to the methods [MOG⁺06, LKK03] described before. In the foreground layer, in order to treat relatively complicated motion in the missing region, the motion of objects are modeled

based on the similarity of the motion between missing regions and the rest of the video and the moving objects are reproduced based on the motion model as shown in Figure 1.8. However, in the conventional methods, the motion model is designed for 2D cyclic motion and these methods cannot treat complex 3D motion of the target in an omnidirectional video.

1.1.3 3D Surface Completion

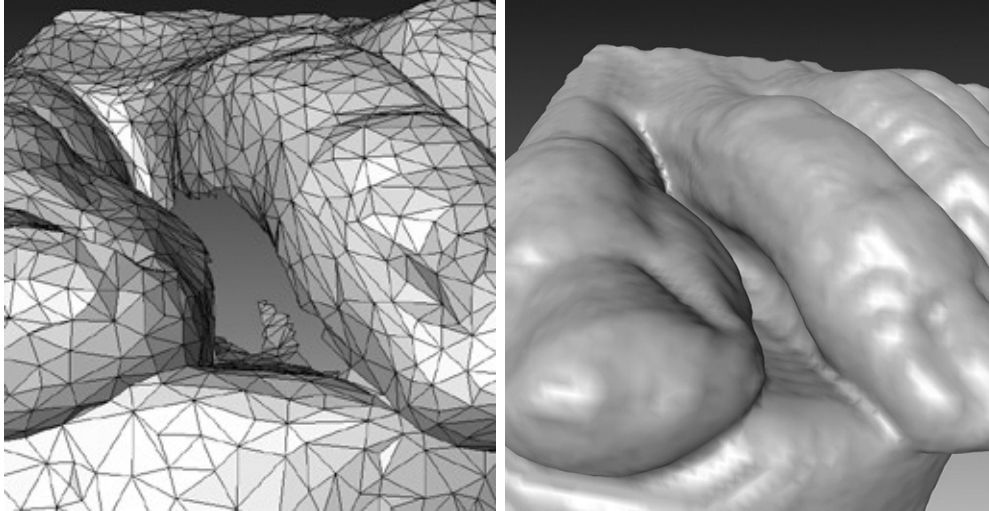
In this section, 3D surface completion methods for 3D mesh models generated from range images and video are reviewed. As the most basic method, the method which generates surfaces in a missing region by connecting vertices on the boundary of the missing region has been used. However, surfaces as a whole become unnatural when a large missing region is filled in because the method can generate only a flat surface. Therefore, methods using the information in the 3D model have been proposed. These surface completion methods are classified into two categories:

- Method using information around missing regions
- Method using the similarity of shapes

Surface completion method using information around missing regions

Castellani et al. [CLF02] have proposed a method which linearly extends edges from the boundary to a missing region. This method can generate edges and corners but not smooth curves. Against this problem, methods using partial differential equation, moving least squares and Willmore flow have been proposed [VCBS03, WO03, XMQ04]. These methods can fill in the missing regions with smooth surface patches as shown in Figure 1.9. Although they are effective for small holes in a 3D model, unnatural shapes may be generated when the missing regions are large and the surrounding shapes are complex because the methods cannot generate a complex surface.

As another approach using the information around missing regions, volumetric-based methods have been proposed [CL96, DMGL02, Mas04, FIMK07, SI08]. Curless et al. [CL96] have proposed a method which classifies each voxel into



Input model with a hole

Completed model

Figure 1.9. Example of surface completion by Verdera et al. [VCBS03].

'Unseen', 'Nearsurface' and 'Empty' and generates meshes between 'Unseen' and 'Empty'. Davis et al. [DMGL02] have filled holes by diffusing a signed distance function. Masuda [Mas04] has successively fitted a quadratic surface on a signed distance function and generate smooth surfaces in missing regions. Results by these methods largely depend on the location and the number of measuring points. Thus, it is difficult to obtain good results when the location is grossly one-sided and the number is small. For such problems, Furukawa et al. [FIMK07] have determined whether a voxel is inside or outside an object using Bays estimation and Sagawa et al. [SI08] have flipped the sign of the signed distance of the voxel considering the relationships among adjacent voxels. These methods have solved the problem that results largely depend on the location and the number of measuring points and have generated smooth surfaces in the missing regions as shown in Figure 1.10. However, it is difficult for them to generate complex shapes in the missing regions because these methods also use only the information around the missing regions.

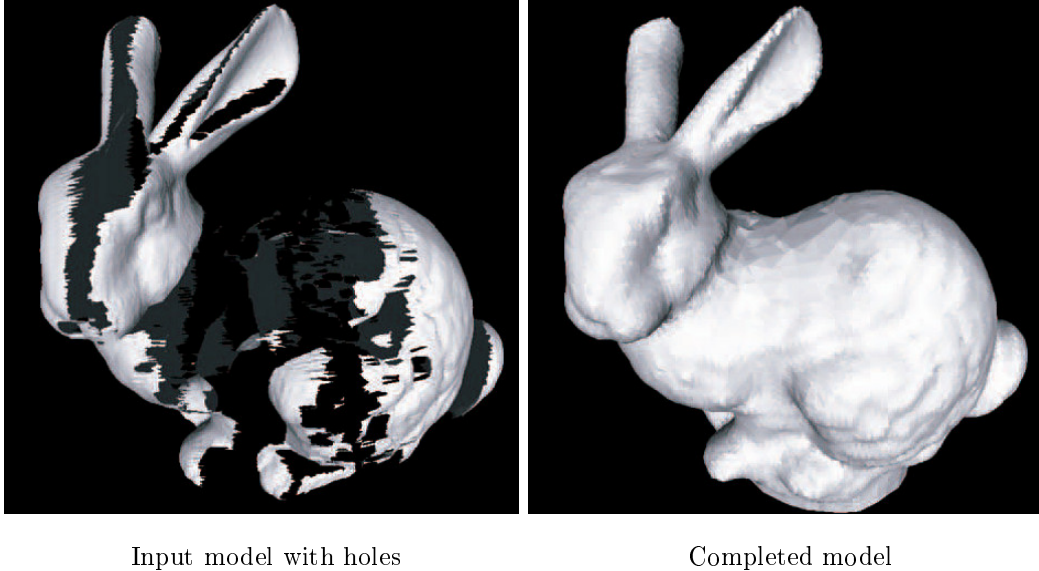


Figure 1.10. Example of surface completion by Sagawa et al. [SI08]

Surface completion method using the similarity of shapes

Methods using shapes in all regions in a 3D model (data region) or shapes in a model similar to the target model in a database have been proposed [KS05, Pau05, PMW⁺08, PGSQ06, SACO04, BF08]. These methods can generate complex shapes in missing regions because abundant shapes are used as exemplars for completion. Kraevoy et al. [KS05] and Pauly et al. [Pau05] have proposed methods that fill in missing regions using a similar 3D model in a database. Concretely, the similar model is deformed so that the model fits the target model and the surface in the similar model corresponding to the missing region in the target model is copied to the missing region. In these methods, a similar model is required to be modeled beforehand because these methods need a similar model to the target model. Therefore, there is the problem that it requires inordinate man hours to assemble such a database.

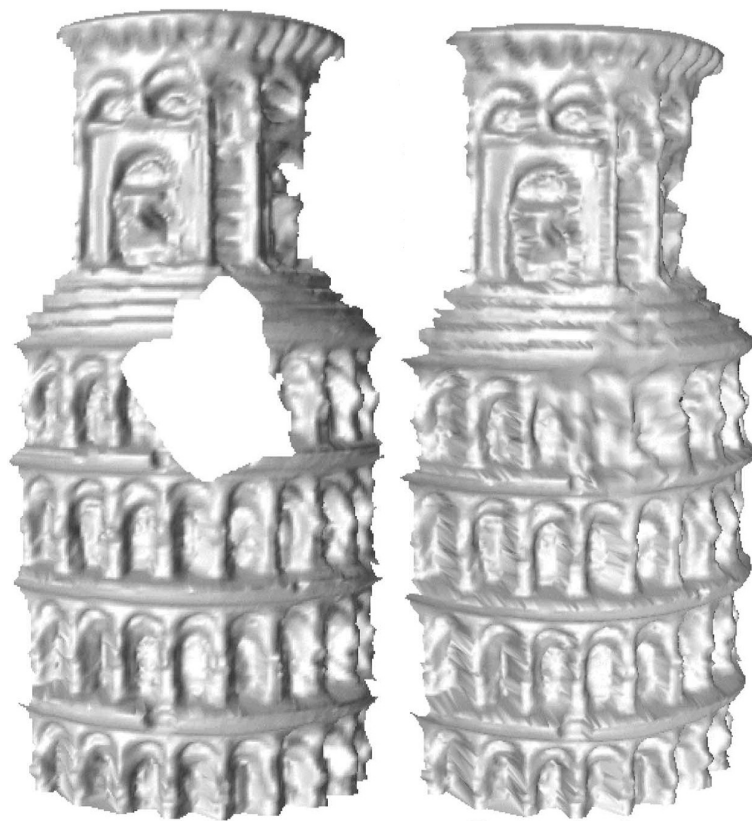
On the other hand, methods using not a special database but example shapes in a data region in the target object have been proposed [PMW⁺08, SACO04, BF08, PGSQ06]. These methods can be classified into the method which es-

estimates shapes by modeling the structure around a missing region [PMW⁺08] and the methods which copy the similar surface to a missing region by using surfaces in a data region as exemplars [SACO04, BF08, PGSQ06]. The former method [PMW⁺08] estimates the shape in missing regions by modeling the element shape and the cycle on the assumption that the target object has a cyclic structure around the missing regions. Therefore, this method is effective for the model such as a cluster housing, which has a cyclic structure. However, it is difficult for the method to successfully complete missing regions in a model without cyclic structures because the method is based on the condition that cyclic structures are automatically detected. The latter methods [SACO04, BF08, PGSQ06] calculate the similarity between the local surface shape around the missing regions and that in the data region and fill missing regions by copying the most similar surface patches successively. These methods can generate plausible and complex shapes for the model which does not have cyclic structures using many kinds of surfaces in whole the data region as shown in Figure 1.11. However, an inconsistent surface is sometimes generated on the seam in the completed model because the successive copy makes the optimum shape in the local area but does not always make the optimum surface as a whole.

1.2. Positioning of this Study

In the previous sections, conventional works in completion for multi-dimensional images (still images, videos, 3D mesh models from range images) have been reviewed and we have described that there are still some problems in the conventional completion methods. In this thesis, we propose novel completion methods for multi-dimensional images based on a unified framework of energy minimization using pattern similarity measures considering the remaining problems.

As for image completion, although the methods based on global optimization have obtained good results for many images, unnatural images, especially those with unnatural brightness changes and undesirable blurs, are still generated due to two factors: (1) available samples in a data region are quite limited, and (2) pattern similarity is one of the necessary conditions but is not sufficient for reproducing natural textures. In order to improve the image quality, these two factors



Input model with a hole

Completed model

Figure 1.11. Example of surface completion by Breckon et al. [BF08].

should be considered. Until now, there have already been some attempts at this. For (1), the scale and orientation of textures have been considered to obtain effective samples [DCOY03]. As for (2), Sun et al. [SYJS05] and Jia et al. [JT03] have proposed techniques that use explicit constraints for texture boundaries. These methods synthesize textures preserving the edges of textures by specifying them beforehand in the missing region either manually or automatically. However, it is difficult to determine automatic and effective explicit constraints when textures around missing regions are complex. In order to settle such problems, we employ a new approach different from the heretofore mentioned conventional ones. This study extends the conventional energy function [WSI07] considering brightness changes in sample textures to obtain effective samples (for (1)) and introducing spatial locality of texture patterns as a constraint that is usually satisfied in many of real scenes (for (2)).

As for video completion, conventional methods have successfully completed missing regions in video captured with a monocular camera. However, in order to complete a blind side in omnidirectional video in which the appearance of textures largely changes between different frames, consideration of the appearance change is an essential factor. As mentioned earlier, the change in appearance of textures by a freely-moving camera has not been considered in conventional methods. In addition, 2D optical flows that are used in the conventional methods cannot correctly estimate the positions of corresponding pixels in omnidirectional vision because a blind side in omnidirectional video is relatively large and the motion of corresponding pixels in omnidirectional vision is quite complex due to a characteristic of hard distortion. In order to resolve the problems, the proposed method utilizes the 3D information from the motion of a camera and the structure around a blind side which are then simultaneously estimated by a structure-from-motion technique for omnidirectional video. From the 3D information, appropriate appearance of textures for completion can be generated by compensating for the change of the appearance of textures. In addition, the relatively accurate correspondence of pixels between quite different frames can be determined by considering the projection model of the omnidirectional camera.

As for 3D surface completion, the conventional methods based on successive copy of local shapes can make complex surfaces. However, a discontinuous surface

is sometimes generated. In this thesis, in order to solve the problem, missing regions are filled in by minimizing an energy function that is defined based on the similarity of local shapes between missing and data regions. As a result, the proposed surface completion method can generate complex and consistent shapes in missing regions as an optimal solution.

1.3. Organization of this Thesis

The rest of this thesis is organized as follows. Chapter 2 proposes an image completion method considering brightness changes and spatial locality of textures by extending a conventional energy function. In experiments, 100 images with missing regions are completed and the resultant images are qualitatively and quantitatively evaluated. Chapter 3 proposes a video completion method for omnidirectional video using the shape around a missing region and extrinsic camera parameters. In experiments, omnidirectional videos with missing regions are completed and the resultant videos are qualitatively evaluated. Chapter 4 proposes a 3D surface completion method based on similarity of local shapes. Experiments are performed using three models with a missing region and the resultant models are qualitatively evaluated by comparing those by a conventional method. Finally, Chapter 5 summarizes the present study.

Chapter 2

Image Completion Considering Brightness Changes and Spatial Locality of Textures

2.1. Introduction

This chapter describes an image completion method for still images. At present, digital images are widely used in web pages and magazines for entertainment and information. However, original images can sometimes not be used since the images often contain undesired objects. In order to increase the utility value of such images, it is required to remove undesirable parts in an image and plausibly fill in the missing regions. For this task, recently, the methods based on global optimization using the similarity of textures [KT06, AP06, WSI07] have been intensively developed because the methods can generate consistent and complex textures in missing regions. However, unnatural images, especially those with unnatural brightness changes and undesirable blurs, are still generated due to two factors: (1) available samples in a data region are quite limited; and (2) pattern similarity is one of the necessary conditions but is not sufficient for reproducing natural textures.

In the proposed method, in order to obtain good results for many images, the conventional method [WSI07], which is based on energy minimization using pattern similarity, is extended considering brightness changes of sample textures

to obtain effective samples (for (1)) and introducing spatial locality of texture patterns as a constraint that is usually satisfied in many of real scenes (for (2)).

In the following sections, first, the overview of the proposed image completion method is given in Section 2.2. After introducing the conventional energy function [WSI07] which is defined based on the similarity of textures in Section 2.3, an energy function considering brightness changes and spatial locality is newly defined by extending the conventional energy function in Section 2.4. In Section 2.5, the energy minimization method is described. The order of the computational cost is discussed in Section 2.6. Next, experiments using 100 images are performed in Section 2.7. Finally, Section 2.8 concludes this chapter.

2.2. Overview of the Method

Figure 2.1 shows the flow of the proposed method. First, a user manually specifies regions to be completed such as physically damaged regions and undesired object regions in an image. Next, initial pixel values are given to the missing regions. Finally, overall the missing regions are optimally completed by repeating two processes: (i) searching for a similar texture and (ii) parallel updating of all pixel values. In the following sections, the energy function based on pattern similarity, its extension and the energy minimization method are described in detail.

2.3. Definition of Energy Function Based on Pattern Similarity

As illustrated in Figure 2.2, first, an image is divided into region Ω' , including missing region Ω specified by a user, and data region Φ , which is the rest of the image. The plausibility in region Ω' , including missing region Ω , is defined using image patterns in data region Φ . Here, Ω' is the expanded area of missing region Ω in which there is a central pixel, \mathbf{x}_i , of square window W overlapping region Ω . Conventional energy function E_{conv} that represents the implausibility in the missing region is defined as the weighted sum of SSD (Sum of Squared Differences) between pixels around pixel \mathbf{x} in region Ω' and those around pixel $\hat{\mathbf{x}}_i$

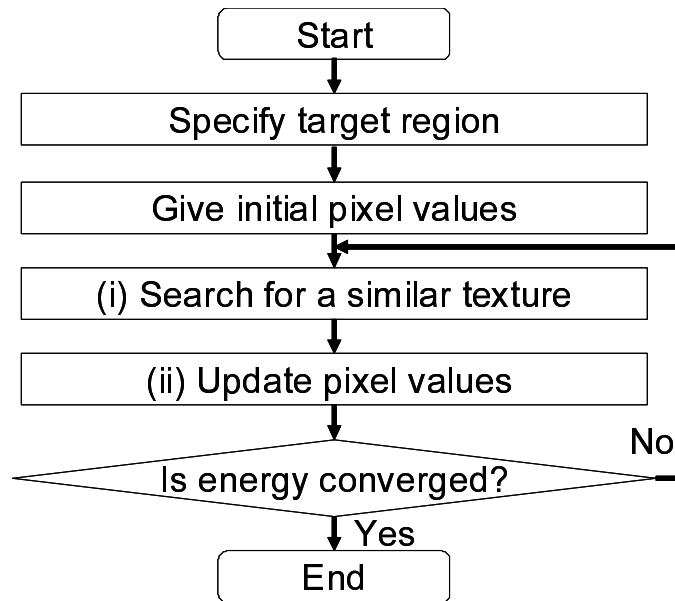


Figure 2.1. Flow diagram of the proposed image completion method.

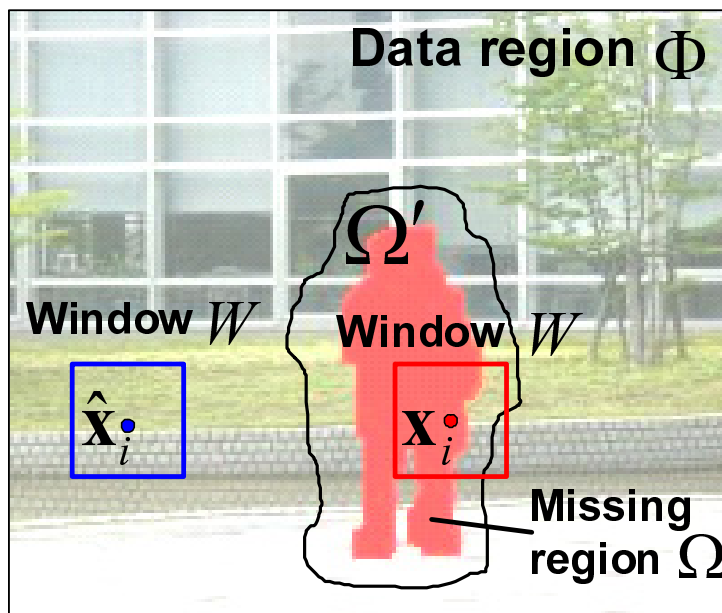


Figure 2.2. Missing and data regions in an image.

in region Φ as follows:

$$E_{conv} = \sum_{\mathbf{x}_i \in \Omega'} w_{\mathbf{x}_i} SSD(\mathbf{x}_i, \hat{\mathbf{x}}_i), \quad (2.1)$$

where $\hat{\mathbf{x}}_i$ in data region Φ denotes the pixel around which the texture pattern is the most similar to that around \mathbf{x}_i in region Ω' , and $SSD(\mathbf{x}_i, \hat{\mathbf{x}}_i)$ is defined as follows:

$$SSD(\mathbf{x}_i, \hat{\mathbf{x}}_i) = \sum_{\mathbf{p} \in W} \{I(\mathbf{x}_i + \mathbf{p}) - I(\hat{\mathbf{x}}_i + \mathbf{p})\}^2. \quad (2.2)$$

Here, $I(\mathbf{x})$ represents the intensity of pixel \mathbf{x} . Pixel $\hat{\mathbf{x}}_i$ for minimizing E_{conv} is decided as follows:

$$\hat{\mathbf{x}}_i = \operatorname{argmin}_{\mathbf{x}' \in \Phi} SSD(\mathbf{x}_i, \mathbf{x}'). \quad (2.3)$$

Note that weight $w_{\mathbf{x}_i}$ is set as 1 if \mathbf{x}_i is in region $\Omega' \cap \overline{\Omega}$ because pixel values in this region are fixed; otherwise $w_{\mathbf{x}_i}$ is set as c^{-d} (d is the distance from the boundary of Ω and c is a positive constant) because pixel values around the boundary have higher confidence than those in the center of the missing region. In Wexler's work [WSI07], the missing region is completed by calculating pixel value $I(\mathbf{x}_i)$ in the missing region and the position of pixel $\hat{\mathbf{x}}_i$ that minimizes energy function E_{conv} .

2.4. Extension of Energy Function Considering Brightness Changes and Spatial Locality

In this study, conventional energy function E_{conv} defined in Eq. (2.1) is extended by considering brightness changes and spatial locality of texture patterns. Concretely, a modification coefficient is introduced to allow for linear brightness changes in the texture pattern. For considering spatial pattern locality, the cost function based on the distance between a pixel in a missing region and the corresponding pixel in a data region is added to the conventional energy function. The extended energy function is defined as follows:

$$E_{ic} = \sum_{\mathbf{x}_i \in \Omega'} w_{\mathbf{x}_i} \{SSD'(\mathbf{x}_i, \hat{\mathbf{x}}_i) + w_{dis} SD(\mathbf{x}_i, \hat{\mathbf{x}}_i)\}, \quad (2.4)$$

where $SSD'(\mathbf{x}_i, \hat{\mathbf{x}}_i)$ means the pattern similarity considering brightness changes, and $SD(\mathbf{x}_i, \hat{\mathbf{x}}_i)$ means the cost term for the spatial locality. w_{dis} is the weight

representing the strength of spatial locality. In the following sections, definitions of $SSD'(\mathbf{x}_i, \hat{\mathbf{x}}_i)$ and $SD(\mathbf{x}_i, \hat{\mathbf{x}}_i)$ are described in detail.

2.4.1 Pattern Similarity Considering Brightness Changes

In general images, there are many texture pairs that have the same texture pattern but different brightness under the non-uniform illumination condition. Although it can be effective to use such texture patterns for image completion, that was not considered in similarity SSD (Eq. (2.2)) in the conventional method [WSI07]. In this study, by introducing a modification coefficient to allow textures to change brightness, more texture patterns can be used for image completion to improve the quality of results.

Concretely, similarity measure SSD' considering brightness changes of textures is defined as follows:

$$SSD'(\mathbf{x}_i, \hat{\mathbf{x}}_i) = \sum_{\mathbf{p} \in W} \{I(\mathbf{x}_i + \mathbf{p}) - \alpha_{\mathbf{x}_i \hat{\mathbf{x}}_i} I(\hat{\mathbf{x}}_i + \mathbf{p})\}^2. \quad (2.5)$$

Here, $\alpha_{\mathbf{x}_i \hat{\mathbf{x}}_i}$ is an intensity modification coefficient to allow the brightness change of textures. In this thesis, the ratio of average pixel values around pixel \mathbf{x}_i and pixel $\hat{\mathbf{x}}_i$ is employed as modification coefficient $\alpha_{\mathbf{x}_i \hat{\mathbf{x}}_i}$ to adjust the brightness of textures in the data region to that in the missing region. However, an unnatural image is easily generated if large brightness change is approximated by linear transformation. Therefore, the range of the value of $\alpha_{\mathbf{x}_i \hat{\mathbf{x}}_i}$ is limited as given in Eq. (2.6):

$$\alpha_{\mathbf{x}_i \hat{\mathbf{x}}_i} = \begin{cases} 1 - D & (\beta_{\mathbf{x}_i \hat{\mathbf{x}}_i} < 1 - D) \\ \beta_{\mathbf{x}_i \hat{\mathbf{x}}_i} & (1 - D \leq \beta_{\mathbf{x}_i \hat{\mathbf{x}}_i} \leq 1 + D) \\ 1 + D & (\beta_{\mathbf{x}_i \hat{\mathbf{x}}_i} > 1 + D), \end{cases} \quad (2.6)$$

where D is a constant ($0 < D < 1$) and $\beta_{\mathbf{x}_i \hat{\mathbf{x}}_i}$ is defined as follows:

$$\beta_{\mathbf{x}_i \hat{\mathbf{x}}_i} = \frac{\sqrt{\sum_{\mathbf{q} \in W} I(\mathbf{x}_i + \mathbf{q})^2}}{\sqrt{\sum_{\mathbf{q} \in W} I(\hat{\mathbf{x}}_i + \mathbf{q})^2}}. \quad (2.7)$$

Modification coefficient $\alpha_{\mathbf{x}\hat{\mathbf{x}}}$ makes the brightness of generated textures smooth while preserving texture patterns and enables utilization of textures that have different brightnesses but the same pattern.

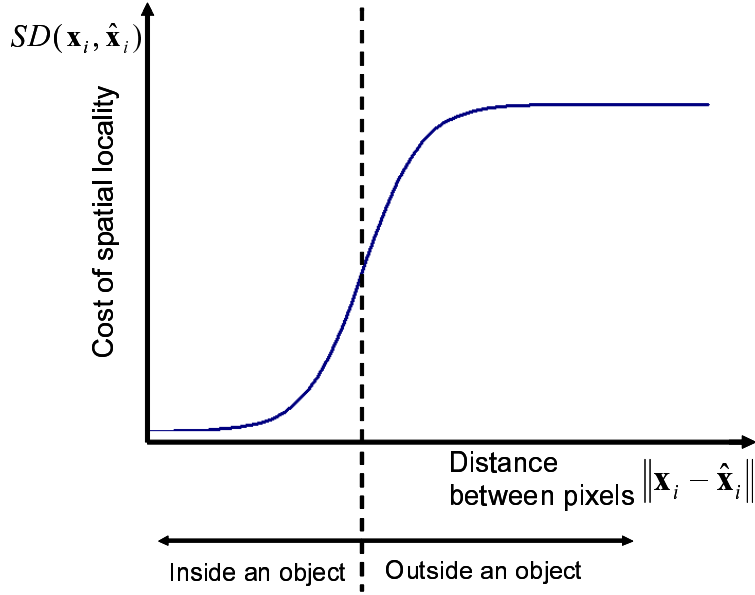


Figure 2.3. Sigmoid function for cost of spatial locality.

2.4.2 Cost Term for Spatial Locality

In many images, it is highly possible that similar textures exist in neighboring areas. Here, spatial locality of texture patterns SD considering such a characteristic is defined by using a sigmoid function as follows:

$$SD(\mathbf{x}_i, \hat{\mathbf{x}}_i) = \frac{N_W}{1 + e^{\{-K(\|\mathbf{x}_i - \hat{\mathbf{x}}_i\| - X_0)\}}}, \quad (2.8)$$

where K and X_0 are constant and N_W is the number of pixels in window W . As illustrated in Figure 2.3, this cost function is defined based on the assumption that the probability of similar texture existence for a certain pixel is uniformly high for the object region where the pixel exists. On the other hand, outside the object region, the probability can be assumed to be uniformly low. Therefore, the cost function as shown in Eq. (2.8) is defined using a sigmoid function, which gives uniform values inside and outside a certain range. Note that a constant-sized object region is currently assumed in Eq. (2.8) because we could not know the range of the object in the missing region.

By adding the constraint of spatial locality, even when the deformation of

texture pattern exists around the target region, appropriate textures that exist near the target region are preferentially selected. Thus, textures are less likely to blur by not selecting blurry textures of low frequency far from the missing region.

2.5. Energy Minimization

Energy function E_{ic} defined in Eq. (2.4) is minimized using a framework of greedy algorithm similar to Wexler’s EM approach [WSI07]. In the proposed method, it should be noted to the fact that energy function E_{ic} for each pixel can be treated independently if similar pattern pairs $(\mathbf{x}_i, \hat{\mathbf{x}}_i)$ can be fixed and the change of coefficient $\alpha_{\mathbf{x}\hat{\mathbf{x}}}$ in an iteration is much smaller than the change of pixel values in missing regions. Thus, the following two processes are repeated until energy function E_{ic} converges: (i) searching for a texture pattern while keeping all pixel values in missing regions fixed; and (ii) parallel updating of all pixel values in the missing regions while keeping similar pairs of windows between the missing regions and the data regions fixed. In the following sections, each process is described in detail.

2.5.1 Searching for Similar Texture Pattern

In process (i), data regions are searched for a similar texture pattern keeping the pixel values in missing regions fixed. Basically, the position $\hat{\mathbf{x}}_i$ of the most similar texture pattern can be updated by calculating SSD' and SD that satisfy the following equation:

$$\mathbf{f}(\mathbf{x}_i) = \hat{\mathbf{x}}_i = \underset{\mathbf{x}' \in \Phi}{\operatorname{argmin}}(SSD'(\mathbf{x}_i, \mathbf{x}') + w_{dis}SD(\mathbf{x}_i, \mathbf{x}')). \quad (2.9)$$

However, there is a high cost for calculating SSD' and SD for all the pixels in data region Φ . For this problem, two methods are used in this research. One is SSDA (Sequential Similarity Detection Algorithm) [BS72], which can skip the calculation of SSD' whose value is quite larger than the minimum of SSD' . The other method is making lists of the positions of similar patterns. Making lists and searching only the lists can reduce the searching region. Figure 2.4 shows the flow of searching for the most similar patterns and generating a list for each

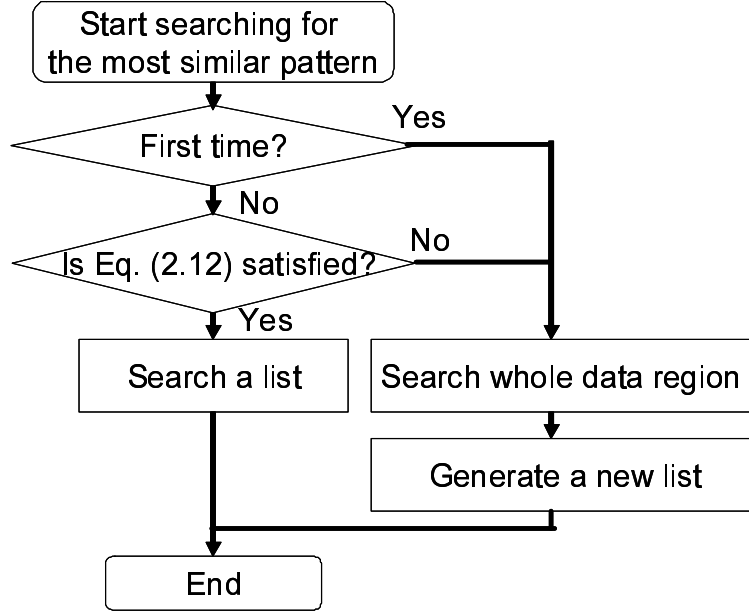


Figure 2.4. Flow of searching for the most similar pattern and generating list.

pixel. In the first time of process (i), the entire data region Φ is searched for the most similar pattern and a list is generated. After that, the list is searched for the most similar pattern if the condition described below is satisfied. Otherwise the list is updated by searching the entire data region. This process is performed pixel-by-pixel.

First, the way to generate a list for each pixel is described. Here, the smallest evaluation value of $SSD + w_{dis}SD$ by exhaustive search is set as S_{min} as follows:

$$S_{min} = SSD'(\mathbf{x}_i, f(\mathbf{x}_i)) + w_{dis}SD(\mathbf{x}_i, f(\mathbf{x}_i)). \quad (2.10)$$

Next, \mathbf{x}' which satisfies the following condition is listed as the candidate of the position of the most similar pattern:

$$SSD'(\mathbf{x}_i, \mathbf{x}') + w_{dis}SD(\mathbf{x}_i, \mathbf{x}') < T_1 S_{min}, \quad (2.11)$$

where T_1 is a constant.

The computational cost is reduced by using the list because only the list is searched for the most similar pattern. However, the update of pixel values around

pixel \mathbf{x}_i by iterative processing leads to the absence of the most similar pattern in the list. Therefore, the list is updated by exhaustively searching if the following condition is not satisfied:

$$\sum_{\mathbf{p} \in W} \{I'(\mathbf{x}_i + \mathbf{p}) - I(\mathbf{x}_i + \mathbf{p})\}^2 < T_2 S_{min}, \quad (2.12)$$

where I' is the pixel value when the list is generated and I is the current pixel value after updating the value. T_2 is a constant and the value of T_2 should be set considering the trade-off between the possible existence of the most similar pattern and the computational cost of updating the list through exhaustive search.

2.5.2 Parallel Updating of Pixel Values

In process (ii), all the pixel values $I(\mathbf{x}_i)$ in missing regions are updated in parallel so as to minimize energy function E_{ic} by keeping all the similar pairs fixed. In the following, the method for calculating pixel values $I(\mathbf{x}_i)$ for fixed similar pairs of windows is described in detail.

First, energy function E_{ic} is resolved into element energy $E_{ic}(\mathbf{x}_i)$ for each pixel \mathbf{x}_i in missing regions. As shown in Figure 2.5, the target pixel to be updated is \mathbf{x}_i , and a pixel inside a window can be expressed as $\mathbf{x}_i + \mathbf{p}$ ($\mathbf{p} \in W$) and is corresponded to $f(\mathbf{x}_i + \mathbf{p})$ by Eq. (2.9). Thus, the pixel corresponding to pixel \mathbf{x}_i becomes $f(\mathbf{x}_i + \mathbf{p}) - \mathbf{p}$. Now, element energy $E_{ic}(\mathbf{x}_i)$ can be defined in terms of the pixel values of \mathbf{x}_i and $f(\mathbf{x}_i + \mathbf{p}) - \mathbf{p}$, the coefficient α and the Euclid distance between \mathbf{x}_i and $f(\mathbf{x}_i)$ as follows:

$$E_{ic}(\mathbf{x}_i) = \sum_{\mathbf{p} \in W} w_{(\mathbf{x}_i + \mathbf{p})} \{I(\mathbf{x}_i) - \alpha_{(\mathbf{x}_i + \mathbf{p})(f(\mathbf{x}_i + \mathbf{p}))} I(f(\mathbf{x}_i + \mathbf{p}) - \mathbf{p})\}^2 + w_{dis} \frac{N_W}{1 + e^{\{-K(\|\mathbf{x}_i - f(\mathbf{x}_i)\| - X_0)\}}}. \quad (2.13)$$

The relationship between total energy E_{ic} for whole the missing regions and element energy $E_{ic}(\mathbf{x}_i)$ for each pixel \mathbf{x}_i can be written as follows:

$$E_{ic} = \sum_{\mathbf{x}_i \in \Omega} E_{ic}(\mathbf{x}_i) + C. \quad (2.14)$$

C is the energy for the pixels in region $\bar{\Omega} \cap \Omega'$, and is treated as a constant because pixel values in this region and all similar pairs of windows are fixed here.

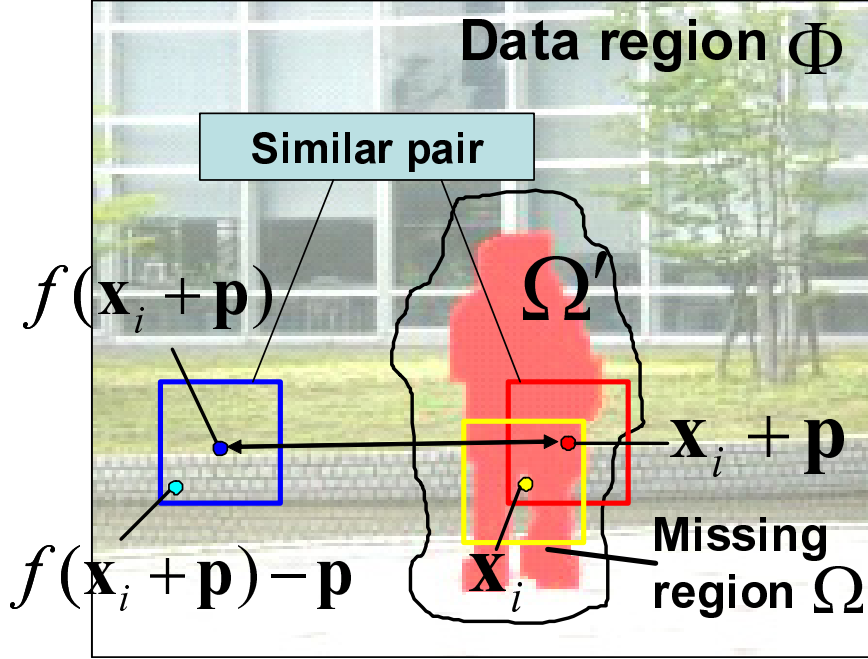


Figure 2.5. Relationship between pixels in energy calculation.

By differentiating E_{ic} with respect to $I(\mathbf{x}_i)$ in the missing region, the requirement for minimizing energy E_{ic} can be obtained as follows:

$$\frac{\partial E_{ic}}{\partial I(\mathbf{x}_i)} = \sum_{\mathbf{x}_j \in \Omega} \frac{\partial E_{ic}(\mathbf{x}_j)}{\partial I(\mathbf{x}_i)} = 0. \quad (2.15)$$

Here, if it is assumed that the change of intensity modification coefficient α is much smaller than that of pixel value $I(\mathbf{x}_i)$, the following equation is obtained:

$$\frac{\partial \alpha_{\mathbf{x}_j} f(\mathbf{x}_j)}{\partial I(\mathbf{x}_i)} = 0 \quad (\forall \mathbf{x}_j \in \Omega'). \quad (2.16)$$

From this equation, equation $\partial E_{ic}(\mathbf{x}_j)/\partial I(\mathbf{x}_i) = 0$ ($j \neq i$) is formed. Thus, energy E_{ic} is minimized by calculating $I(\mathbf{x}_k)$, which satisfies the following equation:

$$\frac{\partial E_{ic}}{\partial I(\mathbf{x}_i)} = \frac{\partial E_{ic}(\mathbf{x}_i)}{\partial I(\mathbf{x}_i)} = 0. \quad (2.17)$$

Here,

$$\frac{\partial E_{ic}(\mathbf{x}_i)}{\partial I(\mathbf{x}_i)} = \sum_{\mathbf{p} \in W} w_{(\mathbf{x}_i + \mathbf{p})} 2 \{ I(\mathbf{x}_i) - \alpha_{(\mathbf{x}_i + \mathbf{p})(f(\mathbf{x}_i + \mathbf{p}))} I(f(\mathbf{x}_i + \mathbf{p}) - \mathbf{p}) \} \\ \{ 1 - \frac{\partial \alpha_{(\mathbf{x}_i + \mathbf{p})(f(\mathbf{x}_i + \mathbf{p}))}}{\partial I(\mathbf{x}_i)} I(f(\mathbf{x}_i + \mathbf{p}) - \mathbf{p}) \}. \quad (2.18)$$

Therefore, from Eqs. (2.16), (2.17) and (2.18), pixel value $I(\mathbf{x}_i)$ so as to minimize E_{ic} can be calculated as follows:

$$I(\mathbf{x}_i) = \frac{\sum_{\mathbf{p} \in W} w_{(\mathbf{x}_i + \mathbf{p})} \alpha_{(\mathbf{x}_i + \mathbf{p})(f(\mathbf{x}_i + \mathbf{p}))} I(f(\mathbf{x}_i + \mathbf{p}) - \mathbf{p})}{\sum_{\mathbf{p} \in W} w_{(\mathbf{x}_i + \mathbf{p})}}. \quad (2.19)$$

Eq. (2.19) is an approximate solution when Eq. (2.16) is satisfied. We can obtain a good solution as the energy converges because the value of intensity modification coefficient α converges as $I(\mathbf{x}_i)$ converges.

2.5.3 Coarse-to-fine Approach

In order to avoid local minima efficiently, a coarse-to-fine approach is also employed as shown in the left part of Figure 2.6. Specifically, an image pyramid is generated and energy minimization processes (i) and (ii) are repeated from higher-level to lower-level layers successively using a certain size of window. Good initial values are given to the lower layer by projecting results from the higher layer. This makes it possible to decrease computational cost and avoid local minima.

In the lowest layer (original size), as shown in the right part of Figure 2.6, the energy minimization process is repeated while successively reducing the size of the window. This enables reproduction of more detailed textures.

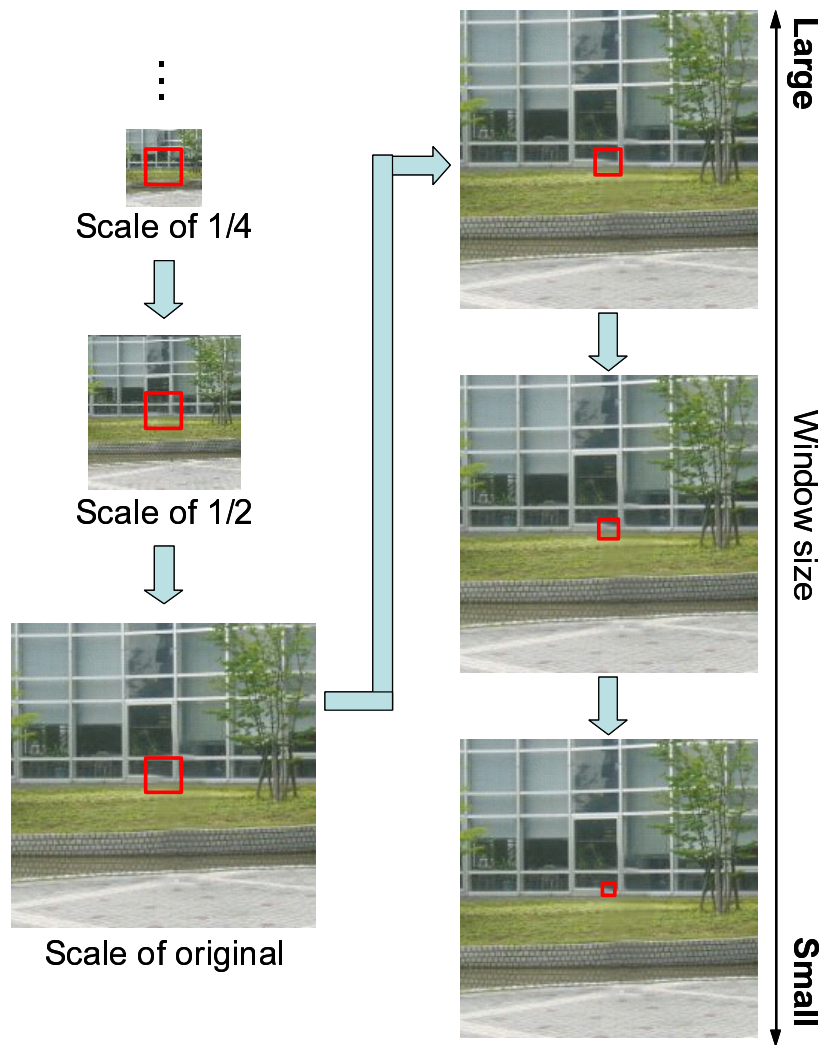


Figure 2.6. Coarse-to-fine Approach.

2.6. Order of Computational Cost

In the proposed method, missing regions are completed by iterating two processes: (i) searching for a texture pattern keeping all pixel values in missing regions fixed; and (ii) parallel updating of all pixel values in the missing regions keeping similar pairs of windows between the missing regions and the data regions fixed. Therefore, the order of the computational cost can be expressed as follows:

$$O((C_{search} + C_{update})N_{iteration}), \quad (2.20)$$

where C_{search} is the cost for searching for similar texture patterns, C_{update} is the cost for updating pixel values and $N_{iteration}$ is the number of iterations.

C_{search} consists of the costs for calculating SSD' (Eq. (2.5)) and SD (Eq. (2.8)). The cost for SSD' between two pixels in expanded missing region Ω' and data region Φ is expressed using cost C_{dif} for calculating squared differences of two pixel values, cost C_α for calculating intensity modification coefficient α from two pixel values in region Ω' and in data region Φ , and the number of pixels N_W in window W as follows:

$$(C_\alpha + C_{dif})N_W. \quad (2.21)$$

The cost for SD between two pixels in region Ω' and data region Φ is expressed as C_{sd} . Therefore, cost C_{search} , which consists of the costs for calculating SSD' and SD between all pixel pairs in region Ω' and data region Φ , can be expressed as follows:

$$C_{search} = \{(C_\alpha + C_{dif})N_W + C_{sd}\}N_{\Omega'}N_\Phi, \quad (2.22)$$

where N_Ω and N_Φ are the number of pixels in region Ω' and region Φ , respectively. Here, the sum of number of pixels in regions Ω' and Φ equals the number of pixels N_{image} in the image. Thus Eq. (2.22) can be described using the number of pixels N_{image} in the image as follows:

$$C_{search} = \{(C_\alpha + C_{dif})N_W + C_{sd}\}N_{\Omega'}(N_{image} - N_{\Omega'}). \quad (2.23)$$

C_{update} is expressed using cost C_{pv} for calculating a pixel value in missing region Ω and the number of pixels N_Ω in missing region Ω as follows:

$$C_{update} = C_{pv}N_\Omega. \quad (2.24)$$

From Eqs. (2.20), (2.23) and (2.24), the order of the computational cost of the proposed method is described as follows:

$$O(N_W N_{\Omega'} N_{image} N_{iteration}). \quad (2.25)$$

From this, we can confirm that the processing time of the proposed method is proportional to the number of pixels N_W in Window W , the number of pixels $N_{\Omega'}$ in region Ω' , the number of pixels N_{image} in an image and the number of iterations $N_{iteration}$. $N_{\Omega'}$ and N_{image} of the four parameters depend on user input. For example, when the width and height of an image is doubled and the ratio of region Ω' in the image is unchanged, the processing time becomes 2^4 times longer.

2.7. Experiments

In this section, in order to demonstrate the effectiveness of the proposed method, we have applied five kinds of image completion methods, including the conventional methods and the proposed methods, to 100 images (200×200 pixels) as shown in Figure 2.7. The methods used in the experiments are as follows:

Method A Our implemented Criminisi’s method [CPT04], which is based on successive copy of similarity texture

Method B Our implemented Wexler’s method [WSI07], which is based on energy minimization using pattern similarity

Method C Proposed method considering only brightness changes

Method D Proposed method considering only spatial locality

Method E Proposed method considering considering both brightness changes and spatial locality

Here, in order to confirm the effectiveness of considering the brightness changes and spatial locality, respectively, the method considering only brightness changes (Method C) and the method considering only spatial locality (Method D) were also evaluated as well as the proposed method considering both brightness changes and spatial locality (Method E).

In these experiments, we used a PC whose specifications were Xeon 3.2 GHz of CPU and 8 GB of memory. Each parameter in the energy function was set as shown in Table 2.1 and the same parameters are used for all 100 images. Missing regions were manually specified as shown in red regions in each image in Figure 2.7, and the average of pixel values on the boundary of the missing region was given as an initial value to the missing region.

In this section, first, the effectiveness of the proposed method is qualitatively demonstrated by questionnaire evaluation in Section 2.7.1 and the resultant images are discussed based on the questionnaire evaluation in Section 2.7.2. Then, the results of the five methods are quantitatively evaluated in Section 2.7.3. The computational costs of the proposed and conventional methods are compared in

Table 2.1. Parameters in image completion experiments.

| Multi-scale Level | 1 | 2 | 3 | 4 | 5 | 6 |
|--|------------------|-----|-----|-----|-----|-----|
| Image size | 1/4 | 1/2 | 1 | 1 | 1 | 1 |
| Window size | 9×9 | 9×9 | 9×9 | 7×7 | 5×5 | 3×3 |
| c in weight for pixel $w_{\mathbf{x}}$ | 1.3 | | | | | |
| Weight w_{dis} for spatial locality | 120 | | | | | |
| Parameter in sigmoid function K | 0.4 | | | | | |
| Parameter in sigmoid function X_0 | 20 | | | | | |
| Range D of coefficient α | 0.1 | | | | | |
| Parameter T_1 for list | 4 | | | | | |
| Parameter T_2 for list | 0.5 | | | | | |
| Condition for convergence (Ratio of decrease of energy) | Less than 0.01 % | | | | | |

Section 2.7.4. The local minima problem is discussed in Section 2.7.5. Finally, the reliability of the evaluation methods are discussed in Section 2.7.6.

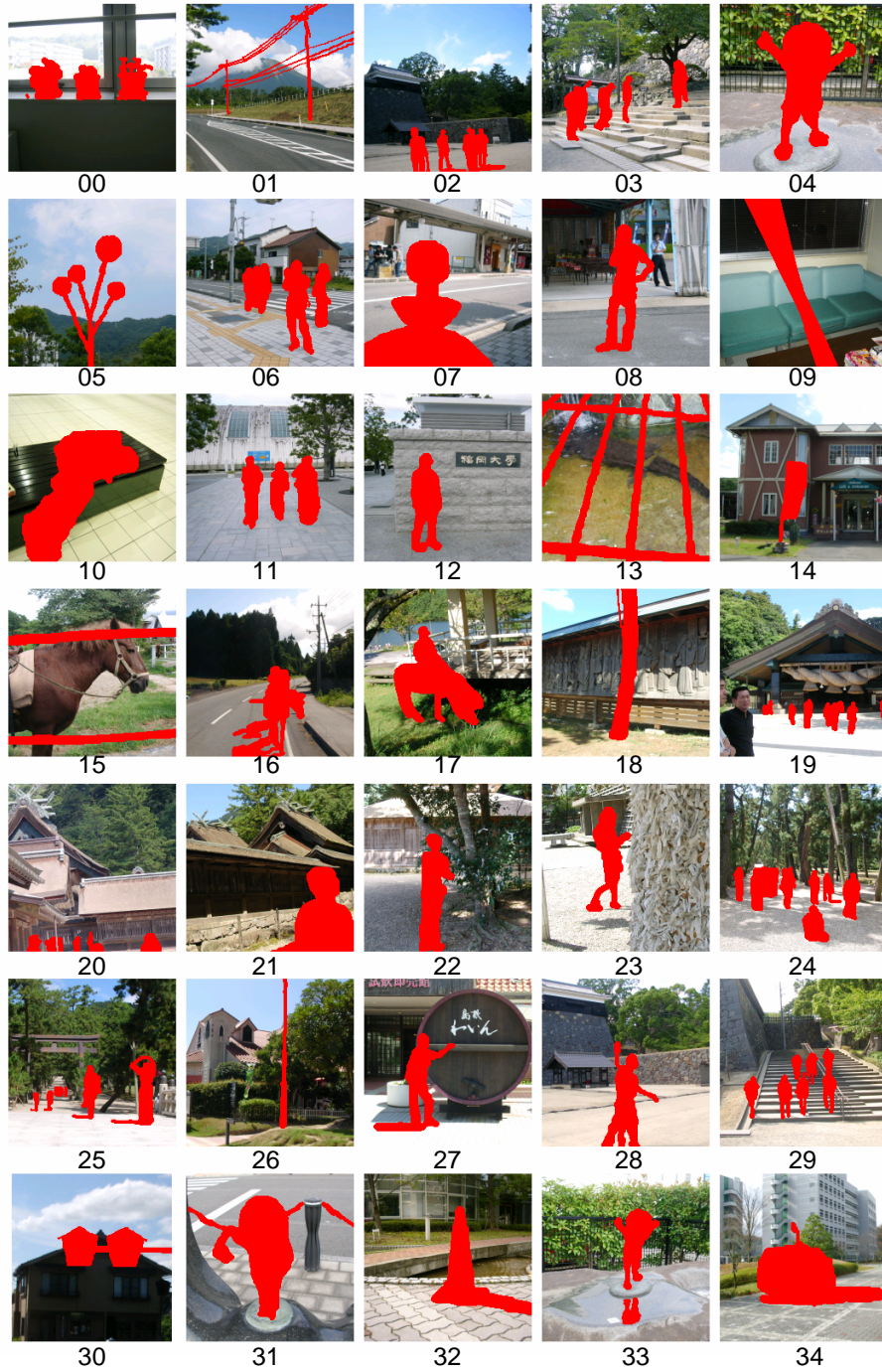


Figure 2.7. Input images with missing regions (1/3)

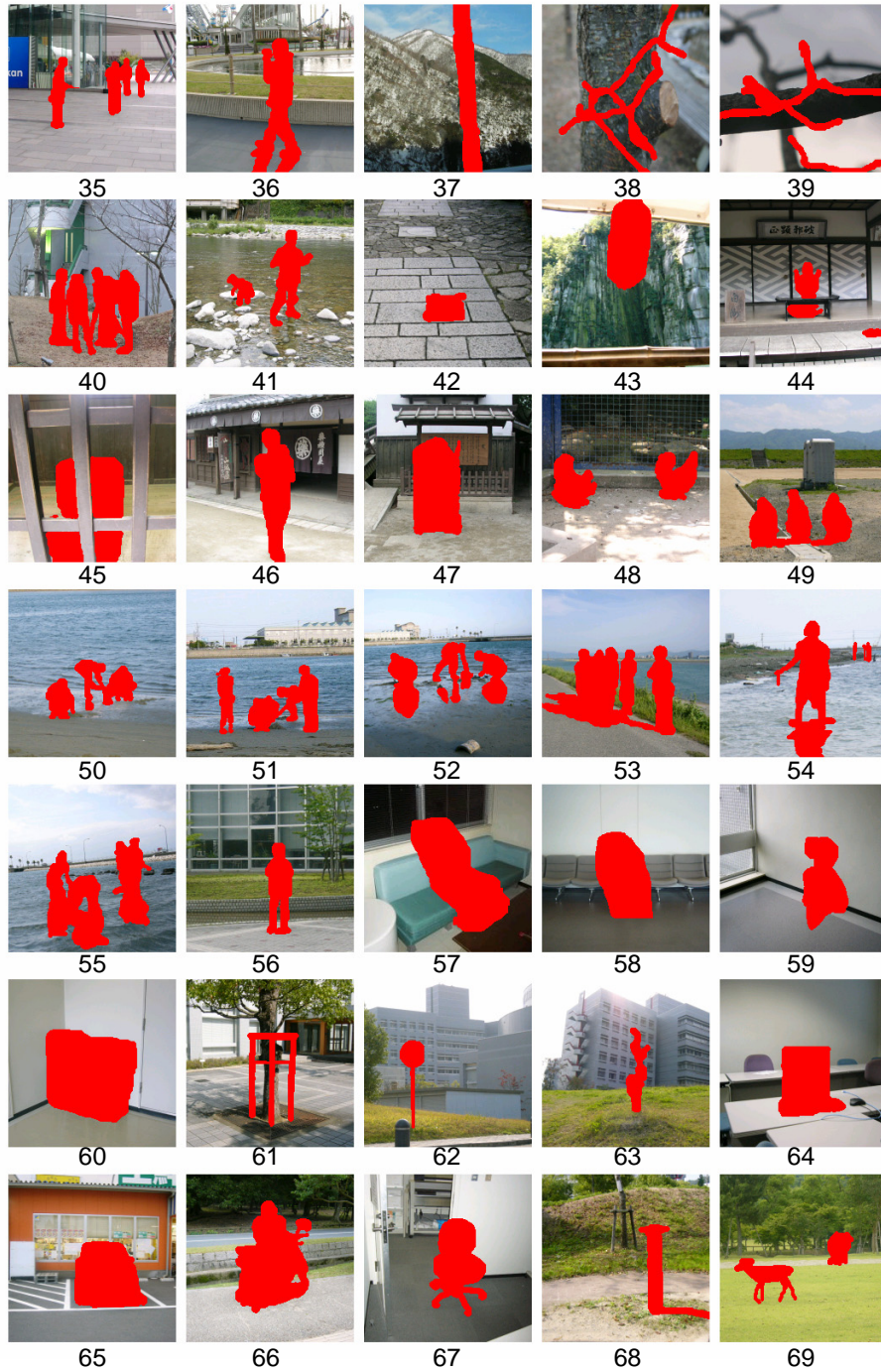


Figure 2.7. Input images with missing regions (2/3)

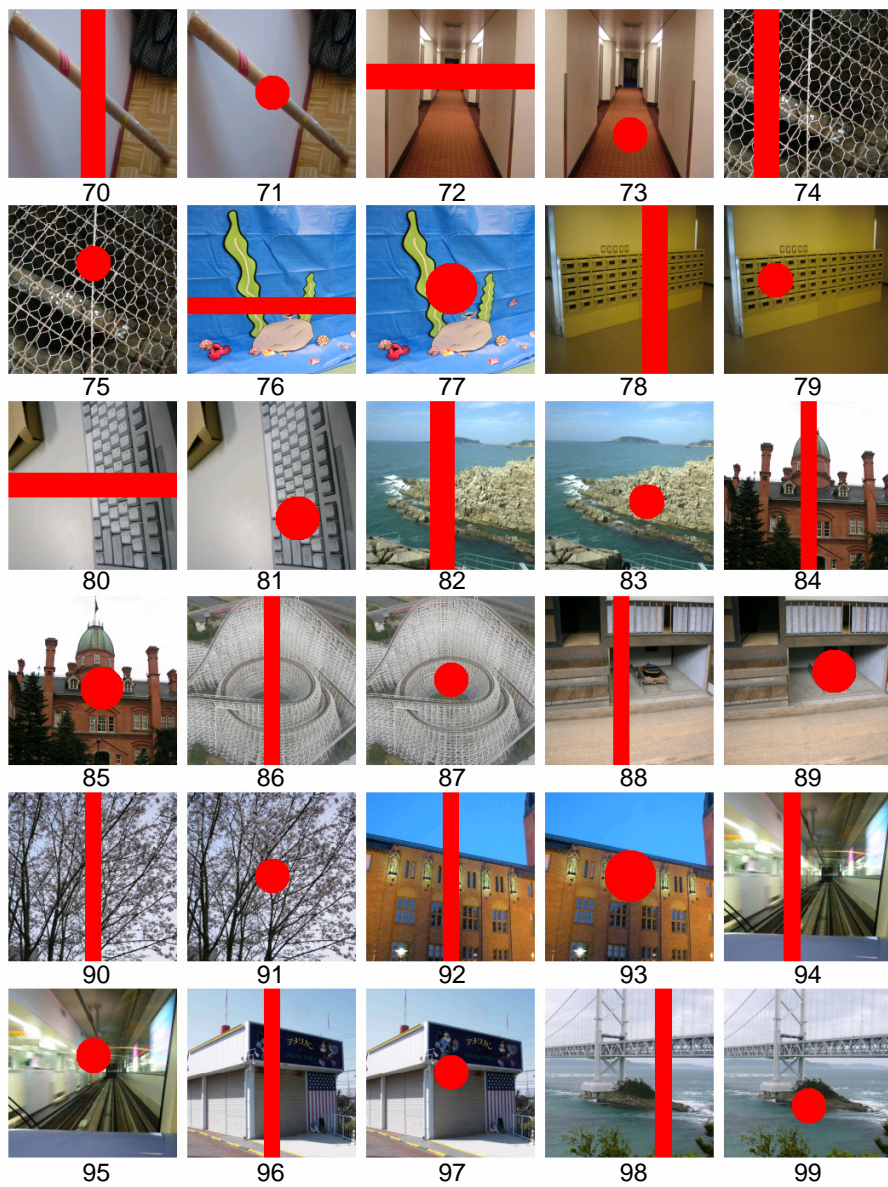


Figure 2.7. Input images with missing regions (3/3)

2.7.1 Qualitative Evaluation

In the qualitative evaluation, all the completed images by five methods (Methods A, B, C, D and E) were subjectively evaluated by 37 subjects (students of Nara Institute of Science and Technology). The subjects were requested to access the web page and complete a questionnaire evaluation as shown in Figure 2.8 and evaluate the resultant images arranged in random order by giving a score of 1 (bad) to 5 (good).

The average of the scores given by the 37 subjects for each of the 100 images are shown in Figure 2.9. The average scores for the 100 resultant images are shown in Table 2.2 for each method. The number of times each method got the highest average score is shown in Table 2.3. From Tables 2.2 and 2.3, the average score by the proposed method (Method E) is higher than the scores by the conventional methods (Methods A and B) and the proposed method got the highest score for the most images. In this experiment, in order to evaluate the effectiveness of the proposed method statistically, the scores between each two methods out of five methods were compared by using the t-test with a 5% significant level. Table 2.4 shows the amount of statistics and whether or not a significant difference is observed. From these results, significant differences were observed between the scores for any two methods except Methods C and D. Therefore, considering brightness changes and spatial locality is respectively effective the proposed method (Method E) considering both brightness changes and spatial locality can be statistically verified to be the best of the five methods.

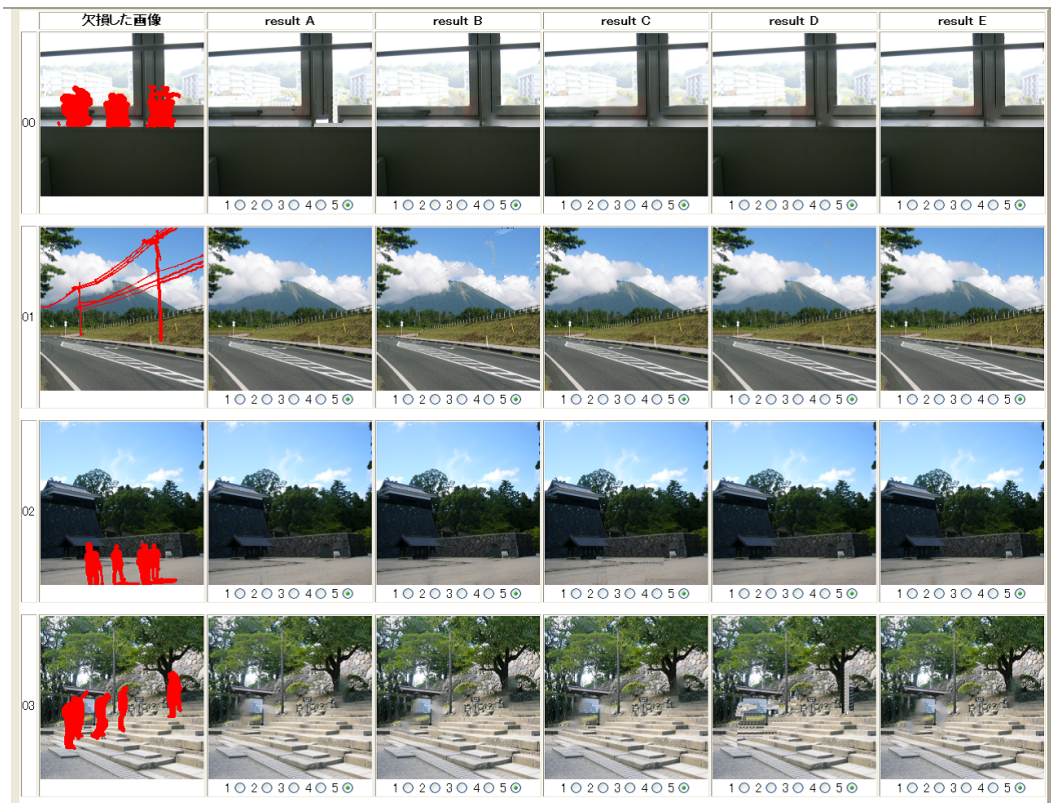


Figure 2.8. Web page for questionnaire evaluation.

Table 2.2. Average score from 100 images.

| Method | A | B | C | D | E |
|---------------|------|------|------|------|------|
| Average score | 2.21 | 3.24 | 3.39 | 3.42 | 3.60 |

Table 2.3. The number of times the method obtained the highest score.

| Method | A | B | C | D | E |
|---------------|---|---|----|----|----|
| Average score | 7 | 7 | 26 | 23 | 45 |

Table 2.4. T-value produced by the t-test between a pair of methods with a 5% significant level. (If significant difference is observed, the t-value is underlined.)

| Method | B[WSI07] | C | D | E |
|----------|--------------|--------------|---------------|---------------|
| A[CPT04] | <u>9.053</u> | <u>9.138</u> | <u>10.870</u> | <u>11.046</u> |
| B[WSI07] | - | <u>2.225</u> | <u>4.575</u> | <u>5.919</u> |
| C | - | - | 0.403 | <u>4.397</u> |
| D | - | - | - | <u>3.351</u> |

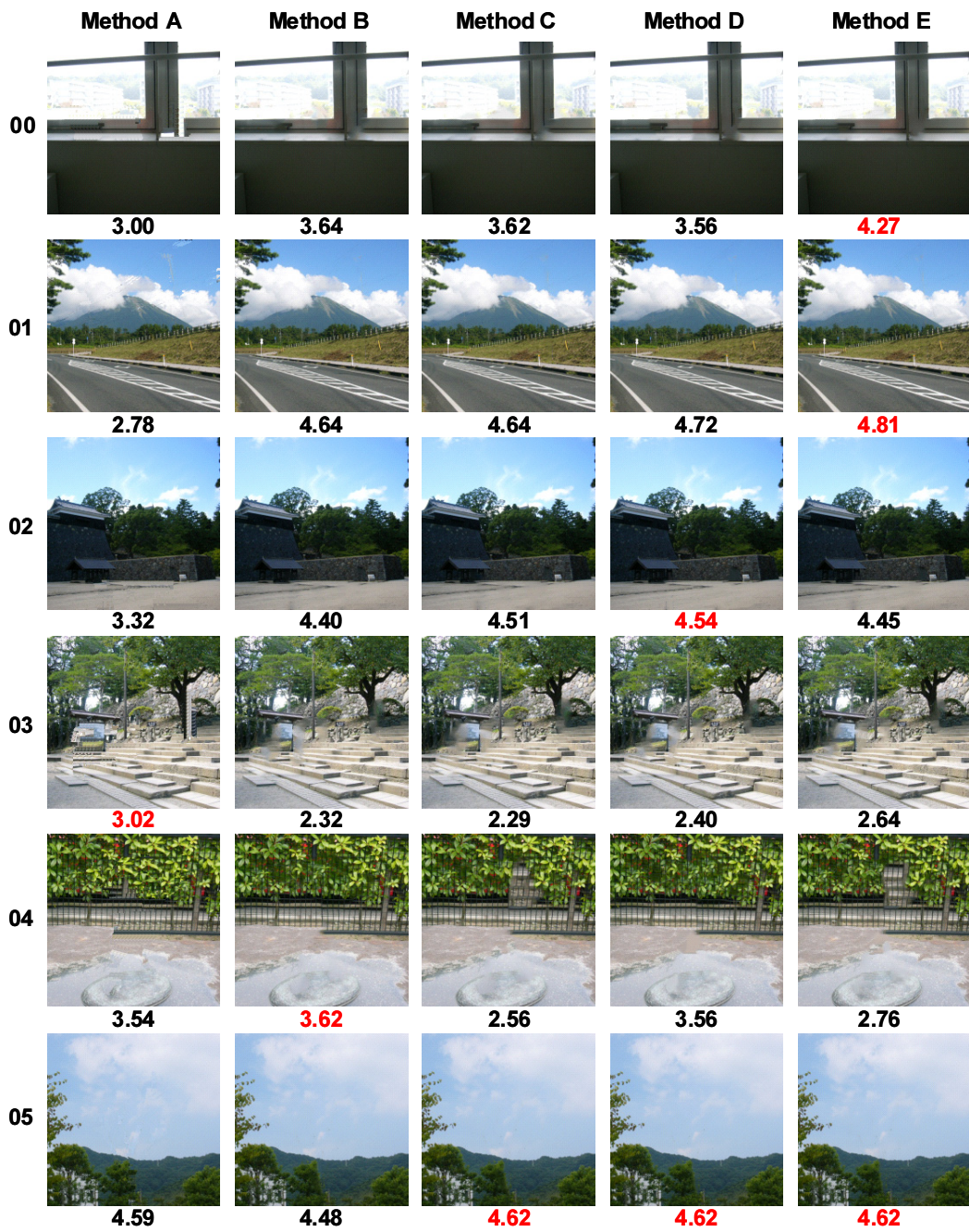


Figure 2.9. Resultant images and scores (1/17)











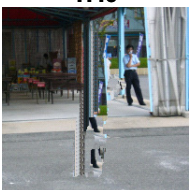
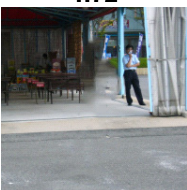
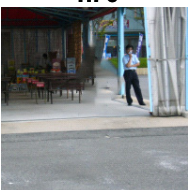
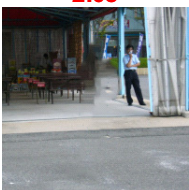
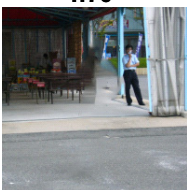
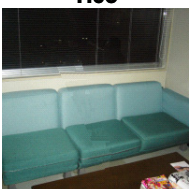











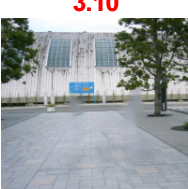
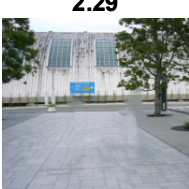
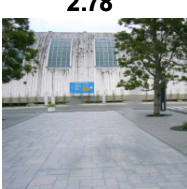
| | Method A | Method B | Method C | Method D | Method E |
|----|---|---|--|---|--|
| 06 |  1.89 |  2.45 |  1.81 |  2.37 |  2.76 |
| 07 |  1.40 |  1.72 |  1.70 |  2.08 |  1.70 |
| 08 |  1.05 |  2.91 |  3.16 |  3.27 |  2.72 |
| 09 |  2.62 |  3.56 |  4.45 |  3.16 |  4.83 |
| 10 |  1.21 |  2.62 |  3.10 |  2.29 |  2.78 |
| 11 |  3.51 |  2.56 |  2.62 |  2.75 |  3.64 |

Figure 2.9. Resultant images and scores (2/17)











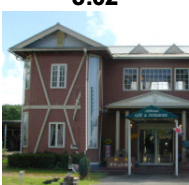


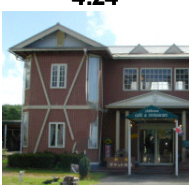



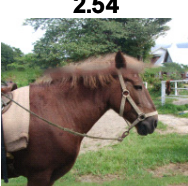
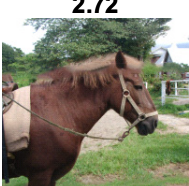
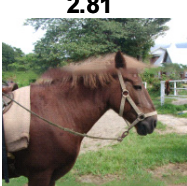










| | Method A | Method B | Method C | Method D | Method E |
|----|---|---|---|--|---|
| 12 |  2.81 |  3.05 |  3.62 |  4.05 |  4.43 |
| 13 |  3.02 |  4.00 |  4.10 |  4.24 |  4.27 |
| 14 |  3.37 |  2.48 |  2.54 |  2.72 |  2.81 |
| 15 |  1.89 |  3.21 |  3.24 |  3.64 |  2.97 |
| 16 |  1.13 |  3.32 |  3.48 |  3.27 |  3.91 |
| 17 |  1.67 |  1.37 |  2.08 |  1.45 |  1.78 |

Figure 2.9. Resultant images and scores (3/17)












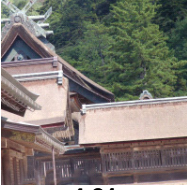
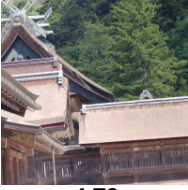

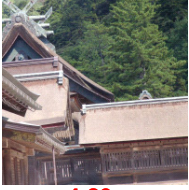















| | Method A | Method B | Method C | Method D | Method E |
|----|---|--|--|---|--|
| 18 |  4.35 |  4.70 |  3.40 |  4.48 |  3.81 |
| 19 |  4.70 |  4.78 |  4.72 |  4.83 |  4.83 |
| 20 |  3.89 |  4.81 |  4.78 |  4.86 |  4.89 |
| 21 |  1.43 |  2.43 |  2.70 |  2.56 |  2.51 |
| 22 |  1.78 |  2.43 |  2.70 |  2.56 |  2.62 |
| 23 |  2.78 |  3.83 |  3.43 |  4.02 |  3.59 |

Figure 2.9. Resultant images and scores (4/17)

| | Method A | Method B | Method C | Method D | Method E |
|----|---|---|---|--|---|
| 24 |  2.54 |  3.24 |  3.48 |  3.48 |  3.45 |
| 25 |  2.83 |  4.16 |  3.16 |  4.02 |  3.29 |
| 26 |  4.00 |  4.89 |  4.91 |  4.94 |  4.89 |
| 27 |  1.67 |  2.75 |  2.64 |  2.81 |  2.70 |
| 28 |  2.13 |  3.62 |  2.81 |  4.62 |  4.13 |
| 29 |  1.94 |  3.21 |  3.08 |  3.35 |  3.35 |

Figure 2.9. Resultant images and scores (5/17)











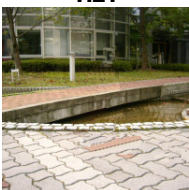
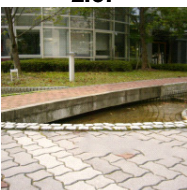
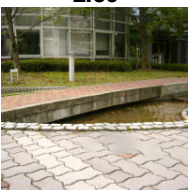
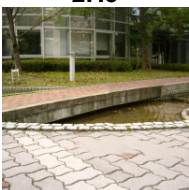



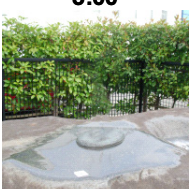






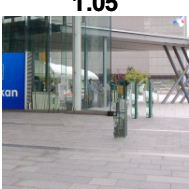

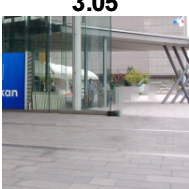
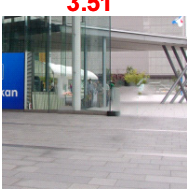
| | Method A | Method B | Method C | Method D | Method E |
|----|---|---|---|--|---|
| 30 |  1.48 |  4.32 |  1.56 |  4.45 |  4.67 |
| 31 |  1.21 |  2.37 |  2.59 |  2.43 |  3.27 |
| 32 |  3.67 |  3.21 |  3.72 |  3.00 |  3.43 |
| 33 |  3.59 |  4.35 |  2.94 |  4.16 |  3.59 |
| 34 |  1.05 |  3.02 |  3.24 |  3.05 |  3.51 |
| 35 |  1.43 |  3.59 |  3.64 |  4.00 |  3.72 |

Figure 2.9. Resultant images and scores (6/17)

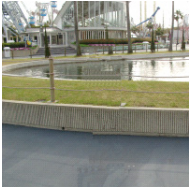
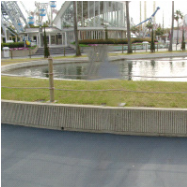
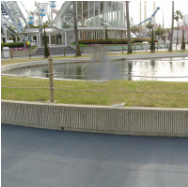
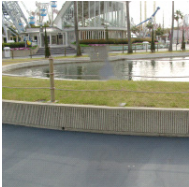
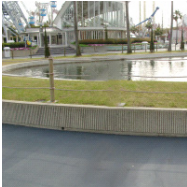
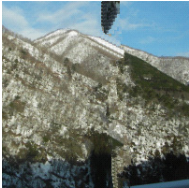
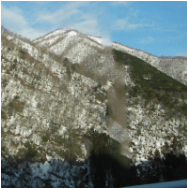
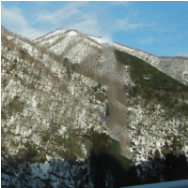
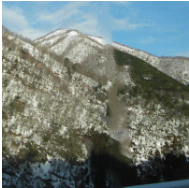
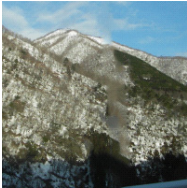






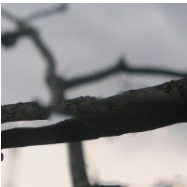
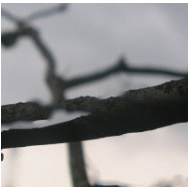
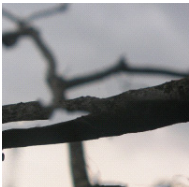
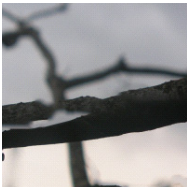
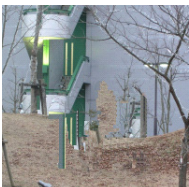
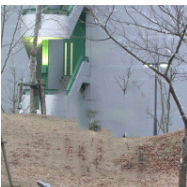
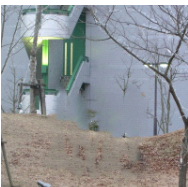
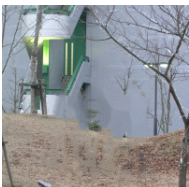
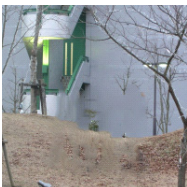

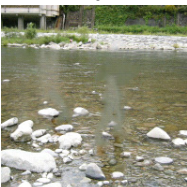

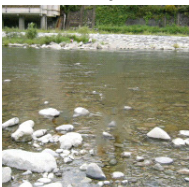
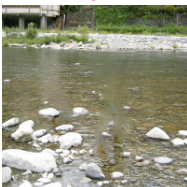
| | Method A | Method B | Method C | Method D | Method E |
|----|---|---|---|---|--|
| 36 |  4.05 |  3.24 |  3.21 |  3.40 |  4.29 |
| 37 |  1.37 |  2.86 |  2.97 |  3.45 |  2.97 |
| 38 |  3.54 |  3.32 |  3.21 |  4.05 |  4.51 |
| 39 |  1.08 |  2.18 |  2.24 |  2.18 |  2.27 |
| 40 |  1.48 |  2.37 |  2.48 |  2.29 |  2.62 |
| 41 |  3.16 |  2.62 |  2.97 |  3.89 |  3.51 |

Figure 2.9. Resultant images and scores (7/17)
























| | Method A | Method B | Method C | Method D | Method E |
|----|---|---|---|--|---|
| 42 |  2.94 |  3.81 |  4.16 |  4.75 |  4.21 |
| 43 |  2.16 |  2.45 |  4.08 |  2.59 |  4.00 |
| 44 |  1.35 |  2.78 |  3.64 |  3.10 |  3.67 |
| 45 |  1.27 |  2.27 |  2.05 |  2.32 |  2.21 |
| 46 |  3.13 |  3.10 |  3.48 |  3.97 |  3.91 |
| 47 |  1.29 |  2.08 |  2.64 |  2.48 |  2.56 |

Figure 2.9. Resultant images and scores (8/17)
























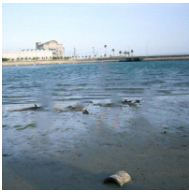
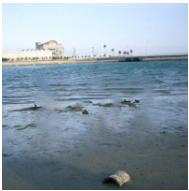



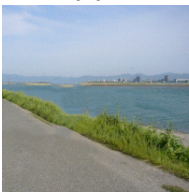

| | Method A | Method B | Method C | Method D | Method E |
|----|---|---|---|--|---|
| 48 |  1.72 |  4.18 |  4.35 |  3.86 |  4.16 |
| 49 |  2.29 |  2.54 |  2.64 |  2.70 |  2.78 |
| 50 |  2.27 |  3.94 |  3.86 |  4.00 |  4.18 |
| 51 |  2.00 |  3.08 |  3.18 |  2.91 |  2.78 |
| 52 |  2.05 |  3.27 |  3.05 |  3.32 |  3.43 |
| 53 |  1.32 |  3.37 |  3.91 |  3.51 |  3.64 |

Figure 2.9. Resultant images and scores (9/17)











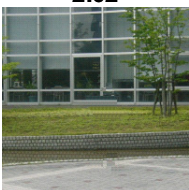

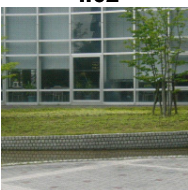
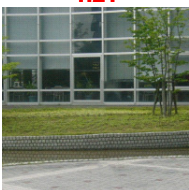
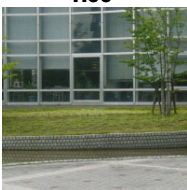





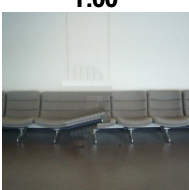
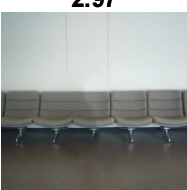
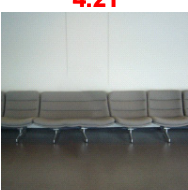


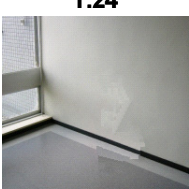


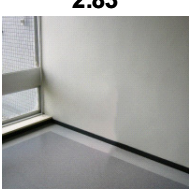
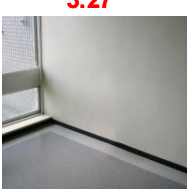
| | Method A | Method B | Method C | Method D | Method E |
|----|---|--|--|---|--|
| 54 |  2.40 |  4.62 |  3.21 |  4.10 |  3.75 |
| 55 |  2.02 |  3.94 |  4.02 |  4.21 |  4.00 |
| 56 |  3.10 |  4.72 |  4.86 |  4.67 |  4.83 |
| 57 |  1.00 |  2.97 |  4.21 |  2.51 |  3.75 |
| 58 |  1.24 |  3.00 |  3.08 |  2.83 |  3.27 |
| 59 |  1.64 |  4.27 |  4.78 |  3.70 |  4.64 |

Figure 2.9. Resultant images and scores (10/17)












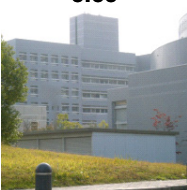

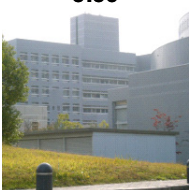
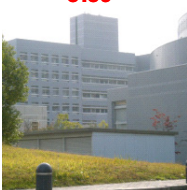
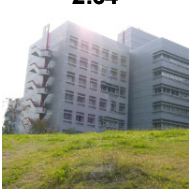
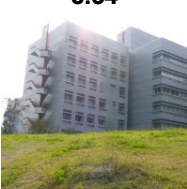


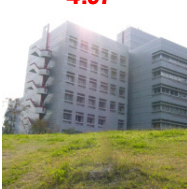
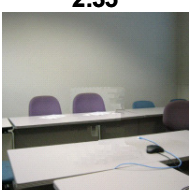

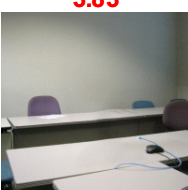
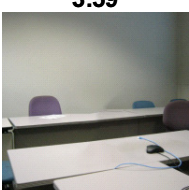
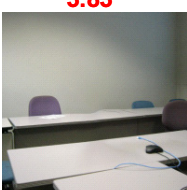





| | Method A | Method B | Method C | Method D | Method E |
|----|---|---|---|--|---|
| 60 |  1.10 |  1.78 |  1.72 |  1.56 |  1.59 |
| 61 |  1.21 |  3.35 |  3.48 |  3.56 |  3.59 |
| 62 |  2.64 |  3.54 |  4.94 |  4.00 |  4.97 |
| 63 |  2.35 |  2.64 |  3.83 |  3.59 |  3.83 |
| 64 |  2.18 |  4.43 |  3.89 |  3.64 |  3.21 |
| 65 |  1.16 |  2.29 |  2.51 |  2.45 |  2.48 |

Figure 2.9. Resultant images and scores (11/17)

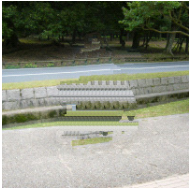


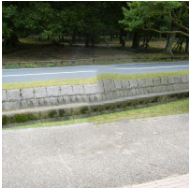






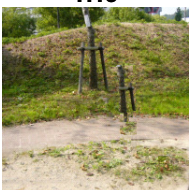


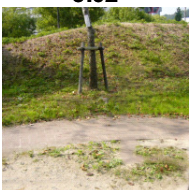
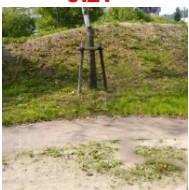
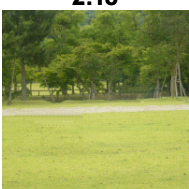


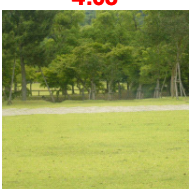




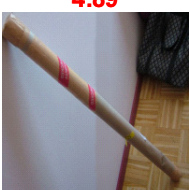

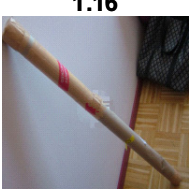




| | Method A | Method B | Method C | Method D | Method E |
|----|---|---|---|--|---|
| 66 |  1.05 |  2.75 |  3.75 |  2.86 |  4.35 |
| 67 |  1.13 |  2.86 |  2.94 |  3.02 |  3.21 |
| 68 |  2.18 |  3.78 |  3.27 |  4.08 |  3.35 |
| 69 |  4.32 |  4.67 |  4.64 |  4.89 |  4.62 |
| 70 |  1.16 |  4.21 |  4.89 |  3.91 |  4.91 |
| 71 |  1.81 |  4.51 |  4.83 |  4.64 |  4.78 |

Figure 2.9. Resultant images and scores (12/17)












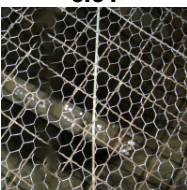
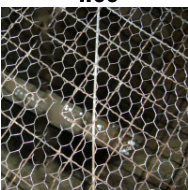
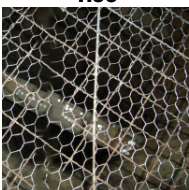
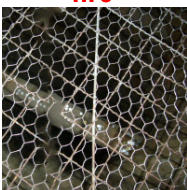















| | Method A | Method B | Method C | Method D | Method E |
|----|---|---|---|--|---|
| 72 |  1.21 |  3.32 |  3.51 |  3.10 |  3.40 |
| 73 |  2.40 |  3.54 |  4.00 |  4.35 |  4.75 |
| 74 |  2.70 |  2.81 |  3.48 |  3.21 |  3.48 |
| 75 |  4.05 |  2.81 |  3.18 |  3.02 |  3.24 |
| 76 |  1.32 |  4.59 |  4.78 |  4.72 |  4.75 |
| 77 |  1.21 |  3.91 |  3.81 |  4.18 |  4.16 |

Figure 2.9. Resultant images and scores (13/17)











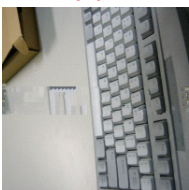
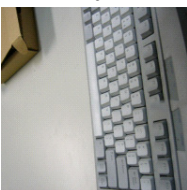
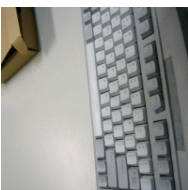
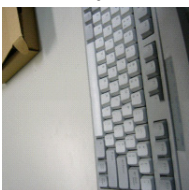
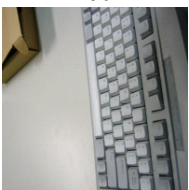
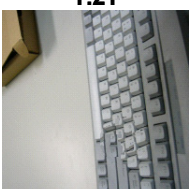
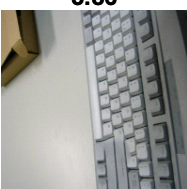
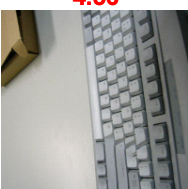

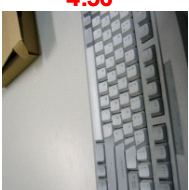
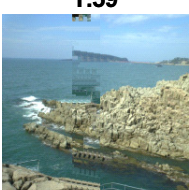
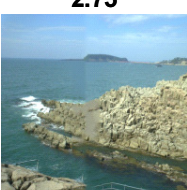
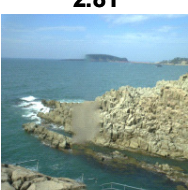
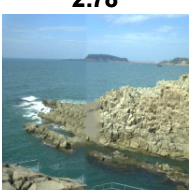
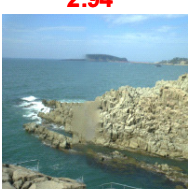
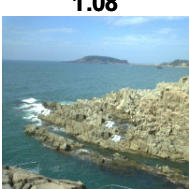
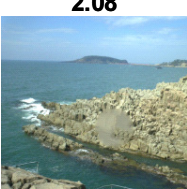
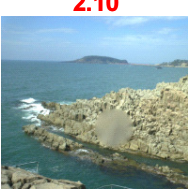
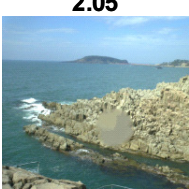
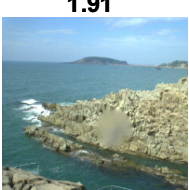
| | Method A | Method B | Method C | Method D | Method E |
|----|---|---|---|--|---|
| 78 |  1.02 |  2.83 |  2.97 |  3.56 |  3.91 |
| 79 |  3.32 |  1.91 |  1.72 |  2.02 |  1.86 |
| 80 |  1.21 |  3.86 |  4.56 |  3.91 |  4.56 |
| 81 |  1.59 |  2.75 |  2.81 |  2.78 |  2.94 |
| 82 |  1.08 |  2.08 |  2.10 |  2.05 |  1.91 |
| 83 |  3.64 |  1.81 |  1.54 |  1.70 |  1.51 |

Figure 2.9. Resultant images and scores (14/17)



Figure 2.9. Resultant images and scores (15/17)

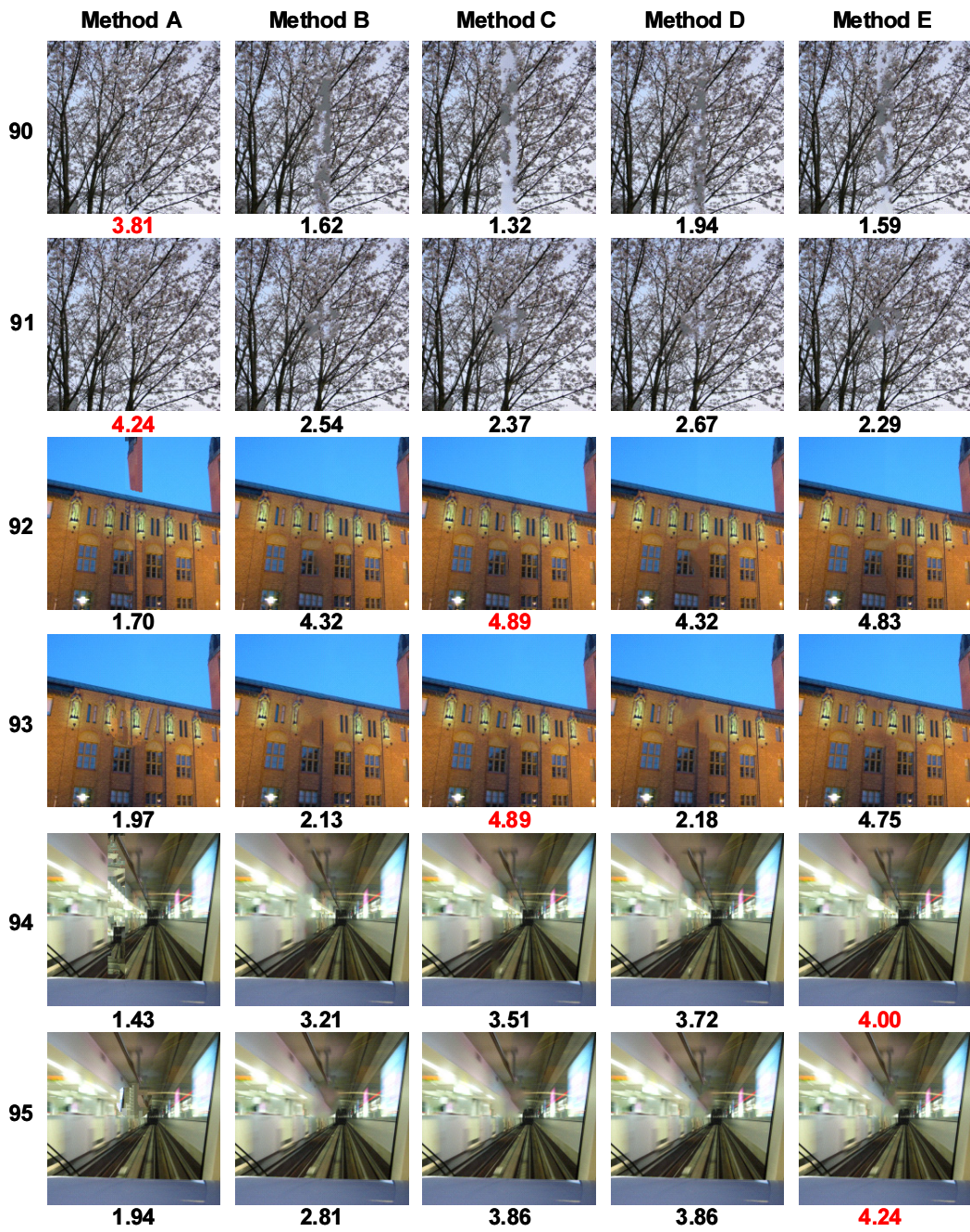


Figure 2.9. Resultant images and scores (16/17)

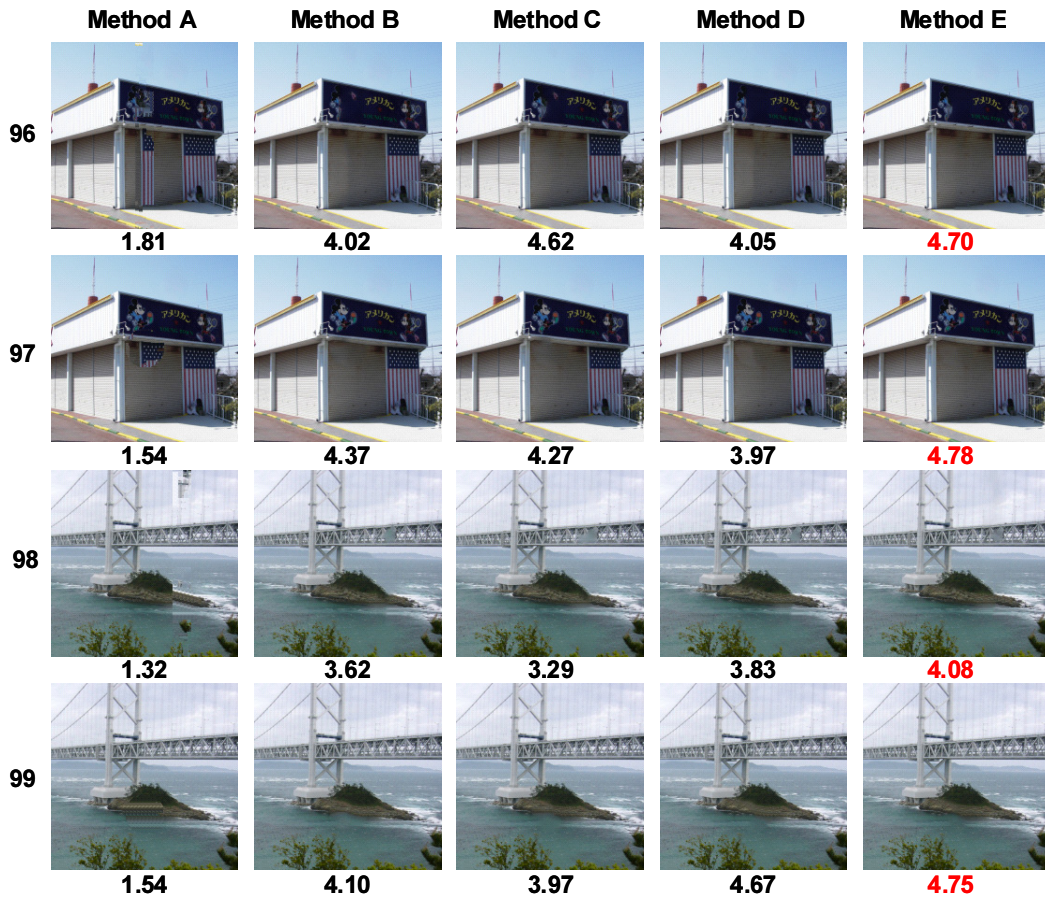


Figure 2.9. Resultant images and scores (17/17)

2.7.2 Discussion of Resultant Images

In this section, we discuss the resultant images in detail for the following cases from the result of questionnaire evaluation.

- Cases when the proposed method obtained a better score than conventional methods [CPT04, WSI07].
- Cases when the proposed method obtained a worse score than conventional methods [CPT04, WSI07].

In the following, each case is described in detail.

Cases when the proposed method obtained a better score than conventional methods [CPT04, WSI07]

From Table 2.3, the proposed method (Methods C or D or E) obtained better scores than conventional methods (Methods A and B) for 86 images. In this section, three examples (Images 09, 62 and 78) out of the 86 images for which the proposed method had a better score than conventional methods [CPT04, WSI07] are shown in Figures 2.10 to 2.12. Figures 2.13 and 2.14 show finally determined correspondences of pixels in Images 62 and 78 by Wexler’s and the proposed methods. In the following, each resultant image is discussed.

Image 09 (Figure 2.10(b)) includes relatively large difference in brightness between the left part and the right part in the image because of nonuniform illumination conditions. Figures 2.10(c), 2.10(d) and 2.10(e) are results by Criminisi’s method, Wexler’s method and the proposed method, respectively. The resultant images (Figures 2.10(c) and 2.10(d)) by the conventional methods look unnatural because sudden brightness changes appear at the seat and the seat back. On the other hand, the sudden brightness change is suppressed by the proposed method in Figure 2.10(e). This is because the allowance of the brightness changes of sample textures makes smooth changes in brightness in the missing region.

Image 62 (Figure 2.11(b)) includes the difference in texture pattern around the missing region because of the perspective effect. Figures 2.11(c), 2.11(d) and 2.11(e) are results by Criminisi’s method, Wexler’s method and the proposed

method, respectively. Figure 2.13(a) and 2.13(b) show finally determined correspondences of pixels by Wexler’s and the proposed methods. In the resultant image (Figure 2.11(c)) by the conventional method [CPT04], unnatural texture appears in the missing region. The most similar texture existed in the tree region in the left side of the image by coincidence and the texture was copied. As a result, the mixture of textures of the window and tree caused an unnatural appearance. In the resultant image (Figure 2.11(d)) by the conventional method, a part of the missing region is blurred. This is because the pixels in the missing region correspond to those in the untextured area far from the missing regions as shown in Figure 2.13(a). On the other hand, neighboring textures are preferentially selected as shown in Figure 2.13(b) by considering spatial locality of texture patterns in the proposed method. Thus the resultant image is successfully completed as shown in Figure 2.11(e).

Image 78 (Figure 2.12(b)) includes relatively large difference in brightness and texture patterns between the left side and the right side of the missing region because of nonuniform illumination conditions and the perspective projection effect. Figures 2.12(c), 2.12(d) and 2.12(e) are results by Criminisi’s method, Wexler’s method and the proposed method, respectively. Figure 2.14(a) and 2.14(b) show finally determined correspondences of pixels by Wexler’s and the proposed methods. In the resultant image (Figure 2.12(c)) by the conventional method [CPT04], the square black texture appears in the missing region by successive copy and whole the image looks unnatural. In the resultant image (Figure 2.12(d)) by the conventional method [WSI07], an unnatural image is generated due to the blurs on the textured area with black squares. This is because pixels in the area correspond to those in the untextured wall area as shown in Figure 2.14(a). In addition, there are discontinuous brightness changes on the wall and floor. On the other hand, in the resultant image (Figure 2.12(e)) by the proposed method, neighboring textures are selected for completion by using the constraint of the spatial locality of textures as shown in Figure 2.14(b). As a result, the black squares are produced. In addition, brightness changes inside the missing region becomes smoother and more natural by allowing brightness changes in sample textures.



(a) Original image

(b) Input image with a missing region



(c) Result by Criminisi's method [CPT04]

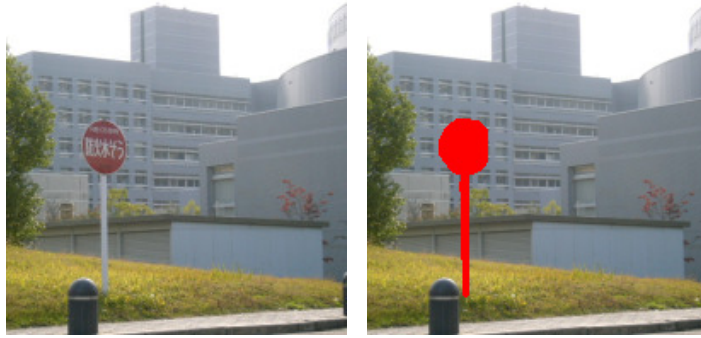


(d) Result by Wexler's method [WSI07]



(e) Result by the proposed method

Figure 2.10. Experiment for Image 09.



(a) Original image

(b) Input image with a missing region



(c) Result by Criminisi's method [CPT04]

(d) Result by Wexler's method [WSI07]

(e) Result by the proposed method

Figure 2.11. Experiment for Image 62.



(a) Original image

(b) Input image with a missing region



(c) Result by Criminisi's method [CPT04]



(d) Result by Wexler's method [WSI07]



(e) Result by the proposed method

Figure 2.12. Experiment for Image 78.

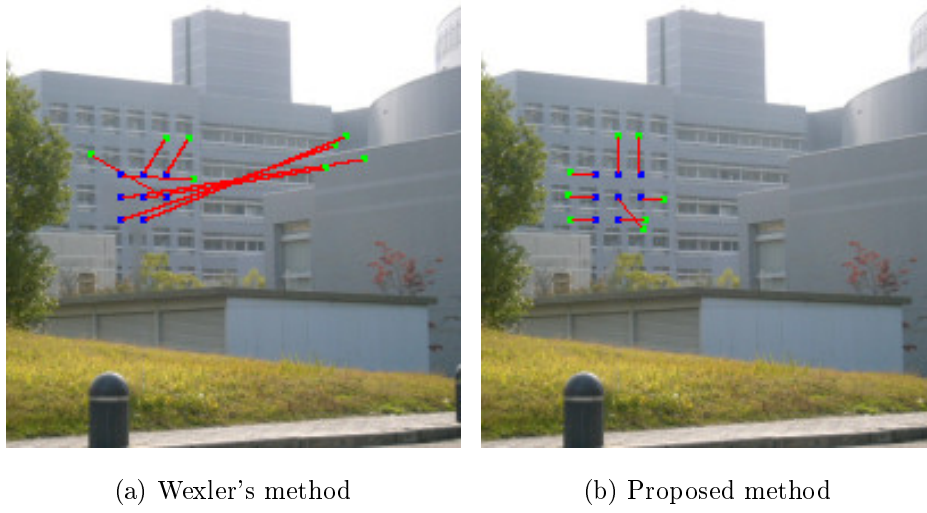


Figure 2.13. Finally determined correspondences of pixels in Image 62.

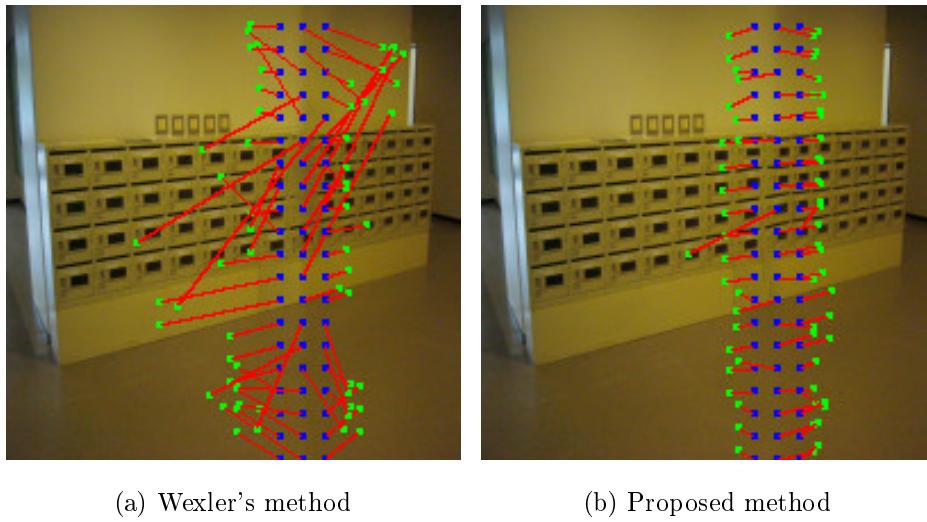


Figure 2.14. Finally determined correspondences of pixels in Image 78.

Cases when the proposed method obtained a worse score than conventional methods [CPT04, WSI07]

The proposed method obtained worse scores for 14 of the 100 images. In this section, we analyze the results and show the limitation of the proposed method. Figures 2.15 and 2.16 show the images for which Criminisi’s method [CPT04] and Wexler’s method [WSI07] obtained the better scores than the proposed method, respectively. In these figures, the number of image and the difference of the scores of the proposed method (Method E) and the conventional method (Method A or B) are written below the images.

First, we analyze the images shown in Figure 2.15. Images 75, 79, 83, 90 and 91 have high-frequency textures around the missing regions and the textures do not include cyclic patterns. For such images, it is difficult for the proposed method, in which values in the missing regions are determined by the mixture of textures, to prevent blurs. On the other hand, the conventional method [CPT04] that is based on simple copy can generate non-blurred textures. The non-blurred textures give the appearance of a natural image. Images 03 and 14 have various kinds of textures around the missing regions and do not have appropriate exemplars for connecting different kinds of textures in the data region. For such images, the missing regions also tend to be blurred by the mixture of various kinds of textures in the proposed method. On the other hand, non-blurred textures by the conventional method look better than blurred ones.

Next, we analyze the images shown in Figure 2.16. As for Image 60, all the methods obtained relatively bad scores. Image 60 has the texture of black edges in the data region but does not have the good exemplar for intersection of black edges. Therefore, it is difficult for both the proposed and conventional methods to generate the texture of intersection of the black edges. Images 04, 25, 33, 54 and 64 have similar textures not only around the missing regions but also in whole the images and the differences in scores between the proposed and conventional methods are relatively small for these images. The proposed method can obtain better results especially for images in which similar textures exist only around the missing region and brightness changes of similar textures appears. However, for images which do not have such properties, results by Wexler’s method often produce outcomes better than or as good as those of the proposed method.

From these results, we can confirm that it is difficult for the proposed method to successfully complete missing regions around which complex and noncyclic patterns or various kinds of textures exist in an image. For such images, the conventional method [CPT04] based on simple texture copy often obtains good results. In addition, the proposed method does not have an advantage against the conventional method [WSI07] for images in which similar textures exist in whole the image and the brightnesses of the same kinds of textures are uniform.

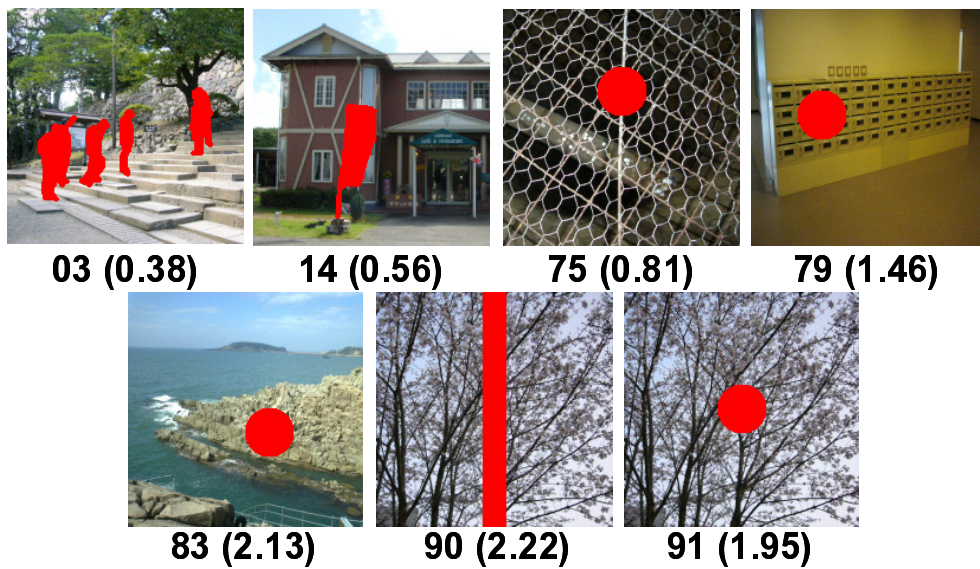


Figure 2.15. Images for which Criminisi's method obtained the best score.

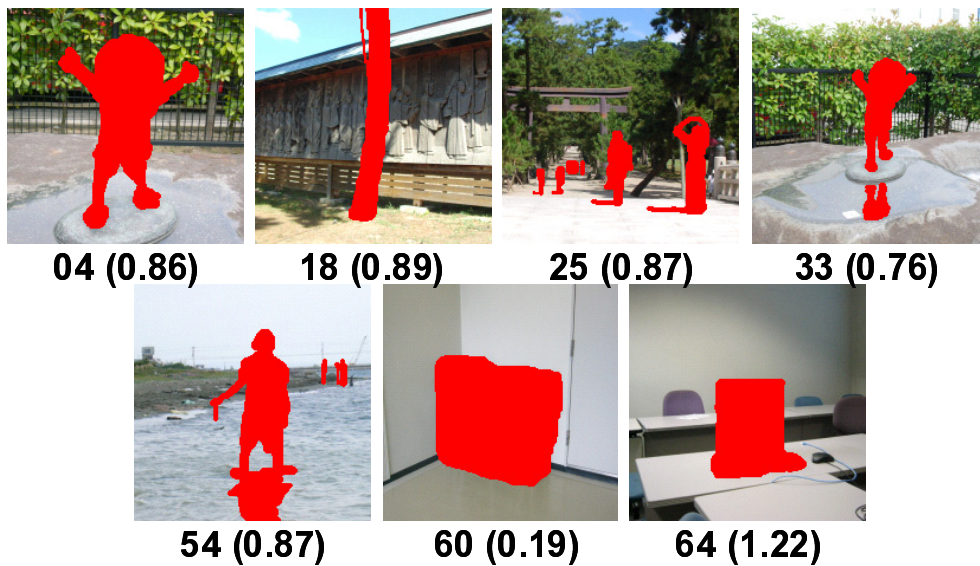


Figure 2.16. Images for which Wexler's method obtained the best score.

2.7.3 Quantitative Evaluation

In this experiment, as a quantitative measure, RMSE (Root Mean Squared Error), which is calculated based on pixel-wise differences between original and completed images, was computed for 29 images whose missing regions were specified regardless of the object regions, and we did not use the remaining 71 images because the missing regions of the 71 images are specified so as to remove certain objects. Table 2.5 shows the average RMSE. From the results, the proposed method is the best of all five methods and significant differences between the proposed (E) and conventional methods (A and B) were observed by using the t-test with a 5% significant level, respectively. From this result, we can confirm that the effectiveness of the proposed method is demonstrated quantitatively.

2.7.4 Discussion about Computational Cost

The proposed and conventional methods are compared with respect to computational time in this experiment. Table 2.6 shows the average processing time of 100 images and the average number of iterations of the conventional and proposed methods. From this table, we can confirm that the processing time of Criminisi’s method, which is based on successive copy of similar texture, is extremely fast. This is because pixel values are fixed once the most similar texture is copied in Criminisi’s method unlike the proposed and Wexler’s methods, which are based on iterative processing.

As for the comparison of Wexler’s and the proposed methods, the proposed method requires about 3.5 times as much time as the conventional method. This is partially because of cost C_α for calculating intensity modification coefficients and cost C_{sd} for calculating SD . In addition, we can confirm that the number of iterations of the proposed method is larger than that of the conventional

Table 2.5. Average RMSE from 29 images.

| Method | A | B | C | D | E |
|--------|-------|-------|-------|-------|-------|
| RMSE | 42.95 | 28.40 | 27.83 | 28.36 | 27.44 |

method from Table 2.6. We consider that increase of the texture patterns by the brightness changes of textures slows the convergence of energy.

2.7.5 Discussion about Local Minima Problem

The energy minimization in the proposed method is performed by a greedy algorithm based on the iterative processing. In general, such an algorithm does not always give the global minimum and the finally converged energy depends on initial pixel values. In this section, we discuss the energies and results with respect to different initial pixel values using Images 04, 09, 62 and 78. Here, five kinds of initial values (average pixel values of the boundary of the missing region, result by Criminisi’s method, black color, gray color and white color) are given to the missing regions.

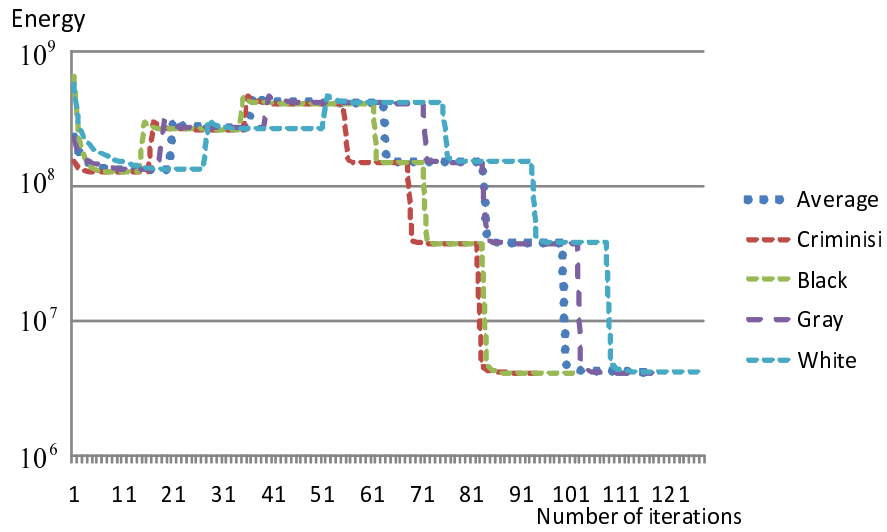
Figures 2.17(a) to 2.20(a) show the changes of energies when the five kinds of initial values are given to missing regions in Images 04, 09, 62 and 78. In these figures, the energy discontinuously changes when the level in the coarse-to-fine approach changes. Figures 2.17(b) to 2.20(b) show the resultant images and the captions below the images show the initial pixel value and the final energy. As for Images 09, 62 and 78, the initial energies are quite different from each other as shown in Figures 2.18(a), 2.19(a) and 2.20(a). Especially, initial energies when the initial colors are black and white are higher. Nevertheless, the finally converged energies are almost the same and the qualities of the resultant images using different initial values are also almost the same, as shown in Figures 2.18(b), 2.19(b) and 2.20(b). As for Image 04 whose missing region is relative large, the resultant images are quite different according to the initial values as shown in Figure 2.17(b). The energies are also relatively largely different according to

Table 2.6. Comparison with respect to computational cost.

| | Criminisi’s method [CPT04] | Wexler’s method [WSI07] | Proposed method |
|------------------------|-------------------------------|----------------------------|--------------------|
| Processing time (sec.) | 2.7 | 214.6 | 706.9 |
| Number of iterations | - | 106.6 | 119.9 |

the values. From these results, we can confirm that the result with the smallest energy cost is better than that with the largest energy cost. We can also confirm that the result with the smaller energy cost using gray color is worse than that with larger energy cost using the result of Criminisi's method. Therefore, the small difference of the energies does not always represent a difference in quality, though good quality of a completed image tends to have a small energy cost.

From these results, the energy sometimes falls into local minima and bad results are obtained by the proposed method when a missing region is relatively large. However, as the missing regions get smaller, the proposed method tends to be able to obtain almost the same or better results regardless of initial values.



(a) Change of energy



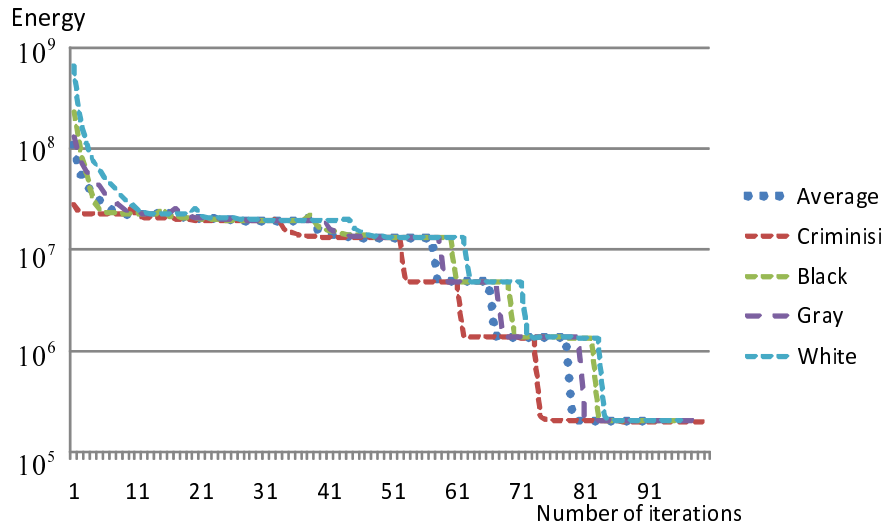
Average color (4.190×10^6) Result [CPT04] (4.106×10^6)



Black (4.058×10^6) Gray (4.092×10^6) White (4.183×10^6)

(b) Resultant images and final energies

Figure 2.17. Change of energy and results of Image 04 with respect to different initial values.



(a) Change of energy



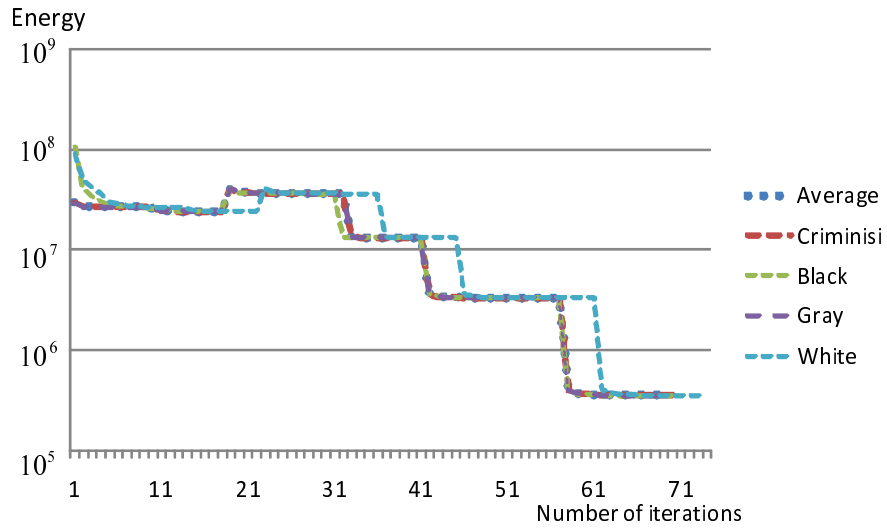
Average color (2.011×10^5) Result [CPT04] (1.987×10^5)



Black (2.016×10^5) Gray (2.010×10^5) White (2.017×10^5)

(b) Resultant images and final energies

Figure 2.18. Change of energy and results of Image 09 with respect to different initial values.



(a) Change of energy



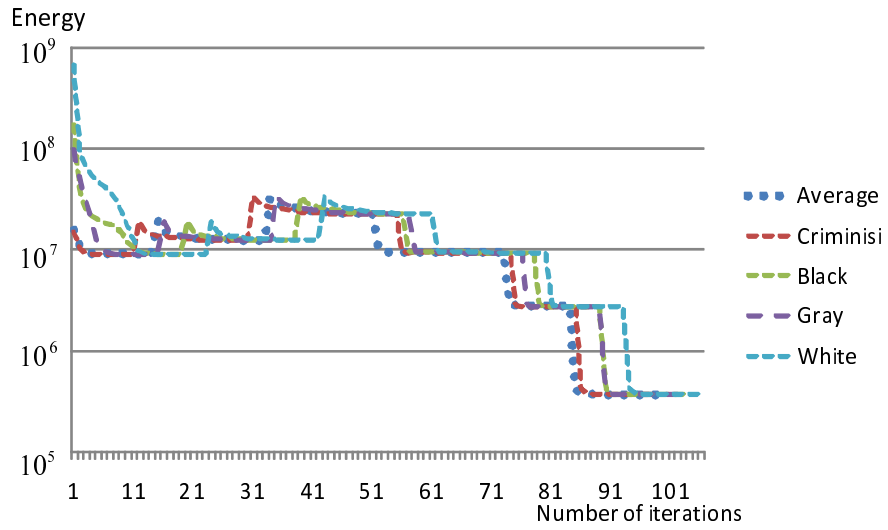
Average color (3.554×10^5) Result [CPT04] (3.554×10^5)



Black (3.554×10^5) Gray (3.556×10^5) White (3.556×10^5)

(b) Resultant images and final energies

Figure 2.19. Change of energy and results of Image 62 with respect to different initial values.



(a) Change of energy



Average color (3.660×10^5)



Result [CPT04] (3.659×10^5)



Black (3.655×10^5)



Gray (3.662×10^5)



White (3.672×10^5)

(b) Resultant images and final energies

Figure 2.20. Change of energy and results of Image 78 with respect to different initial values.

2.7.6 Discussion about Evaluation Methods

In this thesis, although qualitative evaluations using a questionnaire and quantitative evaluations using RMSE were performed in order to verify the effectiveness of the proposed method, the reliability of the evaluation methods has not been well discussed in the literature of image completion. In this section, qualitative and quantitative evaluation methods are analyzed using the evaluation results.

Reliability of qualitative evaluation

In the qualitative evaluation using a questionnaire, the evaluation using a few images and a few subjects may not be able to validate the effectiveness of image completion methods due to the bias of images and subjects. In this section, the reliability of the result when the images and subjects are decreased in a questionnaire is analyzed in a simulation. Here, the ranking of the methods decided by the average of scores given by 37 subjects in the questionnaire using 100 images were used as the ground truth.

In this thesis, as the reliability measure of evaluation results, RE (Ranking Error) is defined as:

$$RE(n, h) = \begin{cases} 0 & (\forall i, g(i, 100, 37) - g(i, n, h) = 0) \\ 1 & (otherwise), \end{cases} \quad (2.26)$$

where $g(i, n, h)$ represents the rank order of the method i determined by n images and h subjects. In this study, we compute $RE(n, h)$ with respect to random selection of n images and h subjects from 100 images and 37 subjects. $ARE(n, h)$ is the average score of $RE(n, h)$ by 10,000 times selection.

Figure 2.21 shows the relationship between ARE and the number of images and subjects. In this figure, (a) and (b) illustrate isolines of ARE for the methods A and E, and D and E, respectively. In this figure, isolines are drawn every 0.05 units.

The scores of the Methods A and E are significantly and largely different. From Figure 2.21(a), when $h = 2$, the ranking corresponds to the ground truth more than 95% of the time ($ARE = 0.05$) if $n > 3$. Therefore, for methods that generate significantly different results, the ranking is usually equivalent to the

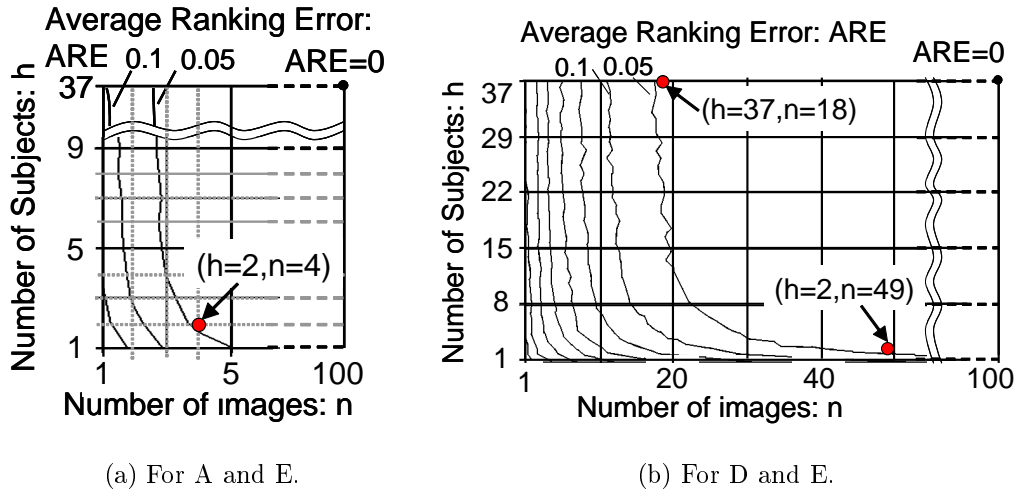


Figure 2.21. Relationships between ARE and the number of images and subjects.

ground truth even if the number of subjects and images are small. On the other hand, the scores of the Methods D and E show a significant albeit small difference. From Figure 2.21(b), in order for the ranking to correspond to the ground truth more than 95% of the time, more than 49 images are needed if $h = 2$. Even if $h = 37$, at least 18 images are needed. Therefore, if the number of subjects and images are small, the ranking may not correspond to the ground truth and the results of subjective evaluation are not reliable. Therefore, we can confirm that many images and subjects (for example, 40 images and 5 subjects) are needed to evaluate image inpainting methods persuasively.

Relationship between quantitative and qualitative evaluation

In the literature on image completion, as the quantitative evaluation, MSE (Mean Squared Error), RMSE (Root Mean Squared Error) and PSNR (Peak Signal-to-Noise Ratio), which are based on computing pixel-wise differences between original and completed images, have been sometimes used. In this experiment, the relationship between RMSE and scores given by subjects is discussed.

Figure 2.22 illustrates the distribution of the correlation coefficient between RMSE and the subjective score for 29 images: (a) is by five methods; (b) is

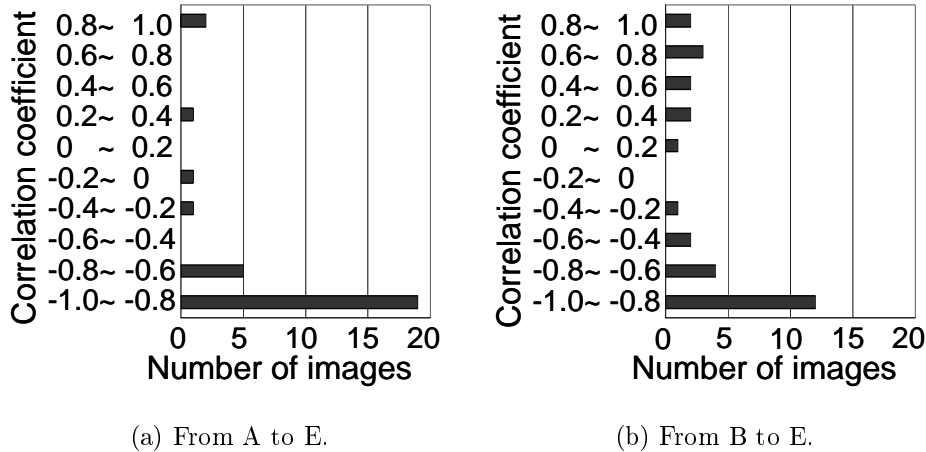


Figure 2.22. Correlation coefficient between RMSE and subjective score.

by four methods, except for Method A [CPT04]. Correlation coefficient Cc is computed as follows:

$$Cc = \frac{N_m \sum_i^{N_m} R_i S_i - \sum_i^{N_m} R_i \sum_i^{N_m} S_i}{\sqrt{N_m \sum_i^{N_m} R_i^2 - (\sum_i^{N_m} R_i)^2} \sqrt{N_m \sum_i^{N_m} S_i^2 - (\sum_i^{N_m} S_i)^2}}, \quad (2.27)$$

where N_m is the number of methods, R_i is RMSE and S_i is the average score by subjects. From Table 2.5 and Figure 2.22(a), there is a clear negative correlation between RMSE and subjective score. From this, although the evaluation result by RMSE may correspond to the ground truth, there also exist three images (Figure 2.23) where positive correlation exists. As shown in Figure 2.23, these images have very high-frequency components. It is well known that pixel correlation-based similarity measures including RMSE are sensitive to pixel phase shift for a high frequency component. However, such a pixel phase shift does not always affect the quality of images. Thus, RMSE cannot appropriately evaluate images with a high frequency component. Figure 2.22(b) illustrates the distribution of the correlation coefficient by the four methods, except for Method A, whose score is extremely low. In Figure 2.22(b), one third of the coefficients indicate positive correlation. This means that the small difference in RMSE does not always represent that of image quality. Therefore, RMSE cannot be used as an absolute criterion of image quality but can be used for a rough evaluation.

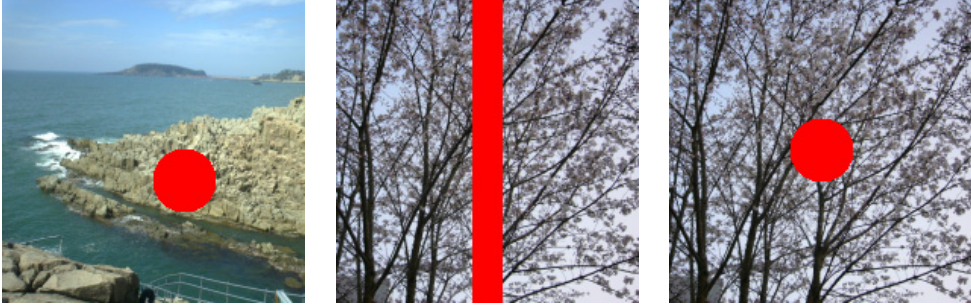


Figure 2.23. Images with inverse correlation between RMSE and subjective evaluations.

2.8. Conclusion

In this chapter, a new energy function for image completion has been proposed. In order to obtain good results for many images, two factors were considered based on the conventional energy function using pattern similarity: (1) brightness changes of sample textures was allowed by introducing an intensity modification coefficient to an energy function; and (2) spatial locality was introduced as a new constraint by adding a cost term to an energy function.

In experiments, we have demonstrated the effectiveness of these extensions by qualitative evaluation using a questionnaire and quantitative evaluation using RMSE. We have also showed the limitation that it is difficult to complete missing regions in an image in which textures around missing regions are complex and noncyclic.

In addition, we have discussed the reliability of evaluation methods for image completion. From the discussion, we confirmed that many images and subjects are required to evaluate image completion methods persuasively in a questionnaire and quantitative evaluation based on pixel-wise difference such as RMSE cannot be used as an absolute criterion but can be used for a rough evaluation.

Chapter 3

Video Completion for Generating Omnidirectional Video without Invisible Areas

3.1. Introduction

This chapter describes a video completion method for omnidirectional video. In these days, omnidirectional video is often used for various purposes and one of these is telepresence, which allows users to feel as if they are present at a remote site [ISY05, HKY07]. However, ordinary omnidirectional cameras for capturing omnidirectional video cannot capture entire fields of view due to a blind side as shown in Figure 3.1. This invisible part in the omnidirectional video decreases the realistic sensation in telepresence. In order to achieve telepresence with highly realistic sensation, it is required to plausibly fill in this missing region.

Until now, many video completion methods have been proposed for video captured with an monocular camera and have successfully completed the missing regions. However, the methods assume that the appearance of objects does not change largely between different frames. Therefore, it is difficult for the methods to successfully complete missing regions in omnidirectional video caused by the blind side of omnidirectional cameras because the appearance of the texture appropriate for a missing region in a frame changes in the different frames of a moving omnidirectional camera.



Figure 3.1. Omnidirectional panoramic image with a missing region (black region) caused by a blind side.

In this thesis, we propose a new method that completes the missing region in omnidirectional video by compensating for the change in the appearance of textures. Concretely, by assuming that the ground exists in the direction of the blind side of a moving camera and the shape of the blind side of the target scene is planar, the change in the appearance of the texture caused by the camera motion is compensated by projecting omnidirectional images onto the planar surface fitted to the 3D positions of natural feature points on the ground acquired by structure-from-motion (SFM). In addition, by using fitted planes and camera motion, data regions where appropriate textures for missing regions may exist are determined. Finally, missing regions are completed by minimizing an energy function based on pattern similarity. In this research, we employ an omnidirectional multi-camera system (OMS) that is composed of radially arranged multiple cameras as an omnidirectional camera. We assume that textures in the blind side are captured from different viewpoints.

3.2. Overview of the Method

The flow of the proposed method is given in Figure 3.2. (A) The position and posture of an OMS and 3D positions of natural feature points are estimated using SFM for omnidirectional video. (B) A plane for each frame is fitted to natural feature points near the ground by using the 3D positions of natural feature points. (C) An image sequence projected on the fitted plane is generated from the omnidirectional video. (D) The missing region in the projected image plane of each frame is completed frame-by-frame by minimizing an energy function based on the similarity between textures in the missing region and data regions. In this process, first, (D-i) data regions in which appropriate textures for missing regions may exist are specified on the projected image plane using the position and posture of the OMS and the fitted planes. Next, the energy function is minimized by repeating two processes: (D-ii) searching for similar textures; and (D-iii) parallel updating of all pixel values. (E) Omnidirectional video without invisible areas is generated by re-projecting the completed image onto the spherical panoramic video with a missing region. In the following sections, each process is described in detail.

3.3. Estimation of Extrinsic Camera Parameters and Positions of Natural Feature Points

The position and posture of an OMS and 3D positions of natural feature points are estimated by SFM for omnidirectional video [TSN04] in process (A). In this method, first, a target scene is captured with a moving OMS. Next, initial extrinsic camera parameters and 3D positions of feature points are estimated by tracking natural feature points in a video, which are detected by Harris operator. Finally, the accumulative errors of the camera parameters and the 3D positions of feature points are minimized by bundle adjustment for whole the video.

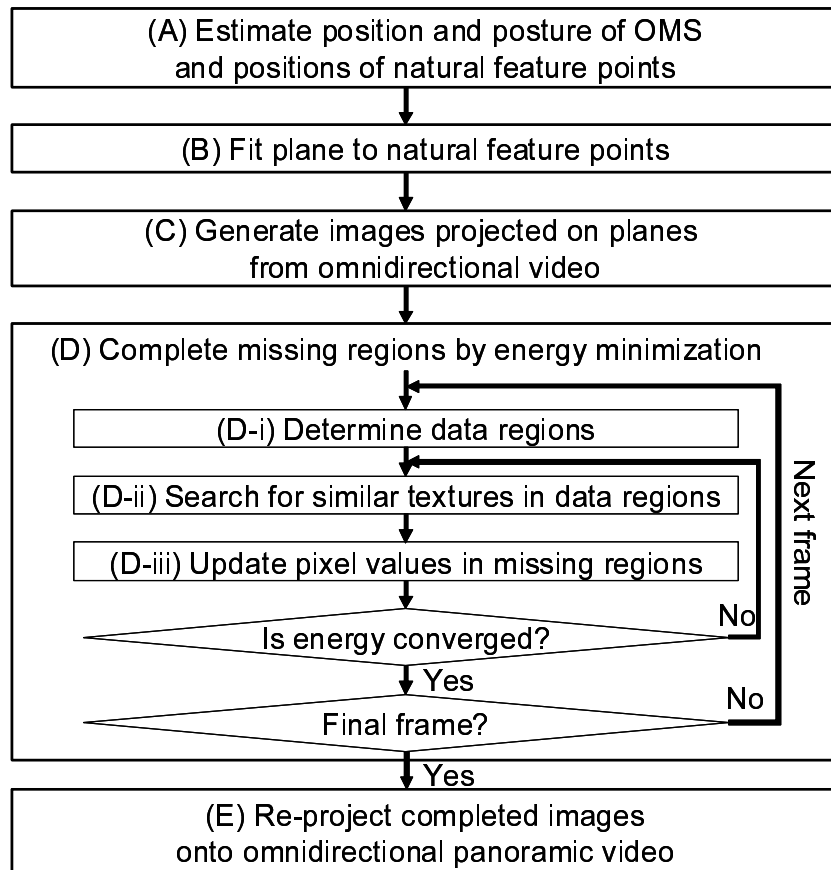


Figure 3.2. Flow diagram of the proposed video completion method for an omnidirectional video.

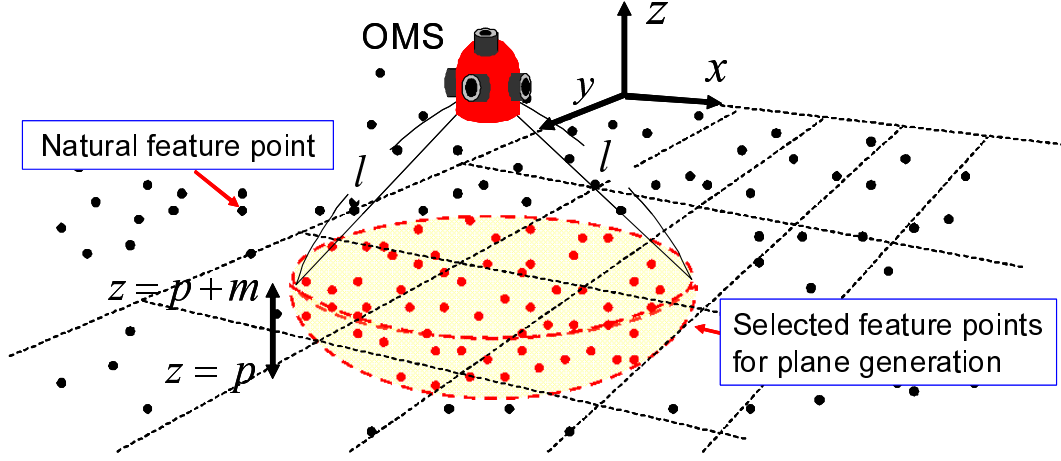


Figure 3.3. Selection of feature points around missing region.

3.4. Generation of Images Projected on Planes

In ordinary omnidirectional video, it is difficult to successfully complete missing regions using original captured textures because the appearance of textures largely changes due to camera motion. In this research, on the assumption that an omnidirectional video is captured while moving on the ground and the shape around a missing region is planar, an image sequence that includes missing regions is generated by projecting the omnidirectional video to the planes in order to compensate for the change in the appearance of textures caused by the camera motion.

First, a plane for each frame which represents the ground is fitted to natural feature points in process (B). Concretely, natural feature points for plane fitting are selected from the points obtained by SFM described in Section 3.3. Here, the points that satisfy the following conditions are selected: (i) a point exists in the spherical area whose center is a projection center of a representative camera unit of an OMS and radius is l ; and (ii) the height z of a point in the world coordinate system is $(p < z < p + m)$ (p and m are constants) as shown in Figure 3.3. Then the expression of the plane of the f -th frame that represents the ground in the

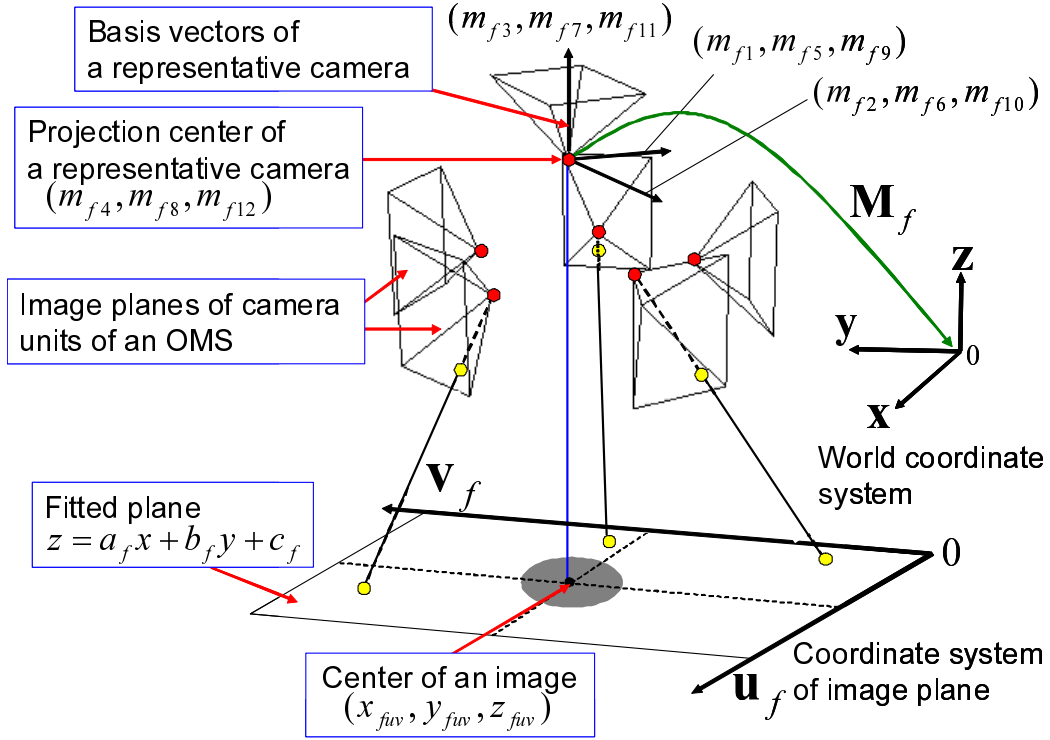


Figure 3.4. Generation of an image projected on a plane.

world coordinate system is calculated. When the expression is set as

$$z = a_f x + b_f y + c_f, \quad (3.1)$$

parameters (a_f, b_f, c_f) are determined by the least-squares method so as to minimize the following cost function L :

$$L = \sum_{i=1}^n (a_f x_i + b_f y_i + c_f - z_i)^2, \quad (3.2)$$

where (x_i, y_i, z_i) are the coordinates of a feature point and n is the number of selected feature points.

Next, an image sequence is generated by successively projecting the omnidirectional video to the estimated plane for each frame in process (C), as shown in Figure 3.4. In order for a missing region to be the center of the projected image,

an intersection point of the plane with the straight line that goes just under an OMS through the projection center of a representative camera of the OMS is set as the center of the image. Concretely, transformation matrix \mathbf{M}_f from the representative camera coordinate system of the f -th frame to the world coordinate system is expressed as follows:

$$\mathbf{M}_f = \begin{pmatrix} m_{f1} & m_{f2} & m_{f3} & m_{f4} \\ m_{f5} & m_{f6} & m_{f7} & m_{f8} \\ m_{f9} & m_{f10} & m_{f11} & m_{f12} \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (3.3)$$

Then the central coordinate of the projected image (x_f, y_f, z_f) is expressed using parameter t as follows:

$$\begin{pmatrix} x_f \\ y_f \\ z_f \end{pmatrix} = \begin{pmatrix} m_{f4} \\ m_{f8} \\ m_{f12} \end{pmatrix} + t \begin{pmatrix} m_{f3} \\ m_{f7} \\ m_{f11} \end{pmatrix}. \quad (3.4)$$

From Eqs. (3.4) and (3.1), the central coordinate is calculated by removing parameter t as follows:

$$\begin{pmatrix} x_f \\ y_f \\ z_f \end{pmatrix} = \begin{pmatrix} m_{f4} \\ m_{f8} \\ m_{f12} \end{pmatrix} + \frac{a_f m_{f4} + b_f m_{f8} + c_f - m_{f12}}{m_{f11} - a_f m_{f3} - b_f m_{f7}} \begin{pmatrix} m_{f3} \\ m_{f7} \\ m_{f11} \end{pmatrix}. \quad (3.5)$$

Here, in order to prevent the rotation of textures in the projected image plane among different frames, basis vectors $(\mathbf{u}_f, \mathbf{v}_f)$ of the image of the f -th frame in the world coordinate system are set so as to satisfy the following equation:

$$\mathbf{u}_f \cdot \mathbf{y} = 0, \quad (3.6)$$

where \mathbf{y} is one of the basis vectors of the world coordinate system. From Eqs. (3.6) and (3.1), basis vectors $(\mathbf{u}_f, \mathbf{v}_f)$ are determined as follows:

$$\mathbf{u}_f = \begin{pmatrix} \frac{1}{\sqrt{1+a_f^2}} \\ 0 \\ \frac{a_f}{\sqrt{1+a_f^2}} \end{pmatrix}, \quad \mathbf{v}_f = \begin{pmatrix} \frac{-a_f b_f}{\sqrt{(1+a_f^2)(1+a_f^2+b_f^2)}} \\ \frac{1+a_f^2}{\sqrt{(1+a_f^2)(1+a_f^2+b_f^2)}} \\ \frac{b_f}{\sqrt{(1+a_f^2)(1+a_f^2+b_f^2)}} \end{pmatrix}. \quad (3.7)$$

From the center position (Eq. (3.5)) and the basis vectors (Eq. (3.7)), the relationships between coordinates (u_f, v_f) on the projected image and 3D positions $(x_{fuv}, y_{fuv}, z_{fuv})$ in the world coordinate system are expressed as follows:

$$\begin{pmatrix} x_{fuv} \\ y_{fuv} \\ z_{fuv} \end{pmatrix} = \begin{pmatrix} x_f \\ y_f \\ z_f \end{pmatrix} + r(u_f - \frac{U}{2})\mathbf{u}_f + r(v_f - \frac{V}{2})\mathbf{v}_f, \quad (3.8)$$

where the size of a pixel is $r \times r$ and the resolution of a projected image is $U \times V$. Note that the scale change of textures in the projected image are also prevented by fixing the size of a pixel. Finally, pixel values in the projected image are determined by re-projecting the 3D position of each pixel on the image plane of the OMS camera unit.

3.5. Video Completion by Minimizing Energy Function

In process (D), we basically apply the image completion method for still images described in Chapter 2 to video frames. In this completion for video frames, in order to preserve the temporal continuity, data regions in which the appropriate texture for a missing region exist are determined pixel-by-pixel from multiple frames using geometric information. The missing region in an image projected on the plane of each frame is then completed frame-by-frame by minimizing the energy function based on the similarity of textures between the missing and data regions. In the following, the definition of the energy function, a method determining data regions and a method to minimize the energy function are described in detail.

3.5.1 Definition of Energy Function Based on Pattern Similarity

In this study, we employ pattern similarity allowing brightness changes based on SSD (Sum of Squared Differences) for an energy function. The similarity allowing brightness changes generates natural structure and brightness of textures

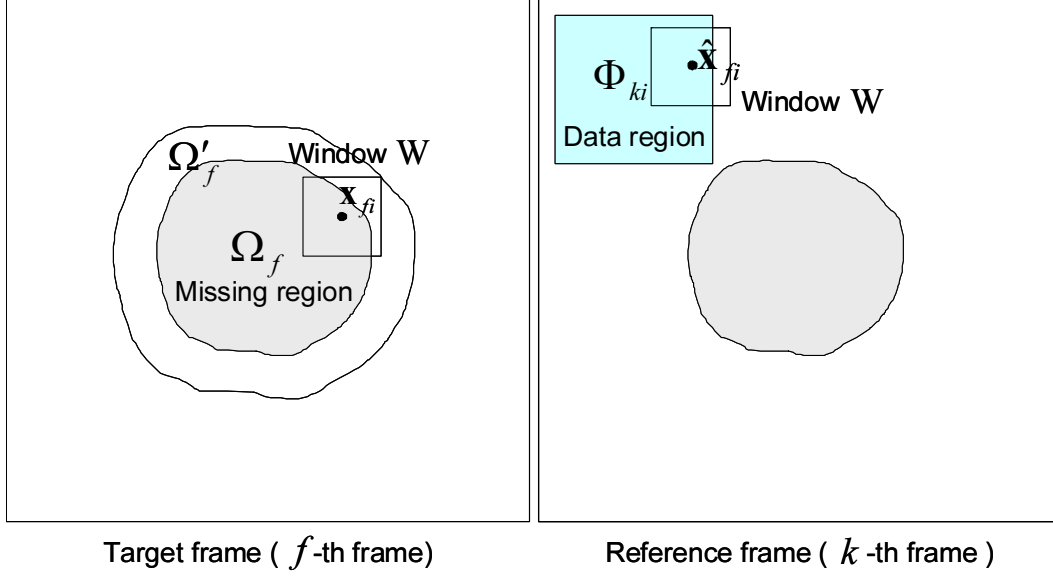


Figure 3.5. Missing and data regions in projected images for completion process.

in missing regions. As shown in Figure 3.5, a missing region in the projected image of the target (f -th) frame is completed using an energy function based on the similarity of textures between in region Ω'_f , including missing region Ω_f , in the f -th frame and data region Φ_{ki} in the reference (k -th) frame ($k \neq f$). Here, Ω'_f is the expanded area of missing region Ω_f in which there is a central pixel, \mathbf{x}_{fi} , of a square window W overlapping region Ω_f . Each data region Φ_{ki} corresponding to each pixel \mathbf{x}_{fi} in the f -th frame is individually determined in Section 3.5.2. Energy function E_{vc} is defined as the weighted sum of SSD between the textures around pixel \mathbf{x}_{fi} in region Ω'_f and $\hat{\mathbf{x}}_{fi}$ in data region Φ_{ki} .

$$E_{vc} = \sum_{\mathbf{x}_{fi} \in \Omega'_f} w_{\mathbf{x}_{fi}} SSD(\mathbf{x}_{fi}, \hat{\mathbf{x}}_{fi}), \quad (3.9)$$

where $w_{\mathbf{x}_{fi}}$ is the weight for pixel \mathbf{x}_{fi} and is set as 1 if \mathbf{x}_{fi} is inside region $\Omega'_f \cap \overline{\Omega_f}$ because pixel values in this region are fixed; otherwise $w_{\mathbf{x}_{fi}} = g^{-d}$ (d is the distance from the boundary of Ω_f and g is a constant), because pixel values around the boundary have higher confidence than those in the center of the missing region.

$SSD(\mathbf{x}_{fi}, \hat{\mathbf{x}}_{fi})$, which represents the similarity of textures around pixel \mathbf{x}_{fi} and $\hat{\mathbf{x}}_{fi}$, is defined as follows:

$$SSD(\mathbf{x}_{fi}, \hat{\mathbf{x}}_{fi}) = \sum_{\mathbf{q} \in W} \{I(\mathbf{x}_{fi} + \mathbf{q}) - \alpha_{\mathbf{x}_{fi}\hat{\mathbf{x}}_{fi}} I(\hat{\mathbf{x}}_{fi} + \mathbf{q})\}^2, \quad (3.10)$$

where $I(\mathbf{x})$ represents the pixel value of pixel \mathbf{x} . $\alpha_{\mathbf{x}_{fi}\hat{\mathbf{x}}_{fi}}$ is the intensity modification coefficient. Note that textures around a missing region may change due to the reflection of the light on the ground and the shade of the camera and operator. Therefore, by using this coefficient, the brightness of textures in data regions is adjusted to that in the missing region. In this research, $\alpha_{\mathbf{x}_{fi}\hat{\mathbf{x}}_{fi}}$ is defined as the ratio of average pixel values around pixels \mathbf{x}_{fi} and $\hat{\mathbf{x}}_{fi}$ as follows:

$$\alpha_{\mathbf{x}_{fi}\hat{\mathbf{x}}_{fi}} = \frac{\sqrt{\sum_{\mathbf{q} \in W} I(\mathbf{x}_{fi} + \mathbf{q})^2}}{\sqrt{\sum_{\mathbf{q} \in W} I(\hat{\mathbf{x}}_{fi} + \mathbf{q})^2}}. \quad (3.11)$$

In the defined energy function, the parameters are pixel value $I(\mathbf{x}_{fi})$ of each pixel in missing region Ω_f and position $\hat{\mathbf{x}}_{fi}$ corresponding to pixel \mathbf{x}_{fi} in region Ω'_f . These parameters are calculated so as to minimize the energy function.

3.5.2 Determination of Data Region

In process (D-i), data regions in which the appropriate texture for a missing region in a target frame exist are determined pixel-by-pixel using the position and posture of a moving OMS estimated in process (A) and the planes generated in process (B). In this research, appropriate textures for the missing region in the target frame are assumed to be captured from different viewpoints in an omnidirectional video. Additionally, the parameters of the plane and the position and posture of the OMS in each frame are known. Therefore, regions in which the appropriate texture pattern for the missing region exists can be determined in multiple frames by using the geometric relationships of a moving OMS and the ground. Here, in order to use textures with high quality as exemplars for completion, one frame is selected from the multiple frames considering the resolution of the texture pattern and the difference of frame numbers. In addition, as for frames other than the initial frame, the positions of selected exemplars for

completion of the previous frame are considered in order to preserve the temporal continuity of generated textures. In the following, we describe the way to determine a data region (a region and a frame) corresponding to each pixel \mathbf{x}_{fi} in region Ω'_f in the target (f -th) frame that is used for the energy minimization process. Methods to determine a data region for an initial frame and for frames except for the initial frame are successively described.

Determination of Data Region for Initial Frame

Data regions for an initial frame are determined pixel-by-pixel considering the resolution of the textures and the difference of frame numbers between the initial and reference frames. A region in a frame is determined as a data region. Concretely, first, the 3D coordinate of pixel \mathbf{x}_{fi} in region Ω'_f in the target (f -th) projected image is re-projected on the image plane of a camera unit of the OMS in the reference (k -th) frame. Then the pixel coordinate $p_k(\mathbf{x}_{fi})$, which is the intersection point of the k -th projected image with the straight line that goes through the re-projected pixel on the image plane of the camera unit and pixel \mathbf{x}_{fi} on the f -th projected image, is calculated as shown in Figure 3.6. In the same way, pixel coordinate $p_k(\mathbf{x}_{fi})$ in each frame k corresponding to pixel \mathbf{x}_{fi} is calculated.

Next, one frame is selected by considering the position of $p_k(\mathbf{x}_{fi})$ in a projected image and the difference of frame numbers between the target and reference frames. In projected images, the resolution of texture becomes lower the farther a pixel is from the center of the image because textures of objects remote from the camera become small in input images of an OMS. Thus textures near the center of the image should be used as exemplars for completion in order to prevent the generation of blurred textures. In addition, it is highly possible that temporally close frames have similar brightness of textures. Therefore, the appropriate frame $s(\mathbf{x}_{fi})$ is selected from candidate frames $\mathbf{K} = (k_1, \dots, k_n)$ by the following equation:

$$s(\mathbf{x}_{fi}) = \underset{k \in \mathbf{K}}{\operatorname{argmin}} (\| p_k(\mathbf{x}_{fi}) - \mathbf{x}_{center} \| + \lambda |k - f|), \quad (3.12)$$

where candidate frames \mathbf{K} are picked up so that the texture pattern of exemplar whose center is $p_k(\mathbf{x}_{fi})$ does not include the missing region. \mathbf{x}_{center} is the central pixel in the k -th planar projected image and λ is the weight for the difference of

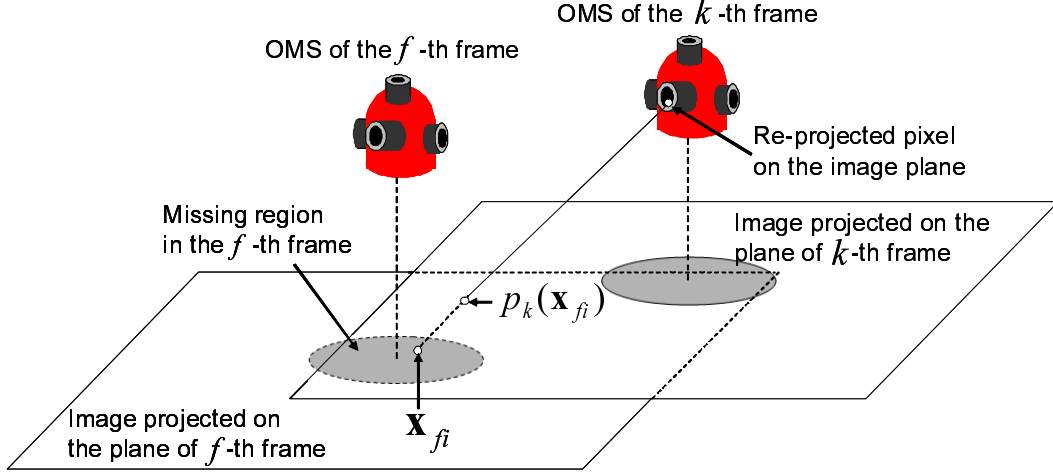


Figure 3.6. Projection to the other frame.

frame numbers. Finally, the fixed square area whose center is pixel $p_{s(\mathbf{x}_{fi})}(\mathbf{x}_{fi})$ is set as data region $\Phi_{s(\mathbf{x}_{fi})i}$. In the same way, each data region $\Phi_{s(\mathbf{x}_{fi})i}$ corresponding to each pixel \mathbf{x}_{fi} in region Ω'_f is individually determined.

Determination of Data Region for Frames Except for Initial Frame

Data regions for frames except for the initial frame are determined considering not only the resolution of textures and the difference of frame numbers but also the positions of finally selected exemplars for the completion of the previous frame in order to preserve the temporal continuity of generated textures. Here, all pixels in region Ω'_f in the target (f -th) frame are divided into two groups based on whether or not the pixel can use the position of the selected exemplar for completion of the previous frame.

Concretely, first, the relationship of the positions of corresponding pixels in regions Ω'_f and Ω'_{f-1} of the target (f -th) frame and the previous ($(f-1)$ -th) frame is determined in the same way as shown in Figure 3.6. Here, as shown in Figure 3.7, the pixel in the previous frame corresponding to \mathbf{x}_{fi} in the target frame can be expressed as $p_{(f-1)}(\mathbf{x}_{fi}) (= \mathbf{x}_{(f-1)j})$. Next, frame $s(\mathbf{x}_{fi})$ corresponding to \mathbf{x}_{fi} in region Ω'_f of the target frame is determined by Eq. (3.12). Then pixels in region

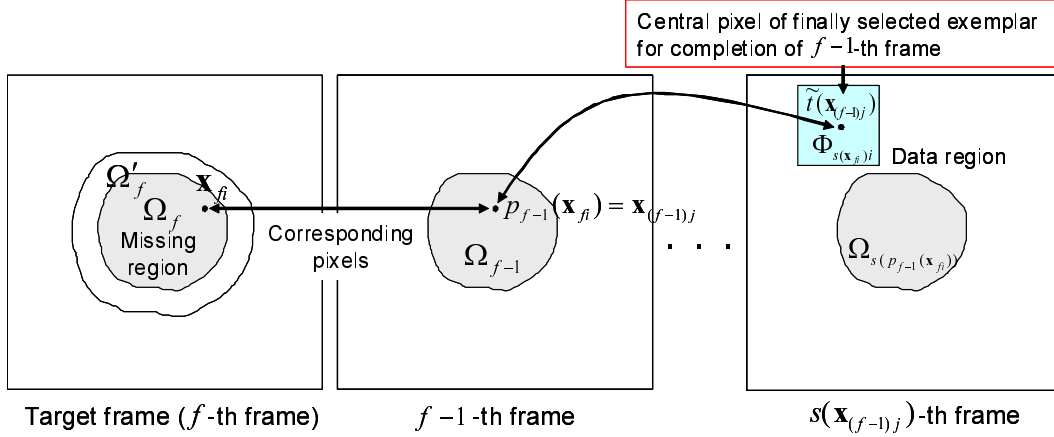


Figure 3.7. Projection to the other frame.

Ω'_f are divided into two groups. When the reference frames of \mathbf{x}_{fi} and $\mathbf{x}_{(f-1)j}$ are different ($s(\mathbf{x}_{fi}) \neq s(\mathbf{x}_{(f-1)j})$), pixel \mathbf{x}_{fi} belong to group (a); otherwise, pixel \mathbf{x}_{fi} belong to group (b).

As for pixel \mathbf{x}_{fi} in group (a), data regions are set in the same way as the initial frame because the positions of the selected exemplars for completion of the previous frame can not be used. As for pixel \mathbf{x}_{fi} in group (b), the positions of the selected exemplars can be used. Therefore, the fixed size of square area whose center is pixel $\tilde{t}(\mathbf{x}_{(f-1)j})$, which is a central pixel of the finally selected exemplar for the completion of the previous frame, is set as data region $\Phi_{s(\mathbf{x}_{fi})i}$, as shown in Figure 3.7. In addition, we make the size of the data region smaller than that for pixels in group (a). This makes it possible to not only preserve the temporal continuity of textures but also decrease the computational cost.

3.5.3 Energy Minimization

Energy function E_{vc} in Eq. (3.9) is minimized by using a framework of greedy algorithm. In our definition of energy E_{vc} , the energy for each pixel can be treated independently if pattern pairs $(\mathbf{x}_{fi}, \hat{\mathbf{x}}_{fi})$ can be fixed and the change of coefficient $\alpha_{\mathbf{x}_{fi}, \hat{\mathbf{x}}_{fi}}$ in the iterative process of energy minimization is much smaller than the change of pixel values in the missing region. Thus we repeat the following two

processes until the energy converges: (D-ii) searching for the most similar pattern keeping pixel values fixed; and (D-iii) parallel updating of all pixel values keeping pattern pairs fixed.

(D-ii) Searching for Similar Texture Pattern

In process (D-ii), data region Φ_{ki} ($k = s(\mathbf{x}_{fi})$) is searched for position $t(\mathbf{x}_{fi})$ of the most similar pattern corresponding to pixel \mathbf{x}_{fi} while keeping pixel values $I(\mathbf{x}_{fi})$ fixed. $t(\mathbf{x}_{fi})$ is determined pixel-by-pixel in parallel as follows:

$$t(\mathbf{x}_{fi}) = \hat{\mathbf{x}}_{fi} = \underset{\mathbf{x} \in \Phi_{ki}}{\operatorname{argmin}}(SSD(\mathbf{x}_{fi}, \mathbf{x})). \quad (3.13)$$

Here, SSDA (Sequential similarity detection algorithm) [BS72] is used for reducing the computational cost.

(D-iii) Parallel Updating of Pixel Values

In process (D-iii), all pixel values $I(\mathbf{x}_{fi})$ are updated in parallel so as to minimize the energy keeping the similar pattern pairs fixed. In the following, the method for calculating pixel values $I(\mathbf{x}_{fi})$ is described. First, energy E_{vc} is resolved into element energy $E(\mathbf{x}_{fi})$ for each pixel \mathbf{x}_{fi} in missing region Ω_f . Element energy $E_{vc}(\mathbf{x}_{fi})$ can be expressed in terms of the pixel values of \mathbf{x}_{fi} and $f(\mathbf{x}_{fi} + \mathbf{q}) - \mathbf{q}$, coefficient α as follows:

$$E_{vc}(\mathbf{x}_{fi}) = \sum_{\mathbf{q} \in W} w_{(\mathbf{x}_{fi} + \mathbf{q})} \{I(\mathbf{x}_{fi}) - \alpha_{(\mathbf{x}_{fi} + \mathbf{q})t(\mathbf{x}_{fi} + \mathbf{q})} I(t(\mathbf{x}_{fi} + \mathbf{q}) - \mathbf{q})\}^2. \quad (3.14)$$

The relationship between energy E_{vc} and element energy $E_{vc}(\mathbf{x}_{fi})$ for each pixel can be written as follows:

$$E_{vc} = \sum_{\mathbf{x}_{fi} \in \Omega} E(\mathbf{x}_{fi}) + C. \quad (3.15)$$

C is the energy of pixels in region $\Omega'_f \cap \overline{\Omega_f}$, and is treated as a constant because pixel values in the region and all the pattern pairs are fixed in process (D-iii). Therefore, by minimizing element energy $E_{vc}(\mathbf{x}_{fi})$ respectively, total energy E_{vc} can be minimized. Here, if it is assumed that the change of $\alpha_{\mathbf{x}_{fi}t(\mathbf{x}_{fi})}$ is much

smaller than that of pixel value $I(\mathbf{x}_{fi})$, by differentiating $E_{vc}(\mathbf{x}_{fi})$ with respect to $I(\mathbf{x}_{fi})$, each pixel value $I(\mathbf{x}_{fi})$ in missing region Ω_f can be calculated in parallel as follows:

$$I(\mathbf{x}_{fi}) = \frac{\sum_{\mathbf{q} \in W} w(\mathbf{x}_{fi} + \mathbf{q}) \alpha(\mathbf{x}_{fi} + \mathbf{q}) t(\mathbf{x}_{fi} + \mathbf{q}) I(t(\mathbf{x}_{fi} + \mathbf{q}) - \mathbf{q})}{\sum_{\mathbf{q} \in W} w(\mathbf{x}_{fi} + \mathbf{q})}. \quad (3.16)$$

3.5.4 Coarse-to-fine Approach

In order to avoid local minima efficiently, a coarse-to-fine approach is also employed for energy minimization. Concretely, an image pyramid is generated and energy minimization processes (D-ii) and (D-iii) are repeated from higher-level to lower-level layers successively. Good initial values are given to the lower layer by projecting results from the higher layer. In addition, the correspondences of pixels are inherited and new data regions are set so that the corresponding pixels are the center of the new data regions. This makes it possible to decrease computational cost and avoid local minima.

3.6. Re-projection of Images Projected on Planes to Panoramic Images

In process (E), an omnidirectional video without invisible areas is generated by re-projecting the projected images completed in process (D) onto spherical panoramic images with a missing region. Concretely, the coordinate of the intersection of the plane with the straight line that goes through the projection center of a camera unit and each pixel in the missing region in the sphere is calculated. Then the pixel value of the calculated coordinate in the projected image is copied to the corresponding pixel in the spherical panoramic image.

3.7. Experiments

In this section, the effectiveness of the proposed method is demonstrated using two image sequences by completing missing regions caused by a blind side of an OMS and generating omnidirectional videos without invisible areas. In experiments, we used a PC (CPU: Xeon 3.0GHz, Memory: 8GB) and Ladybug [Poi] as an OMS as shown in Figure 3.8. Ladybug has radially located six camera units and their positions and postures are fixed. Each camera unit can acquire 768×1024 pixels resolution image at 15 fps. Parameters in the experiments were empirically determined as shown in Table 3.1. For completion, a missing region in each image projected on a plane was determined by manually specifying the unneeded regions (equipment of Ladybug and a shadow of an operator of an OMS) in six images of the first frame and a blind region in the projected image is also specified as the missing region.

In the following, first, we describe the acquisition of omnidirectional video and the estimation results of the extrinsic camera parameters. Then, the experiment of completion for images projected on images and a prototype telepresence system using the omnidirectional video without invisible areas is presented.

3.7.1 Acquisition of Input Information

In this experiment, first, two image sequences were captured while a person on which Ladybug was mounted was walking. Each video consists of 301 frames of



Figure 3.8. Omnidirectional multi-camera system "Ladybug".

Table 3.1. Parameters in experiments.

| For generation of panoramic and projected images | | | |
|--|--|-------|-------|
| Resolution of panoramic image | 2048 × 1024 | | |
| Resolution of projected image | 1200 × 1200 | | |
| Range for selecting feature points | -1000 (mm) < z < height of camera $l = 6000$ (mm) | | |
| Size of a pixel | 10 (mm) × 10 (mm) | | |
| Weight λ | 2.0 | | |
| For energy minimization | | | |
| Multi-scale level | 1 | 2 | 3 |
| Image size | 1/4 | 1/2 | 1 |
| g in weight w | 1.1 | | |
| Window size | 51 × 51 | | |
| Data region (case (a)) | 11 × 11 | 5 × 5 | 5 × 5 |
| Data region (case (b)) | 3 × 3 | 3 × 3 | 3 × 3 |

an omnidirectional video (1806 images). Figures 3.9 and 3.10 show images of the 1st frame captured by Ladybug. In scene (1) of Figure 3.9, the ground is almost planar through all the frames. In scene (2) of Figure 3.10, the ground is slightly rugged due to various stones and there are steps near missing regions in the beginning and the end of the scene. Next, the extrinsic camera parameters and the positions of natural feature points were estimated by Structure-from-Motion described in Section 3.3 as shown in Figures 3.11 and 3.12.

3.7.2 Completion for Images Projected on Planes

In this section, first, we generated the images projected on planes by the method described in Section 3.4. The nine frames out of 300 frames of scenes (1) and (2) are shown in Figures 3.13 and 3.14. Round black regions in the images are missing regions caused by the blind side of Ladybug. As shown in the figures of scene (1), textures of tiles on the ground are uniform regardless of the position of pixels and textures of the same objects do not rotate in each frame. As a result, we can confirm that appropriate projected images used for completion were generated. As shown in the figures of scene (2), textures of stones are also the same and do not rotate in each frame. However, as for scene (2), the brightness of the stones differs between the right and left sides of the missing regions and the brightness of the same stones greatly changes between different frames. In this scene, strong sunlight comes from the right of the images. Therefore, the brightness of objects changes due to the reflection of the sunlight on the stones according to the position of the OMS.

Next, a missing region in each projected image was completed. Figures 3.15(d) and 3.16(d) show a close-up of the completed results for the projected image of the 11th frame of scene (1) and the 71st frame of scene (2), respectively. Figures 3.15(a) and 3.16(a) show the target frame in which the missing region (red region) was specified, and Figures 3.15(b) and 3.16(b) show the central pixels of the data regions in the reference frames corresponding to pixel (600,600) in the target frames. Figures 3.15(c) and 3.16(c) show the close-up of the results by copying values of central pixels in data regions without the energy minimization process described in Section 3.5. As for Figure 3.15(c), the geometrical and optical disconnection of textures in the boundary of the missing region appears.



Figure 3.9. 1st frame of input image sequence captured by six camera units (scene (1)).



Figure 3.10. 1st frame of input image sequence captured by six camera units (scene (2)).

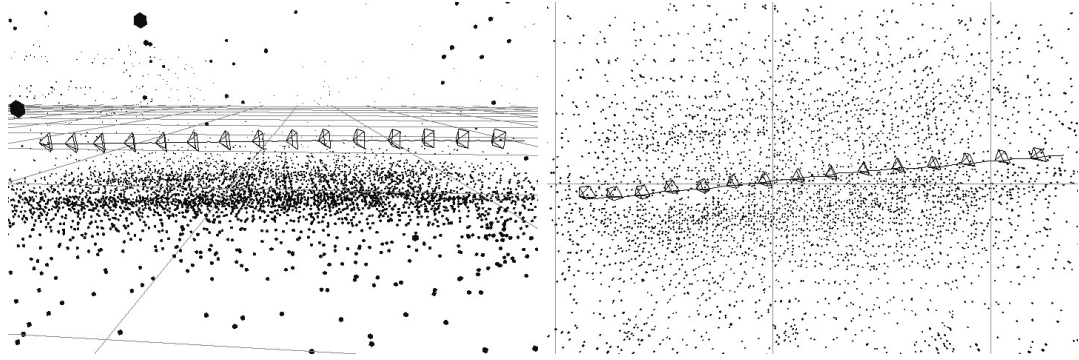


Figure 3.11. Estimated positions and postures of an OMS and positions of natural feature points (scene (1)).

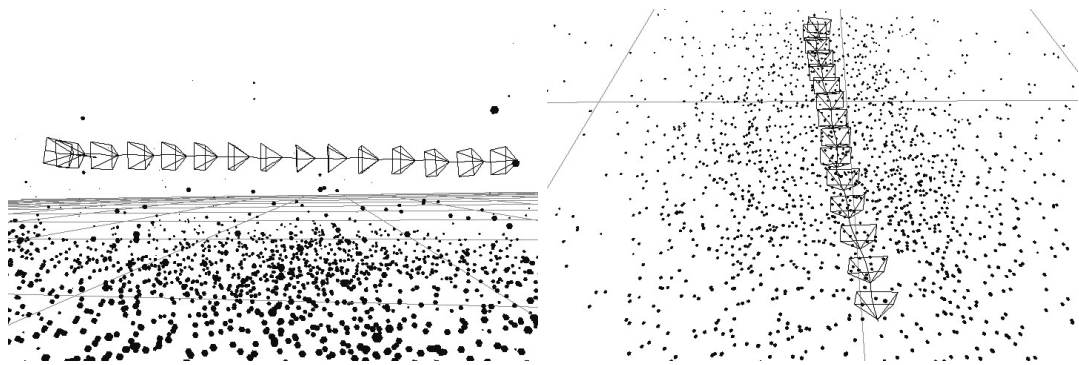


Figure 3.12. Estimated positions and postures of an OMS and positions of natural feature points (scene (2)).

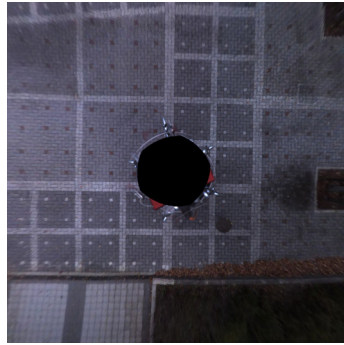
We consider this is because of the errors of the estimation of camera parameters by SFM and errors of plane fitting. As for Figure 3.16(c), we can also confirm that the geometrical disconnection of textures in the middle of the missing region. In this scene, textures in the left side of the missing region are copied from earlier frames and those of the right side are copied from later frames. Therefore, the large disconnection appears in the region where the frame numbers of data regions greatly changes. On the other hand, in the resultant images (d) in Figures 3.15 and 3.16, textures are continuously connected on the boundary and the middle of the missing region. As a result, natural textures are generated in the missing regions. From this result, we can confirm that the energy minimization process using pattern similarity is effective for generating natural textures.

Figures 3.17 and 3.18 and Figures 3.19 and 3.20 show projected images with a missing region of successive frames (12th to 20th frames of scene (1)) and (72nd to 80th frames of scene (2)), respectively. From these figures, the missing region in each frame was successfully completed. In addition, textures in the missing region change smoothly between successive frames and plausible video is generated. However, as for scene (2), unnatural changes of textures appeared from 298th to 301st frames as shown in Figure 3.21. The completed textures are gradually distorted through these frames. We consider that correspondences of pixels were not accurate because the angle of the plane of each frame became relatively greatly different from the ground truth due to steps near the missing region. In order to overcome the problems, we should consider that the range for selecting natural features is adaptively determined. In addition, to obtain good results for more complex scenes, detailed surfaces should be fitted to natural features in future work.

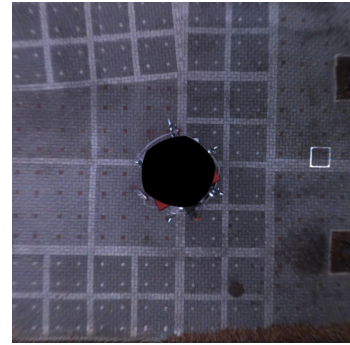
Finally, we discuss the processing time. As for scene (1), it took 722 seconds to complete the initial frame (determine data regions and minimize the energy function) and 513 seconds on averages to complete the frames except for the initial frame. As for scene (2), it took 776 seconds to complete the initial frame and 523 seconds on averages to complete the frames except for the initial frame. In frames except the initial frame, completion results for the previous frame are used to determine data regions and the size of the data regions is made smaller. Therefore, the computational cost is decreased.



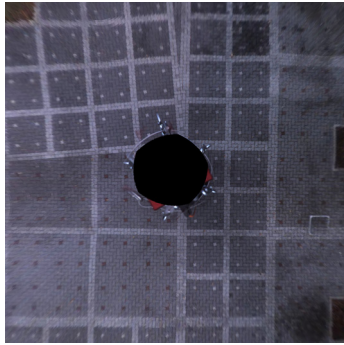
1st frame



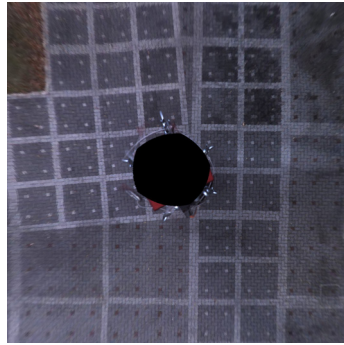
41st frame



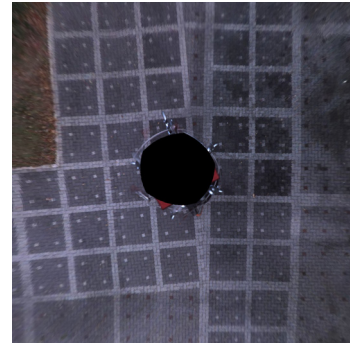
81st frame



121st frame



161st frame



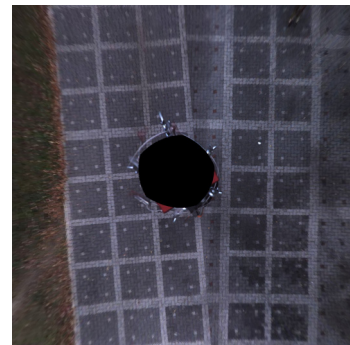
201st frame



241st frame



281st frame



301st frame

Figure 3.13. Images projected on planes (scene (1)).

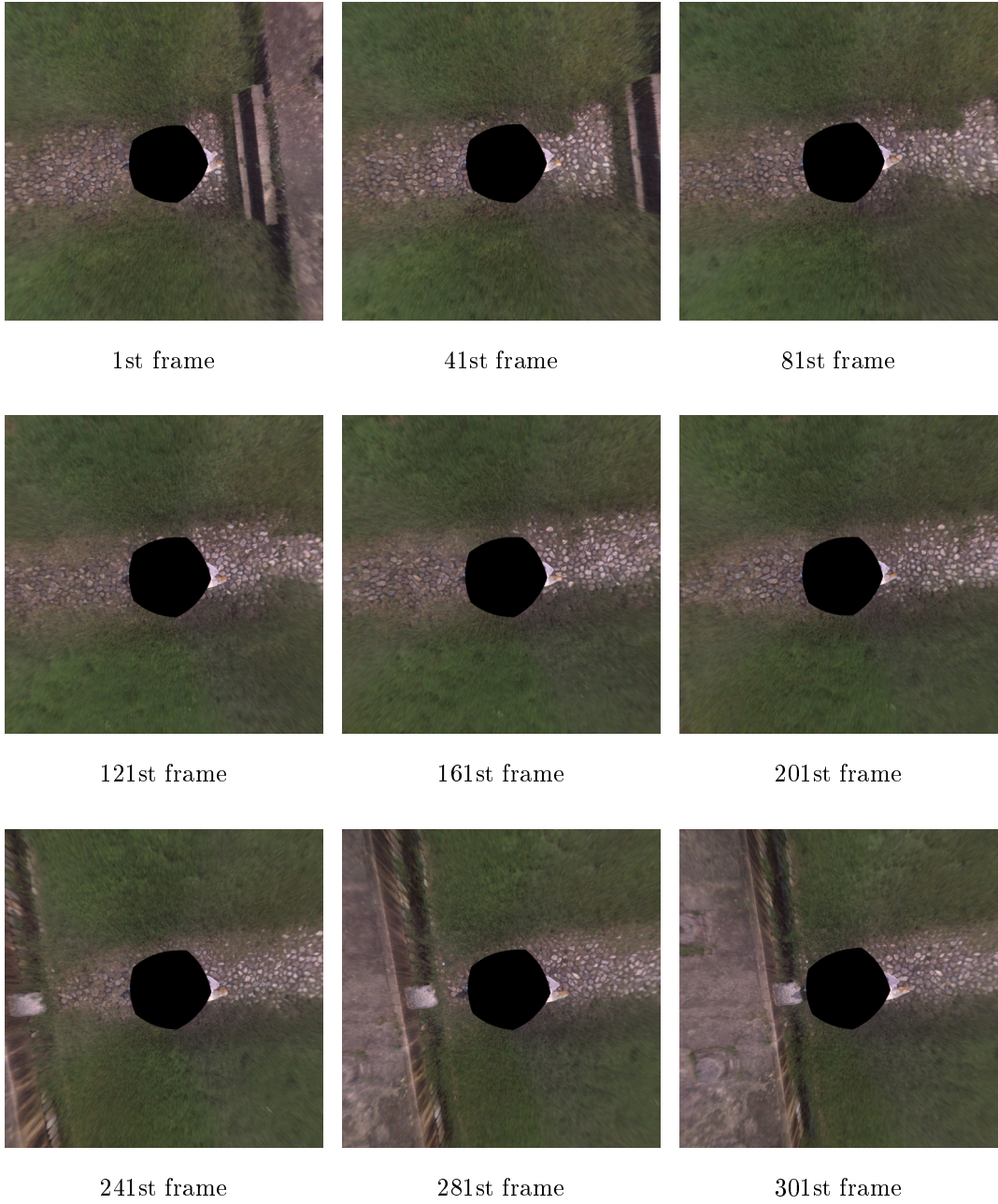
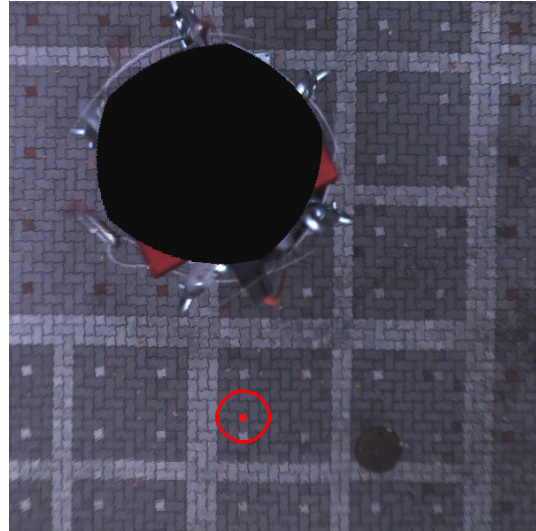


Figure 3.14. Images projected on planes (scene (2)).



(a) Target frame (11th frame)



(b) Center of the data region in the 67th frame corresponding to pixel (600,600) in the target frame

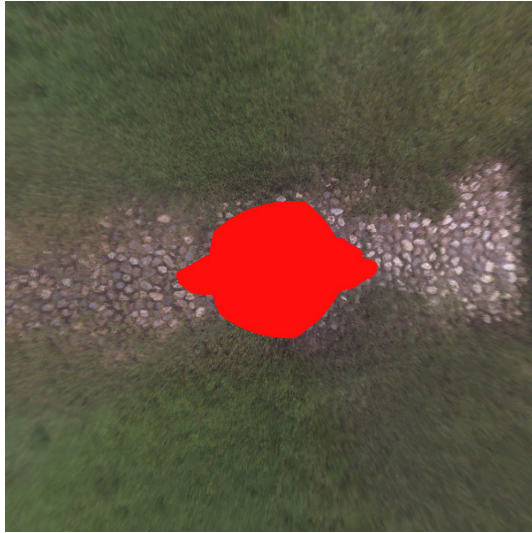


(c) Result by copying values of central pixels in data regions

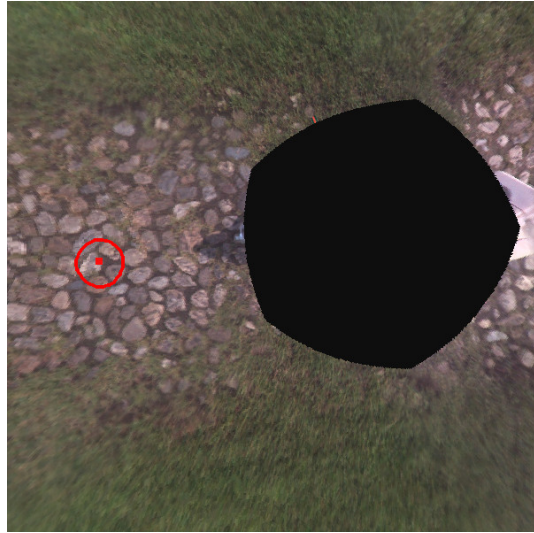


(d) Result by the proposed method

Figure 3.15. Comparison of results by projection using a plane and proposed method.



(a) Target frame (71st frame)



(b) Center of the data region in the 15th frame corresponding to pixel (600,600) in the target frame



(c) Result by copying values of central pixels in data regions



(d) Result by the proposed method

Figure 3.16. Comparison of results by projection using a plane and proposed method.

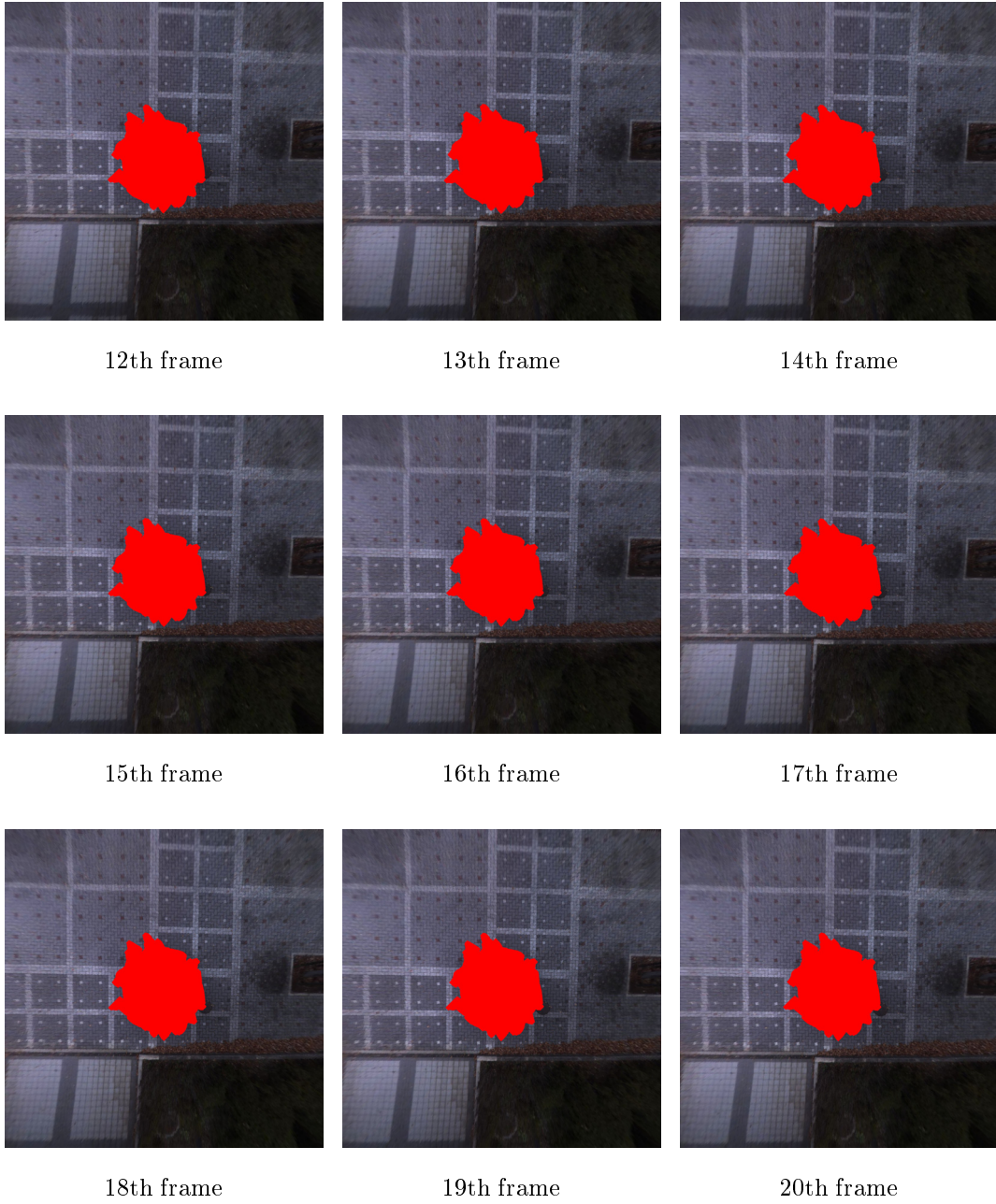
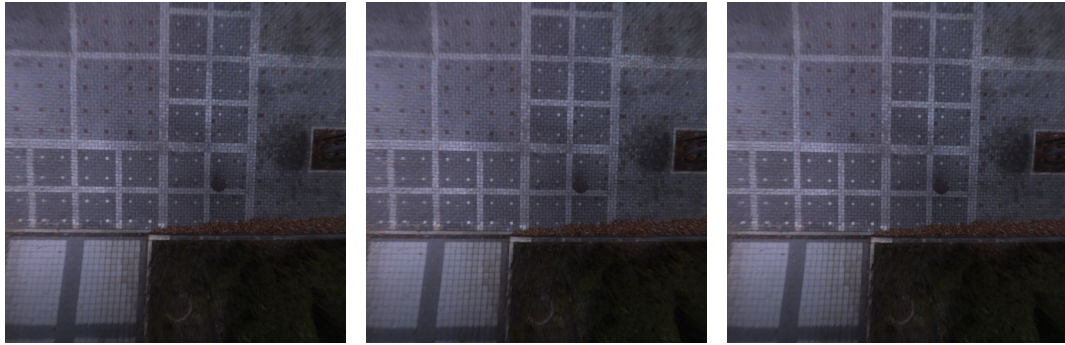


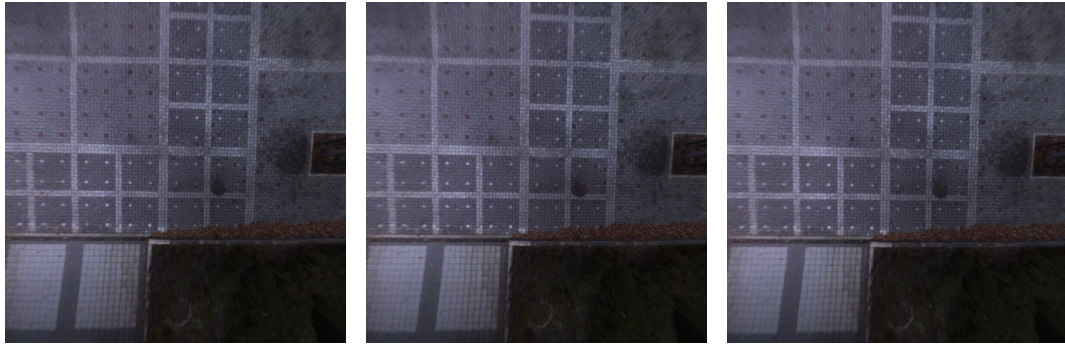
Figure 3.17. Successive frames with a missing region (scene (1)).



12th frame

13th frame

14th frame



15th frame

16th frame

17th frame



18th frame

19th frame

20th frame

Figure 3.18. Successive completed projected images (scene (1)).

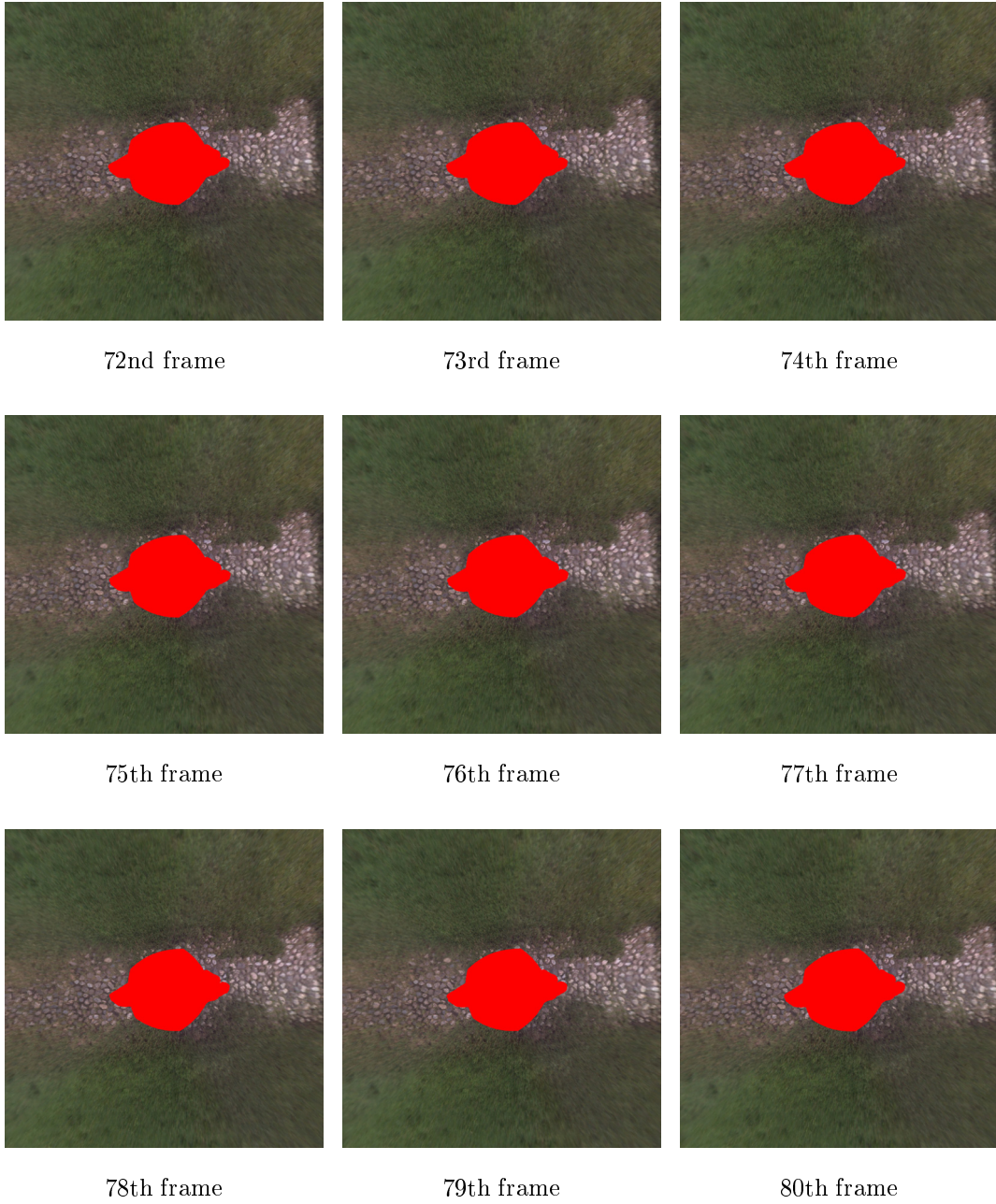


Figure 3.19. Successive frames with a missing region (scene (2)).

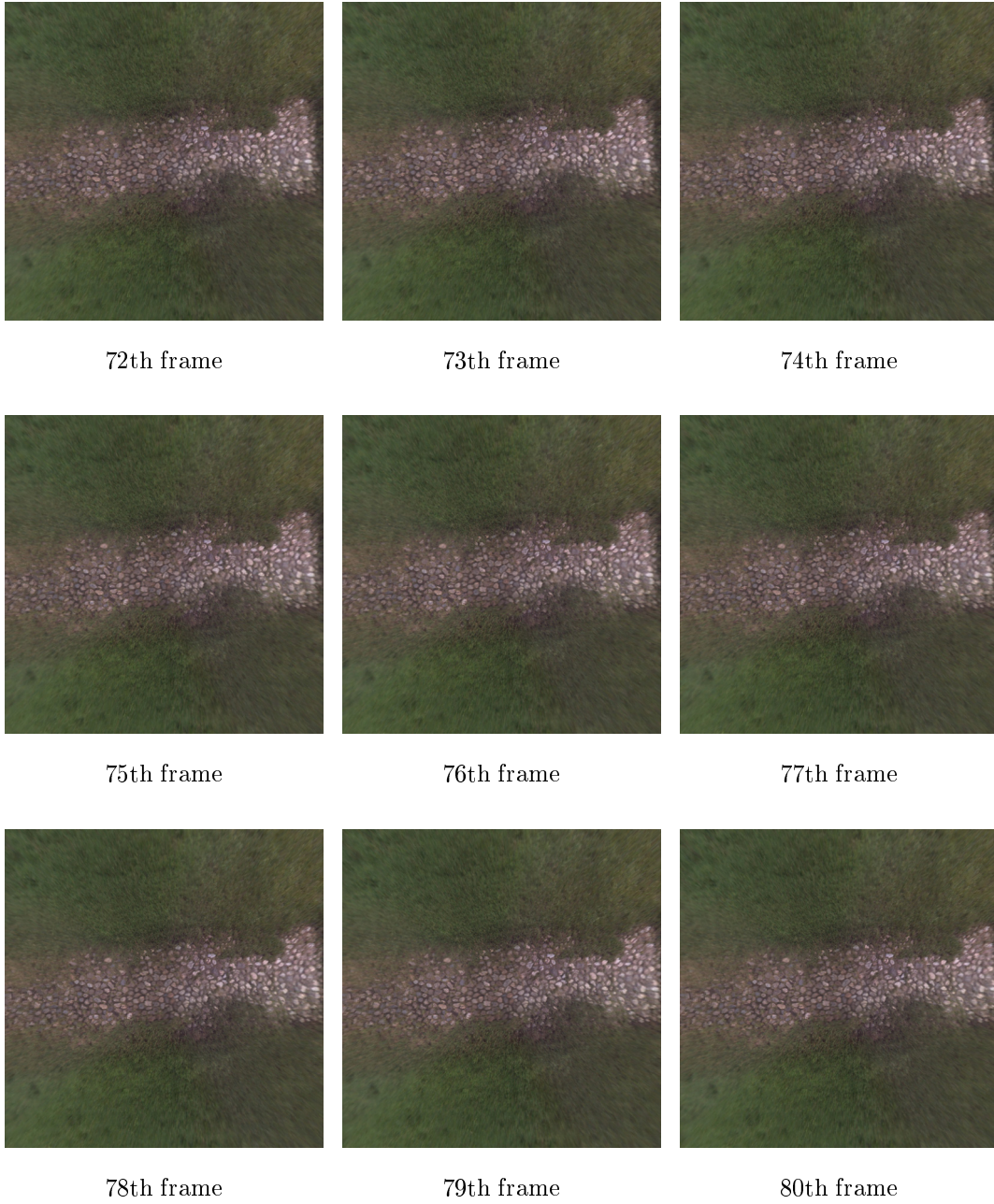


Figure 3.20. Successive completed projected images (scene (2)).

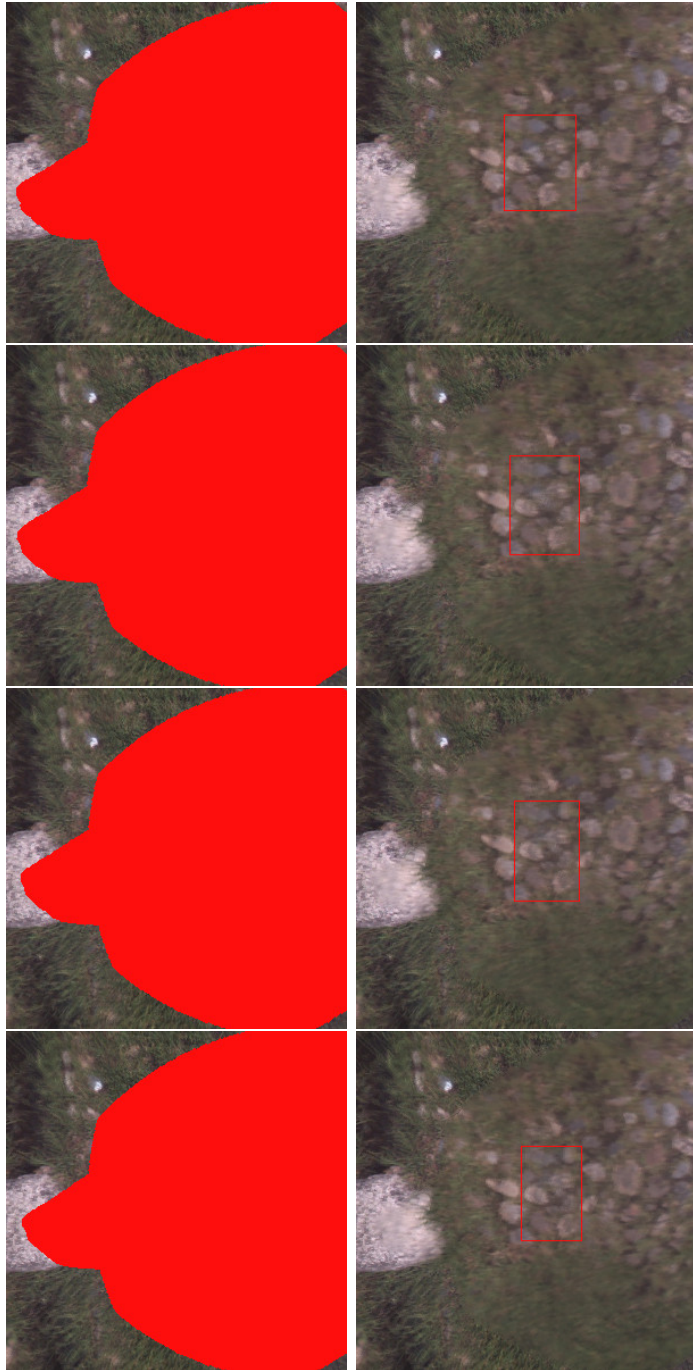


Figure 3.21. Example of failure results. From 298th frame to 301st frame.

3.7.3 Omnidirectional Telepresence without Invisible Areas

In this experiment, the effectiveness of the proposed method is demonstrated by making the telepresence system using omnidirectional videos in which missing regions are filled in with completed images shown in the previous section.

Figures 3.22(a) and 3.23(a) show omnidirectional panoramic images with a missing region and Figures 3.22(b) and 3.23 show the omnidirectional panoramic images without invisible areas generated by projecting the completed image (Figures 3.15(d) and 3.16(d)) onto the panoramic images (2048×1024 pixels). Figures 3.24 to 3.27 show examples of user's views in the telepresence system using the panoramic images. By comparing the images with and without the missing regions, we can confirm that realistic sensation is drastically increased by the proposed method. However, we can also confirm that textures in the missing regions slightly blur. In the completion of the proposed method, the resolution of textures for exemplars is small because the textures are captured from different viewpoints away from the camera position of the target frame. In order to increase the resolution of generated textures, we consider that a super-resolution method is useful.



(a) Panoramic image with a missing region



(b) Filled panoramic image

Figure 3.22. Filled panoramic image of 11th frame of scene (1).

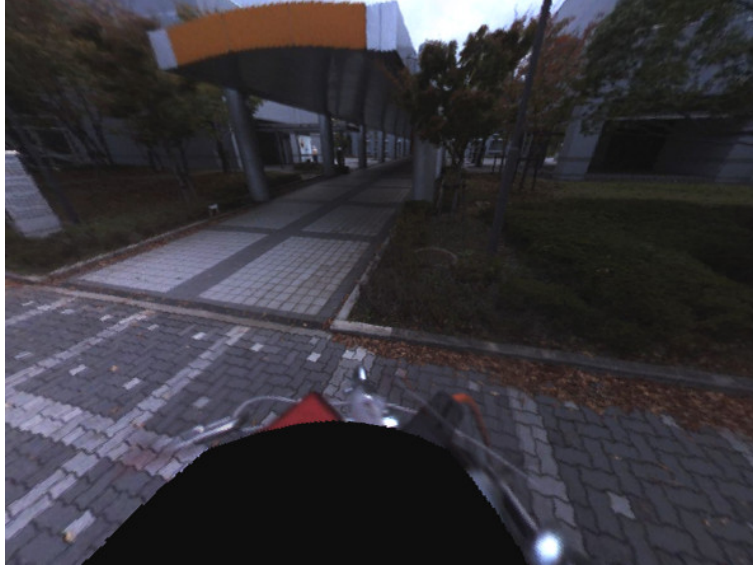


(a) Panoramic image with a missing region



(b) Filled panoramic image

Figure 3.23. Filled panoramic image of 71st frame of scene (2).



With a missing region.



Without a missing region.

Figure 3.24. Example of looking around using omnidirectional video (11th frame of scene (1)).



With a missing region.



Without a missing region.

Figure 3.25. Example of looking around using omnidirectional video (101st frame of scene (1)).



With a missing region.



Without a missing region.

Figure 3.26. Example of looking around using omnidirectional video (71st frame of scene (2)).



With a missing region.



Without a missing region.

Figure 3.27. Example of looking around using omnidirectional video (101st frame of scene (2)).

3.8. Conclusion

In this chapter, we have proposed a method that generates omnidirectional video without invisible areas by filling in missing regions. In the proposed method, in order to compensate for the change in the appearance of textures caused by camera motion, omnidirectional video images were projected onto the planes which were generated from natural feature points acquired by structure-from-motion. Then, appropriate data regions for missing regions were determined using the planes and the position and the posture of an omnidirectional camera. Finally, the missing regions were completed by minimizing an energy function based on pattern similarity between in the missing regions and data regions in different frames.

In experiments, missing regions in images projected on planes of two image sequences were successfully completed and the effectiveness of an energy minimization based on pattern similarity was demonstrated by comparing results by the proposed method with and without an energy minimization process. In addition, we confirmed that textures in missing regions change smoothly between successive frames. The omnidirectional telepresence without missing regions was also achieved and an improvement of realistic sensation in telepresence by filling in missing regions was confirmed. However, we also confirmed that the proposed method still has the problem that textures in the missing regions blur slightly.

Chapter 4

Surface Completion By Minimizing Energy Function Based on Similarity of Local Shape

4.1. Introduction

This chapter describes a surface completion method for 3D mesh models. 3D models of real environments are widely used for entertainment and 3D maps. Thus, automatic generation of 3D models using range scanners and video images has been investigated [ZS03, FZ04, AKY05, SKYT02]. Such methods can obtain whole 3D models by measuring the target objects from multiple view points and integrating the partial shapes. However, when the target is a large and complex environment, such as an outdoor scene, it is difficult to generate complete 3D models without holes due to various occlusions. For this problem, several surface completion methods, which fill in missing regions in a 3D mesh model, have been proposed. Recently, methods using the similarity of shape have been intensively developed [SACO04, BF08, PGSQ06] because such methods can generate complex surfaces in missing regions. However, inconsistent surfaces are often generated in the seams of the completed models because methods based on successive copying

of similar surface patches do not always generate optimum surfaces for the whole.

In this thesis, to solve such a problem, an energy function, which represents the surface shape implausibility, is defined using the similarity between the local surface shapes in the missing and data regions. By minimizing the energy function for whole of the missing region, a model with consistent surface shapes can be generated as an optimal solution.

In the following sections, first, the overview of the proposed method is given in Section 4.2. In Section 4.3, an energy function based on the similarity of local shapes is defined. In Section 4.4, a minimization method for the energy function is described. Next, experiments using three models are performed in Section 4.5. Finally, this chapter is concluded in Section 4.6.

4.2. Overview of the Method

Figure 4.1 illustrates the flow diagram of the proposed method. First, a user manually selects missing regions to be repaired in a 3D model. Next, initial points and faces are generated to the missing regions. Then, the whole surface is optimally completed by repeating three processes: (i) searching for similar local shapes; (ii) parallel updating of positions of vertices; and (iii) adding and integrating vertices considering density of those vertices. In the following sections, first, the energy function is defined based on local shape similarity in Section 4.3. Next, Section 4.4 describes the procedure that minimizes the energy function.

4.3. Definition of Energy Function Based on Similarity of Local Shapes

In this section, first, the definition of an energy function based on the similarity between local shapes in a missing region and a data region is described. Next, the similarity measure used in the energy function is presented.

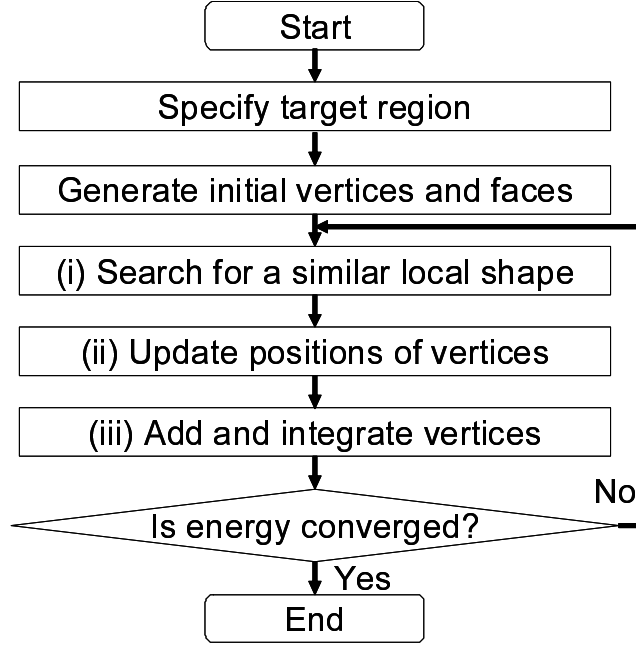


Figure 4.1. Flow diagram of the proposed method.

4.3.1 Definition of Energy Function

As illustrated in Figure 4.2, first, a 3D model is divided into region Ω' , including missing region Ω selected by a user, and data region Φ , that is the rest of the object. Region Ω' is determined so that a spherical area A with a constant radius whose central vertex is in region Ω' includes at least one of the initial vertices in region Ω . Energy function E_s is defined as the weighted sum of SSD (Sum of Squared Distances) between the vertices around vertex \mathbf{p}_i in region Ω' and the surface around point $\hat{\mathbf{p}}_i$ in region Φ as follows:

$$E_s = \frac{\sum_{\mathbf{p}_i \in \Omega'} w_{\mathbf{p}_i} SSD_s(\mathbf{p}_i, \hat{\mathbf{p}}_i)}{\sum_{\mathbf{p}_i \in \Omega'} w_{\mathbf{p}_i}}. \quad (4.1)$$

Here, weight $w_{\mathbf{p}_i}$ for each vertex \mathbf{p}_i is set as 1 when \mathbf{p}_i is inside of region $\Omega' \cap \bar{\Omega}$ because the positions of the vertices in this region are fixed; otherwise, $w_{\mathbf{p}_i}$ is set as s^{-m} (m is the minimum number of links from vertices of the boundary to vertex \mathbf{p}_i , and s is a positive constant) because vertices near the boundary have higher

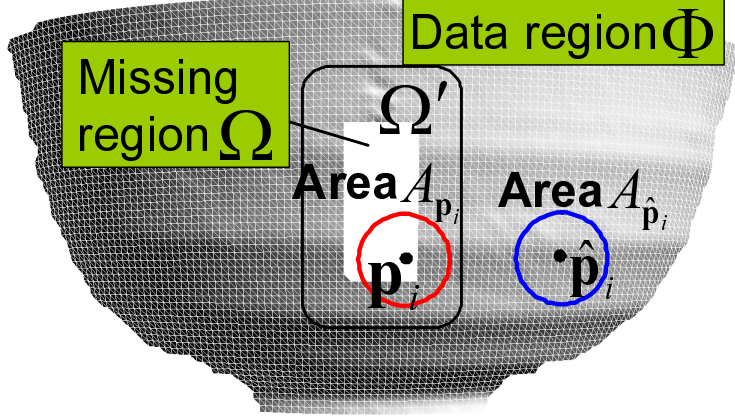


Figure 4.2. Missing and data regions in a 3D model.

confidence than those near the center of the missing region. In Eq. (4.1), the energy is normalized with respect to the sum of the weights because the weight of each vertex changes due to the updating of vertices positions. In the following section, the similarity measure SSD_s is described in detail.

4.3.2 Similarity of Local Shape

In general, the density of vertices depends on position in a 3D model because of the nonuniformity of measuring points. Therefore, this study defines the density-independent similarity SSD using vertices in a missing region and surfaces in a data region. Concretely, $SSD_s(\mathbf{p}_i, \hat{\mathbf{p}}_i)$, which represents the similarity between local shapes in missing and data regions, is defined as the sum of squared distances between the vertices in spherical area $A_{\mathbf{p}_i}$ whose central vertex is \mathbf{p}_i and the aligned surface around $\hat{\mathbf{p}}_i$ as follows:

$$SSD_s(\mathbf{p}_i, \hat{\mathbf{p}}_i) = \sum_{\mathbf{p}_k \in A_{\mathbf{p}_i}} \frac{\|\mathbf{p}_k - \mathbf{M}_{\hat{\mathbf{p}}_i, \mathbf{p}_i} \mathbf{g}_i(\mathbf{p}_k)\|^2}{N(A_{\mathbf{p}_i)}}, \quad (4.2)$$

where $\mathbf{M}_{\hat{\mathbf{p}}_i, \mathbf{p}_i}$ denotes the transform matrix for surface alignment as shown in Figure 4.3. $\mathbf{M}_{\hat{\mathbf{p}}_i, \mathbf{p}_i} \mathbf{g}_i(\mathbf{p}_k)$ is a point on the aligned surface in the data region that exists in the normal direction of point \mathbf{p}_k ($\in A_{\mathbf{p}_i}$). $N(A_{\mathbf{p}_i})$ is the number of

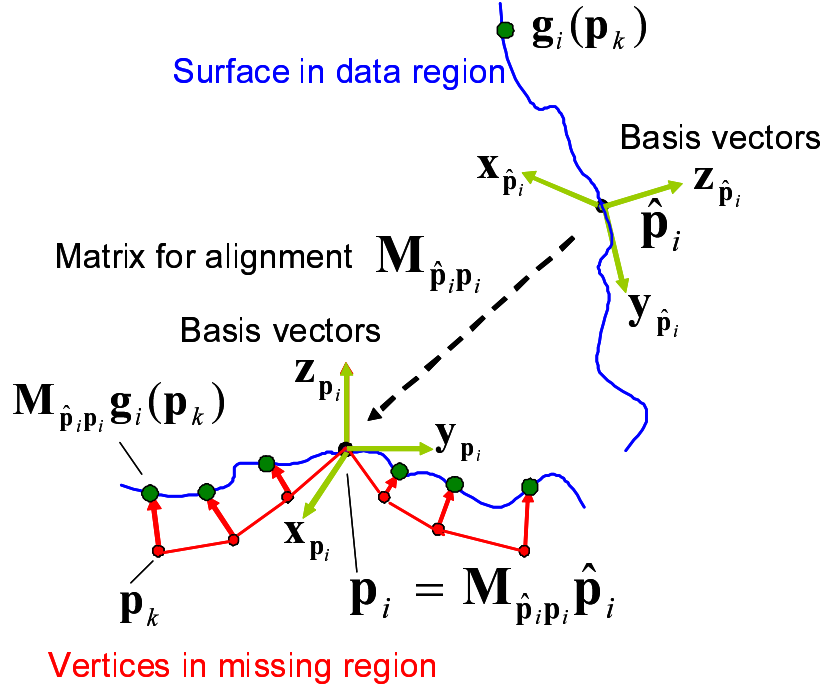


Figure 4.3. Alignment of vertex clouds and surface.

vertices in spherical area $A_{\mathbf{p}_i}$.

Transform matrix $\mathbf{M}_{\hat{\mathbf{p}}_i \mathbf{p}_i}$ for surface alignment consists of the 3D positions of $\mathbf{p}_i = (x_{\mathbf{p}_i}, y_{\mathbf{p}_i}, z_{\mathbf{p}_i})$ and $\hat{\mathbf{p}}_i = (x_{\hat{\mathbf{p}}_i}, y_{\hat{\mathbf{p}}_i}, z_{\hat{\mathbf{p}}_i})$, and the basis vectors for \mathbf{p}_i and $\hat{\mathbf{p}}_i$: $(\mathbf{x}_{\mathbf{p}_i}, \mathbf{y}_{\mathbf{p}_i}, \mathbf{z}_{\mathbf{p}_i})$ and $(\mathbf{x}_{\hat{\mathbf{p}}_i}, \mathbf{y}_{\hat{\mathbf{p}}_i}, \mathbf{z}_{\hat{\mathbf{p}}_i})$ as follows:

$$\mathbf{M}_{\hat{\mathbf{p}}_i \mathbf{p}_i} = \begin{pmatrix} \mathbf{x}_{\mathbf{p}_i} & -x_{\mathbf{p}_i} \\ \mathbf{y}_{\mathbf{p}_i} & -y_{\mathbf{p}_i} \\ \mathbf{z}_{\mathbf{p}_i} & -z_{\mathbf{p}_i} \\ 0 & 0 & 0 & 1 \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{x}_{\hat{\mathbf{p}}_i} & -x_{\hat{\mathbf{p}}_i} \\ \mathbf{y}_{\hat{\mathbf{p}}_i} & -y_{\hat{\mathbf{p}}_i} \\ \mathbf{z}_{\hat{\mathbf{p}}_i} & -z_{\hat{\mathbf{p}}_i} \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (4.3)$$

Here, the similarity measure SSD_s in energy function E_s largely depends on basis vectors $(\mathbf{x}, \mathbf{y}, \mathbf{z})$ for each vertex. Conventional methods inefficiently calculated the similarity for every surface that is rotated every fixed degree around a normal axis. Instead of such an inefficient search, in this research, basis vectors are uniquely determined using the directions of normal and principal curvatures of

local surface. In the following, the way to determine basis vectors ($\mathbf{x}_{\mathbf{p}_i}, \mathbf{y}_{\mathbf{p}_i}, \mathbf{z}_{\mathbf{p}_i}$) for each point \mathbf{p}_i is described.

First, through principal component analysis for 3D coordinates of vertices in spherical area $B_{\mathbf{p}_i}$ whose central vertex is \mathbf{p}_i , the coordinate system for each vertex is set so that its x, y and z coordinates are the directions of the eigenvectors of the first, second and third eigenvalues. Next, the following quadratic surface function is fitted to vertices in area $B_{\mathbf{p}_i}$:

$$z(x, y) = ax^2 + by^2 + cxy + dx + ey + f, \quad (4.4)$$

where each parameter (a, b, c, d, e, f) is determined using the least-squares method so as to minimize the following cost $Q_{\mathbf{p}_i}$:

$$Q_{\mathbf{p}_i} = \sum_{\mathbf{p}_k \in B_{\mathbf{p}_i}} \{z(\bar{x}_k, \bar{y}_k) - \bar{z}_k\}^2, \quad (4.5)$$

where $(\bar{x}_k, \bar{y}_k, \bar{z}_k)$ is the 3D coordinate of \mathbf{p}_k in the coordinate system generated by principal component analysis. From the estimated quadratic surface, the directions of normal and maximum and minimum principal curvatures are calculated. Concretely, Hessian matrix H of quadratic function $z(x, y)$ is used to calculate the directions of maximum and minimum principal curvatures:

$$\mathbf{H} = \begin{pmatrix} \frac{\partial^2 z(x, y)}{\partial x^2} & \frac{\partial^2 z(x, y)}{\partial x \partial y} \\ \frac{\partial^2 z(x, y)}{\partial y \partial x} & \frac{\partial^2 z(x, y)}{\partial y^2} \end{pmatrix} = \begin{pmatrix} 2a & c \\ c & 2b \end{pmatrix}. \quad (4.6)$$

Here, eigenvectors of matrix H indicate the directions of the maximum and minimum principal curvatures of vertex \mathbf{p}_i in the coordinate system generated by principal component analysis. Normal vector $\bar{\mathbf{n}}_i$ of vertex \mathbf{p}_i in the coordinate system is also calculated as follows:

$$\bar{\mathbf{n}}_i = \left(-\frac{\partial z(0, 0)}{\partial x}, -\frac{\partial z(0, 0)}{\partial y}, 1 \right) = (-d, -e, 1). \quad (4.7)$$

The directions of the principal curvatures and normal are converted into the directions in the original coordinate system, and the unit vectors are set to $\mathbf{x}_{\mathbf{p}_i}$, $\mathbf{y}_{\mathbf{p}_i}$ and $\mathbf{z}_{\mathbf{p}_i}$, respectively. Basis vectors ($\mathbf{x}_{\hat{\mathbf{p}}_i}, \mathbf{y}_{\hat{\mathbf{p}}_i}, \mathbf{z}_{\hat{\mathbf{p}}_i}$) are also determined in the same way. However, if directions of principal curvatures are not uniquely determined in the case when the surface shape is rotationally symmetric, $\mathbf{x}_{\mathbf{p}_i}$ and $\mathbf{y}_{\mathbf{p}_i}$ are determined using any directions so that directions of the vectors are orthogonal.

4.4. Energy Minimization

This section describes the method to minimize energy function E_s in Eq. (4.1) using a framework of greedy algorithm after generating initial vertices and faces in missing regions. In our definition of energy function E_s , the energy for each vertex can be treated independently if all similar shape pairs $(\mathbf{p}_i, \hat{\mathbf{p}}_i)$ are fixed. Thus, the following three processes are repeated until the energy converges: (i) searching for similar local shape keeping positions of vertices fixed; (ii) parallel updating of all the positions of vertices keeping similar shape pairs fixed; and (iii) adding and integrating vertices considering the density of vertices. In the following sections, each process is described in detail.

4.4.1 Searching for Similar Local Shape

In process (i), a data region is searched for similar local shapes while keeping the positions of all vertices in missing regions fixed. Basically, SSD_s is calculated for all the vertices in data region Φ , and the vertex which gives a minimum value is determined as the most similar vertex $\hat{\mathbf{p}}_i$ as follows:

$$\mathbf{f}(\mathbf{p}_i) = \hat{\mathbf{p}}_i = \underset{\mathbf{p}' \in \Phi}{\operatorname{argmin}}(SSD_s(\mathbf{p}_i, \mathbf{p}')). \quad (4.8)$$

However, the required cost for calculating SSD_s for all the vertices in region Φ is great. For this problem, two methods are used in this research. One is SSDA (Sequential similarity detection algorithm) [BS72], which can skip the calculation of SSD_s whose value is much larger than the minimum of SSD_s . In the other method, the calculation of SSD_s is skipped for a vertex around which the local shape may not be similar to that around the target vertex using maximum and minimum principal curvatures calculated in surface fitting described in section 4.3.2.

Concretely, first, among eigenvalues of Hessian matrix H expressed in Eq. (4.6), the higher value is set as the maximum principal curvature and the lower value is set as the minimum principal curvature. Here, maximum and minimum curvatures of target vertex \mathbf{p}_i and corresponding vertex $\hat{\mathbf{p}}_i$ are represented as $k1_{\mathbf{p}_i}$ and $k2_{\mathbf{p}_i}$, $k1_{\hat{\mathbf{p}}_i}$ and $k2_{\hat{\mathbf{p}}_i}$ respectively. Then, cost K , that is the dissimilarity measure

of curvatures, is calculated for all the vertices in the data region as follows:

$$K_{\mathbf{p}_i \hat{\mathbf{p}}_i} = (k1_{\mathbf{p}_i} - k1_{\hat{\mathbf{p}}_i})^2 + (k2_{\mathbf{p}_i} - k2_{\hat{\mathbf{p}}_i})^2, \quad (4.9)$$

Here, there is a high possibility that both SSD_s and K render low values if surface shapes in missing and data regions are similar. Therefore, by sorting the vertices in the data region in ascending order according to K and calculating SSD_s only for the top $n\%$ vertices out of all the vertices in the data region, the calculation cost is significantly reduced.

In addition, the proposed method uses plane-symmetrical local surfaces for completion by paying attention to the fact that an ordinary 3D object has many plane-symmetrical local shapes. Concretely, by reversing the sign of basis vector $\mathbf{x}_{\mathbf{p}_i}$ or $\mathbf{y}_{\mathbf{p}_i}$ in matrix $\mathbf{M}_{\hat{\mathbf{p}}\mathbf{p}}$ for surface alignment, SSD_s for the plane-symmetrical shape is also calculated.

4.4.2 Parallel Updating of Positions of Vertices

In process (ii), the positions of all vertices \mathbf{p}_i in missing regions are updated in parallel so as to minimize energy E_s defined in Eq. (4.1). In the following, the method for calculating the positions of points \mathbf{p}_i for fixed similar shape pairs is described in detail. First, energy function E_s is resolved into element energy $E_s(\mathbf{p}_i)$ for each vertex in the missing region. Here, as shown in Figure 4.4, the target vertex to be updated is \mathbf{p}_i , and the position of the k -th vertex inside area $A_{\mathbf{p}_i}$ is expressed as \mathbf{p}_k and is corresponded to $\mathbf{f}(\mathbf{p}_k)$ by Eq. (4.8). In this case, the point corresponding to vertex \mathbf{p}_i is $\mathbf{g}_k(\mathbf{p}_i)$. Now, element energy $E_s(\mathbf{p}_i)$ can be defined in terms of \mathbf{p}_i , $\mathbf{g}_k(\mathbf{p}_i)$ and transform matrix $\mathbf{M}_{\mathbf{f}(\mathbf{p}_k)\mathbf{p}_k}$ for surface alignment as follows:

$$E_s(\mathbf{p}_i) = \sum_{\mathbf{p}_k \in A_{\mathbf{p}_i}} \frac{w_{\mathbf{p}_k}}{N(A_{\mathbf{p}_i})} \|\mathbf{p}_i - \mathbf{M}_{\mathbf{f}(\mathbf{p}_k)\mathbf{p}_k} \mathbf{g}_k(\mathbf{p}_i)\|^2. \quad (4.10)$$

The relationship between total energy E_s and element energy $E_s(\mathbf{p}_i)$ for each vertex can be written as follows:

$$E_s = \sum_{\mathbf{p}_i \in \Omega} E_s(\mathbf{p}_i) + C. \quad (4.11)$$

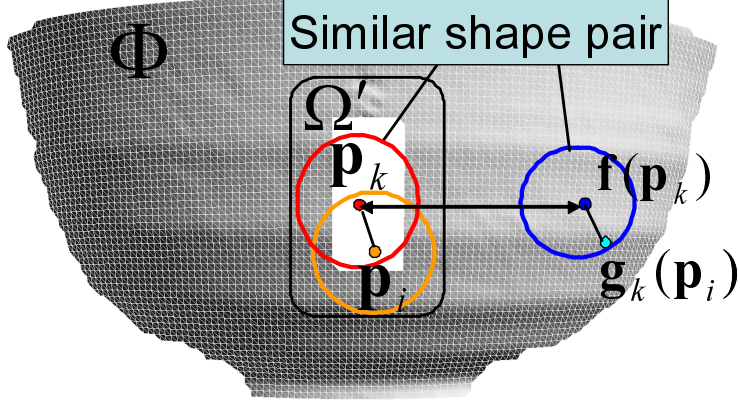


Figure 4.4. Relationship between vertices in energy calculation.

C is the energy for the vertices in region $\bar{\Omega} \cap \Omega'$, and is treated as a constant because positions of vertices and all the similar shape pairs are fixed in this region in process (ii).

Here, it should be noted that all the corresponding points $\mathbf{M}_{\mathbf{f}(\mathbf{p}_k)\mathbf{p}_k} \mathbf{g}_k(\mathbf{p}_i)$ ($\forall \mathbf{p}_k \in A_{\mathbf{p}_i}$) exist in the direction of the normal vector of vertex \mathbf{p}_i . Thus, as illustrated in Figure 4.5, vertex \mathbf{p}_i and point $\mathbf{M}_{\mathbf{f}(\mathbf{p}_k)\mathbf{p}_k} \mathbf{g}_k(\mathbf{p}_i)$ can be expressed using unit normal vector $\mathbf{n}_{\mathbf{p}_i}$ of vertex \mathbf{p}_i and arbitrary point \mathbf{p}_0 in the direction of normal vector of \mathbf{p}_i as follows:

$$\mathbf{p}_i = \mathbf{p}_0 + t_{\mathbf{p}_i} \mathbf{n}_{\mathbf{p}_i}, \quad (4.12)$$

$$\mathbf{M}_{\mathbf{f}(\mathbf{p}_k)\mathbf{p}_k} \mathbf{g}_k(\mathbf{p}_i) = \mathbf{p}_0 + t_{(\mathbf{p}_k, \mathbf{p}_i)} \mathbf{n}_{\mathbf{p}_i}. \quad (4.13)$$

By substituting these equations into Eq. (4.10), the following equation is obtained:

$$E(\mathbf{p}_i) = \sum_{\mathbf{p}_k \in A_{\mathbf{p}_i}} \frac{w_{\mathbf{p}_k}}{N(A_{\mathbf{p}_i})} (t_{\mathbf{p}_i} - t_{(\mathbf{p}_k, \mathbf{p}_i)})^2. \quad (4.14)$$

Here, on the assumption that normal vector $\mathbf{n}_{\mathbf{p}_i}$ does not change after updating the position of \mathbf{p}_i , the parameter of element energy $E_s(\mathbf{p}_i)$ is only $t_{\mathbf{p}_i}$ and the change in parameter $t_{\mathbf{p}_i}$ does not influence the element energies of the other vertices. Therefore, overall energy E_s can be minimized by minimizing element energy $E_s(\mathbf{p}_i)$ individually. The value of parameter $t_{\mathbf{p}_i}$ that minimizes $E_s(\mathbf{p}_i)$ is

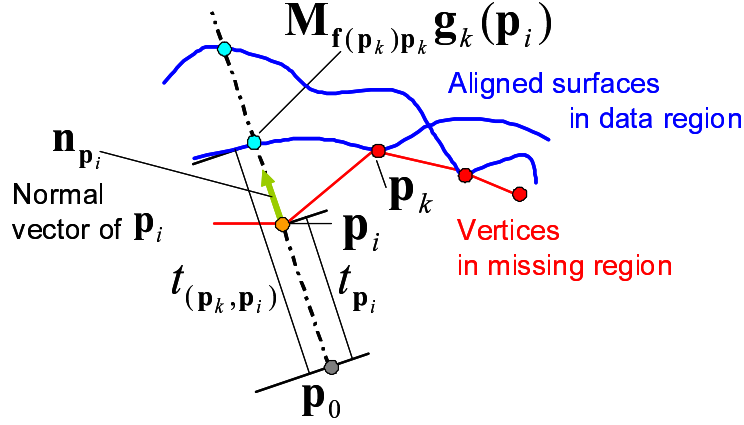


Figure 4.5. Conversion of parameters.

calculated as follows:

$$t_{\mathbf{p}_i} = \frac{\sum_{\mathbf{p}_k \in A_{\mathbf{p}_i}} w_{\mathbf{p}_k} t_{(\mathbf{p}_k, \mathbf{p}_i)}}{\sum_{\mathbf{p}_k \in A_{\mathbf{p}_i}} w_{\mathbf{p}_k}}. \quad (4.15)$$

Finally, the position of vertex \mathbf{p}_i is calculated from Eqs. (4.12), (4.13) and (4.15) as follows:

$$\mathbf{p}_i = \frac{\sum_{\mathbf{p}_k \in A_{\mathbf{p}_i}} w_{\mathbf{p}_k} \mathbf{M}_{f(\mathbf{p}_k)\mathbf{p}_k} \mathbf{g}_k(\mathbf{p}_i)}{\sum_{\mathbf{p}_k \in A_{\mathbf{p}_i}} w_{\mathbf{p}_k}}, \quad (4.16)$$

where, in fact, the value obtained in Eq. (4.16) is an approximate solution because the direction of normal vector $\mathbf{n}_{\mathbf{p}_i}$ and the position of $\mathbf{M}_{f(\mathbf{p}_k)\mathbf{p}_k} \mathbf{g}_k(\mathbf{p}_i)$ change due to the updating of the position of each vertex. However, a satisfactory solution can be obtained as the energy converges because the change in the direction of the normal vector of each vertex gradually decreases.

4.4.3 Adding and Integrating Vertices

The movement of vertices in the energy minimization process described in the previous section causes bias in the distribution of vertices in the missing region. As a result, the energy minimization process becomes inefficient if the density of vertices is higher than necessary, and detailed surface shapes can not be produced if the density is low. For this problem, the proposed method adds and integrates

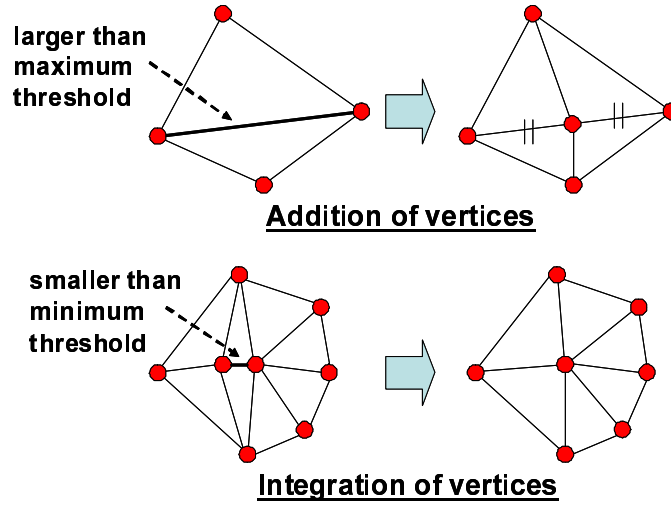


Figure 4.6. Addition and integration of vertices.

vertices in the missing region in order to keep the density constant in the iterative process.

Concretely, as illustrated in Figure 4.6, the maximum and minimum thresholds of the line fragment length between vertices are set and the following two processes are performed: (1) add a vertex in the middle of the line segment whose length is larger than the maximum threshold; and (2) integrate both end vertices of the line segment whose length is smaller than the minimum threshold and put the integrated vertex in the middle of the line segment.

4.4.4 Coarse-to-fine Approach

In this research, in order to efficiently complete the missing regions and avoid local minima, a coarse-to-fine approach, in which the range of spherical area $A_{\mathbf{p}}$ used for calculating SSD_s and the number of vertices in region Ω' used for calculating energy are changed every time the energy converges, is employed for energy minimization. Concretely, the radius of spherical area $A_{\mathbf{p}}$ is decreased and the number of vertices in region Ω' is increased step-by-step. Here, in missing region Ω in region Ω' , the density of vertices is increased by gradually decreasing the

thresholds for the addition and integration of vertices described in the previous section. This leads to the generation of coarse surface shapes in the initial step and detailed surface shapes in the final step. In the remaining region $\Omega' \cap \overline{\Omega}$ in region Ω' , the number of vertices and the positions are fixed because the region is not the missing region. Therefore, the energy is calculated by skipping vertices at a constant rate in the initial step. By repeating the energy minimization processes as the skipping rate is gradually decreased, the missing region is efficiently completed.

4.5. Experiments

In this section, first, in order to demonstrate the effectiveness of the proposed 3D surface completion method, the proposed and conventional methods are applied to three surface models and the results are qualitatively compared. Next, the influence on completion results of parameters in the proposed method is discussed.

4.5.1 Qualitative Evaluation

In this experiment, three surface models (I), (II), and (III) with a hole as shown in Figure 4.7(a), 4.8(a), and 4.10(a) are completed. The effectiveness of the proposed method is demonstrated by comparing the completion results by the proposed method and our implemented conventional method [BF08], which fills in missing regions by successively copying similar local shapes to the missing regions after giving initial vertices and faces in a similar way as the proposed method. The missing regions in Models (I) and (II) were given on purpose and the missing region in Model (III) was caused by the occlusion in practical measurement of a building.

The specifications of the PC in this experiment were Xeon 3.0 GHz of CPU and 8 GB of memory. In completion of the proposed method, each parameter in the proposed method was set as shown in Table 4.1. As initial vertices in a missing region, as shown in each figure (c), the gravity point of the boundary vertices of the missing region and the median points between the gravity point and each boundary point were given. Faces were generated so as to connect these vertices. As for completion of the conventional method [BF08], it is required that a smooth surface be given to a missing region as an initial surface. Therefore, in order to give an appropriate initial surface for the conventional method, the smoothed surface of the original model was given as the initial surface in Models (I) and (II), and the smoothed surface of the completion result by the proposed method was given as the initial surface in Model (III) because the original surface does not exist in the missing region of Model (III). Here, each surface model was completed changing the radius of area A , which is a parameter in the conventional method, from $9l_{ave}$ to $5l_{ave}$ and the best results are illustrated in each figure (d).

Model (I) (Figure 4.7(a)) is a relatively simple bowl model that has a smooth

Table 4.1. Parameters in the proposed method in experiments (l_{ave} indicates average length between vertices in data region).

| Multi-scale level | 1 | 2 | 3 |
|--|---|--------------|--------------|
| s in weight w | 2.0 (Model (I) and (II)) 3.0 (Model (III)) | | |
| Radius of area A | $9l_{ave}$ | $7l_{ave}$ | $5l_{ave}$ |
| Radius of area B | $12l_{ave}$ (Model (I)) $26l_{ave}$ (Model (II)) $8l_{ave}$ (Model (III)) | | |
| Rate n in searching | 10% | | |
| Skipping rate of vertices in $\Omega' \cap \bar{\Omega}$ | 8/9 | 3/4 | 0 |
| Maximum threshold | $4l_{ave}$ | $2l_{ave}$ | $1.3l_{ave}$ |
| Minimum threshold | $1.5l_{ave}$ | $0.8l_{ave}$ | $0.4l_{ave}$ |

curved surface and raised edges along the curve around the missing region. As illustrated in Figure 4.7(b), the model completed by the conventional method has distorted edges in the vicinity of the center of the missing region. On the other hand, by refining the initial model as shown in Figure 4.7(c) with the proposed method, the missing region is filled in with a smooth curved surface similar to the surface in the data region. Raised edges are also constructed and on the whole the completed model looks very natural.

Model (II) (Figure 4.8(a)) is Stanford Bunny that has a rugged surface and a dented edge between the body and the leg of the bunny model around the missing region. As illustrated in Figure 4.8(b), the model completed by the conventional method has connected edges in the vicinity of the center of the missing region. However, the shapes of the edges are not smooth even though the shape in the data region is. On the other hand, by refining the surface model from the initial model as shown in Figure 4.8(c), a natural rugged surface is generated and a dented and smooth edge is connected in the missing region as shown in Figure 4.8(d). Figure 4.9 shows changes in shape with iterations. From this figure, we can confirm that the shape is gradually completed according to the number of

iterations. In addition, a dented shape is minutely generated by repeating the processes while increasing the density of vertices step-by-step as shown in Figures 4.9(c) and (d).

Model (III) (Figure 4.10) is a model of a real building and the ground which has a hole caused by the occlusion of a street lamp in a practical measurement of outdoor environments. In the resultant model completed by the conventional method as shown in Figure 4.10(b), an implausible shape is generated near a window. On the other hand, the resultant model generated by the proposed method from the initial model (Figure 4.10(c)) has a more plausible shape in the context of the whole model.

Figure 4.11 shows changes in energy with respect to iterations for each model. Here, energy in this figure is normalized so that the initial energy is 1. From this figure, we can confirm that the energy for each model gradually decreases as the process is repeated from the initial model. It should be noted that the energy discontinuously changes when the density of vertices and the range of area A for calculating SSD_s are changed (After 17 and 25 times iterations for Model (I), 19 and 23 times iterations for Model (II), and 7 and 11 times iterations for Model (III)).

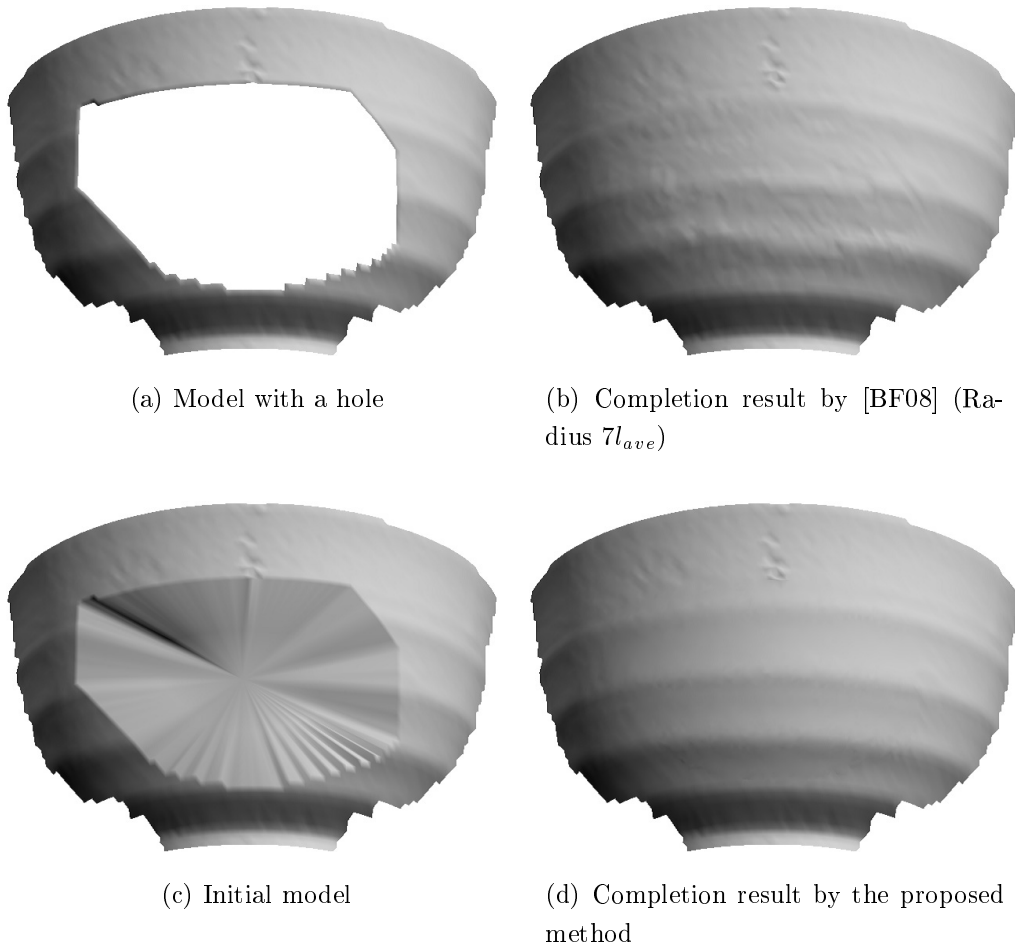


Figure 4.7. Surface completion for Model (I).
 (The number of vertices in model (a) is 3404, the number of vertices in missing region in model (d) is 1579, and the processing time is 234 seconds.)

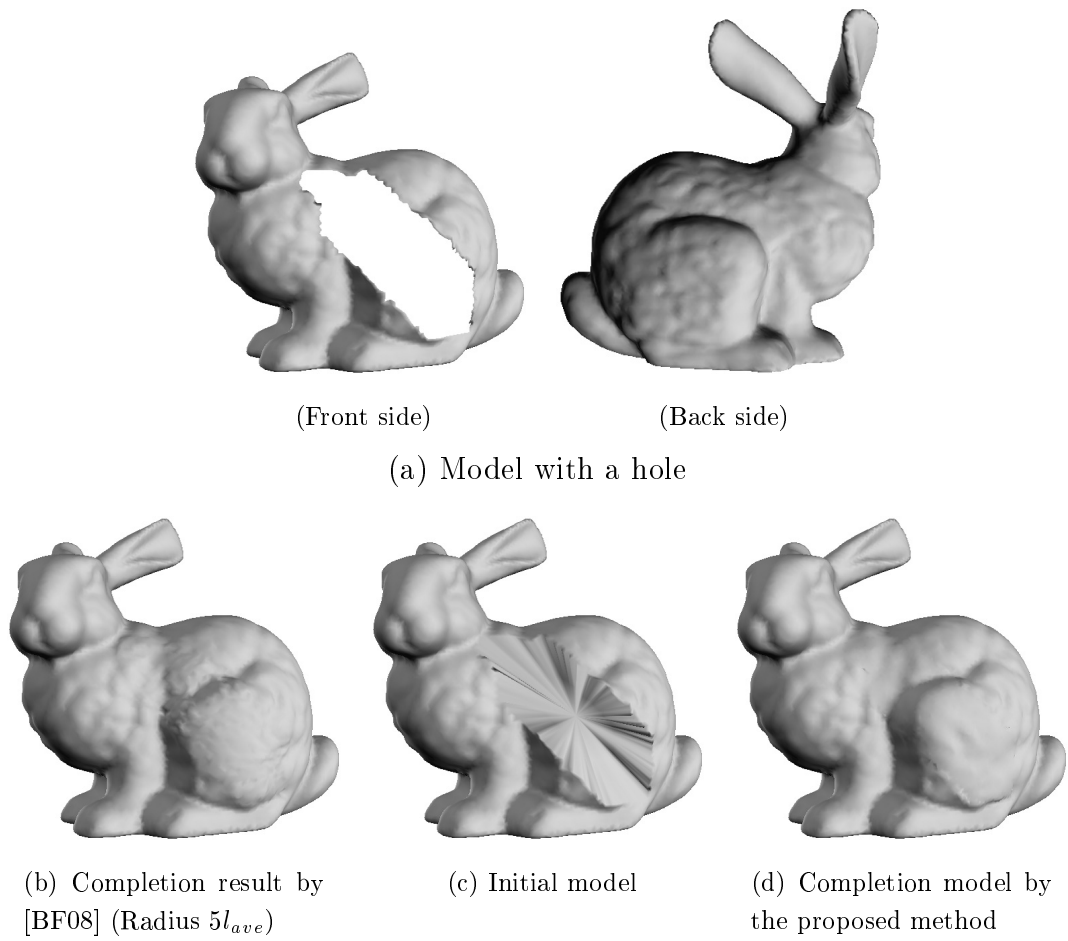
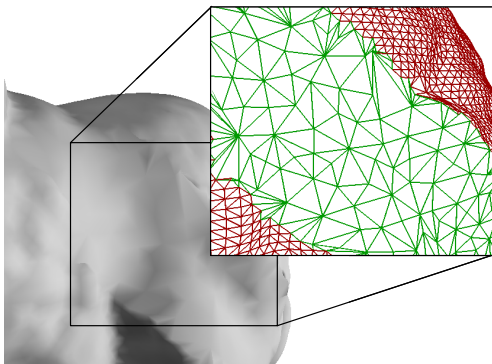
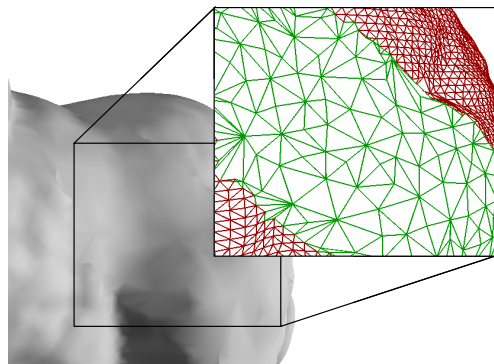


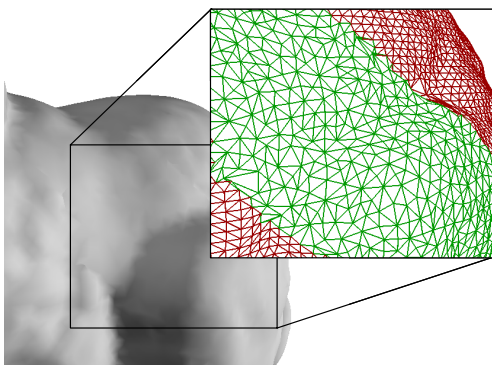
Figure 4.8. Surface completion for Model (II).
 (The number of vertices in model (a) is 17522, the number of vertices in missing region in model (d) is 1505, and the processing time is 1682 seconds.)



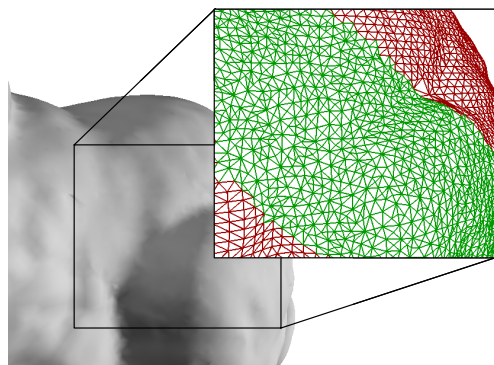
(a) After 2 iterations



(b) After 12 iterations

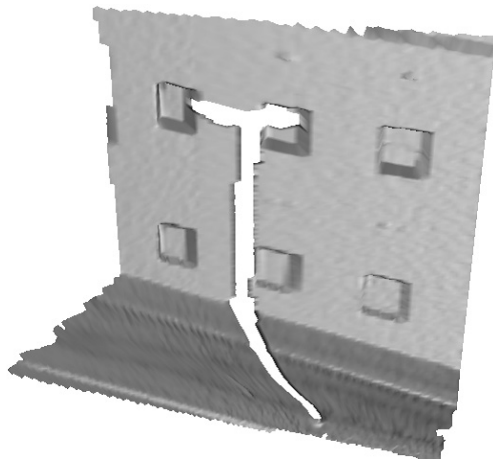


(c) After 22 iterations (2 iterations in the second stage)

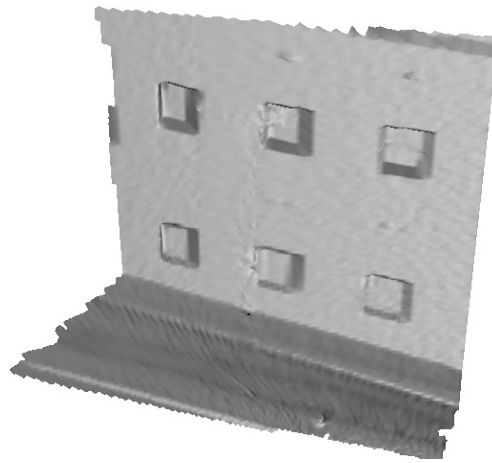


(d) After 26 iterations (2 iterations in the third stage)

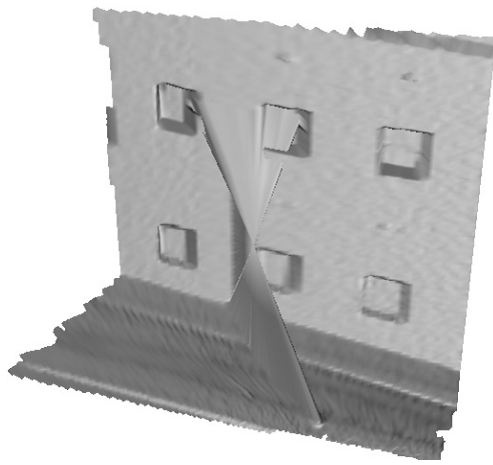
Figure 4.9. Changes in shape with iterations for Model (II).



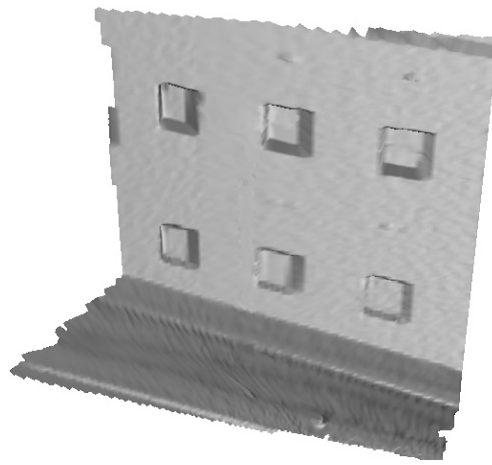
(a) Model with a hole



(b) Completion result by [BF08] (Radius $6l_{ave}$)



(c) Initial model



(d) Completion result by the proposed method

Figure 4.10. Surface completion for Model (III).

(The number of vertices in model (a) is 12749, the number of vertices in missing region in model (d) is 474, and the processing time is 1688 seconds.)

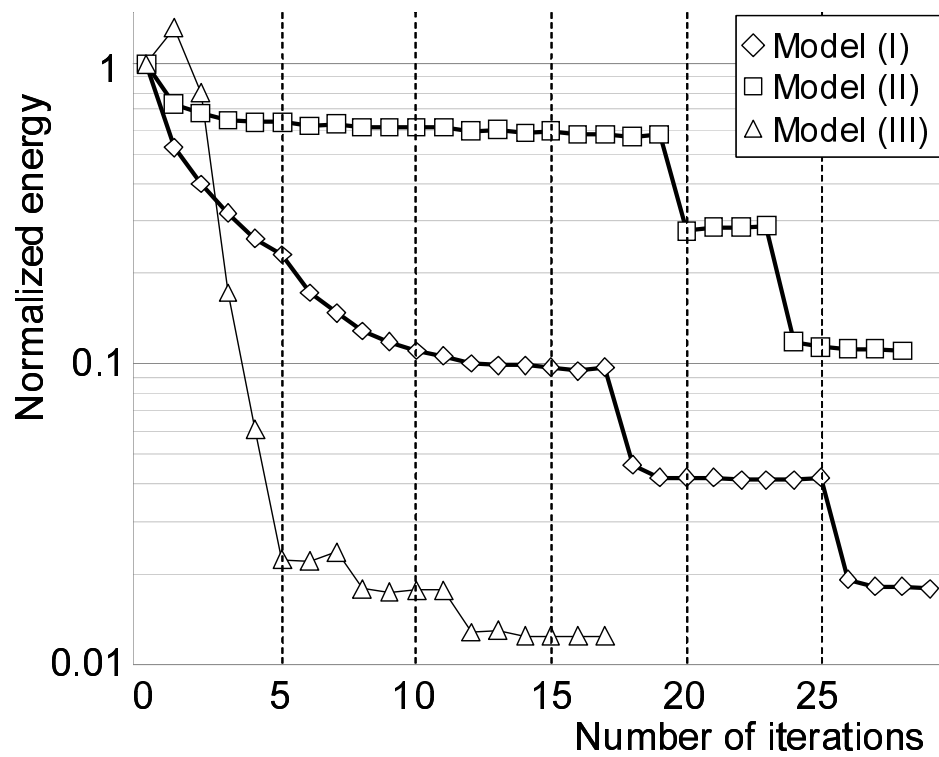


Figure 4.11. Changes in energy with respect to iterations for each model.

4.5.2 Discussion about Influence of Parameters

In the proposed method, it is necessary to determine multiple parameters as shown in Table 4.1 to perform surface completion. In this section, processing time and completion results for Models (I) and (II) with different values of parameter s in weight w in energy function E_s , which may especially have a significant influence on results, are discussed. Table 4.2 and Figures 4.12 and 4.13 show processing time and completion results for Models (I) and (II) when the value of parameter s is changed.

As for processing time, from Table 4.2, we can confirm that the processing time tends to decrease according to the increase of s , which means that the change rate of the weight between vertices based on the number of links from the boundary of the missing region increases. When positions of vertices are updated, the influence of vertices near the boundary of the missing region on those in more internal areas of the missing region increases as the change rate of the weight increases. Therefore, the large change of positions of vertices in one iteration makes the number of iterations decrease. As a result, the processing time is reduced.

As for completion results, as shown in Figure 4.12, the influence of the difference in the weight on the completion result of Model (I), which has a relatively simple shape, was small. However, the large influence on the completion result of Model (II) can be seen. Concretely, when s is relatively small ($s \leq 1.5$), it can be seen that a dented edge does not connect and the whole surface in the missing region tends to be smooth. On the other hand, when s is relatively large ($s \geq 3.0$), a dented edge connects and a ragged surface is generated but the shape in the vicinity of the missing region is unnatural because such shape does not exist in

Table 4.2. Processing time with different values of parameter s for each model.

| | Processing time (second) | | | | |
|------------|--------------------------|------|------|------|-----|
| s | 1.1 | 1.5 | 2.0 | 3.0 | 5.0 |
| Model (I) | 467 | 303 | 234 | 215 | 174 |
| Model (II) | 1891 | 1307 | 1682 | 1237 | 958 |

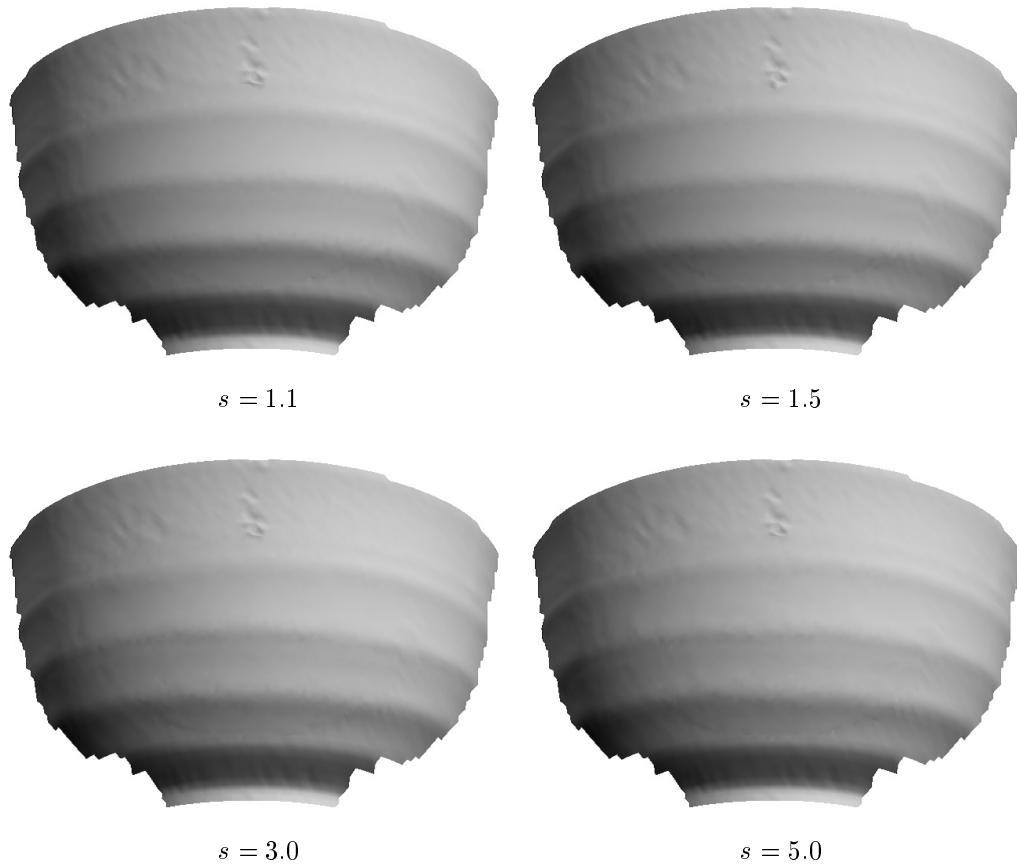


Figure 4.12. Results of Model (I) with different values of parameter s in weight w .

the data region.

From these results, it is considered that the increase of s in weight w makes the energy converge rapidly and the completion efficient. However, as for completion results, it is difficult to automatically determine optimal parameters and adjustment of the parameters by trial and error is required. Therefore, in future work, it will be necessary to establish a method which adaptively determines optimal parameters by analyzing the characteristics of the target model and the change of the shape in the completion process in order to realize efficient and effective completion.

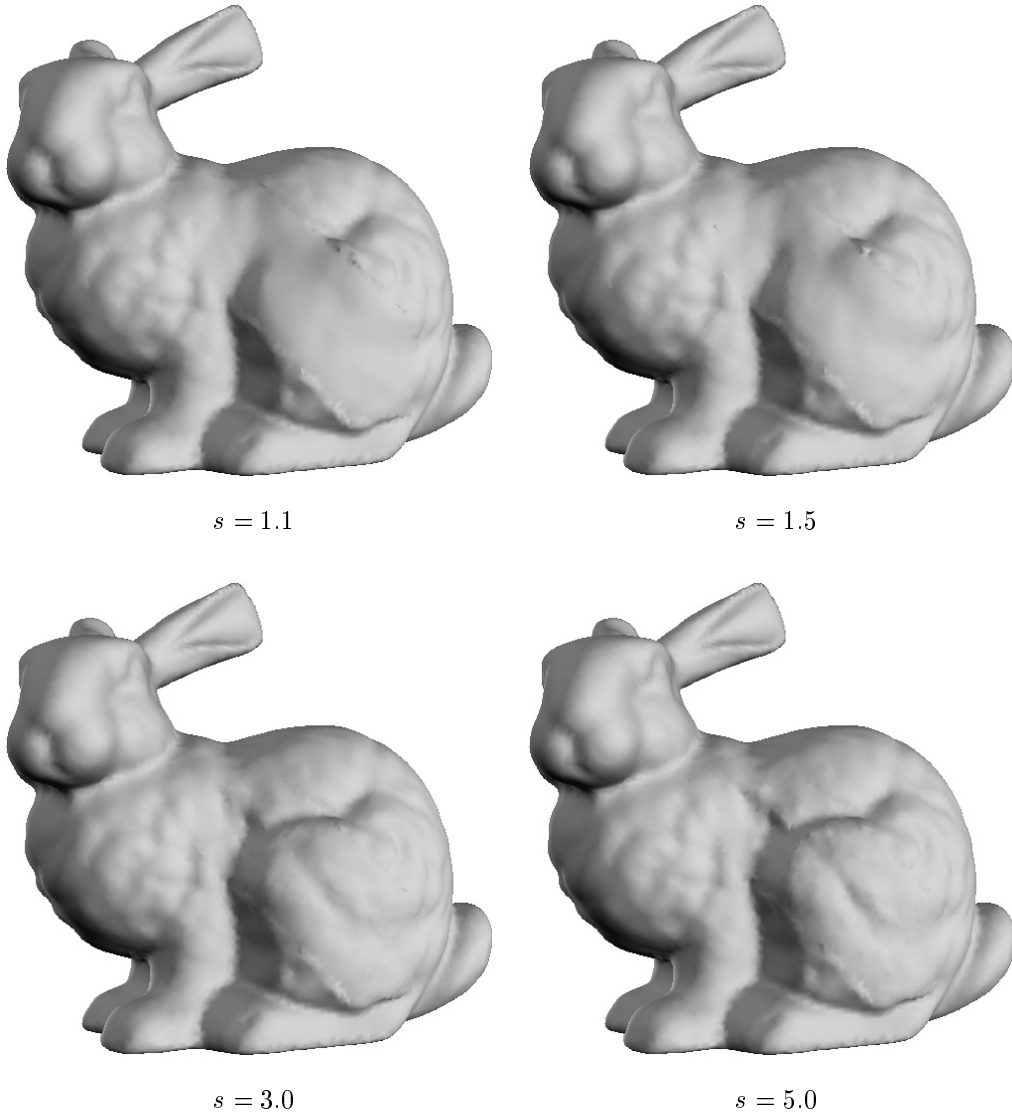


Figure 4.13. Results of Model (II) with different values of parameter s in weight w .

4.6. Conclusion

In this capture, a novel 3D surface completion method was presented. In the proposed method, an energy function, which represents implausibility of surface shape in missing regions, was defined based on pattern similarity SSD_s between local shapes in data and missing regions, and the missing region was plausibly filled in by minimizing the energy function.

Experiments were performed using three mesh models: a relatively simple bowl model with smooth surface shape; a relatively complex bunny model with rugged surface shape and a dented edge; and a model of a real building and the ground. In the experiments, the effectiveness of the proposed method was demonstrated by comparing the completed results of three mesh models by the proposed method and the conventional method. We have also confirmed that completion results by the proposed method were largely influenced by parameters in the energy function.

Chapter 5

Conclusion

5.1. Summary

In this thesis, novel methods for image, video and 3D surface completion based on a unified framework of energy minimization using pattern similarity measures have been proposed.

In Chapter 2, we have presented a novel image completion method for still images. In the proposed method, in order to obtain better results for many images, two factors were considered: (1) brightness changes of sample textures were allowed to obtain effective samples; and (2) spatial locality of textures was introduced as a constraint usually satisfied in many of real scenes. By considering these two factors, missing regions were successfully completed for many images with few unnatural brightness changes and few blurs. In experiments, the effectiveness of the proposed method has been demonstrated by qualitative evaluation using a questionnaire and quantitative evaluation using RMSE. We have also confirmed that the proposed method has some limitations. In addition, we have discussed the local minima problem. Then the reliability of the evaluation methods for image completion is discussed. From the discussion, we confirmed that many images and subjects are required to evaluate image completion methods persuasively in a questionnaire, and quantitative evaluation based on pixel-wise difference such as RMSE cannot be used as an absolute criterion, though it can be used for a rough evaluation.

In Chapter 3, we have presented a novel video completion method for om-

nidirectional video. In the proposed method, first, images projected on planes used for completion were generated from an omnidirectional video to compensate for the change in the appearance of textures considering the shapes around the missing region and extrinsic camera parameters. Then, missing regions of the projected images were completed by minimizing an energy function based on pattern similarity. Finally, omnidirectional video without invisible areas was generated using the completed images. In experiments, missing regions in images projected on planes were successfully completed and the effectiveness of energy minimization based on pattern similarity was demonstrated by comparing results of the proposed method with and without an energy minimization process. In addition, we confirmed that textures in missing regions change smoothly between successive frames. An omnidirectional telepresence without missing regions was also achieved and an improvement of realistic sensation in telepresence by filling in missing regions was confirmed.

In Chapter 4, we have presented a novel 3D surface completion method for 3D mesh models. In the proposed method, an energy function, which represents implausibility of surface shape in missing regions, was defined based on pattern similarity between local shapes in data and missing regions and the missing region was plausibly filled in by minimizing the energy function. In experiments, missing regions in three mesh models were completed. Then, the effectiveness of the proposed method was demonstrated by comparing results of three mesh models by the rendering of the proposed method and the conventional method. We have also confirmed that completion results by the proposed method were largely influenced by the parameters of the energy function.

5.2. Future work

This thesis has proposed novel methods for image, video and 3D surface completion. In order for completion methods to be used more practically, the following items should be investigated further.

- **Reduction of computational cost**

In the proposed methods for image, video and 3D surface completion, the

cost for searching for similar pattern is quite large. In order for the completion methods to be used practically for high-resolution images, videos and models, it is necessary to reduce computational costs. As a technique for reducing the computational cost, it is considered that using Graphics Processing Unit (GPU) could drastically decrease the search cost because the process of searching data regions for similar pattern is performed in parallel per pixel or vertex. As another technique, efficient searching methods could be applied to the proposed method to reduce computational costs for searching data regions for similar textures.

- **Surface completion for textured models**

In the proposed method for a 3D mesh model, only the structure is completed. We consider that surface completion for textured models could also be realized based on the same framework of energy minimization using the similarity measures of both textures and structures in order for completed models to be used practically.

The following items should be also investigated.

- Automatic determination of optimal parameters
- Video completion for omnidirectional video without planar assumption
- Super-resolution for increasing the resolution of textures in missing regions in omnidirectional video

Acknowledgements

This work was completed under the supervision of Professor Naokazu Yokoya and Assistant Professor Tomokazu Sato of the Graduate School of Information Science at the Nara Institute of Science and Technology.

I would especially like to thank my advisor, Professor Naokazu Yokoya, who has been an immeasurable source of support and guidance throughout my Ph.D. studies. Without the continuous encouragement and advice of Professor Yokoya, I could never have completed this thesis. I would also like to thank Assistant Professor Tomokazu Sato, who has also been a wonderful source of support and guidance throughout this study.

I am deeply grateful to Professor Kunihiro Chihara and Associate Professor Kazumasa Yamazawa, who have offered insightful comments on this research as members of the thesis committee.

I have also benefited greatly through being able to meet and work with a number of people during my doctoral work at the Nara Institute of Science and Technology. Assistant Professor Masayuki Kanbara of the Graduate School of Information Science at the Nara Institute of Science and Technology provided helpful comments and invaluable discussions. Secretaries Tomoyo Moriya and Mio Takahashi of the Vision and Media Computing Laboratory gave me invaluable support in daily affairs. I would also like to thank all the members of the Vision and Media Computing Laboratory of the Graduate School of Information Science at the Nara Institute of Science and Technology for their friendship and support.

Finally, I especially want to thank my family for their tireless understanding, support and encouragement during my Ph.D. studies.

References

- [AKY05] T. Asai, M. Kanbara, and N. Yokoya. 3D Modeling of Outdoor Environments by Integrating Omnidirectional Range and Color Images. *Proc. Int. Conf. on 3-D Digital Imaging and Modeling*, pp. 447–454, 2005.
- [AP06] C. Allène and N. Paragios. Image Renaissance Using Discrete Optimization. *Proc. IEEE Int. Conf. on Pattern Recognition*, pp. 631–634, 2006.
- [AS07] T. Amano and Y. Sato. Image Interpolation Using BPLP Method on the Eigenspace. *Systems and Computers in Japan*, Vol. 38, No. 1, pp. 87–96, 2007.
- [BBMN04] C.A. Barcelos, M.A. Batista, A.M. Martins, and A.C. Nogueira. Level Lines Continuation based Digital Inpainting. *Proc. IEEE Brazilian Symposium on Computer Graphics and Image Processing*, pp. 50–57, 2004.
- [BBS01] M. Bertalmio, A.L. Bertozzi, and G. Sapiro. Navier-Stokes, Fluid Dynamics, and Image and Video Inpainting. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 355–362, 2001.
- [BBSV01] C. Ballester, M. Bertalmio, V. Sapiro, and J. Verdera. Filling-In by Joint Interpolation of Vector Fields and Gray Levels. *IEEE Trans. on Image Processing*, Vol. 10, No. 8, pp. 1200–1211, 2001.

- [BCV⁺01] C. Ballester, V. Caselles, J. Verdera, M. Bertalmio, and G. Sapiro. A Variational Model for Filling-In Gray Level and Color Images. *Proc. IEEE Int. Conf. on Computer Vision*, pp. 10–16, 2001.
- [BF08] T.P. Breckon and R.B. Fisher. Three-Dimensional Surface Relief Completion via Nonparametric Techniques. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 30, No. 12, pp. 2249–2255, 2008.
- [BLLC02] R. Bornard, E. Lecan, L. Laborelli, and J. Chenot. Missing Data Correction in Still Images and Image Sequences. *Proc. ACM Int. Conf. on Multimedia*, pp. 355–361, 2002.
- [BS72] D.I. Barnea and H.F. Silverman. A Class of Algorithms for Fast Digital Image Registration. *IEEE Trans. on Computers*, Vol. c-21, No. 2, pp. 179–186, 1972.
- [BSCB00] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. Image Inpainting. *ACM Trans. on Graphics*, pp. 417–424, 2000.
- [BVSO03] M. Bertalmio, L. Vese, G. Sapiro, and S. Osher. Simultaneous Structure and Texture Image Inpainting. *IEEE Trans. on Image Processing*, Vol. 12, No. 8, pp. 882–889, 2003.
- [CFJ08] V. Cheung, B. Frey, and N. Jojic. Video Epitomes. *Int. Journal of Computer Vision*, Vol. 76, No. 2, pp. 141–152, 2008.
- [CKSO02] T. Chan, S. Kang, J. Shen, and S. Osher. Euler’s Elastica and Curvature Based Inpaintings. *SIAM Journal of Applied Mathematics*, Vol. 63, No. 2, pp. 564–592, 2002.
- [CL96] B. Curless and M. Levoy. A Volumetric Method for Building Complex Models from Range Images. *ACM Trans. on Graphics*, pp. 303–312, 1996.
- [CLF02] U. Castellani, S. Livatino, and R.B. Fisher. Improving Environment Modelling by Edge Occlusion Surface Completion. *Proc. Int. Symp.*

on *3D Data Processing, Visualization and Transmission*, pp. 672–675, 2002.

- [CPT04] A. Criminisi, P. Pérez, and K. Toyama. Region Filling and Object Removal by Exemplar-Based Image Inpainting. *IEEE Trans. on Image Processing*, Vol. 13, No. 9, pp. 1200–1212, 2004.
- [CS01a] T. Chan and J. Shen. Mathematical Models for Local Non-texture Inpaintings. *SIAM Journal of Applied Mathematics*, Vol. 62, No. 3, pp. 1019–1043, 2001.
- [CS01b] T. Chan and J. Shen. Non-texture Inpainting by Curvature-Driven Diffusions (CDD). *Journal of Visual Communication and Image Representation*, Vol. 12, No. 4, pp. 436–449, 2001.
- [DCOY03] I. Drori, D. Cohen-Or, and H. Yeshurun. Fragment-Based Image Completion. *ACM Trans. on Graphics*, pp. 303–312, 2003.
- [DMGL02] J. Davis, S.R. Marschner, M. Garr, and M. Levoy. Filling Holes in Complex Surfaces Using Volumetric Diffusion. *Proc. Int. Symp. on 3D Data Processing, Visualization and Transmission*, pp. 428–438, 2002.
- [EL99] A.A. Efros and T.K. Leung. Texture Synthesis by Non-parametric Sampling. *Proc. IEEE Int. Conf. on Computer Vision*, pp. 1033–1038, 1999.
- [ES03] S. Esedoglu and J. Shen. Digital Inpainting Based on the Mumford-shah-euler Image Model. *European Journal of Applied Mathematics*, Vol. 13, pp. 353–370, 2003.
- [FIMK07] R. Furukawa, T. Itano, A. Morisaka, and H. Kawasaki. Improved Space Carving Method for Merging and Interpolation Multiple Range Images Using Information of Light Sources of Active Stereo. *Proc. Asian Conf. on Computer Vision*, pp. 206–216, 2007.

- [FZ04] C. Früh and A. Zakhor. An Automated Method for Large-scale, Ground-based City Model Acquisition. *Int. Journal on Computer Vision*, Vol. 60, No. 1, pp. 5–24, 2004.
- [Goo] Google Inc. Google Street View. <http://www.google.com/help/maps/streetview/>.
- [Har01] P. Harrison. A Non-Hierarchical Procedure for Re-synthesis of Complex Textures. *Proc. Int. Conf. in Central Europe Computer Graphics, Visualization and Computer Vision*, pp. 190–197, 2001.
- [HKY07] M. Hori, M. Kanbara, and N. Yokoya. Novel Stereoscopic View Generation by Image-Based Rendering Coordinated with Depth Information. *Proc. Scandinavian Conf. on Image Analysis*, pp. 193–202, 2007.
- [HT96] A.N. Hirani and T. Totsuka. Combining Frequency and Spatial Domain Information for Fast Interactive Image Noise Removal. *ACM Trans. on Graphics*, pp. 269–276, 1996.
- [IKK02] S. Izoë, Y. Kenmochi, and K. Kotani. Image Restoration Using Vector Subspaces Based on Self-similarity. *Proc. Meeting on Image Recognition and Understanding*, Vol. 1, pp. 89–94, 2002 (in Japanese).
- [ISY05] S. Ikeda, T. Sato, and N. Yokoya. Immersive Telepresence System with a Locomotion Interface Using High-resolution Omnidirectional Videos. *Proc. IAPR Conf. on Machine Vision Applications*, pp. 602–605, 2005.
- [JT03] J. Jia and C. Tang. Image Repairing: Robust Image Synthesis by Adaptive ND Tensor Voting. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 643–650, 2003.
- [JTWT06] J. Jia, Y. Tai, T. Wu, and C. Tang. Video Repairing under Variable Illumination Using Cyclic Motions. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 28, No. 5, pp. 832–839, 2006.

- [KS05] V. Kraevoy and A. Sheffer. Template-based Mesh Completion. *Proc. Eurographics Symp. on Geometry Processing*, pp. 13–22, 2005.
- [KT06] N. Komodakis and G. Tziritas. Image Completion Using Global Optimization. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 442–452, 2006.
- [LKK03] A. Litvin, J. Konrad, and W. Karl. Probabilistic Video Stabilization Using Kalman Filtering and Mosaicking. *Proc. SPIE Electronic Imaging*, Vol. 5022, pp. 663–674, 2003.
- [LQS05] B. Li, Y. Qi, and X. Shen. An Image Inpainting Method. *Proc. IEEE Int. Conf. on Computer Aided Design and Computer Graphics*, pp. 531–536, 2005.
- [LZW03] A. Levin, A. Zomet, and Y. Weiss. Learning How to Inpaint from Global Image Statistics. *Proc. IEEE Int. Conf. on Computer Vision*, Vol. 1, pp. 305–312, 2003.
- [Mas04] T. Masuda. Filling the Signed Distance Field by Fitting Local Quadrics. *Proc. Int. Symp. on 3D Data Processing, Visualization and Transmission*, pp. 1003–1010, 2004.
- [Min80] M. Minsky. Telepresence. *OMNI magazine*, Vol. 2, pp. 45–52, 1980.
- [MM98] S. Masnou and J.M. Morel. Level Lines Based Disocclusion. *Proc. IEEE Int. Conf. on Image Processing*, Vol. 3, pp. 259–263, 1998.
- [MOG⁺06] Y. Matsushita, E. Ofek, W. Ge, X. Tang, and H. Shum. Full-Frame Video Stabilization with Motion Inpainting. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 28, No. 7, pp. 1150–1163, 2006.
- [MTO86] H. Maeda, K. Takahashi, and M. Ohta. A Restoration Scheme for Images with Missing Parts. *Trans. of the Institute of Electronics, Information and Communication Engineers*, Vol. J69-D, No. 1, pp. 91–97, 1986 (in Japanese).

- [OHK05] T. Ogawa, M. Haseyama, and H. Kitajima. Restoration Method of Missing Areas in Still Images Using GMRF Model. *Proc. IEEE Int. Symp. on Circuits and Systems*, Vol. 5, pp. 4931–4934, 2005.
- [OM04] A. Obata and S. Morita. Three Dimensional Image Recovery Using Three Dimensional Inpainting. *Technical report of the Institute of Electronics, Information and Communication Engineers*, PRMU 2004-41, pp. 49–54, 2004 (in Japanese).
- [Pau05] M. Pauly. Example-based 3D Scan Completion. *Proc. Eurographics Symp. on Geometry Processing*, pp. 23–32, 2005.
- [PGSQ06] S. Park, X. Guo, H. Shin, and H. Qin. Surface Completion for Shape and Appearance. *Int. J. of Computer Graphics*, Vol. 22, pp. 168–180, 2006.
- [PMW⁺08] M. Pauly, N.J. Mitra, J. Wallner, H. Pottmann, and L.J. Guibas. Discovering Structural Regularity in 3D Geometry. *ACM Trans. on Graphics*, Vol. 27, No. 3, pp. #43, 1–11, 2008.
- [Poi] Point Grey Research Inc. Ladybug. <http://www.ptgrey.com/products/spherical.asp>.
- [PSB07] K. Patwardhan, G. Sapiro, and M. Bertalmio. Video Inpainting Under Constrained Camera Motion. *IEEE Trans. on Image Processing*, Vol. 16, pp. 545–553, 2007.
- [RRS96] S.D. Rane, J. Remus, and G. Sapiro. Wavelet-Domain Reconstruction of Lost Blocks in Wireless Image Transmission and Packet-Switched. *Proc. Int. Conf. on Image Processing*, Vol. 1, pp. 309–312, 1996.
- [SACO04] A. Sharf, M. Alexa, and D. Cohen-Or. Context-based Surface Completion. *ACM Trans. on Graphics*, pp. 878–887, 2004.
- [Sho88] K. Shoji. A Method for Restoration of Missing Regions in Texture Images. *Trans. of the Institute of Electronics, Information and Communication Engineers*, Vol. J71-D, No. 9, pp. 1701–1708, 1988 (in Japanese).

- [SI08] R. Sagawa and K. Ikeuchi. Hole Filling of a 3D Model by Flipping Signs of a Signed Distance Field in Adaptive Resolution. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 30, No. 4, pp. 686–699, 2008.
- [SKYT02] T. Sato, M. Kanbara, N. Yokoya, and H. Takemura. Dense 3-D Reconstruction of an Outdoor Scene by Hundreds-baseline Stereo Using a Hand-held Video Camera. *Int. Journal on Computer Vision*, Vol. 47, No. 1-3, pp. 119–129, 2002.
- [SLCF06] Y. Shen, F. Lu, X. Cao, and H. Foroosh. Video Completion for Perspective Camera Under Constrained Motion. *Proc. IEEE Int. Conf. on Pattern Recognition*, pp. 63–66, 2006.
- [SYJS05] J. Sun, L. Yuan, J. Jia, and H. Shum. Image Completion with Structure Propagation. *ACM Trans. on Graphics*, pp. 861–868, 2005.
- [Tsc06] D. Tschumperlé. Curvature-Preserving Regularization of Multi-valued Images Using PDE's. *Proc. European Conf. on Computer Vision*, Vol. 2, pp. 295–307, 2006.
- [TSN04] T. Sato, S. Ikeda, and N. Yokoya. Extrinsic Camera Parameter Recovery from Multiple Image Sequences Captured by an Omnidirectional Multi-camera System. *Proc. European Conf. on Computer Vision*, Vol. 2, pp. 326–340, 2004.
- [VABF04] E. Villéger, G. Aubert, and L. Blanc-Féraud. Image Disocclusion Using a Probabilistic Gradient Orientation. *Proc. IEEE Int. Conf. on Pattern Recognition*, Vol. 2, pp. 52–55, 2004.
- [VCBS03] J. Verdera, V. Caselles, M. Bertalmio, and G. Sapiro. Inpainting Surface Holes. *Proc. Int. Conf. on Image Processing*, Vol. 2, pp. 903–906, 2003.
- [WO03] J. Wang and M.M. Oliveira. A Hole-filling Strategy for Reconstruction of Smooth Surfaces in Range Images. *Proc. SIBGRAPI03*, pp. 11–18, 2003.

- [WSI07] Y. Wexler, E. Shechtman, and M. Irani. Space-Time Completion of Video. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 29, No. 3, pp. 463–476, 2007.
- [XMQ04] H. Xie, K.T. McDonnell, and H. Qin. A Finite Element Method for Surface Restoration with Smooth Boundary Conditions. *Computer Aided Geometric Design Archive*, Vol. 21, No. 5, pp. 427–445, 2004.
- [YFKM08] A. Yamashita, I. Fukuchi, T. Kaneko, and T. Miura. Removal of Adherent Noises from Image Sequences by Spatio-temporal Image Processing. *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 2386–2391, 2008.
- [YOT05] M. Yasuda, J. Ohkubo, and K. Tanaka. Digital Images Inpainting based on Markov Random Field. *Proc. Int. Conf. on Computational Intelligence for Modeling Control and Automation and Int. Conf. on Intelligent Agents, Web Technologies and Internet Commerce*, Vol. 2, pp. 747–752, 2005.
- [ZS03] H. Zhao and R. Shibasaki. Reconstructing a Textured CAD Model of an Urban Environment Using Vehicle-borne Laser Range Scanners and Line Cameras. *Machine Vision and Applications*, Vol. 14, No. 1, pp. 35–41, 2003.
- [ZXS05] Y. Zhang, J. Xiao, and M. Shah. Motion Layer Based Object Removal in Videos. *Proc. IEEE Workshops on Application of Computer Vision*, Vol. 1, pp. 516–521, 2005.

List of Publications

Journal Papers

1. N. Kawai, T. Sato and N. Yokoya. Image Inpainting by Minimizing Energy Function Based on Pattern Similarity Considering Brightness Change and Spatial Locality. *Transactions of the Institute of Electronics, Information and Communication Engineers of Japan*, Vol. J91-D, No. 9, pp. 2293-2304, Sep. 2008 (in Japanese). (Chapter 2).
2. N. Kawai, T. Sato and N. Yokoya. 3D Surface Completion by Minimizing Energy Based on Similarity of Local Shape. *Transactions of the Virtual Reality Society of Japan*, Vol. 15, No. 1, March 2010 (in Japanese). (to appear). (Chapter 4).

Technical Letters

1. N. Kawai, T. Sato and N. Yokoya. A Study on Qualitative and Quantitative Evaluation for Image Inpainting. *Information Technology Letters*, Vol. 6, pp. 245-248, Sep. 2007 (in Japanese). (Chapter 2).

International Conferences

1. N. Kawai, T. Sato and N. Yokoya. Image Inpainting Based on Energy Minimization. *Proc. SPIE Electronic Imaging*, Vol. 6498, pp. 649813-649813-9, Jan. 2007. (Chapter 2).
2. N. Kawai, T. Sato and N. Yokoya. Image Inpainting Considering Brightness Change and Spatial Locality of Textures. *Proc. Int. Conf. on Computer Vision Theory and Applications (VISAPP)*, Vol. 1, pp. 66-73, Jan. 2008. (Chapter 2).
3. N. Kawai, T. Sato and N. Yokoya. Surface Completion by Minimizing Energy Based on Similarity Measure of Local Shape. *MIRU Int. Workshop on Computer Vision 2008 (MIRU-IWCV2008)*, July 2008. (Chapter 4).

4. N. Kawai, T. Sato and N. Yokoya. Surface Completion by Minimizing Energy Based on Similarity of Shape. *Proc. IEEE Int. Conf. on Image Processing (ICIP2008)*, pp. 1532-1535, Oct. 2008. (Chapter 4).
5. N. Kawai, T. Sato and N. Yokoya. Image Inpainting Considering Brightness Change and Spatial Locality of Textures and Its Evaluation. *Proc. Pacific-Rim Symposium on Image and Video Technology (PSIVT2009)*, pp. 271-282, Jan. 2009. (Chapter 2).
6. Y. Awatsu, N. Kawai, T. Sato and N. Yokoya. Spatio-temporal Super-Resolution Using Depth Map. *Proc. Scandinavian Conf. on Image Analysis (SCIA2009)*, pp. 696-705, June 2009.
7. N. Kawai, K. Machikita, T. Sato and N. Yokoya. Generation of an Omnidirectional Video without Invisible Areas Using Image Inpainting. *Proc. Asian Conf. on Computer Vision (ACCV2009)*, pp. 359-370, Sep. 2009. (Chapter 3).
8. N. Kawai, T. Sato and N. Yokoya. Efficient Surface Completion Using Principal Curvature and Its Evaluation. *Proc. IEEE Int. Conf. on Image Processing (ICIP2009)*, pp. 521-524, Nov. 2009. (Chapter 4).

Domestic Conferences

1. N. Kawai, T. Sato and N. Yokoya. Image Inpainting Based on Energy Minimization. *Proc. Forum on Information Technology (FIT2006)*, Vol. 3, pp. 177-178, Sep. 2006 (in Japanese). (Chapter 2).
2. K. Sugie, M. Uenoyama, N. Kawai, T. Suenaga, Y. Matsumoto and T. Ogasawara. Development of Public Guide Robot "IKO-CHAN". *Proc. Annual Conference of the Robotics Society of Japan*, pp. 3I21, Sep. 2006 (in Japanese).
3. N. Kawai, T. Sato and N. Yokoya. Image Inpainting by Minimizing an Energy Function of Pattern Similarity. *Technical Report of the Institute*

of Electronics, Information and Communication Engineers of Japan, Pattern Recognition and Media Understanding (PRMU), PRMU2006-106, Dec. 2006 (in Japanese). (Chapter 2).

4. N. Kawai, T. Sato and N. Yokoya. Image Inpainting by Minimizing Energy Based on Pattern Similarity Considering Brightness Change and Spatial Locality. *Proc. Meeting on Image Recognition and Understanding (MIRU2007)*, pp. 227-234, July 2007 (in Japanese). (Chapter 2).
5. Y. Awatsu, N. Kawai, T. Sato and N. Yokoya. Generation of Super-resolved Image from Video Using Depth Maps. *Proc. Kansai-section Joint Convention of Institutes of Electrical Engineering*, No. G13-23, Nov. 2007 (in Japanese).
6. N. Kawai, T. Sato and N. Yokoya. Surface Completion by Minimizing Energy Based on Pattern Similarity of Shape. *Proc. Symposium on Pattern Measurement*, pp. 19-24, Nov. 2007 (in Japanese). (Chapter 4).
7. N. Kawai, T. Sato and N. Yokoya. Surface Completion by Minimizing Energy Based on Local Shape Similarity. *Proc. the 2008 Institute of Electronics, Information and Communication Engineers, General Conference*, No. D-12-78, March 2008 [Young Researchers' Award] (in Japanese). (Chapter 4).
8. Y. Awatsu, N. Kawai, T. Sato and N. Yokoya. Spatio-temporal Super-Resolution Using Depth Maps. *Technical Report of the Institute of Electronics, Information and Communication Engineers of Japan, Pattern Recognition and Media Understanding (PRMU)*, PRMU2007-311, March 2008 (in Japanese).
9. N. Kawai, T. Sato and N. Yokoya. Surface Completion by Minimizing Energy Based on Similarity Measure of Local Shape. *Proc. Meeting on Image Recognition and Understanding (MIRU2008)*, pp. 272-277, July 2008 (in Japanese). (Chapter 4).
10. Y. Awatsu, N. Kawai, T. Sato and N. Yokoya. Spatio-temporal Super-Resolution Using Depth Maps. *Proc. Meeting on Image Recognition and*

Understanding (MIRU2008), pp. 188-193, July 2008 (in Japanese).

11. N. Kawai, T. Sato and N. Yokoya. Efficiency Improvement and Quantitative Evaluation for 3D Model Completion by Minimizing Energy. *Proc. Symposium on Pattern Measurement*, pp. 49-56, Nov. 2008 [Excellent Paper Award](in Japanese). (Chapter 4).
12. K. Machikita, N. Kawai, T. Sato and N. Yokoya. Generation of Omnidirectional Video without Invisible Areas Using Image Inpainting. *Proc. the 2008 Institute of Image Information and Television Engineers Winter Annual Convention*, No. 9-7, Dec 2008 (in Japanese). (Chapter 3).
13. K. Machikita, N. Kawai, T. Sato and N. Yokoya. Generation of Omnidirectional Video without Invisible Area Based on Image Inpainting. *Technical Report of the Institute of Electronics, Information and Communication Engineers of Japan, Pattern Recognition and Media Understanding (PRMU)*, PRMU2008-245, March 2009 (in Japanese). (Chapter 3).
14. K. Machikita, N. Kawai, T. Sato and N. Yokoya. Realization of Omnidirectional Telepresence without Invisible Areas Using an Omnidirectional Camera Based on Video Inpainting. *Proc. Meeting on Image Recognition and Understanding (MIRU2009)*, pp. 391-397, July 2009 (in Japanese). (Chapter 3).

Awards

1. Young Researchers' Award of the Institute of Electronics, Information and Communication Engineers (IEICE), 2009.
2. Excellent Paper Award of Technical Committee on Pattern Measurement, the Society of Instrument and Control Engineers (SICE), 2008.