Four-dimensional City Modeling using Vehicular Imagery

TOHOKU UNIVERSITY
Graduate School of Information Science

Four-dimensional City Modeling using Vehicular Imagery
(車載画像を用いた都市の４次元モデリング)

A dissertation submitted for the degree of Doctor of Philosophy
(Information Science)

Department of System Information Sciences

by

Ken SAKURADA

January 13, 2015

Four-dimensional City Modeling using Vehicular Imagery

Ken Sakurada

Abstract

This dissertation presents the methods for estimating scene changes of entire cities using vehicular imagery. The estimation target is the areas damaged by tsunami due to Great East Japan Earthquake on March 11, 2011. A large number of architectures are damaged in many urban and residential areas in the coastlines of Iwate, Miyagi, and Fukushima prefectures located in the north-eastern part of Japan. The objective of this study is to develop the methods that can visualize the tsunami-damages and the recovery/reconstruction process.

Since about one month after the earthquake, we started recording the damage and recoveries of the tsunami-damaged areas using a vehicle-mounted omni-directional camera. In the first one month, this activity mostly covered the entire devastated areas across the three prefectures whose total length is almost 400 kilometers. The image archive activity periodically acquires the images of the tsunami-damaged areas to estimate temporal changes of the areas.

To estimate temporal changes of such a wide area using vehicular imagery, there are three challenges to overcome. The first challenge is the limitation of view point of camera. Basically, vehicle-mounted camera only captures the scene alongside street. Three-dimensional reconstruction of a scene requires images taken at various view points. The limitation of view points causes ambiguity of scene depth and makes it difficult to reconstruct dense 3D model of the scene. Second, vehicle image cannot cover occluded areas or unreachable areas. Vehicular camera can capture higher resolution images of vertical structures and has better access to information about covered areas than aerial image. It is also less affected by weather conditions. However, vehicle image is constrained to the ground plane and a single image has limited physical range. The third challenge is the large amount of computation since one city has several thousands to several tens of thousands of image. It is computationally prohibitive to estimate scene change of regional-scale area using 3D model and pixel-level registration.

This dissertation approaches these challenges by the following strategies. First, the 2D-based method roughly but quickly detects scene change of entire areas from an image pair. Next, the 3D-based method detects accurate structural change where detailed analysis is

necessary. Finally, the method of land surface condition analysis estimates city-scale temporal change based on object recognition.

To reduce computational time, unlike 3D-based approach, the 2D-based method detects scene changes from an image pair without 3D model of a scene and pixel-level registration. The method makes it possible to process the entire tsunami-damaged areas with a single workstation. The proposed method detects scene change in grid resolution and refines the result into pixel-level using superpixel segmentation. The method makes use of high discrimination of Convolutional Neural Network (CNN) as a grid feature. To validate the proposed approach, this study introduces Panoramic Change Detection Dataset which is manually created for this task. The experimental results show that the proposed method effectively integrates high discrimination of CNN feature and accurate segmentation of superpixel.

Next, the 3D-based method detects accurate structural change of a scene using multi-view images where the 2D-based methods detect big changes. The method estimates scene structures probabilistically, not deterministically. Based on structure estimates, the method evaluates the probability of structural changes in the scene, where the inputs are the similarity of the local image patches among the multi-view images. The aim of the probabilistic treatment is to maximize the accuracy of change detection under uncertainty. The proposed method is compared to the methods that use multi-view stereo (MVS) to reconstruct the scene structures of the two time points and then differentiate them to detect changes. The experimental results show that the proposed method outperforms such MVS-based methods.

To estimate change of debris distribution in a city, this dissertation presents a unified framework for robustly integrating image data taken at vastly different viewpoints, namely, street-level and aerial images. The method generates large-scale estimates of land surface conditions based on object recognition. The strategy uses macro-level imaging to learn land condition correspondences between land regions that share similar visual characteristics, while micro-level images are used to acquire high resolution statistics of land conditions. The experimental results show that our approach can effectively integrate both macro (aerial) and micro-level (vehicle) images, along with other forms of metadata, to estimate city-scale debris. Furthermore, the experiments show that the detection method can be successfully applied to vegetation estimation.

This dissertation achieved the objective of developing the methods for 4D city modeling in tsunami-damaged area using vehicular imagery. The proposed methods overcome the three challenges mentioned above and make it possible to estimate change of the entire tsunami-damaged area. If scene images of cities will be available in real-time in the future, the proposed method can extend to real-time monitoring of the city.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

On March 11th, 2011, Great East Japan Earthquake brought catastrophic damage to the north east of Japan. The earthquake centered at 70 km offshore of Miyagi prefecture and recorded the magnitude of 9.0. The Tsunami caused by the earthquake reached 40.1 meters height at maximum. The earthquake caused giant Tsunami whose maximum height was 40.1 meter and the Tsunami gave serious damages to the Pacific coast area of the Tohoku. The great earthquake, the giant Tsunami and the aftershocks caused landslide disaster, fire, land subsidence and ground liquefaction. The secondary disasters spread to a very wide area including Fukushima prefecture where first nuclear power plant resulted in the release of radioactive substances after the power loss caused by the Tsunami. The earthquake triggered an all-time wide-area complex disaster.

Accurate understanding of damage and temporal scene change is important to reduce secondary damage and quick recovery and restoration. Aerial image is one of the most frequently used sensory information to investigate a wide area damaged by a disaster. For example, it is possible to observe the land-surface condition using a satellite image of visible light, and estimate an area condition such as inundation using aerial image of infrared light and microwave both during day and night. However, aerial image has some drawbacks, such as low resolution and large variation of illumination condition due to weather change, since aerial image observes the ground from high-altitude in the sky. Furthermore, there are many areas invisible from the sky due to coverage by a roof, an elevated road and a pedestrian bridge (Fig.1.1). Street-view image is essential to supplement such missing observation from aerial image and understand the detail of city condition.

The objective of this study is to visualize the damage and recovery/restoration process of tsunami-damaged area. About one month after the earthquake, we started recording

Figure 1.1: Areas unseen from a satellite

the damages and recoveries of tsunami-damaged areas driving a car on which an omni-directional camera and a GPS receiver are mounted. The time interval of the recording is from 2 to 6 months depending on the recovery progress of the areas. The target area is the costal area of almost 500 kilometer which observed serious Tsunami-damages caused by Great East Japan Earthquake in 2011 (from Aomori to Fukushima prefecture). We have recorded about 40 terabytes of image data so far.

Figure 1.2 describes the overview of 4-dimensional city modeling using vehicular imagery. The image archival activity has been periodically recording the scenes of the cities in the tsunami-damaged areas. From the periodical observation, the 4D modeling method detects scene change, and estimates city-scale and regional-scale temporal changes. The simplest method for estimating temporal change is to directly differentiate results of three dimensional reconstruction. However, to estimate temporal changes of regional-scale area using vehicular imagery, there are challenges to overcome. First, vehicle-mounted camera only captures the scene alongside street. The limited view points causes the depth ambiguity and makes it difficult to densely reconstruct the 3D structure of the scene. Second, vehicle image cannot cover occluded areas and unreachable areas. A single vehicle image has limited physical range. Third, regional-scale change detection requires too much computational resources since one city has several thousands to several tens of thousands of image pairs, especially for 3D reconstruction and pixel-level registration.

The strategy for this dissertation to overcome these challenges is as follows. First, the proposed method roughly but quickly detects 2D scene change of entire areas from an image pair. Next, the method detects accurate structural change where detailed analysis is necessary. Finally, the method estimate city-scale temporal change. This dissertation proposes a novel method for 2D change detection, structural change detection and city-scale land surface condition analysis.

Chapter 2 Digital Image Archive of Tsunami-damaged Area This chapter describes the image dataset built for studying 4D city modeling. The archival process started since about one month after the Japan earthquake of March 11, 2011, and accumulated about 40 TB of data. The proposed methods estimate the recovery process of

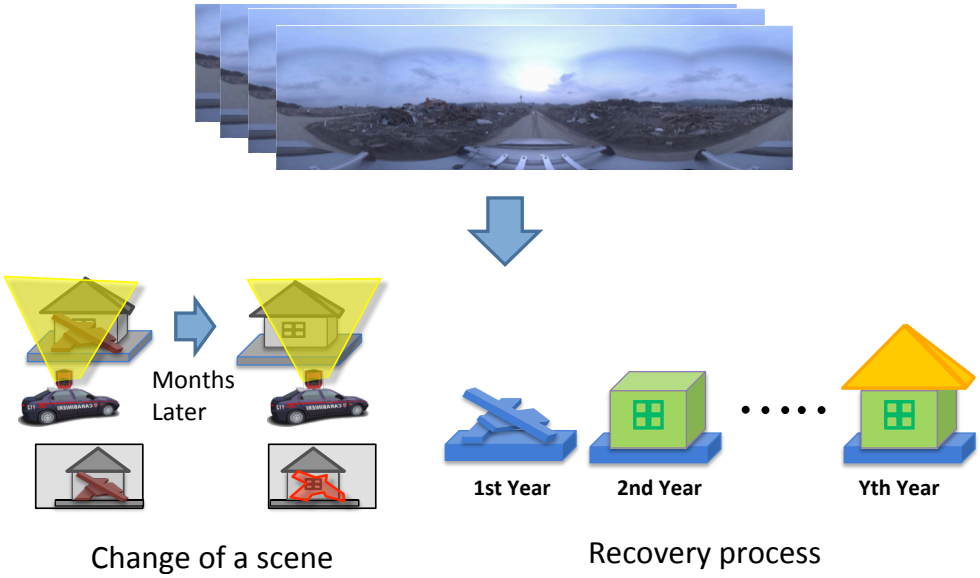


Figure 1.2: Overview of 4-dimensional city modeling using vehicular imagery

the tsunami-damaged area using this image dataset.

Chapter 3 Three-dimensional Reconstruction As preliminary study, this chapter shows the results of sparse and dense 3D reconstructions. For sparse reconstruction, a standard Structure from Motion (SfM) is performed which is extended to omnidirectional image. Using the camera poses of the SfM, Patch-based Multi-view Stereo (PMVS2) [1] generates dense city-models. Furthermore, this chapter shows the result of temporal change detection that naively compares the reconstructed structure over time. The results show that the naive method is not enough to understand the detail of city condition. To overcome this challenge, this dissertation proposes the following three methods for 4D city modeling.

Chapter 4 2D Change Detection This chapter proposes the method to detect 2D scene changes from an image pair using grid feature. Several previous approaches of change detection require 3D model of a scene and pixel-level registration between different time images. In the case that 3D model is not available, it is difficult to directly apply the previous methods to the change detection problem. Furthermore, it is computationally prohibitive to estimate scene change of wide area using 3D model and pixel-level registration. The proposed method can detect scene change without pixel-level registration integrating convolutional neural network (CNN) feature with superpixel segmentation. The method can reduce the computational time and detect change of entire tsunami-damaged areas. The experimental results show that the proposed method effectively integrates high discrimination of CNN feature and accurate segmentation of superpixel.

Chapter 5 3D Change Detection This chapter describes a method for detecting

temporal changes of the three-dimensional structure of an outdoor scene from its multi-view images captured at two separate times. The method estimates scene structures probabilistically, not deterministically, and based on their estimates, the method evaluates the probability of structural changes in the scene, where the inputs are the similarity of the local image patches among the multi-view images. The proposed method is compared to the approach that use multi-view stereo (MVS) to reconstruct the scene structures of the two time points and then differentiate them to detect changes. The experimental results show that the proposed method outperforms MVS-based methods.

Chapter 6 Land Surface Condition Analysis This chapter presents a unified framework for robustly integrating image data taken at vastly different viewpoints to generate large-scale estimates of land surface conditions. To validate the proposed approach, this study attempts to estimate the amount of post-tsunami damage over the entire city of Kamaishi, Iwate Prefecture (over 4 million square-meters). The results show that the proposed approach can effectively integrate both micro and macro-level images, along with other forms of meta-data, to effectively estimate city-scale phenomena. Experiments evaluate the proposed approach on two modes of land condition analysis, namely, city-scale debris and greenery estimation, to show the ability of the proposed method to generalize to a diverse set of estimation tasks.

Chapter 7 Conclusion The thesis concludes with a summary, and discusses a consideration of future extensions of this work, including open and remaining questions.

1.2 Related Work

This section will review the previous work relevant to understanding 4-dimensional city modeling in terms of temporal change detection and city-scale analysis.

1.2.1 City Modeling

The problem of measuring and documenting a city is the objective of photogrammetry, remote sensing and computer vision community [2, 3, 4, 5, 6]. City modeling is, for example, 3D reconstruction, land-use mapping and scene change estimation. There are many input data types to reconstruct a city other than image, for example, light detection and ranging (LiDAR), digital elevation map (DEM), digital terrain model (DTM) and digital surface model (DSM). The followings focus on automatic methods using image and LiDAR. For the method using other data sources and interactive methods, please refer to the paper [7].

There are multiple types of devices to measure a city, for example, digital camera and LiDAR mounted on mobile devices or systems such as smartphone, vehicle, UAV, airplane and satellite. Snavely et al. propose a method to reconstruct an entire city using unstructured images which were captured from a variety of view points using mobile devices and uploaded on the Internet [8]. Pollefeys et al. proposed an approach for dense 3D reconstruction from unregistered Internet-scale photo collections with about 3 million of images within the span of a day on a single PC [9]. Furthermore, Pollefeys developed a system for automatic, georegistered, real-time 3D reconstruction from video of urban scenes [10].

Poullis and You proposed a method for massive city-scale reconstruction using imagery and LiDAR [11]. This system automatically creates lightweight, watertight polygonal 3D models from LiDAR data captured by an airborne scanner[12, 13, 14, 11]. The technique is based on the statistical analysis of the geometric properties of the data, which makes no particular assumptions about the input data. Zhou and Neumann proposed a similar approach [15, 16]. Lafarge and Mallet developed a method for modeling cities from 3D-point data providing a more complete description than existing approaches by reconstructing simultaneously buildings, trees and topologically complex grounds [17, 18]. Cabezas et al. proposed an integrated probabilistic model for multimodal fusion of aerial imagery [19], LiDAR data and GPS measurements. The model of their method allows for analysis and dense reconstruction (in terms of both geometry and appearance) of large 3D scenes. One of its advantages is that it explicitly models uncertainty and allows for missing data. This work takes the advantages of the city modeling methods.

1.2.2 Temporal Change Detection

Many researchers have worked on temporal change detection of a scene. However, most of them consider the detection of 2D changes (i.e., those only in image appearance), whereas the objective of this study is to detect changes in 3D structure of scenes.

The standard problem formulation of 2D change detection [20, 21] is an appearance model of a scene is learned using its n images and then based on $n + 1^{st}$ image, it is determined whether a significant change has occurred. Most of the studies of 3D change detection [22, 23, 24, 20, 25] follow a similar formulation; namely, a model of the scene in a "steady state" is built and a newly-captured image(s) is compared against it to detect changes.

In [20], targeting at aerial images capturing a ground scene, Pollard and Mundy proposed a method that learns a voxel-based appearance model of a 3D scene from its 20–40 images. Crispell et al. later improved method to minimize storage space is presented in [22]. In [23], Ibrahim and David proposed a method that detects scene changes by estimating the appearance or disappearance of line segments in space. All of these studies create an appearance model of the target scene from a sufficiently large number of images, unfortunately, this approach does not work due to lack of images. Such an approach is appropriate for aerial or satellite imagery or the case of stationary cameras, but is not appropriate for the images taken in our setting.

The alternative approach is to obtain a 3D model of the scene from other sensors or methods than the images used for the change detection. In [24], assuming that the 3D model of a building is given, the edges extracted in its aerial images are matched with the projection of the 3D model to detect changes. The recent study of Taneja et al. [25] is of the same type. Their method detects temporal changes of a scene from its multi-view images, and thus it is close to ours from an application point of view. However, their motivation is to minimize the cost needed for updating the 3D model of a large urban area, and thus, they assume that a dense 3D model of the target scene is given.

The proposed method in this dissertation differs from all of the above in the formulation of the problem. In the proposed formulation, the changes of a scene are detected from two sets of images taken at two different time points. The two image sets are "symmetric" in a sense that they have similar sizes and are of the same nature. The proposed method does not assume that a dense 3D model of the scene is given, or created from the input images themselves, as it is difficult for the images captured from a ground vehicle-mounted camera. If the dense model is required, it is necessary to have a large number of multi-view images captured from a variety of viewpoints [8, 26, 10, 27, 28, 29], or to use a range sensor.

In the sense that the input data are symmetric, the proposed method might be close to

the study of Schindler and Dellaert [30]. They propose a method that uses a large number of images of a city that are taken over several decades to perform several types of temporal inferences, such as estimating when each building in the city was constructed. However, besides the necessity for a large number of images, their method represents scene changes only in the form of point clouds associated with image features.

1.2.3 City-scale Surface Condition Analysis

There has been significant advances in the state-of-the-art techniques for quantitative geometric interpretations of large-scale city scenes. Methods for city-scale 3D reconstruction have been proposed using thousands of images gathered from Internet images [27, 8]. Similar techniques have been proposed for images captured by a vehicle-mounted camera [10, 31] or aerial images [32, 33, 34]. Street-view images have also been combined with aerial images for the purpose of improving 3D reconstruction, where 3D point clouds have been projected to the ground plane and aligned with edges of buildings detected from aerial images [35] or building maps [36]. There has also been work using aerial and street view images taken several months or decades apart [24, 21, 20, 30, 37] to understand temporal changes of a scene. The focus of these previous approaches are on a quantitative geometric interpretation of the scene where local visual features are matched directly to estimate camera pose using epipolar geometry [38]. This work aims to push beyond a purely geometric understanding of the scene towards a more qualitative understanding of city conditions. For instance, the aim is not only to estimate the 3D geometry of a building but also the condition of the building or the condition of the ground surrounding a building.

There also has been work focused on the qualitative estimation of land condition over large-scale environments. In the field of remote sensing, coarse land surface conditions have been estimated using aerial color images, aerial infrared light and aerial microwave sensing [39, 40, 41, 42, 43, 44]. Color aerial images have been applied to land condition estimation for vegetation monitoring [45, 46, 47], land cover mapping, and flood risk and damage assessment [48, 49]. For example, forest maps [50, 51, 52] are an important source of information for monitoring and reducing deforestation, allowing environmental scientists to know how forested areas increase or decrease in over the entire earth.

Apart from aerial imaging using color cameras, many other modes of sensing have been proposed for estimating coarse large-scale land surface conditions. Digital elevation map (DEM) [50], Spectroradiometer (MODIS), high resolution radiometer (AVHRR) and Synthetic Aperture Radar (SAR) have been proposed to improve accuracy of estimating large-scale land surface condition. However the resolution of satellite-mounted MODIS and AVHRR only measure surface conditions over a very rough resolution – typically over

a cell size of a several hundred meters. As such, these works do not utilize street-level sensing which are too detailed for their estimation task. However, this work aims at estimating land conditions on a cell size closer to 20 meters wide.

The proposed work fills a void between detailed geometric reconstructions of city-scale structures and coarse qualitative estimation of land conditions. The proposed method uses known techniques to provide an accurate geometric model of the city and use state-of-the-art object recognition results carefully registered to the scene geometry to understand the qualitative conditions of the entire city.

Chapter 2

Tsunami Damage Archive

This chapter discusses about the detail of the image dataset used in this research. We have been recording images in tsunami-devastated areas using a vehicle mounted camera since about one month after the earthquake. This image dataset consists of city-scale street-view images of different times. This research proposed some methods estimating city-condition and temporal change from the dataset.

2.1 Image Acquisition

Since about one month after the earthquake, we started recording the damages and recoveries of these areas mainly using a vehicle-mounted omni-directional camera (Fig. 2.1).

The image archive activity is periodically acquiring the images of the tsunami-devastated areas in the northern-east coast of Japan. The images are captured by a vehicle having an omni-directional camera (Ladybug3 of Point Grey Research Inc.) on its roof. An image is captured at about every 2m on each city street to maintain the running speed of the vehicle under the constraint of the frame rate of the camera.

2.1.1 Measurement Vehicle

Figure 2.1 shows our measurement vehicle which mounts an omni-directional camera (Ladybug3 or Ladybug5 of Point Grey Research Inc.) and a receiver of Differential Global Positioning System (DGPS) (R100 of Hemisphere Inc.). A Ladybug camera has six CCD image sensors. Figure 2.2(a) shows image of each camera of Ladybug. Using these raw images, computational photography method can generate omnidirectional panoramic image (Fig.2.2(b)), perspective image of arbitrary view-direction (Fig.2.2(c)) and image of dome projection (Fig.2.2(d)). In this research, Structure from Motion (SfM) uses the panoramic image and recognition methods use perspective image. Our approach uses perspective image cropped in the left or right direction since images in the left and right

Figure 2.1: Measurement vehicle equipping an omnidirectional camera (Ladybug 3) and GPS.

direction have rich information of city scene.

2.1.2 Measured area

Figure 2.3 shows the measured area period of this image archive activity. In the first one month, this activity mostly covered the entire devastated areas across the three prefectures whose total length is almost 400 kilometers. Figures 2.4 and 2.5 show periodically measured area in Kamaishi and Rikuzentakata, respectively. The color line shows a trajectory of our measurement vehicle. Different color shows different time data. The blue circles show the area where ordinary people could enter because of recovery operations one year after the tsunami. It takes about two weeks to measure the entire devastated areas across the three prefectures. We have gotten about 40 terabytes of image data until December, 2014.

This image archive activity is different from similar activities conducted by other parties such as Google Inc. in that the goal of this activity is to record the temporal changes of these areas and thus we have been periodically recorded these areas.

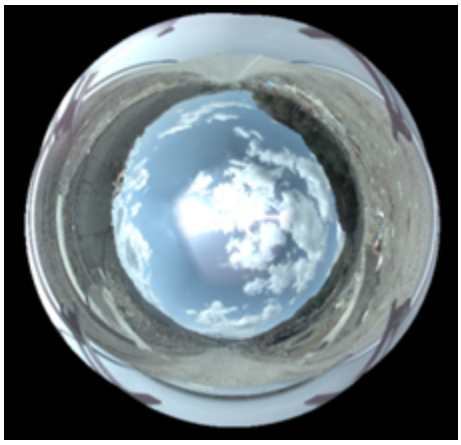
2.2 Temporal changes

Figure 2.6 shows the examples of panoramic images which we periodically captured in the tsunami-devastated areas. It is possible to understand from these images that there are temporal changes. For example, big damages due to tsunami and recovery operations. However, it is not easy to understand damage and recovery process of an entire city only by looking at these images. Furthermore, these images have differences of viewpoint and illumination condition between different time data. The proposed method of this dissertation enables it to automatically estimate and visualize temporal change of an entire city using street-view images and other metadata.

(a) Images of each camera.

(b) Omnidirectional panoramic image.

(c) Image of arbitrary view-direction.



(d) Images of dome projection.

Figure 2.2: Different image type of Ladybug.

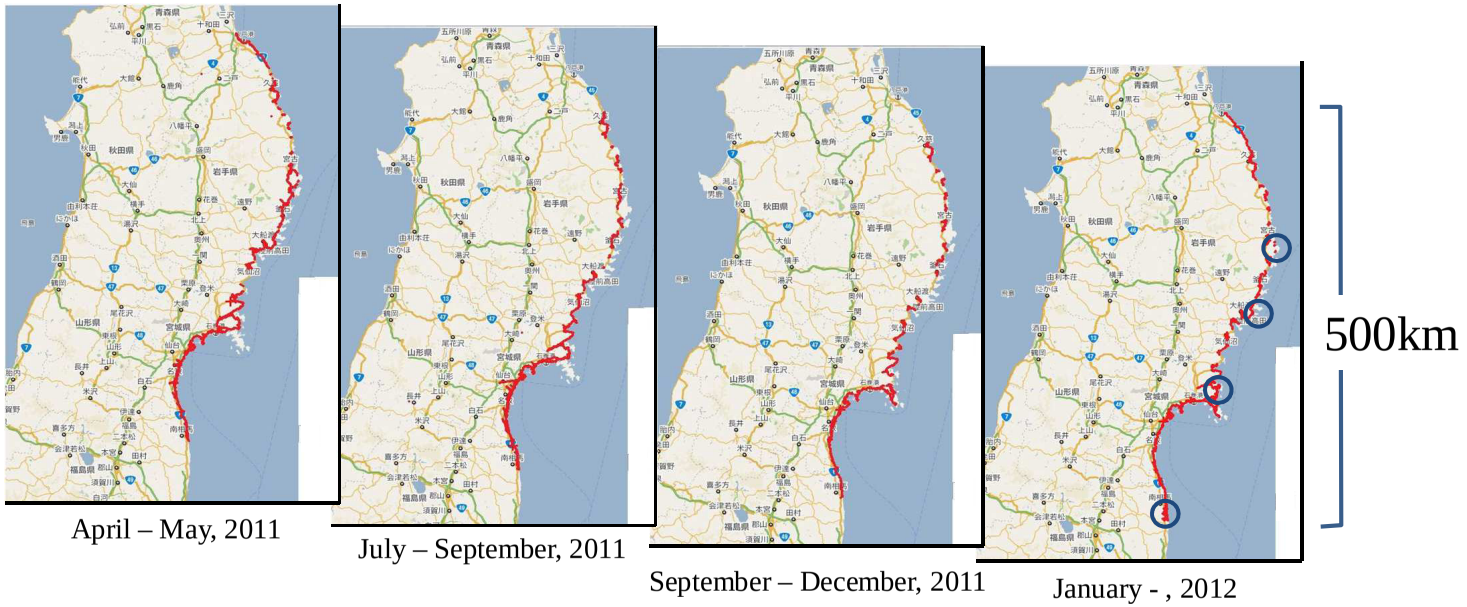


Figure 2.3: Area and period of our image archive activity.

Figure 2.4: Measured area in Kamaishi. The lines show the trajectories of the measurement vehicle. Different color shows different time data.

Figure 2.5: Measured area in Rikuzentakata. The lines show the trajectories of the measurement vehicle. Different color shows different time data.

Figure 2.6: Example of temporal changes in tsunami-devastated areas.

Chapter 3

3D Reconstruction

This chapter explains methods to reconstruct three-dimensional structure of a scene using a sequence of omnidirectional panoramic images and to estimate temporal changes using the reconstruction results.

The simplest baseline for estimating temporal change is to directly differentiate results of three dimensional reconstruction. To differentiate different time data, it is necessary to align the data in a common coordinate. Later, chapter 5 compares this baseline against the proposed approach.

3.1 Structure from Motion

Structure from Motion (SfM) is a general method to estimate camera pose using images [8, 10] (Fig. 3.1). As mentioned in chapter 2, the image archive activity is capturing sequential omnidirectional images in tsunami-devastated areas. Hence, in this dissertation, SfM estimates camera pose using 360° field of view panoramic images [31].

The method is summarized as follows:

(1) Feature points are extracted with the Speed Up Robust Feature (SURF) [53] and tentative matching is obtained for two consecutive images using the descriptors of these feature points.

(2) Essential matrices are calculated with the five point algorithm [54]. At that time, mismatches of feature points are rejected using Random Sample Consensus (RANSAC) [55] (Fig. 3.2).

(3) Camera poses and positions of feature points are calculated using those essential matrices.

(4) Camera poses and position of 3D points are optimized to minimize reprojection errors of feature points.

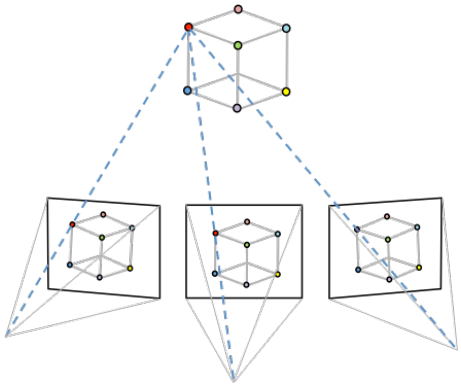


Figure 3.1: Abstract of Structure from Motion (SfM). SfM estimates camera poses and 3D points of a scene structure simultaneously using multi-view images.

3D point clouds of which each point has an image descriptors is generated through (1)-(4) processes.

Figure 3.3 shows a result of city-scale three-dimensional reconstruction using one thousand omnidirectional images (April, 2011, Kamaishi, Iwate). Red line shows trajectories of a camera (i.e. our measurement vehicle). The point clouds show structural objects, such as building, telegraph pole, and tree. This reconstruction result shows the structure of the entire city. However, it is not easy to understand the detail of the structure due to the sparseness of feature points.

Figures 3.4 and 3.5 show reconstruction results consisting of sparse feature points using images captured at a same location in April, 2011 and July, 2011, respectively. Figures 3.6 and 3.7 show dense reconstruction results using Patch-based Multi-view Stereo (PMVS2) [1] corresponding to Figs. 3.4 and 3.5, respectively. Figures 3.8 - 3.11 are enlarged views of Figs. 3.4 - 3.7. The sparse reconstruction results represent the entire shape of the street well. And the dense reconstruction results using PMVS2 represents the detail of the scenes well although they have some lacks of the structures, especially for texture-less area. However, it is difficult to understand the details of the scenes using the sparse results, and, regional-scale 3D reconstruction requires too much computational resources since one city has several thousands to several tens of thousands of image.

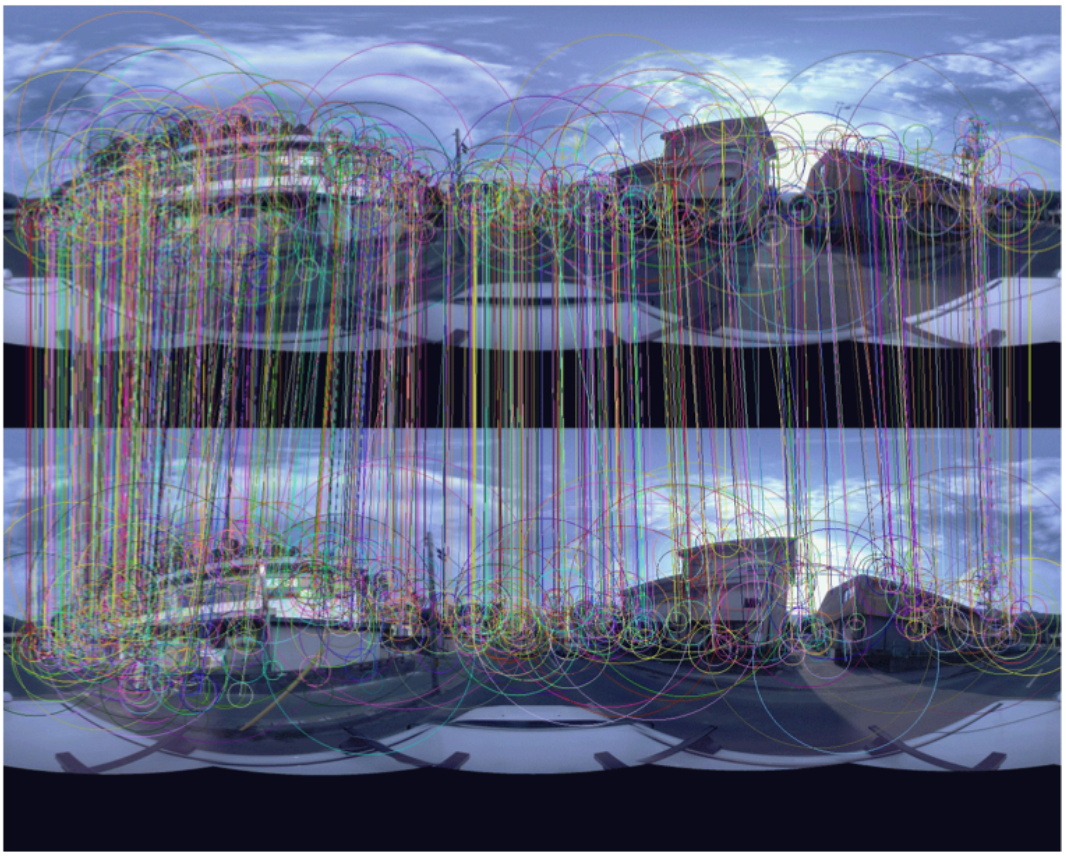


Figure 3.2: Results of feature matching between two consecutive images using local descriptor of feature points. Outliers are removed using RANSAC.

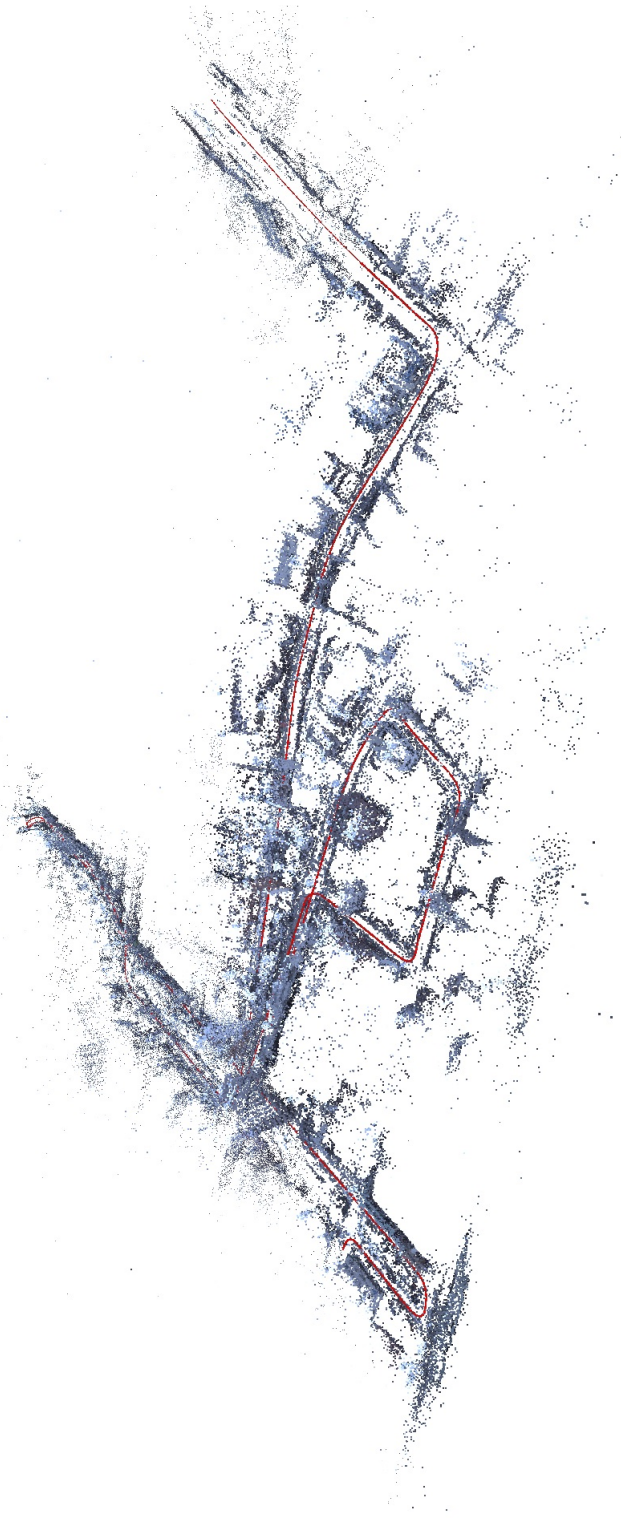


Figure 3.3: Example of city-scale 3D reconstruction (April, 2011, Kamaishi, Iwate). This result consists of one thousand omnidirectional images.

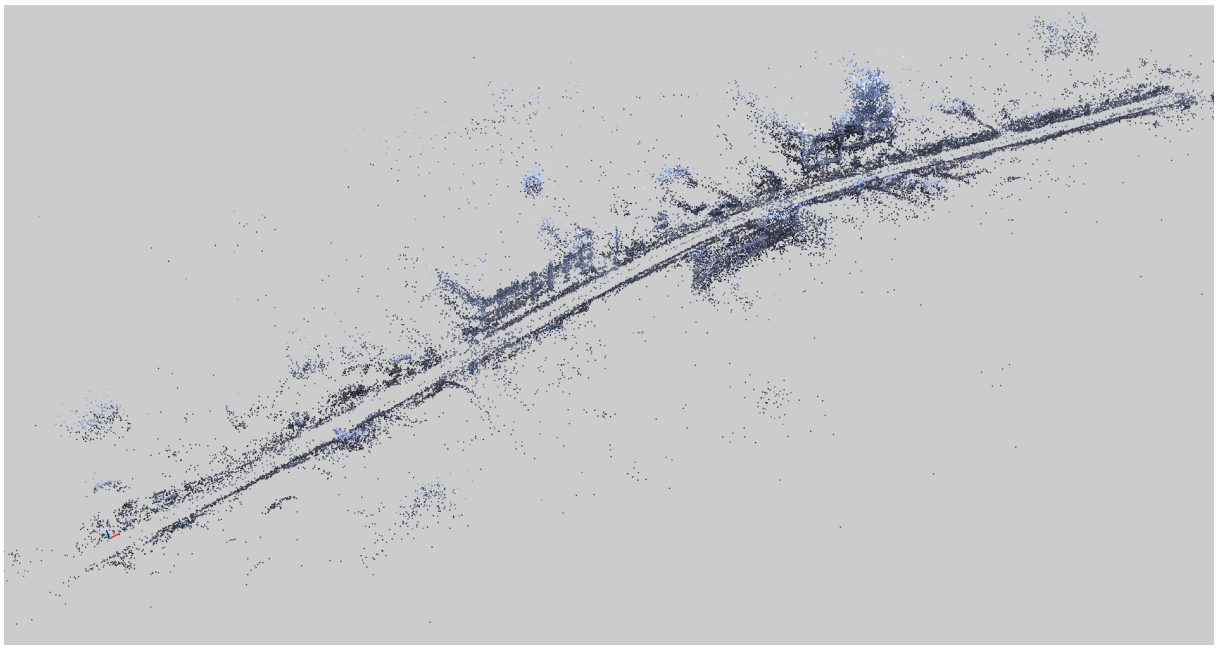


Figure 3.4: Example of 3D reconstruction consisting of sparse feature points (April, 2011, Rikuzentakata, Iwate).

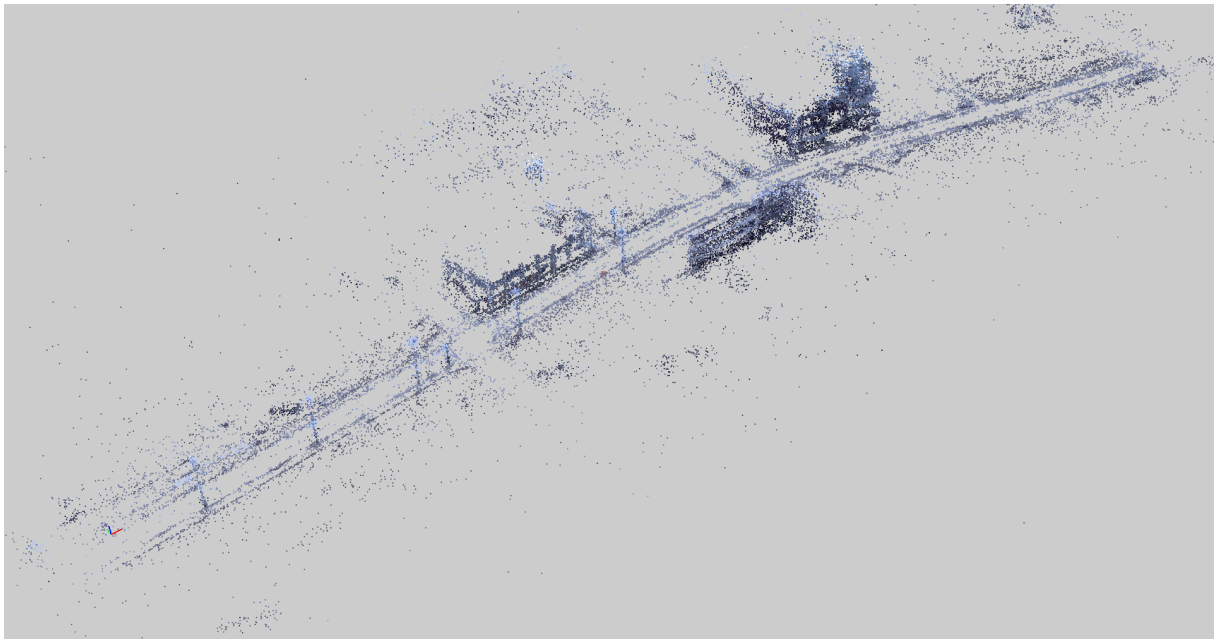


Figure 3.5: Example of 3D reconstruction consisting of sparse feature points (July, 2011, Rikuzentakata, Iwate).

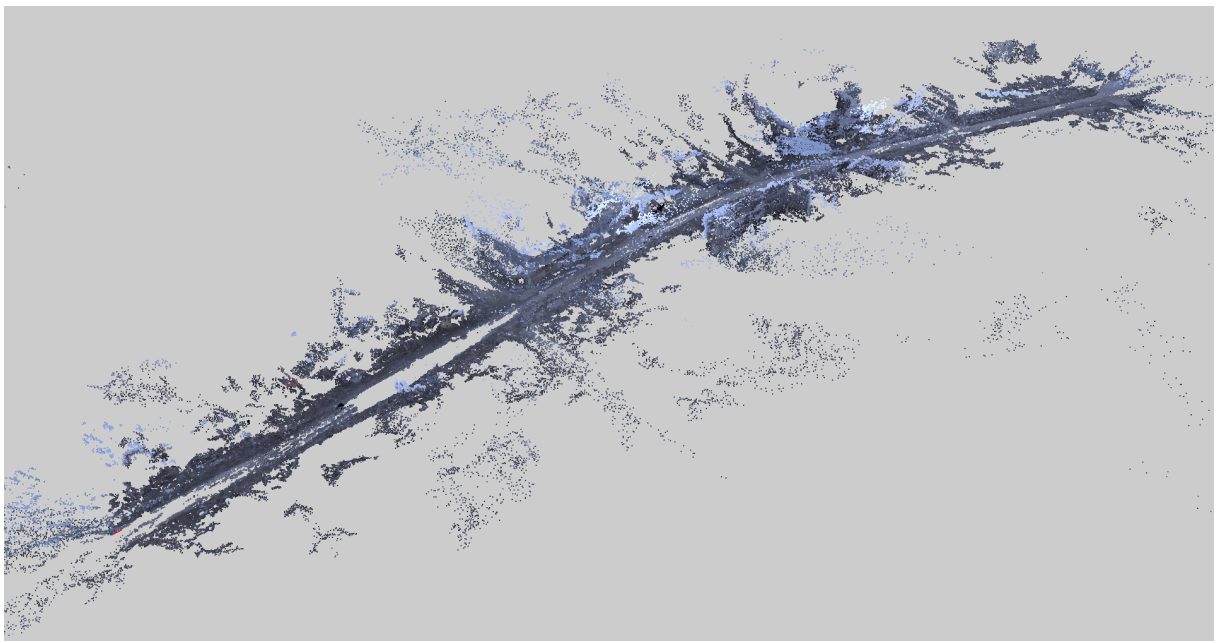


Figure 3.6: Example of dense 3D reconstruction using PMVS2 (April, 2011, Rikuzentakata, Iwate).

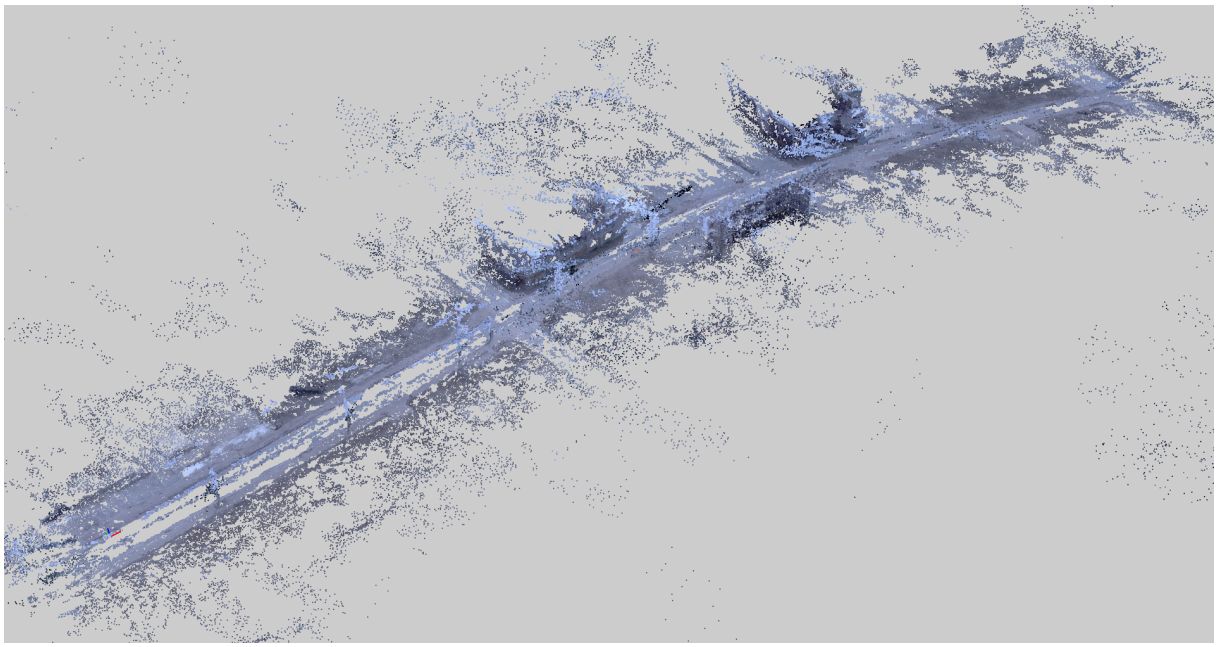


Figure 3.7: Example of dense 3D reconstruction using PMVS2 (July, 2011, Rikuzentakata, Iwate).

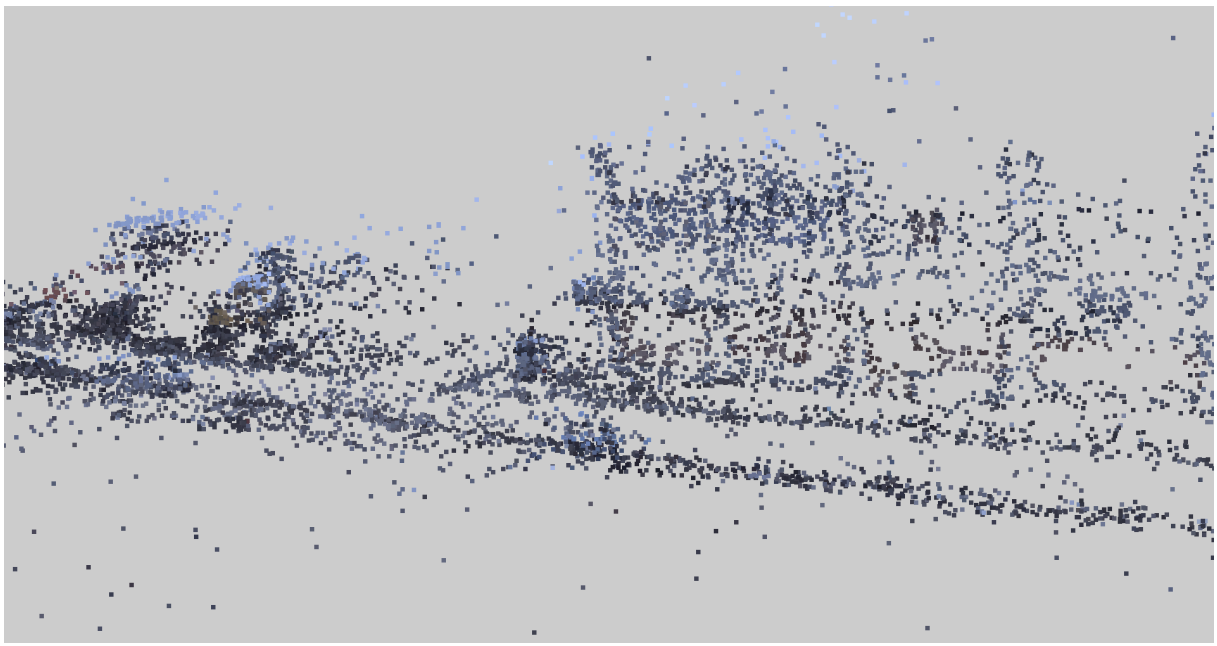


Figure 3.8: Example of 3D reconstruction scene using sparse feature points (April, 2011, Rikuzentakata, Iwate).

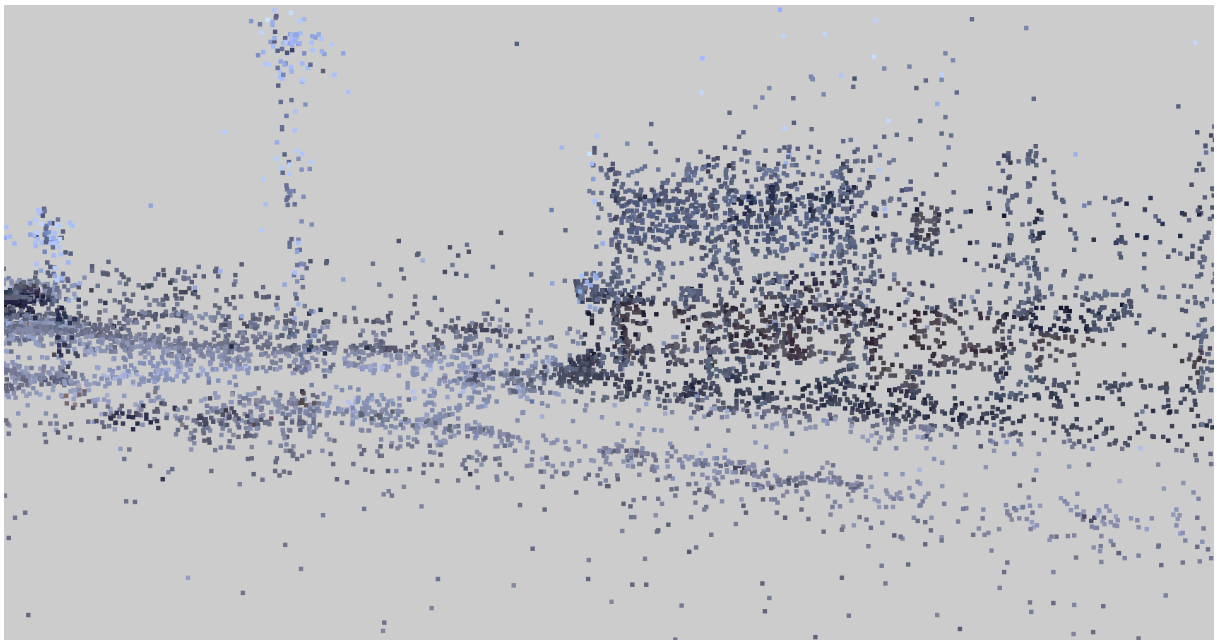


Figure 3.9: Example of 3D reconstruction scene using sparse feature points (July, 2011, Rikuzentakata, Iwate).

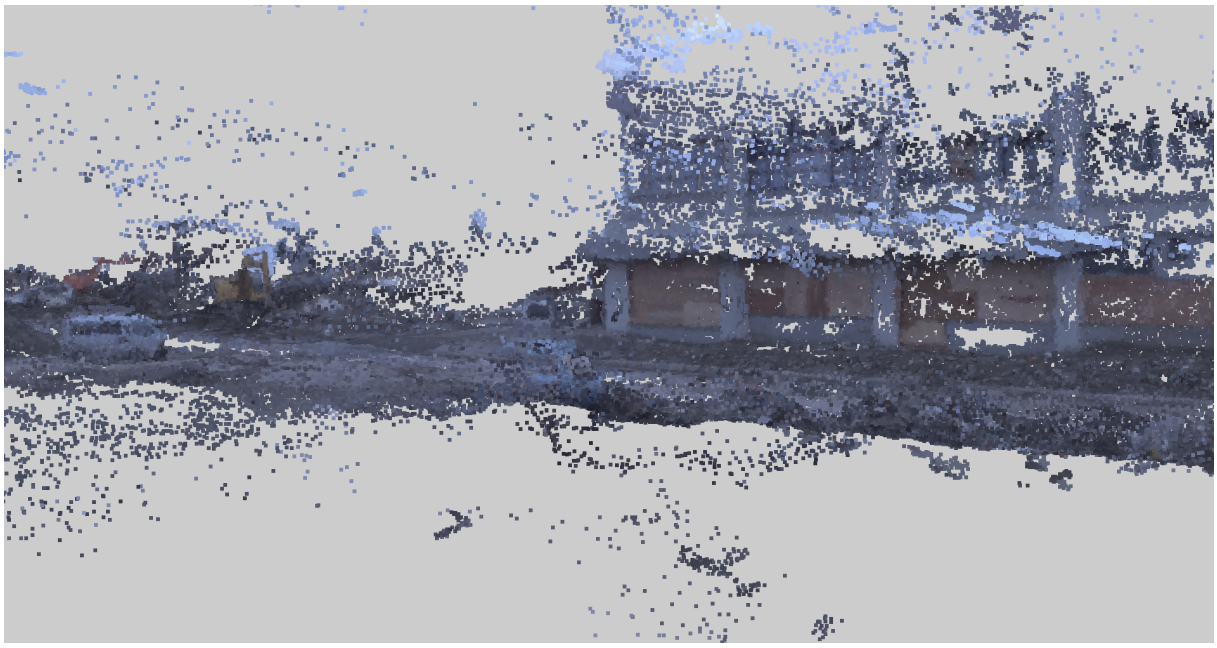


Figure 3.10: Example of dense 3D reconstruction scene using PMVS2 (April, 2011, Rikuzentakata, Iwate).

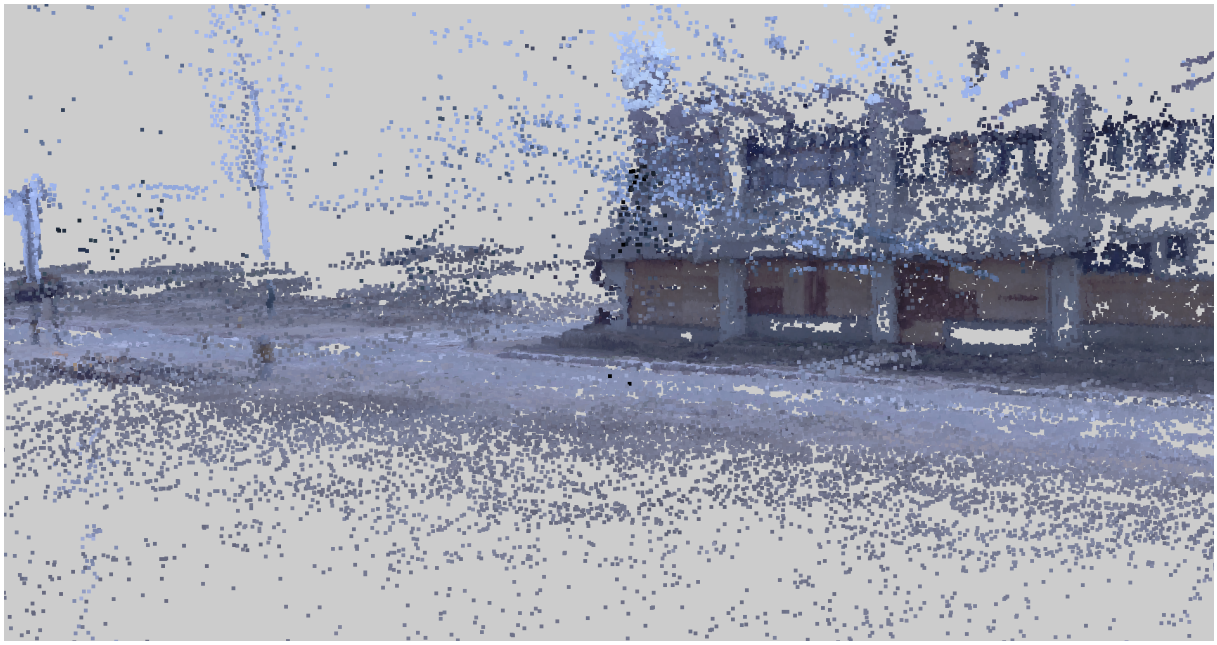


Figure 3.11: Example of dense 3D reconstruction scene using PVS2 (July, 2011, Rikuzentakata, Iwate).

3.2 Baseline Method for Temporal Change Detection

Baseline method for temporal change detection is to directly compare three dimensional structures of different times. In this section, as a baseline of temporal change detection, results of temporal change detection based on point clouds are shown.

To differentiate different time structure, it is necessary to register different time data in the common coordinate. The detail of our registration method is described in sec. 5.2.1. The summary of the method is as follows. First, SfM is performed independently for each sequence. Next, the two reconstructions are roughly aligned with a similarity transform using RANSAC [56]. Finally, bundle adjustment is performed for the extended SfM problem, in which the sum of the reprojection errors for all the correspondences is minimized.

After the alignment, temporal changes is detected by differentiating the two reconstruction. For the change detection based on point clouds, it is necessary to consider difference of point densities because point density reconstructed using SfM is basically in inverse proportional to distances from cameras. Hence, first, the method of this dissertation calculates the average distance d_{same} between the point and the nearest N points of the same time data, and the average distance d_{diff} between the point and the nearest N points of the other time data. The point is labeled as "Change" if $d_{\text{diff}} > 2d_{\text{same}}$, "Not Change" otherwise. If the point is observed in only old or new data, the point is labeled as "Dissapeared" and "Appeared", respectively.

Figures 3.12 and 3.13 show the results of change detection comparing sparse reconstruction results of April, 2011 and July, 2011, Rikuzentakata, Iwate (Figs 3.4 - 3.5 and Figs 3.8 - 3.9, respectively). Red, blue and yellow show disappearance, appearance and no-change, respectively. These figures show important changes of the scene, for example, debris along the street were removed (red), and telegraph poles were built (blue) in an early stage of the recovery operation. Some ground areas are labeled as "Appeared" because those areas are occluded by debris in the images of April.

Figures 3.14 and 3.15 show the results of change detection comparing dense reconstruction results using PMVS2 of the sequences same as Figs. 3.12 and 3.13. The results of change detection using dense reconstructions show the detail of the changes well, especially for texture-less areas (e.g. building wall, the ground).

However, even dense change detection results have some missing parts due to the ambiguity of the estimated scene depth. For getting accurate shape of a scene change, it is necessary to maximize the usage of image information. Our probabilistic method of change detection is explained in chapter 5.

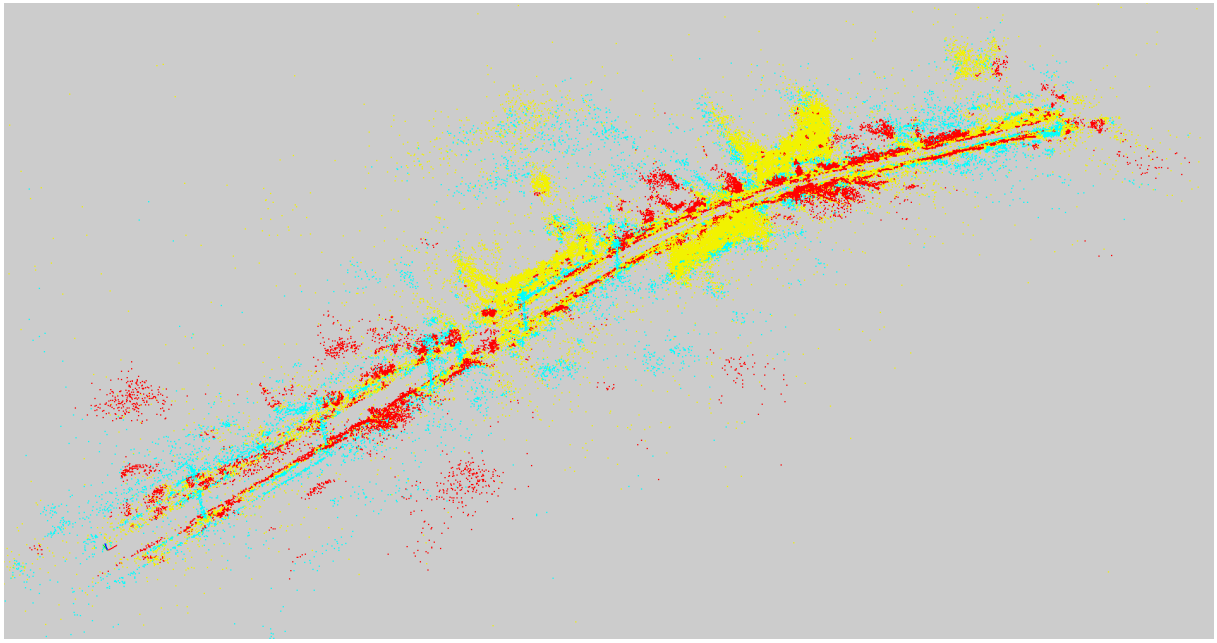


Figure 3.12: Change detection using sparse feature points comparing April, 2011 and July, 2011, Rikuzentakata, Iwate (Figs 3.4 and 3.5). Red, blue and yellow show disappearance, appearance and no-change, respectively.

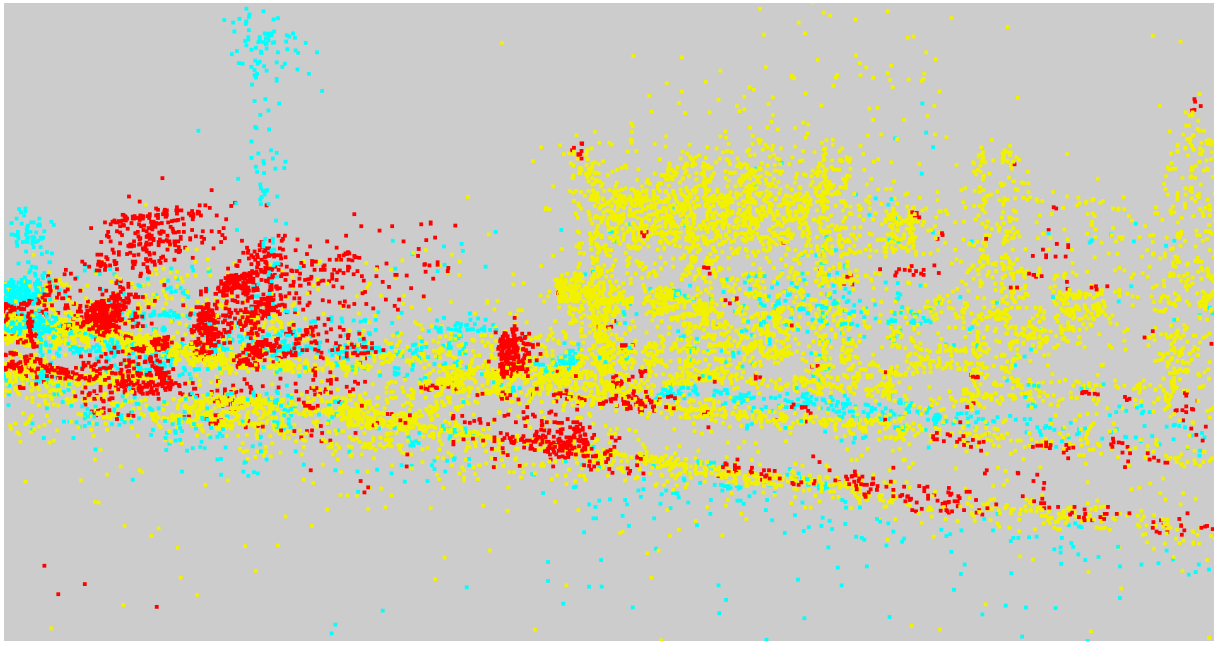


Figure 3.13: A scene of change detection using sparse feature points comparing April, 2011 and July, 2011, Rikuzentakata, Iwate (Figs 3.8 and 3.9). Red, blue and yellow show disappearance, appearance and no-change, respectively.

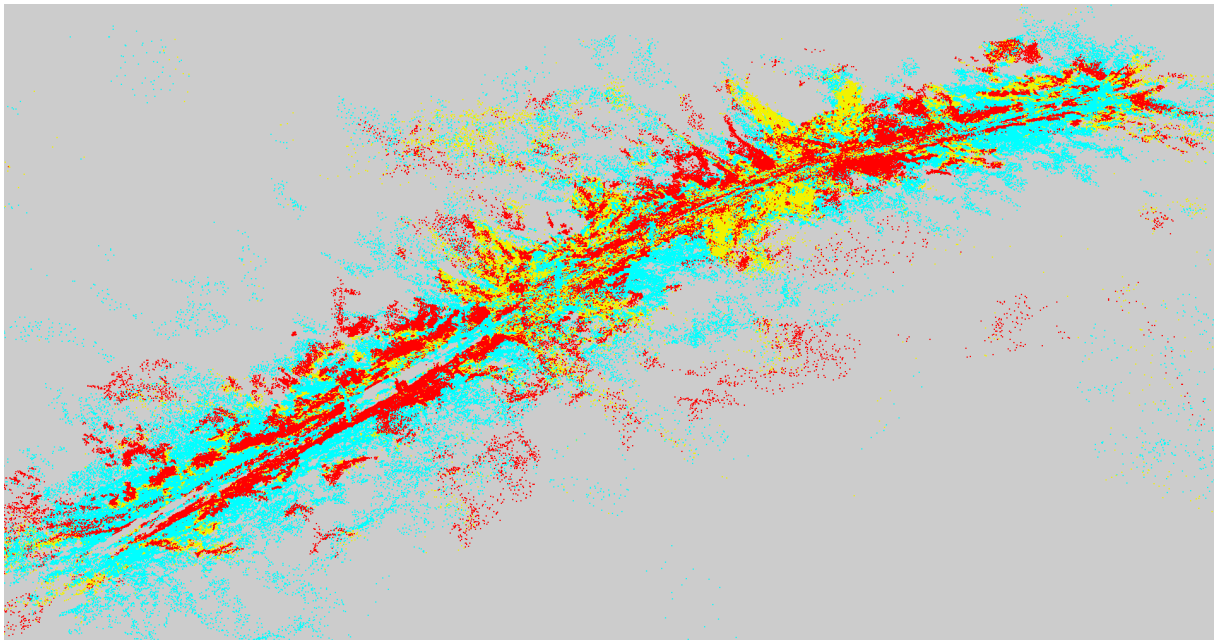


Figure 3.14: Change detection using dense 3D points of PMVS2 comparing April, 2011 and July, 2011, Rikuzentakata, Iwate (Figs 3.6 and 3.7). Red, blue and yellow show disappearance, appearance and no-change, respectively.

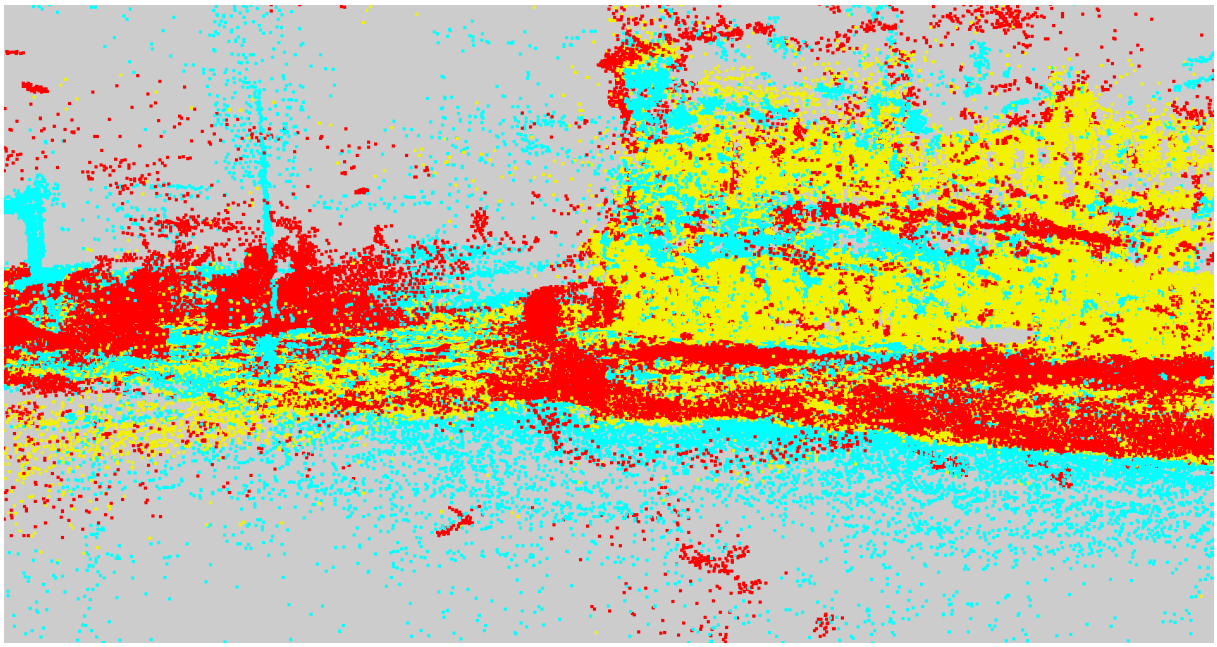


Figure 3.15: A scene of change detection using dense 3D points of PMVS2 comparing April, 2011 and July, 2011, Rikuzentakata, Iwate (Figs 3.10 and 3.11). Red, blue and yellow show disappearance, appearance and no-change, respectively.

Chapter 4

2D Change Detection

This chapter discusses the method detecting scene changes of an image pair using grid feature. Automatic scene change detection is effective for city management, disaster reduction, recovery and restoration. The objective of this dissertation is to detect change of cities using ground-level imagery. Several previous approaches of change detection require 3D model of a scene and pixel-level registration between different time images. Hence, in the case that 3D model is not available, it is difficult to directly apply the previous methods to the change detection. Furthermore, it is computationally prohibitive to estimate scene change of wide area using 3D model and pixel-level registration. This chapter proposes a novel change detection method integrating convolutional neural network (CNN) feature with superpixel segmentation. The proposed method enables it to detect scene change without pixel-level registration. Hence, the method can reduce the computational time and detect change of entire tsunami-damaged areas.

4.1 Motivation

This chapter considers a problem of detecting scene change from a pair of images taken at different time. The goal behind this dissertation is to estimate city-scale scene change of relatively short term due to disaster, for example, earthquake and tsunami. Understanding of scene change only by driving a vehicle is effective for disaster reduction, quick recovery and restoration.

However, there are some challenges for estimating scene change using vehicular imagery due to the differences of camera view points, illumination condition, photographing condition, sky (e.g. cloud) and ground (e.g. dust on the road) between different time images. It is necessary to develop a change detection method method robust for these difficulties.

Some previous approaches of change detection assume either or both a 3D model of a scene and pixel-level registration. In the case of wide-area disaster, it is computationally

prohibitive to reconstruct three dimensional structure of entire areas and to estimate accurate camera pose.

This chapter proposes a novel method to detect scene change without 3D model and pixel-level registration integrating convolutional neural network (CNN) feature with superpixel segmentation. First, the proposed method roughly but quickly estimates scene change of entire areas from image pairs which are aligned using Global Positioning System (GPS) data. Next, the method described in chapter 5 estimates structural scene change of the areas where detailed analysis is necessary. These two steps enable us to quickly and accurately estimate scene change of wide-area.

For change detection without pixel-level registration, the proposed method divides input images into grids and estimates scene change comparing each grid feature between different time images. Furthermore, the method projects the change detection result of each grid into superpixel segments to estimate precise scene change. The key here is discrimination of grid feature. The proposed approach exploits pooling-layer of convolutional neural network (CNN) [57] as a feature. Typically, object recognition approach uses information of fully-connected-layer for classification task, however, the proposed approach exploits feature of pooling-layer which has location information of image space.

Some recent research shows that upper layers in CNN have highly-abstract and wide-area information of input image [58, 59]. Resolution of upper layer is lower than lower layer. It can be assumed that upper pooling-layer can be used to recognize highly-abstract object, conversely, lower pooling-layer can be used to recognize low-level visual feature (e.g. edge, texture). This study evaluates the characteristics of each pooling-layer mentioned above in scene change detection.

4.2 Change Detection using Grid Feature

This section describes the method to estimate scene change using grid feature. The proposed method makes use of the following three information.

(i) Grid Feature

(ii) Superpixel Segmentation

(iii) Geometric Context (Sky and Ground)

Figure 4.1 illustrates the flowchart of the proposed method.

The proposed method detects changes using grid feature to minimize influence of difference of camera view point. Camera view points of images taken at different time points are different since the images are captured by running a vehicle on which a camera is

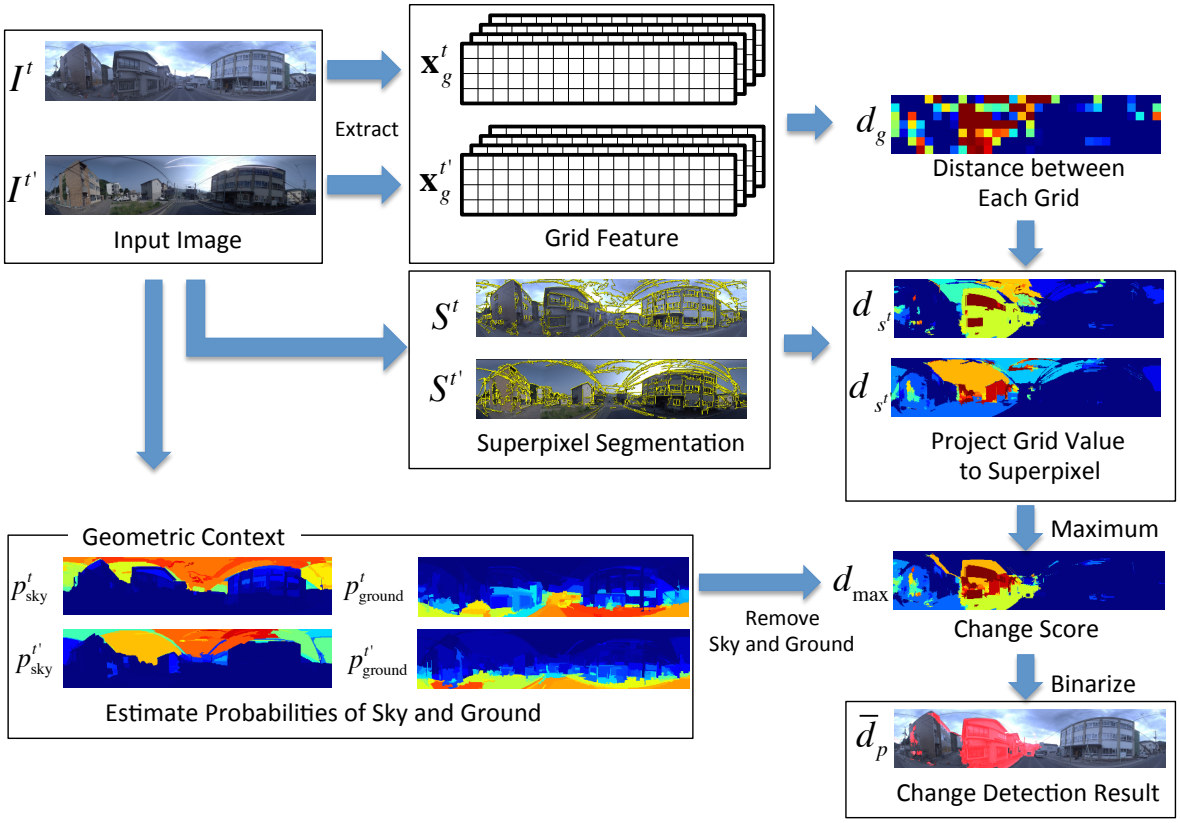


Figure 4.1: Flow chart of change detection using grid feature

mounted. To overcome this difficulty, grid feature coarsely detects scene change in grid resolution, and results of superpixel segmentation and refine the grid result into pixel-level change detection.

The change detection target of this chapter is object-level change (e.g, building, car), not low-level appearance change due to illumination and photographing conditions,. To detect such kind of highly-abstract change, the proposed method exploits feature of CNN-pooling layer as a grid feature.

Furthermore, to remove the differences of the sky (e.g. cloud) and the ground (e.g. dust on the road), the method makes use of Geometric Context [60] which is a segmentation method less affected by illumination and photographing conditions since it is based on only geometric features.

(i) Grid Feature First, the method divides input image I^t into grids, and extract feature ${\mathbf{x}_g}^t$ from each grid ($g = 1, ..., N_g$). The proposed method exploits CNN as feature extractor. The experiments compares the results of CNN with the results of the baseline

methods of SIFT[61, 62, 63] and local-patch. The experimental results show that feature of CNN pooling-layer is effective for scene change detection. Section 4.3 explains about the detail of grid feature. Dissimilarity between each grid at different time images is calculated from the normalized grid features ${\mathbf{x}_g}^t$ ($|{\mathbf{x}_g}^t| = 1$) as in Eq.(4.1).

$$d_g = |{\mathbf{x}_g}^t - {\mathbf{x}_g}^{t'}|. \tag{4.1}$$

Then, the method projects d_g into input images I, I' and calculates dissimilarity of each pixel $d_p(p = 1, ..., N_p)$ where N_p is number of pixel.

(ii) Superpixel Segmentation Superpixel segmentation refines grid dissimilarity d_g estimated from grid feature and estimates precise change. The proposed approach applies superpixel segmentation to input images I^t. The set of the superpixels is S^t. Dissimilarity of superpixel $s^t \in S^t$ is average of all pixels in superpixel s^t.

$$d_{s^t} = \frac{1}{|s^t|} \sum_{p \in s^t} d_p. \tag{4.2}$$

$d_{\max}$ is the maximum value of $d_{s^t}, d_{s^{t'}}$.

$$d_{\max} = \max(d_{s^t}, d_{s^{t'}}) \tag{4.3}$$

(iii) Geometric Context This last step applies Geometric Context [60] to remove segments of the sky and the ground from the target of change detection. Geometric Context estimates probabilities of the sky and the ground of each pixel $({p_{\text{sky}}}^t, {p_{\text{ground}}}^t)$ in input images I^t. Dissimilarity of each pixel removed the sky and the ground is as Eq.(4.4).

$$\overline{d_p} = \begin{cases} 0 & (((p_{\text{sky}}{}^t > a) \wedge ({p_{\text{sky}}}^{t'} > a)) \vee (({p_{\text{ground}}}^t > b) \wedge ({p_{\text{ground}}}^{t'} > b))) \\ d_{\max} & (\text{otherwise}) \end{cases} \tag{4.4}$$

$a = t_{\text{sky}}$ and $b = t_{\text{ground}}$ are constant values within the range of $0 \leq t_{\text{sky}}, t_{\text{ground}} \leq 1$.

4.3 Selection of Grid Feature

The proposed approach exploits coarse grid-level feature for change detection. As a typical grid feature, SIFT and local patch feature have been used for many tasks. Several methods using Bag-of-Visual Words (BOW) and Fisher vector [61, 63, 64, 65, 62, 66] of SIFT perform high recognition accuracy in the object recognition task. In the past few years, convolutional neural networks (CNN) outperformed the alternative in the recognition accuracy.

This study evaluates if the the same high-accuracy applies in change detection. In contradiction to SIFT whose feature extractor is artificially designed, CNN learns network parameters from a large amount of image data. Recent studies show that the architecture of CNNs is similar to the mechanism of human's object recognition [67]. Hence, it can be expected that CNN feature is effective for high abstract classification. However, with respect to the mechanism of CNN, there are many unresolved questions.

To apply CNN to change detection, the proposed method exploits CNN pooling-layer as feature of each grid. The CNN used in this study is one whose padding keeps spatial resolution after convolution (e.g. [68]). Feature scale of each grid is normalized. All elements of the CNN feature in this study are non-negative value because of the Rectified Linear Units (ReLUs) [57]. Hence, dissimilarity between each grid of different time images $d_i \in \mathbf{d}_g$ is within the value of $0 \leq d_i \leq \sqrt{2}$.

The next section evaluates CNN feature against Dense-SIFT [62, 63] and local patch baselines. Dense-SIFT in this study is concatenated SIFT feature of multi-scale whose basic size is the grid size. Local-patch feature is used in gray scale.

4.4 Experimental results

This experiment evaluates the estimation accuracy of the proposed approach for detecting changes in Panoramic Change Detection Dataset. The dataset consists of omnidirectional panoramic images taken at different time in tsunami-damaged areas. Image pair consists of a query image and a different time image which is the closest to the query image in three dimensional space. The proposed method detects scene change using only one image pair.

All images were taken at different view points since the images were taken every two meters by running a vehicle with omnidirectional camera and GPS. Furthermore, illumination condition is different due to weather change. The differences of view point and illumination make it difficult to estimate scene changes even by human vision. It took fifteen minutes on average to manually make the ground-truth of scene changes for one image pair. To scale up the detection to regional-scale, it is essential to make it possible to automatically estimate scene changes since one city has several thousands to several tens of thousands of image pairs.

4.4.1 Panoramic Change Detection Dataset

This dataset is for evaluating estimation accuracy of scene change, and consists of twenty panoramic image pairs. Figure 4.2 shows an example of Panoramic Change Detection Dataset. In this experiment, for evaluation of estimation accuracy, the image pairs and

Panoramic image (query) Panoramic image (database) Ground Truth

Figure 4.2: Example of Panoramic Change Detection Dataset.

the change masks were manually generated.

The target of change detection is both two (e.g. Texture) and three (e.g. building, car) dimensional changes. However, change due to differences of illumination, photographing condition, the sky and the ground are not the target subject. For example, the estimation target includes building, car and debris, but not changes due to specular reflection, cloud and sign on the road surface.

4.4.2 Parameter Settings

In the proposed method, there are several parameters: (1) the threshold of distance between grid features to binarize the change estimation result t_{dist}, (2) thresholds for the probabilities of Geometric Context to detect sky and ground (t_{sky}, t_{ground}), and (3) parameters of superpixel segmentation.

(1) The threshold of distance between grid features t_{dist} is the threshold which takes the best F_1 score for each feature type in the evaluation using the change detection dataset. Table 4.1 shows the thresholds for all features. In the case of feature of pooling-layer and Dense-SIFT, distance of normalized features between each grid $d_i \in \mathbf{d}_g$ takes a value within the range of $0 \leq d_i \leq \sqrt{2}$ because all elements of the features are non-negative values. In the case of gray-scale local-patch, d_i takes a value within the range of $0 \leq d_i \leq 2$. The thresholds of pool-3, 4, 5 and gray-scale local-patch are almost the median values of their range.

(2) The thresholds for the probabilities of the sky and the ground are fixed for all experiments ($t_{\text{sky}} = 0.2$, $t_{\text{ground}} = 0.8$).

(3) The superpixel segmentation method used in this experiment is Felsenszwalb's efficient graph based image segmentation[69]. The parameters of the superpixel segmentation (scale, diameter of a Gaussian kernel, minimum component size) are fixed for all experiments.

The CNN in this experiment is based on VGG model of 19 weight-layers [68] which is one of state-of-the-art CNN model in image recognition task. The proposed method uses the pooling-layer features of the CNN. Furthermore, the spatial resolution of VGG model is preserved after convolution since the padding is 1 pixel for 3×3 convolution-layers.

Table 4.1: F_1 scores of the detected changes and the thresholds of the best F_1 scores. The bottom row shows the dimension of each feature type in the direction of row, column and descriptor. Input image size is 224×1024. The CNN of this experiment is based on VGG model of 19 weight layers [68].

	pool-5	pool-4	pool-3	pool-2	pool-1
F_1 score	0.722	0.722	0.688	0.629	0.592
Threshold	0.75	0.75	0.70	0.65	0.35
Feat Dim (y,x,d)	(7,32,512)	(14,64,512)	(28,128,256)	(56,256,128)	(112,512,64)

	Dense-SIFT	Patch
F_1 scores	0.592	0.599
Threshold	0.25	0.90
Feat Dim (y,x,d)	(7,32,512)	(7,32,256)

The CNN model pads image to keep the consistent of spatial resolution.

The following experiment compares CNN features with Dense-SIFT[61, 62, 63] and gray-scale local patch. In this experiment, the grid size of Dense-SIFT and local patch is the same as the grid size of the pool-5 layer. The Dense-SIFT has features of four different scales for each grid, i.e. the dimension of the Dense-SIFT is $128 \times 4 = 512$. The dimension of the patch feature is 256 since the patch size is resized into 16×16.

4.4.3 Comparison of the results

Table 4.1 shows the F_1 scores of each feature type for Panoramic Change Detection Dataset. The features of pool-4 and 5 performed the best F_1 scores. The high-level pooling-layer performs better than the low-level pooling-layer. F_1 scores of pool 1 is almost the same as that of the baseline methods (Dense-SIFT, gray-scale local-patch). The result clearly indicates CNN feature is effective for the change detection.

Figures 4.3 and 4.4 show the feature distance between each grid and the final estimation results for each feature type. The top row shows the input image pair, the second row shows the ground-truth of change detection, and the other rows show feature distances of each grid. Figure 4.3 indicates that the feature of high-level pooling-layer discriminates the difference of the high abstraction of the scene (e.g. object). On the other hand, the feature of low-level pooling-layer detects the difference of low-level visual feature (e.g. edge). For example, in Fig. 4.3 , the result of pool-3 layer has big errors around the left building due to illumination change and the difference of the view point. However, the pool-3 layer can correctly detect some small changes around the right building.

Figures 4.5 and 4.6 show the results of the scenes of Panoramic Change Detection Dataset. (See appendix for all other results.) The rows show, from top to bottom, input image pair, ground-truth of change detection, final change detection results, superpixel segmentation results, feature distance between each grid using feature of pool-5 layer, feature distance projected to superpixel segmentation result of each input image, probabilities of the sky and the ground estimated using Geometric Context.

The proposed method correctly detects the scene changes, for example, demolished and new buildings, car and debris. In some cases, Geometric Context cannot accurately estimate sky due to electrical wire and pole, and discriminate between the ground and low height object (e.g. debris, car). In contrast, the proposed method can detect object-level scene change well. These results indicate that feature of high-level pooling layer has high discrimination for scene difference and superpixel segmentation can compensate low resolution of the feature of the pooling-layer.

4.5 Summary

This section described the unified framework to detect scene change of an image pair using grid feature. The proposed method can detect scene change without pixel-level registration. To validate the proposed approach, this study introduced Panoramic Change Detection Dataset which is manually created for this task. The experimental results show that the proposed method effectively integrates high discrimination of CNN feature and accurate segmentation of superpixel.

Furthermore, the experiments evaluate the performance of features from high and low level layers in CNN. The study showed that CNN features have selectivity depending on the abstraction level of estimation target. The experimental results indicate that the feature of high-level pooling-layer can discriminates difference of high abstraction of a scene (e.g. object). On the other hand, feature of low-level pooling-layer detects difference of low level visual feature (e.g. edge).

There are remaining work to improve the estimation accuracy of the proposed approach and visualize scene change of vastly wide area.

(i) Integration of high and low level layers in CNN.

(ii) Categorization and statistical processing for change-detected objects

(i) Upper-level feature is appropriate for discriminating high-abstract object, but, feature of low-level layer can discriminate low-level visual feature. Combination of these characteristics might improve change detection accuracy. (ii) Furthermore, the final goal of this study is to visualize the scene change of the entire tsunami-damaged area. To understand

tsunami-damage of the entire area, it is necessary to analyze the amount of scene change for each object category. For the statistical analysis, a system has to extract objects from the change detection results and categorize the objects. The tsunami damages of the entire area can be regressed from the categorization result.

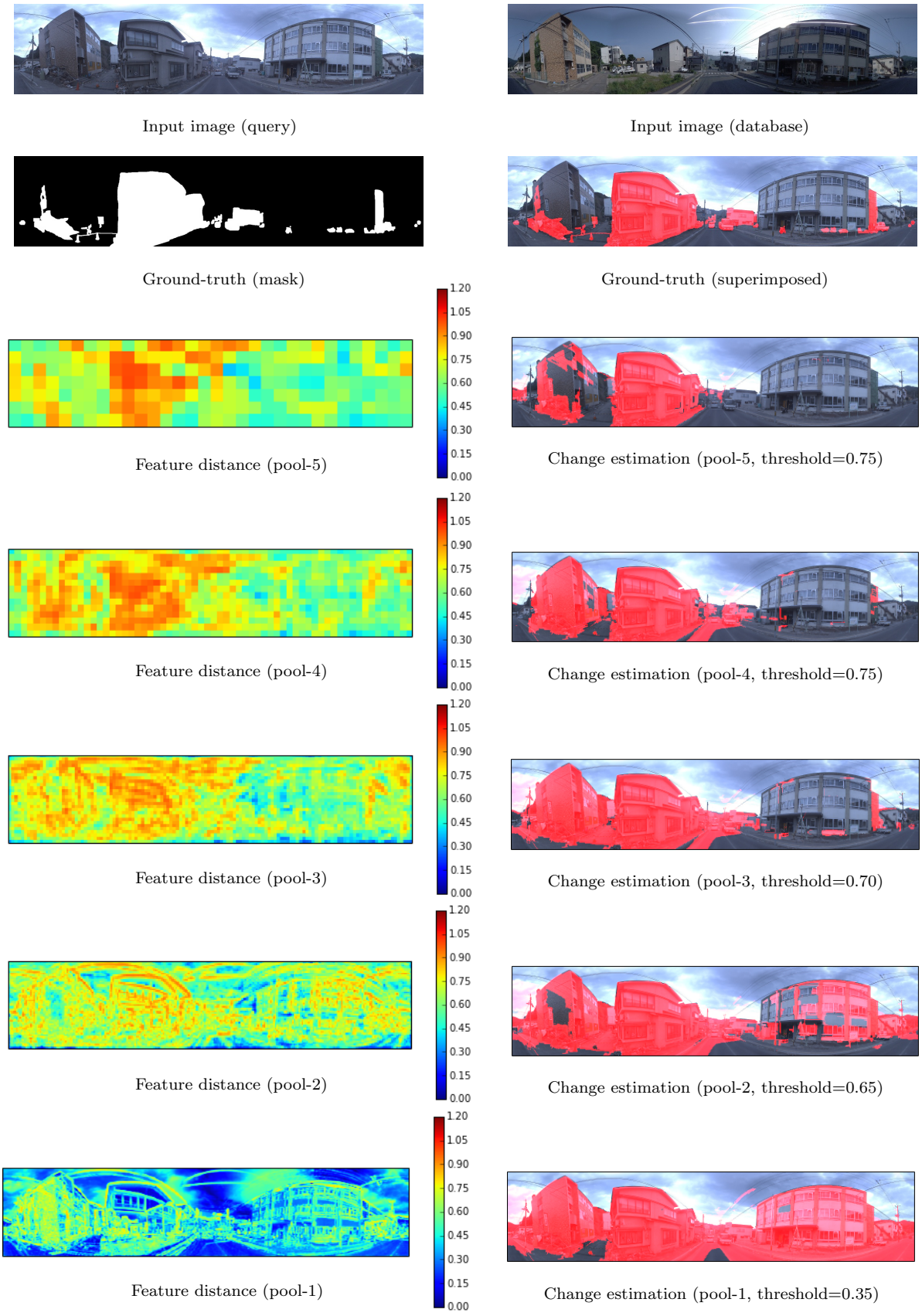


Figure 4.3: Feature distance of each grid (pooling-layers of CNN). The thresholds in the right figures is based on table 4.1. Distance of normalized features between each grid $d_i \in \mathbf{d}_g$ takes a value within the range of $0 \leq d_i \leq \sqrt{2}$ because all elements of pooling-layer feature are non-negative values.

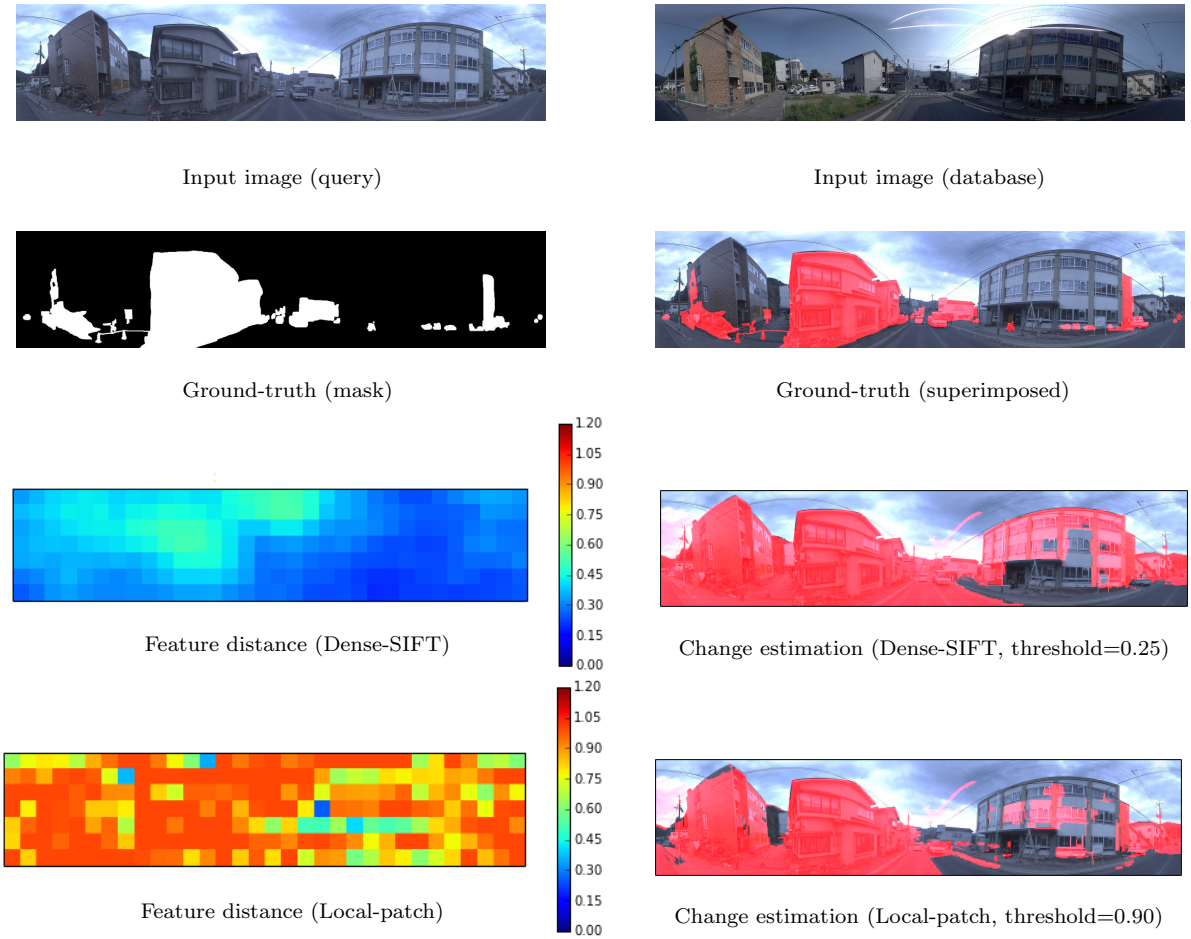


Figure 4.4: Feature distance of each grid (Dense-SIFT and local-patch). The thresholds in the right figures is based on table 4.1. In the case of Dense-SIFT, distance of normalized features between each grid $d_i \in \mathbf{d}_g$ takes a value within the range of $0 \leq d_i \leq \sqrt{2}$ since all elements of SIFT feature are non-negative values. In the case of local-patch feature, d_i takes a value within the range of $0 \leq d_i \leq 2$.

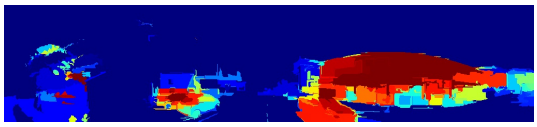
Input image (query)

Input image (database)

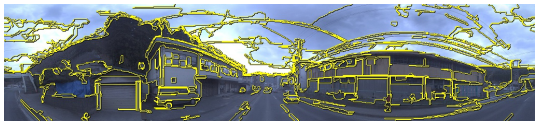
Ground-truth (superimposed)

Ground-truth (mask)

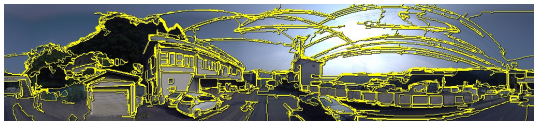
Change estimation (binarized)



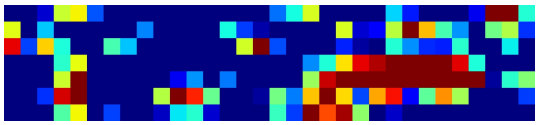
Change estimation (distance)



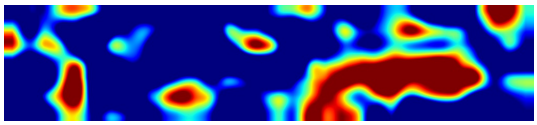
Superpixel segmentation (query)



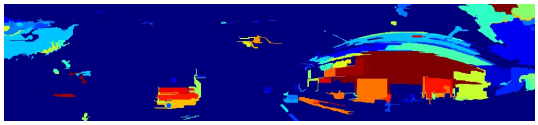
Superpixel segmentation (database)



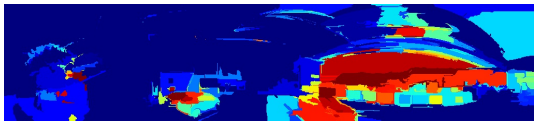
Feature distance (each grid)



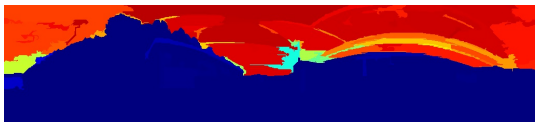
Feature distance (interpolation)



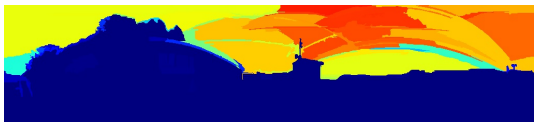
Feature distance in superpixel (query)



Feature distance in superpixel (database)

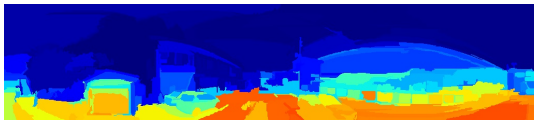


Sky probabilities (query)



Sky probabilities (database)

Ground probabilities (query)



Ground probabilities (database)

Figure 4.5: Results of change detection using pool-5 feature of CNN (Frame No. 0)

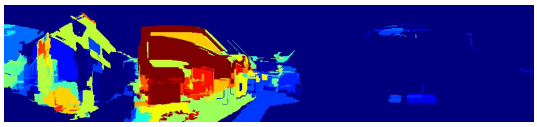
Input image (query)

Input image (database)

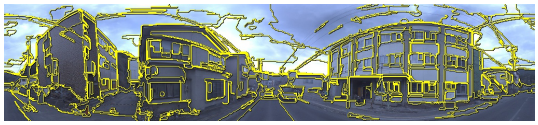
Ground-truth (superimposed)

Ground-truth (mask)

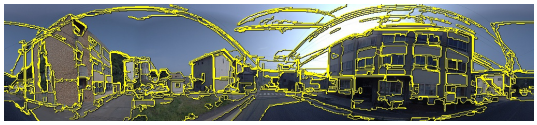
Change estimation (binarized)



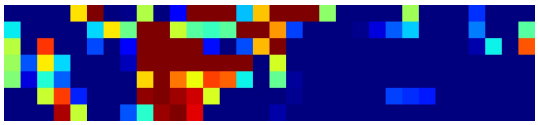
Change estimation (distance)



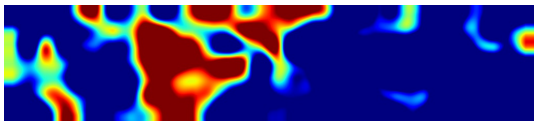
Superpixel segmentation (query)



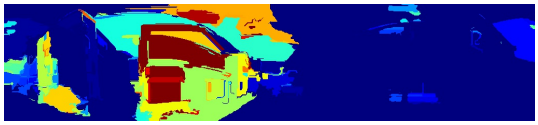
Superpixel segmentation (database)



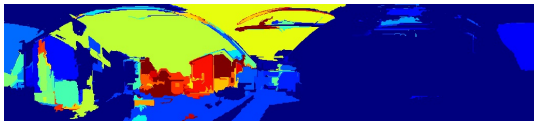
Feature distance (each grid)



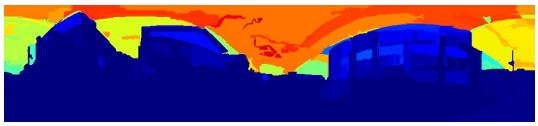
Feature distance (interpolation)



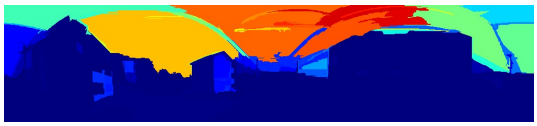
Feature distance in superpixel (query)



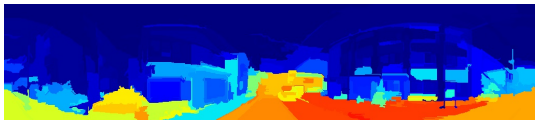
Feature distance in superpixel (database)



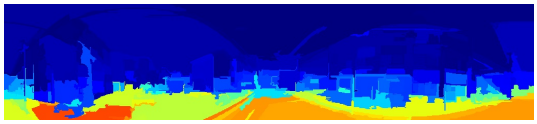
Sky probabilities (query)



Sky probabilities (database)



Ground probabilities (query)



Ground probabilities (database)

Figure 4.6: Results of change detection using pool-5 feature of CNN (Frame No. 1)

Chapter 5

3D Change Detection

This chapter describes a method for detecting temporal changes of the three-dimensional structure of an outdoor scene from its multi-view images captured at two separate times. The proposed method detects accurate structural change of the areas where the result of 2D change detection requests detailed analysis. The method estimates scene structures probabilistically, not deterministically, and based on their estimates, it evaluates the probability of structural changes in the scene, where the inputs are the similarity of the local image patches among the multi-view images. The aim of the probabilistic treatment is to maximize the accuracy of change detection, behind which there is our conjecture that although it is difficult to estimate the scene structures deterministically, it should be easier to detect their changes. The proposed method is compared with the methods that use multi-view stereo (MVS) to reconstruct the scene structures of the two time points and then differentiate them to detect changes. The experimental results show that the proposed method outperforms such MVS-based methods.

5.1 Motivation

This research considers a problem of detecting temporal changes in the three-dimensional structure of a scene, such as an urban area, from a pair of its multi-view images captured at two separate times. The application of this research is, for example, quickly grasping the damages of a city caused by an earthquake by simply running a vehicle with a camera in the area (assuming its pre-earthquake images are also available) and visualizing the processes of short-time recovery or long-time reconstruction from them by similarly capturing images for multiple times (Fig. 5.1).

Exactly for the latter purpose, we are creating the image archives of the urban and residential areas damaged by the tsunami caused by the earthquake happened in Japan in March 2011 (Chapter 2). Figure 5.2 shows examples of these images, which are a pair

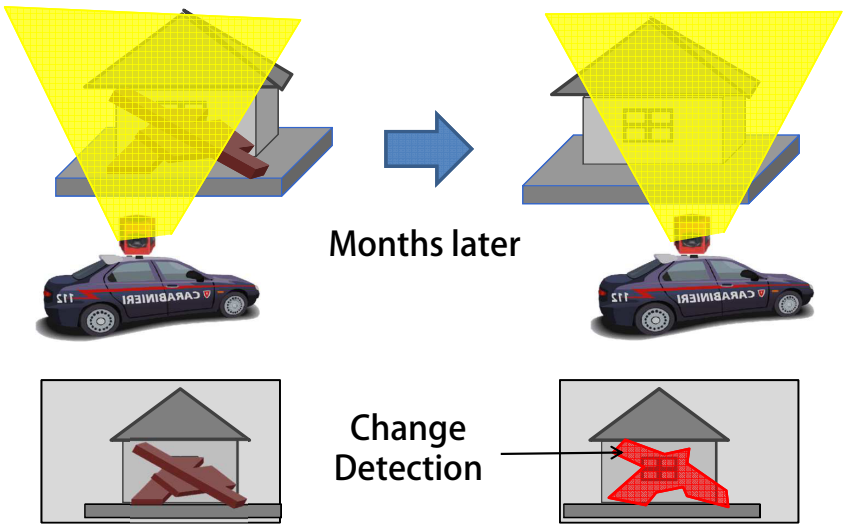


Figure 5.1: Change detection of a scene using a vehicle mounted camera.

of two images of the same scene captured three months apart.

To achieve the goal of detecting temporal 3D scene changes from these images, a naive approach would be to use Multi-View Stereo (MVS) [1, 70] to reconstruct the 3D shapes of the area at different time points from their images and differentiate them to detect changes in 3D structure. Considering the recent success of MVS, this approach is seemingly promising. However, apart from the reconstruction from aerial imagery, which has achieved great success lately, it is still a difficult task to accurately reconstruct the structure of a scene from its images taken by a ground vehicle-mounted camera. Figure 5.3 shows the results of applying PMVS2, one of the state-of-the-art, to our images. It is observed from the results that there are a lot of missing parts in the reconstruction. (Some of the existing ones are also incorrectly reconstructed, although they cannot be judged from this picture alone.) These may be attributable to several reasons, such as the large depth variations which are contrasted with aerial imagery, the limited variety and number of camera poses (i.e., the viewpoints are on a straight line along the vehicle path), and the insufficient scene textures. The differentiation of the two reconstructions thus obtained does not give good results, as will be shown later.

In this study, we propose another approach to this problem. The basic idea is that we want to know not the scene structure of each time point but their changes; thus, we formulate the problem so as to estimate them directly from the images. The core of the formulation, which distinguishes it from the above MVS-based one, is a probabilistic treatment of scene structures. To be specific, we estimate the scene structure (specifically, the scene depths from a selected viewpoint) not deterministically but probabilistically;

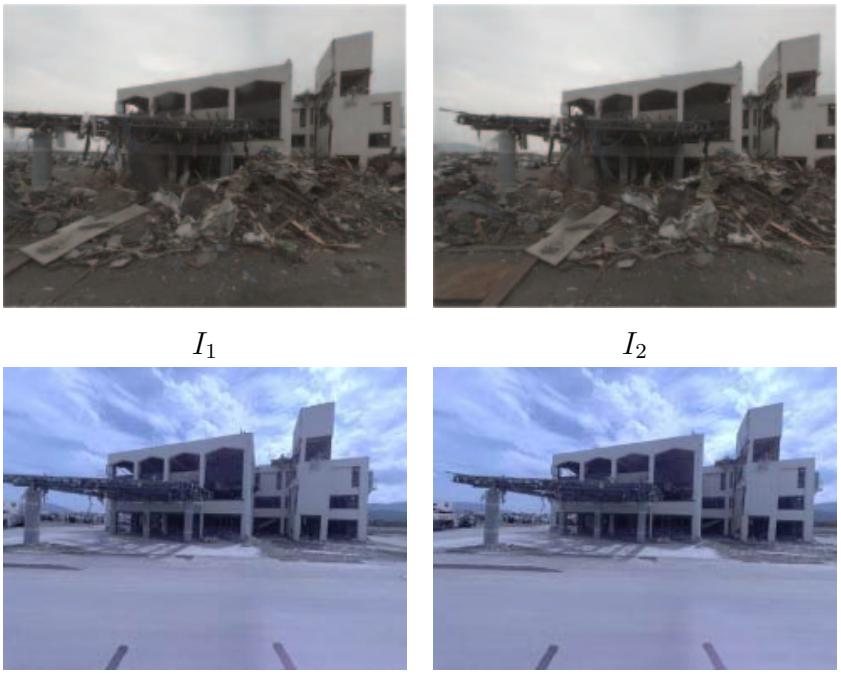


Figure 5.2: A pair of two images of the same scene taken at two separate times. (These are trimmed from omni-directional images.)

namely, we obtain not a point estimate but a probabilistic density of depths; we then estimate whether the scene changes or not by integrating the obtained depth density in such a way that their ambiguity is well reflected in the final estimates. The overall estimation is performed in a probabilistic framework, where the inputs are the similarity of the local image patches among the multi-view images. The camera poses are necessary in this estimation and are estimated in advance by performing SfM for the images of each time point followed by registration of the reconstructions.

Our aim behind this probabilistic treatment of scene structures is to maximize the accuracy of detecting scene changes. If scene structure has to be deterministically determined even though observations give only ambiguous information, the two reconstructions will inevitably have errors, so do the estimated scene changes obtained by differentiating them. Our approach could reduce such errors by appropriately considering the ambiguity of scene structure. As a by-product, we can also reduce the computational time; it might be a waste to spend large computational resources to compute scene structures, as we need only their changes.

The chapter is organised as follows. Section 5.2 explains how data are processed from image capture to change detection. In Section 5.3 , we presents a novel algorithm for

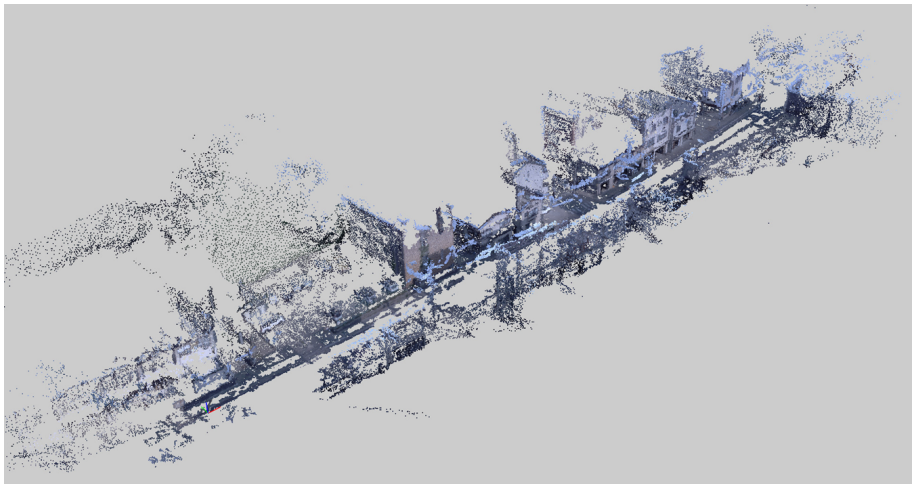


Figure 5.3: A result of applying PMVS2 [1] to our images that are obtained by a vehicle-mounted omni-directional camera at every few meters along a street. The camera poses needed for running PMVS2 are obtained by performing SfM.

change detection. Section 5.4 shows several experimental results. Section 5.5 summarizes the proposed method.

More results of PMVS2 when applied to our image data

Although PMVS2 is known as one of the state-of-the-art methods for dense reconstruction from multi-view images, it does not produce good results for the images of urban areas captured by a camera mounted on a vehicle running in streets, as is mentioned in Sec. 5.1. We show here typical false results which have missing parts due to lacks of texture and number of camera viewpoints.

As mentioned above, we input distortion-corrected versions of the six images captured by the six cameras comprising our omni-directional camera to PMVS2. Figure 5.4 shows these input images. Figure 5.5 shows the results of PMVS2 obtained from the images of two streets. The top row shows the overviews of the reconstructed scene structures, and the middle row shows their magnified portions. Comparing the latter with those of the input images shown in the bottom row of the figure, it is observed that there are many missing and erroneous parts, particularly where there is only limited texture.

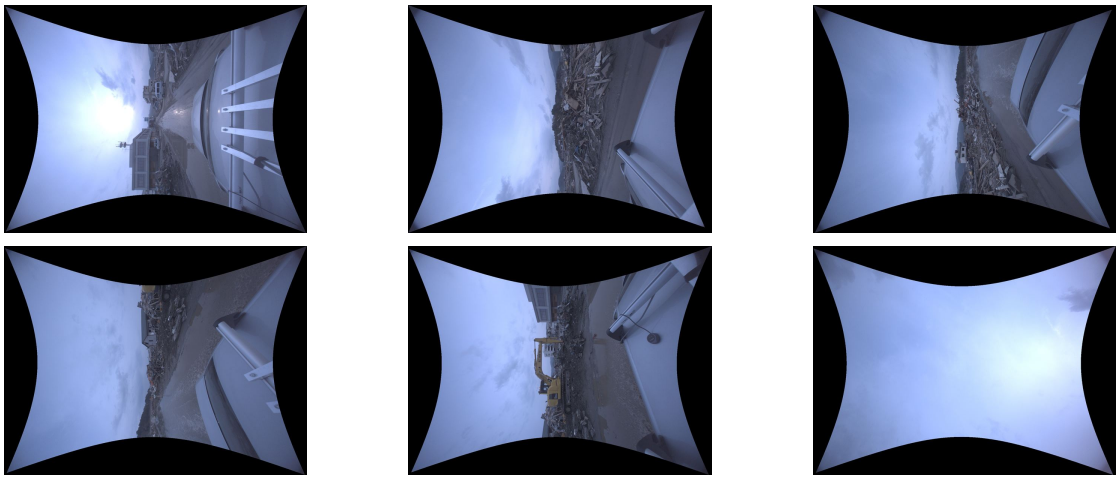


Figure 5.4: An example of the set of six distortion-corrected images that are input to PMVS2 for each viewpoint.

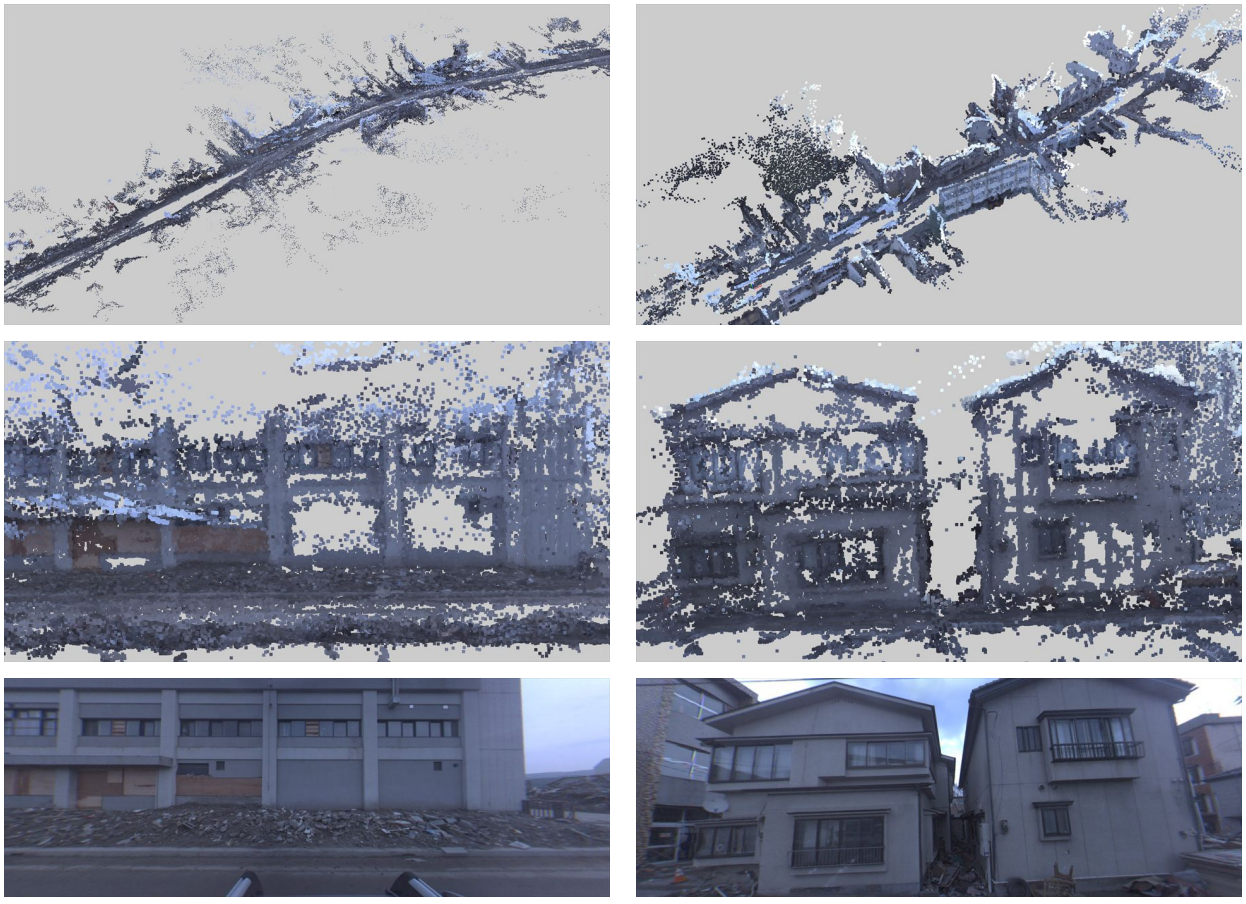


Figure 5.5: Results of PMVS2 when applied to our images. Top rows: The reconstructed structures. Middle rows: Their magnified portions. Bottom rows: One of the input images captured from similar viewpoints.

5.2 From image acquisition to change detection

5.2.1 Image acquisition

As mentioned earlier, we have been periodically acquiring the images of the tsunami-devastated areas in the northern-east coast of Japan. The images are captured by a vehicle having an omni-directional camera (Ladybug3 of Point Grey Research Inc.) on its roof. An image is captured at about every 2m on each city street to minimize the total size of the data as well as to maintain the running speed of the vehicle under the constraint of the frame rate of the camera.

The goal of the present study is to detect the temporal changes of a scene from its images thus obtained at two separate times. Figure 5.6 shows how the input images are processed. For computational simplicity, our algorithm for change detection takes as inputs not the omni-directional images but the perspective images cropped from them. The algorithm also needs the relative camera poses of these images. To obtain them, we perform SfM for each sequence followed by registration of the two reconstructions, which are summarized below.

The algorithm shown in the next section uses only several perspective images to detect changes of a scene. For the reason of accuracy, however, to obtain their camera poses, we perform SfM and registration not with these perspective images alone but with a more number (e.g., 100 viewpoints) of omni-directional images that contain these viewpoints. To be specific, we do this in the following two steps. First, we perform SfM independently for each sequence. We employ a standard SfM method [38, 61, 71] with extensions to deal with omni-directional images [31]. Next, we register the two 3D reconstructions thus obtained as follows. We first roughly align the two reconstructions with a similarity transform; putative matches of the feature points are established between the two sequences based on their descriptor similarity, for which RANSAC is performed [56]. For the aligned reconstructions, we reestablish the correspondences of feature points by incorporating a distance constraint. Using the newly established correspondences along with original correspondences within each sequence, we perform bundle adjustment for the extended SfM problem, in which the sum of the reprojection errors for all the correspondences is minimized. Figure 5.7(a) shows the initial rough alignment of the two reconstructions and (b) shows the final result.

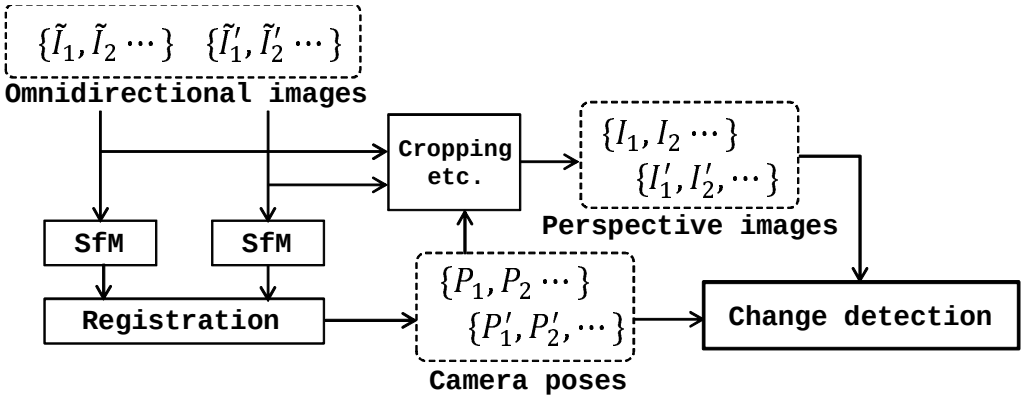
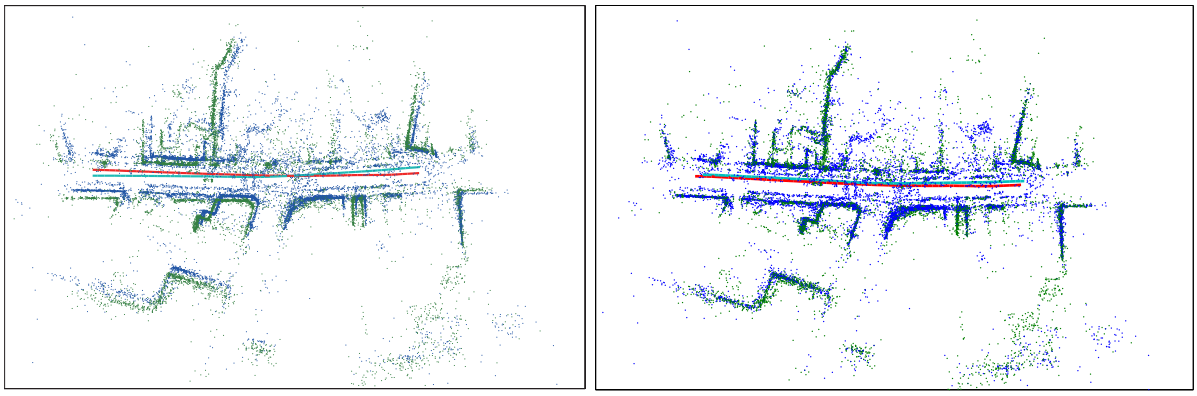


Figure 5.6: Data flow diagram.



(a) (b)

Figure 5.7: Registration of 3D reconstructions from two image sequences taken at different times. (a) Initial estimate. (b) Final result.

5.3 Detection of temporal changes of a scene

5.3.1 Problem

Applying the above methods to two sequences of omni-directional images, we have the camera pose of each image represented in the same 3D space. Choosing a portion of the scene for which we want to detect changes, we crop and warp the original images to have two sets of perspective images covering the scene portion just enough, as shown in Fig. 5.8. In this section, we consider the problem of detecting scene changes from these two sets of multi-view perspective images. For simplicity of explanation, we mainly consider the minimal case where there are two images in each set.

5.3.2 Outline of the proposed method

We denote the first set of images of time t by $\mathcal{I} = \{I_1, I_2\}$ and the second set of time t' by $\mathcal{I}' = \{I'_1, I'_2\}$. As shown in Fig. 5.9, one of the two image sets, $\mathcal{I}$, is used for estimating the depths of the scene, and the other image set $\mathcal{I}'$ is used for estimating changes of the scene depths. (These may be swapped.) Choosing one image from $\mathcal{I}$, say I_1, which we call a *key frame* here, the proposed method considers the scene depth at each pixel of I_1 and estimates whether or not it changes from t to t'. The output of the method is the probability of a depth change at each pixel of I_1.

For the first image set $\mathcal{I}_1$, its images are used to estimate the depth map of the scene at t. To be specific, not the value of the depth d but its probabilistic density $p(d)$ is estimated. For the other set $\mathcal{I}'$, a spatial point having depth d at a certain pixel of the key frame I_1 is projected onto I'_1 and I'_2, respectively, as shown in Fig. 5.8, and then the similarity s'_d of the local patches around these two points is computed. The higher the similarity is, the more the spatial point is likely to belong to the surface of some object in the scene at t', and the inverse is true as well. The similarity s'_d is computed for each depth d, which gives a density function of d that is similar to $p(d)$.

By combining these two estimates, $p(d)$, and s'_d, the proposed method calculates the probability of a depth change. In this process, the change probability evaluated for each depth d is integrated over d to yield the overall probability of a depth change. This makes it unnecessary to explicitly determine the scene depth neither at t nor t'. This is a central idea of the proposed method.

It should also be noted that our method evaluate the patch similarity only within each image set of $\mathcal{I}_1$ and $\mathcal{I}_2$. This makes it free from the illumination changes between the time points of the image capture.

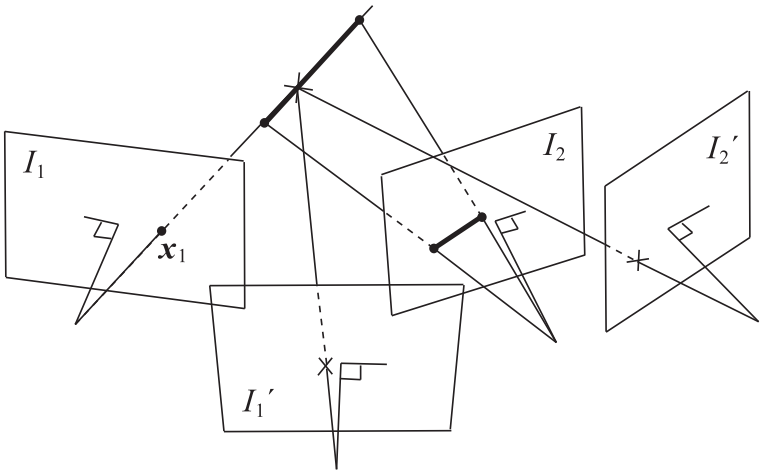


Figure 5.8: Geometry of two sets of multi-view perspective images taken at different times. For each pixel $\mathbf{x}_1$ of I_1, the probability that the scene depth has changed is estimated.

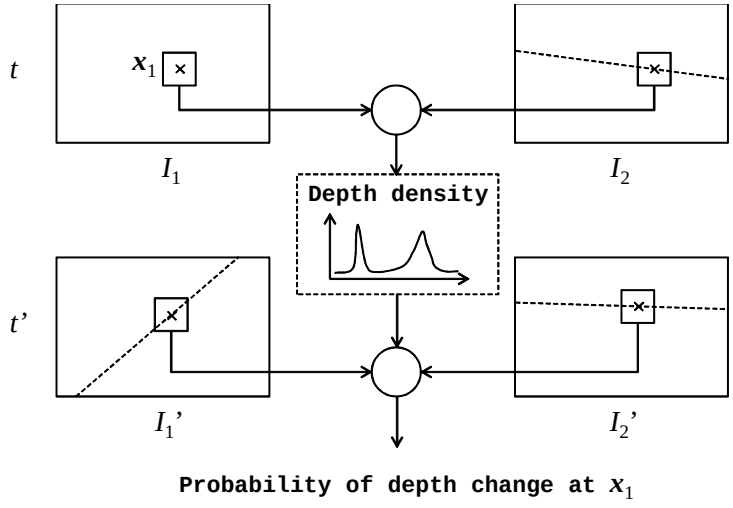


Figure 5.9: Outline of the proposed method. The probability density of the scene depth at a point $\mathbf{x}_1$ of I_1 is estimated from I_1 and I_2. This is combined with the comparison of the local patches of I_1' and I_2' to estimate the probability that the scene depth changes at $\mathbf{x}_1$ between t and t'. Note that the patches are compared only among the images taken at the same time. The broken lines in the images indicate epipolar lines associated with $\mathbf{x}_1$.

5.3.3 Estimation of the density of scene depths

To estimate the density of scene depths, we use the similarity of local patches in the images, as is done in multi-view stereo [1, 72, 10, 70, 73]. By dividing the inverse depth in a certain range from near to far away into n discrete values, we denote the depth by indexes $d = 1, \ldots, n$. For a point $\mathbf{x}_1$ of I_1, we denote the projection onto I_2 of a spatial point lying on the ray of $\mathbf{x}_1$ and having depth d by $\mathbf{x}_2(d)$. The difference between the local patches around $\mathbf{x}_1$ and $\mathbf{x}_2(d)$ is evaluated by the similarity (rigorously dissimilarity) function

$$s_d(\mathbf{x}_1) = \frac{1}{3|\mathcal{W}|} \sum_{r,g,b} \sum_{\delta\mathbf{x} \in \mathcal{W}} \left| I_1(\mathbf{x}_1 + \delta\mathbf{x}) - I_2(\mathbf{x}_2(d) + \delta\mathbf{x}) \right|, \tag{5.1}$$

where $\mathcal{W}$ defines the size of the local patches. (We used 5×5 pixels in the experiment.)

Although s_d for correctly matched points will ideally be 0, it will not in practice because of image noise, shape changes of the patches, etc. Having examined s_d for correctly matched points, we found that its distribution is well approximated by a half Laplace distribution; see the supplementary note for details. Then, we model $p(s)$ as

$$p(d) \propto \exp\left(-s_d/\sigma\right), \quad (d = 1, \ldots, n). \tag{5.2}$$

The probabilities $[p(d = 1), \ldots, p(d = n)]$ are obtained by normalizing the above so that their sum will be 1. We set $\sigma = 1.5$ in the experiments based on the statistics of real images; see the following section.

5.3.4 Estimating probabilities of scene changes

We introduce a binary variable c to represent whether or not the scene depth at a pixel $\mathbf{x}_1$ of the key frame I_1 has changed from t to t'; $c = 1$ indicates it has changed and $c = 0$ it has not.

Suppose projecting onto I'_1 and I'_2 a spatial point lying on the ray of $\mathbf{x}_1$ and having depth d, as shown in Fig. 5.8. We denote these two points by $\mathbf{x}'_1(d)$ and $\mathbf{x}'_2(d)$, respectively. Similarly to Eq. (5.1), the difference of the local patches around these two points is calculated as

$$s'_d = \frac{1}{3|\mathcal{W}|} \sum_{r,g,b} \sum_{\delta\mathbf{x} \in \mathcal{W}} \left| I'_1(\mathbf{x}'_1(d) + \delta\mathbf{x}) - I'_2(\mathbf{x}'_2(d) + \delta\mathbf{x}) \right| \tag{5.3}$$

Computing $s'_1, ..., s'_n$ for the depths $d = 1, ..., n$ from the images, we consider evaluating the following posterior probability given $s'_1, ..., s'_n$ as observations:

$$p(c = 1 | s'_1, ..., s'_n). \tag{5.4}$$

This directly gives the probability that the scene changes its structure at the pixel $\mathbf{x}_1$ of I_1. This can be rewritten by Bayes' rule as

$$p(c=1|s'_1,...,s'_n) = \frac{p(s'_1,...,s'_n|c=1)p(c=1)}{p(s'_1,...,s'_n)}. \tag{5.5}$$

The denominator is given by

$$\begin{aligned} p(s'_1,...,s'_n) &= p(s'_1,...,s'_n|c=1)p(c=1) \\ &\quad + p(s'_1,...,s'_n|c=0)p(c=0). \end{aligned} \tag{5.6}$$

Here, the term $p(c=1)$ is the prior probability that the scene depth changes at this pixel. We set a constant number to $p(c=1)$. Its inverse $p(c=0)$ is given by $p(c=0) = 1-p(c=1)$.

We next evaluate $p(s'_1,...,s'_n|c=1)$ and $p(s'_1,...,s'_n|c=0)$. We assume that $s'_d(d=1,\ldots,n)$ is independent of each other and that

$$p(s'_1,...,s'_n|c=1) = \prod_{d=1}^{n} p(s'_d|c=1), \tag{5.7a}$$

$$p(s'_1,...,s'_n|c=0) = \prod_{d=1}^{n} p(s'_d|c=0). \tag{5.7b}$$

To further analyze $p(s'_d|c=1)$ and $p(s'_d|c=0)$, we introduce a binary variable δ_d to represent whether or not the scene depth (at $\mathbf{x}_1$ of I_1 at time t) is d, that is, whether or not the spatial point having depth d belongs to the surface of some object at t; $\delta_d = 1$ indicates this is the case and $\delta_d = 0$ otherwise. Using δ_d, $p(s'_d|c=1)$ can be decomposed as follows:

$$\begin{aligned} p(s'_d|c=1) &= p(s'_d,\delta_d=1|c=1) + p(s'_d,\delta_d=0|c=1) \\ &= p(s'_d|\delta_d=1,c=1)p(\delta_d=1) \\ &\quad + p(s'_d|\delta_d=0,c=1)p(\delta_d=0), \end{aligned} \tag{5.8}$$

where $p(\delta_d=1|c=1) = p(\delta_d=1)$ and $p(\delta_d=0|c=1) = p(\delta_d=0)$ are used, which is given by the independence of δ_d and c. The density $p(s'_d|c=0)$ can be decomposed in a similar way. The term $p(\delta_d=1)$ in Eq. (5.8) is the probability that the scene depth is d, and thus it is equivalent to $p(d)$ that has been already obtained; thus, $p(\delta_d=1) = p(d)$. The term $p(\delta_d=0)$ is given by $p(\delta_d=0) = 1-p(\delta_d=1) = 1-p(d)$.

To evaluate Eq. (5.8), we need to further consider the conditional densities $p(s'_d|\delta_d=1,c=1)$ and $p(s'_d|\delta_d=0,c=1)$. There are four combinations of (δ_d,c): $(0,0)$, $(0,1)$, $(1,0)$, and $(1,1)$. Each combination can be related to whether the scene depth is d at time t' or not. For example, $(\delta_d,c)=(1,0)$ means that the scene depth is d at time t

Table 5.1: Values of δ'_d for different pairs of c and δ_d. The definition of the variables is as follows: $c = 1$ indicates the scene depth changes from t to t' and $c = 0$ otherwise; $\delta_d = 1$ indicates the scene depth is d at time t and $\delta_d = 0$ otherwise; δ'_d is the same as δ_d but not at t but t'.

c \ δ_d	0	1
0	0	1
1	0 or 1	0

and remains so at t'; $(\delta_d, c) = (1, 1)$ means that the scene depth is d at t and is not so at t'. Let δ'_d be a binary variable indicating whether or not the scene depth is d at time t'; $\delta'_d = 1$ if the scene depth is d at t' and $\delta'_d = 0$ otherwise. Table 5.1 shows the values of δ'_d for all the combinations.

Note that the combination $(\delta_d, c) = (0, 1)$, which means that the scene depth is not d at t and changes at t', does not fully constrain δ'_d. Thus we denote it by δ'_d is either 0 or 1.

From the table, we can rewrite the conditional densities for the four combinations as

$$p(s'_d|\delta_d = 0, c = 0) = p(s'_d|\delta'_d = 0), \tag{5.9a}$$
$$p(s'_d|\delta_d = 0, c = 1) = p(s'_d|\delta'_d = 0 \text{ or } 1), \tag{5.9b}$$
$$p(s'_d|\delta_d = 1, c = 0) = p(s'_d|\delta'_d = 1), \tag{5.9c}$$
$$p(s'_d|\delta_d = 1, c = 1) = p(s'_d|\delta'_d = 0). \tag{5.9d}$$

The densities on the right hand side can be modelled as follows. When $\delta'_d = 0$, which means the scene depth is not d (at t'), s'_d measures the similarity between the patches of two different scene points. Thus, we model $p(s'_d|\delta'_d = 0)$ by a uniform distribution and set

$$p(s'_d|\delta'_d = 0) = \text{const.} \tag{5.10}$$

When $\delta'_d = 1$, on the other hand, s'_d measures the similarity between the patches of the same scene point. Then, this is exactly the same situation as s_d for correctly matched points. Thus, using the same half Laplace distribution as s_d, we set $p(s'_d|\delta'_d = 1) \propto \exp(-s'_d/\sigma')$. In the experiments, we set $\sigma'(= \sigma) = 1.5$.

The conditional density $p(s'_d|\delta_d = 0, c = 1)$ can be factorized as follows:

$$\begin{aligned} p(s'_d|\delta_d = 0, c = 1) = {} & p(s'_d|\delta'_d = 0, \delta_d = 0, c = 1)p(\delta'_d = 0|\delta_d = 0, c = 1) \\ & + p(s'_d|\delta'_d = 1, \delta_d = 0, c = 1)p(\delta'_d = 1|\delta_d = 0, c = 1). \end{aligned} \tag{5.11}$$

The probability $p({\delta'}_d = 1|\delta_d = 0, c = 1)$ is difficult to quantify, but, fortunately, it should be small. Thus, we approximate $p(s'_d|\delta_d = 0, c = 1) \approx p(s'_d|\delta'_d = 0, \delta_d = 0, c = 1)$.

Using the derived equations and the introduced models, the conditional probability $p(c = 1|s'_1, \ldots, s'_n)$ can be evaluated for each $\mathbf{x}_1$. We may judge that if the probability is higher than 0.5, the scene depth has changed at the pixel, and it has not changed, otherwise.

We have considered the minimal case of using a pair of images for each time. When two or more pairs of images are available, we can use them to improve estimation accuracy. In the experiments, we use a naive method, which integrates the observations from the multiple image pairs based on an assumption that they are independent of each other.

5.3.5 Modeling $p(s_d)$ for correctly matched points

As described in the previous section, the proposed method uses models of $p(s_d)$, the density of the patch similarity s_d of the correctly matched pair of points, and also $p(s'_d)$ that is similarly defined for s'_d. Behind this, there is a fact that even for correctly matched points, s_d will not be 0 due to image noises and shape changes of the patches. As mentioned in Section 5.3.3 and 5.3.4, we chose $p(s_d) = \exp(-s_d/\sigma)/\sigma$ and the same model for $p(s'_d)$; for the parameter σ, we chose $\sigma = 1.5$ throughout our experiments.

These choices are made based on the following analysis of real images. As we do not know correct matches of points among the images and thus the true density $p(s_d)$ is difficult to obtain, we instead computes $\tilde{s}_d = \min_d s_d$, the minimum similarity over possible depth d; we generate its frequency histogram for images of scenes without severe occlusion and specular reflections. By excluding scenes with occlusions etc., $\tilde{s}_d$ should be a good substitute for s_d for correctly matched points. Figure 5.10(a) shows the histogram of $\tilde{s_d}$ for about 5 million points from 30 image pairs of such scenes. Figure 5.10(b) shows our model $p(s_d) = \exp(-s_d/\sigma)/\sigma$ with $\sigma = 1.5$. It is seen that the shape of the histogram is well approximated by our model of a half Laplace distribution. We manually chose the parameter as $\sigma = 1.5$ by considering a few differences between the ideal s_d and $\tilde{s}_d$, such as, that the histogram does not have the maximum peak at $s_d = 0$, whereas it ideally should have.

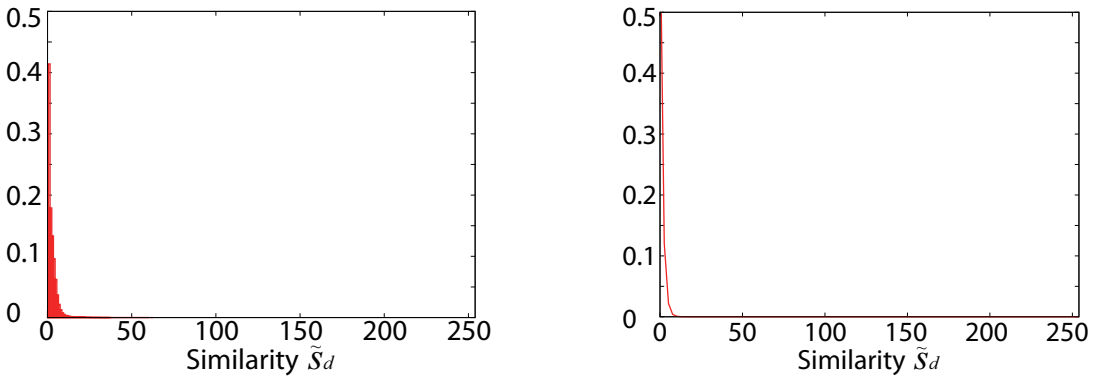


Figure 5.10: (a) Frequency histogram of $\tilde{s}_d$ for 5 million points from 30 pairs of images. (b) Our model of $p(s_d)$ for correctly matched points: a half Laplace distribution $\exp(-s_d/\sigma)/\sigma$ with $\sigma = 1.5$.

5.4 Experimental results

We conducted several experiments to examine the performance of the proposed method. For the experiments, we chose a few scenes and their images from our archives mentioned in Sec.5.2.1. The chosen images are taken at one and four months after the tsunami[1]. Typically, a lot of tsunami debris appear in the earlier images, whereas they disappear in the later ones because of recovery operations. We wish to correctly identify their disappearance in the later images.

The proposed method uses two or more images for each time. In the experiment, we use four images of consecutive viewpoints for each time, i.e., three pairs of images. These are perspective images (cropped from omni-directional images) of 640×480 pixel size. The disparity space is discretized into 128 blocks ($n = 128$). Assuming that there is no prior on the probability of scene changes, we set $p(c = 1) = 0.5$. It is noted, though, that in the experiments, the results are very robust to the choice of this value (Sec.5.4.3) These are fixed for all the experiments.

5.4.1 Compared methods

We compared our method with MVS-based ones, which first reconstruct the structures of a scene based on MVS and differentiate them to obtain scene changes. We consider two MVS algorithms for 3D reconstruction, PMVS2 [1] and a standard stereo matching algorithm for it.

In the former case, PMVS2 is applied to a sufficiently long sequence of images (e.g., 100 viewpoints) covering the target scene. Our omni-directional camera consists of six cameras and records six perspective images at each viewpoint. All these six images per viewpoint are inputted to PMVS2 after distortion correction. PMVS2 outputs point clouds, from which we create a depth map viewed from the key frame. This is done by projecting the points onto the image plane in such a way that each point occupies an image area of 7×7 pixels. Two depth maps are created for the two time points and are differentiated to obtain scene changes. We call the overall procedure PMVS2.

In the latter case, a standard stereo matching algorithm is used, in which a MRF model is assumed that is defined on the four-connected grid graph; the local image similarity is used for the data term and a truncated l_1 norm $f_{ij} = \max(|d_i - d_j|, d_{\max}/10)$ is used for the smoothness term. We use two types of similarity; one is the SAD-based one (Eq. (5.1) and Eq. (5.3)) that is used in our method, and the other is the distance between

[1]The data used in this study (the omni-directional image sequences of the chosen streets and our estimates of their camera poses) are available from our web site: http://www.vision.is.tohoku.ac.jp/us/download/.

SIFT descriptors at the corresponding points [74]. Then, the optimization of the resulting MRF models is performed using graph cuts [75]. Similarly to the above, two depth maps are computed and are differentiated to obtain scene changes. We call these procedures patch-MVS and SIFT-MVS.

5.4.2 Comparison of the results

Figure 5.11 shows the results for a scene. From left to right columns, the input images with a hand-marked ground truth, the results of the proposed method, PMVS2, Patch-MVS, and SIFT-MVS, respectively. For the proposed method, besides the detected changes, the change probability $p(c = 1|\cdots)$ is shown as a grey-scale image; its binarized version by a threshold $p > 0.5$ gives the result of change detection. For each of the MVS-based methods, besides the result, two estimated depths maps for the different times are shown. The detection result is their differences. Whether the scene changes or not is judged by whether the difference in its disparity is greater than a threshold. We chose 6 (disparity ranges in $[0 : 127]$) for the threshold, as it achieves the best results in the experiments. The red patches in the depth maps of PMVS2 indicate that there is no reconstructed point in the space.

Comparing the result of the proposed method with the ground truth, it is seen that the proposed method can correctly detect the scene changes, i.e., the disappearance of the debris and the digger; the shape of the digger arm is extracted very accurately. There are also some differences. The proposed method cannot detect the disappearance of the building behind the digger and of the thin layer of sands on the ground surface. The former is considered to be because the building is occluded by the digger in other viewpoints. The proposed method does not have a mechanism of explicitly dealing with occlusions but using multiple pairs of images, which will inevitably yield some errors. For the layer of sands, its structural difference might be too small for the proposed method to detect it.

The results of the MVS-based methods are all less accurate than the proposed method. As these methods differentiate the two depth maps, a slight reconstruction error in each will results in a false positive. Thus, even though their estimated depths appear to capture the scene structure mostly well, the estimated scene changes tends to be worse than the impression we have for each depth map alone.

There are in general several causes of errors in MVS-based depth estimation. For example, MVS is vulnerable to objects without textures (e.g., the ground surface in this scene). PMVS2 does not reconstruct objects that do not have reliable observations, e.g., textureless objects. As the proposed method similarly obtains depth information from image similarity, the same difficulties will have bad influence on the proposed method.

However, it will be minimized by the probabilistic treatment of the depth map; taking all probabilities into account, the proposed method makes a binary decision as to whether a scene point changes or not.

We obtain precision and recall for each result using the ground truth and then calculate its F_1 score; it is 0.76, 0.59, 0.53, 0.71, in the order of Fig. 5.11, respectively.

Figure 5.12 shows results for other images. From top to bottom rows, I', the ground truths, the results of the proposed method, and those of SIFT-MVS are shown, respectively. It is seen that the proposed method produces better results for all the images. This is quantitatively confirmed by their F_1 scores which are shown in Table 5.2.

Figure 5.14 shows an extended version of Fig.5.12, and Figs.5.15 and 5.16 show results for two different scenes. In these figures, the results obtained by the proposed method, PMVS2, Patch-MVS, and SIFT-MVS are shown, along with the depth maps obtained by PMVS2. Similar to the results of Fig. 5.11 and 5.12, it can be observed that the proposed method performs better than any of the other MVS-based methods.

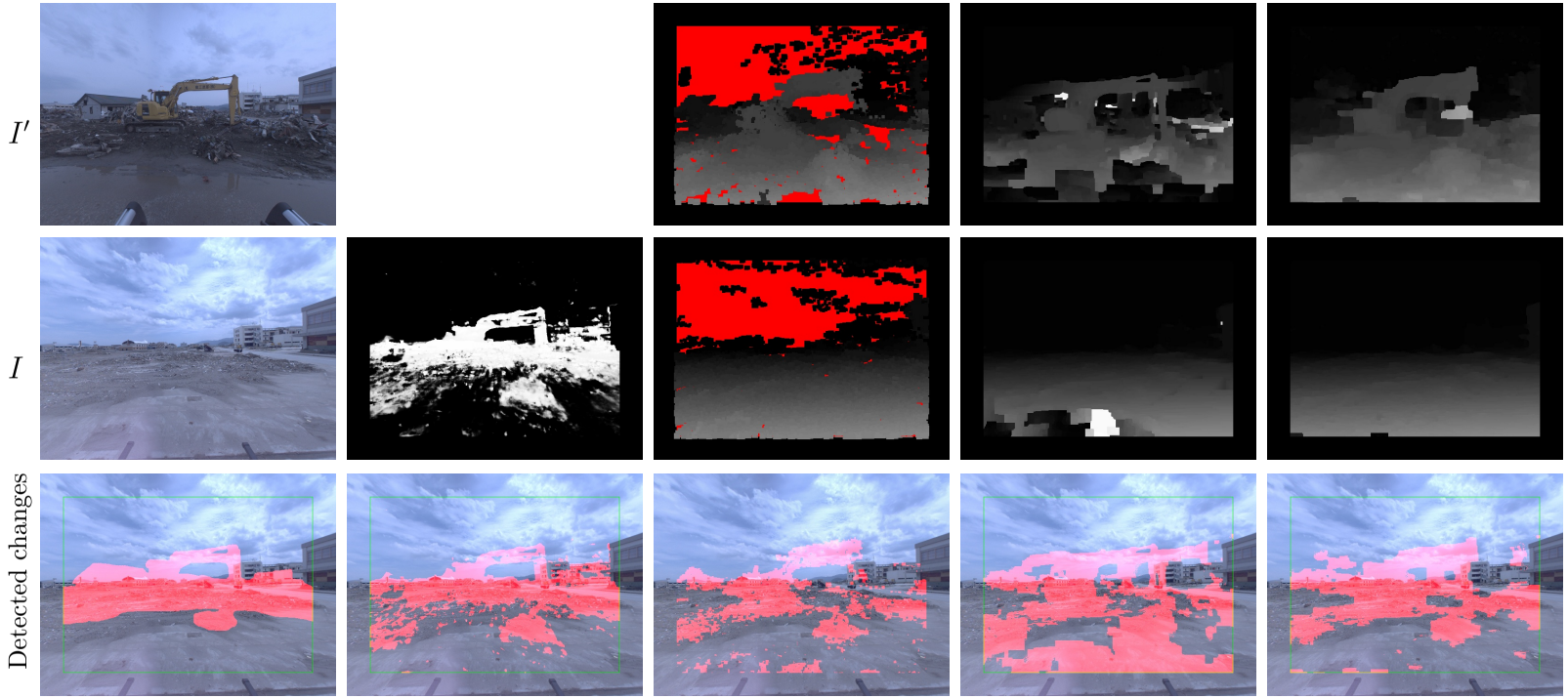


Figure 5.11: Results of the proposed method and the three MVS-based ones for a scene. From left to right columns, the input images and the ground truth, the results of the proposed methods, and those of PMVS2, Patch-MVS, and SIFT-MVS, respectively. The third row shows the detected changes. The numbers in their captions are the F_1 scores representing accuracy of the detection.

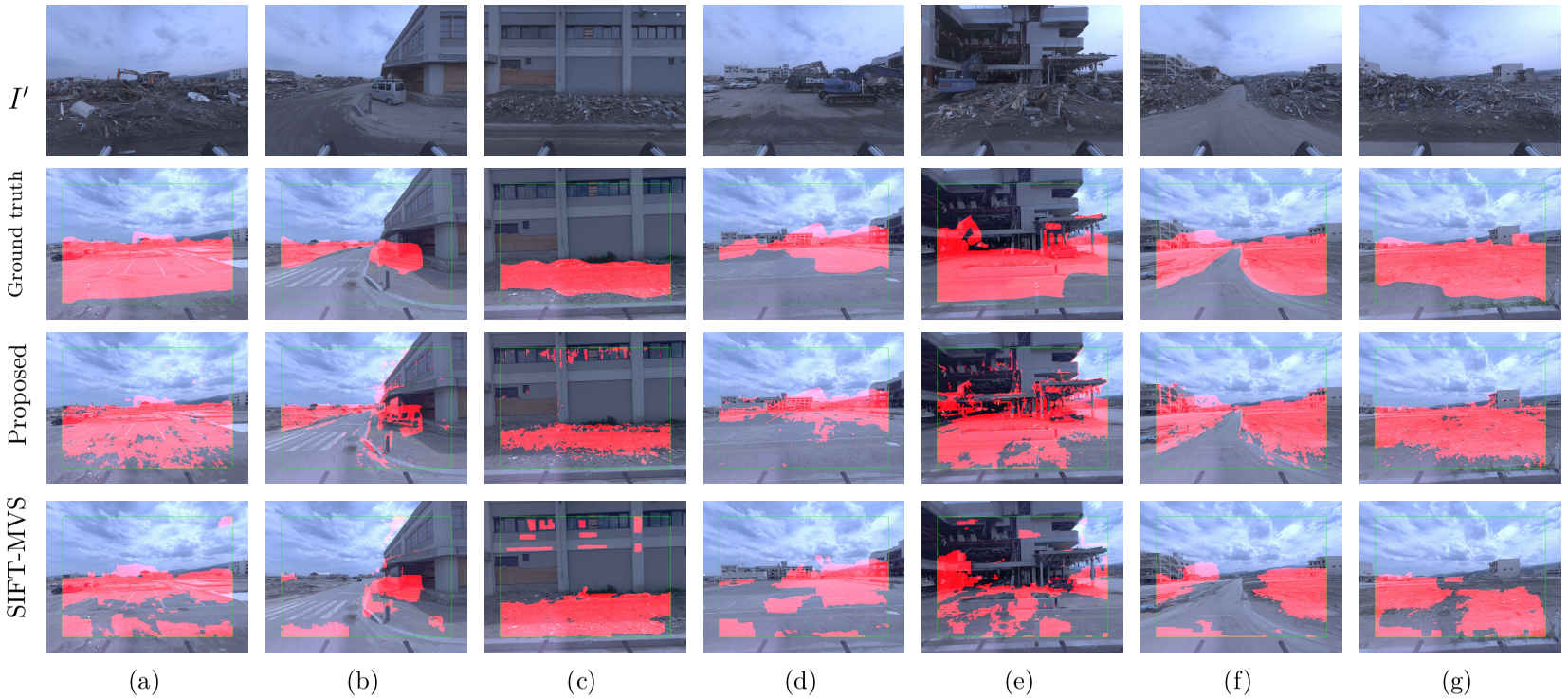


Figure 5.12: Results for other images. From top to bottom rows, I', hand-marked ground truths, results of the proposed method, and those of SIFT-MVS.

Table 5.2: F_1 scores of the detected changes shown in Fig. 5.12.

	(a)	(b)	(c)	(d)	(e)	(f)	(g)	Average
Proposed	0.88	0.67	0.77	0.85	0.82	0.91	0.92	0.83
PMVS2	0.49	0.30	0.65	0.66	0.56	0.58	0.66	0.56
Patch-MVS	0.66	0.28	0.69	0.60	0.70	0.65	0.77	0.62
SIFT-MVS	0.68	0.41	0.73	0.71	0.60	0.67	0.73	0.65

5.4.3 Prior on the probability of scene changes

In the proposed method, $p(c = 1)$, the prior on the probability of scene changes, needs to be specified. As mentioned in Section 5.3.4, we set $p(c = 1) = 0.5$ for all the experiments. We show here that the choice does not affect the results much. Figure 5.13 shows the results obtained when different values of $p(c = 1)$ are used. Table 5.3 shows the accuracy of change detection. It is seen from these that the results tend to be worse only for small $p(c = 1)$, i.e., $p(c = 1) \leq 0.3$.

5.5 Summary

We have described a method for detecting temporal changes of the 3D structure of an outdoor scene from its multi-view images taken at two separate times. These images are captured by a vehicle-mounted camera running in a city street. The method estimates the scene depth probabilistically, not deterministically, and judges whether or not the scene depth changes in such a way that the ambiguity of the estimated scene depth is well reflected in the final estimates. We have shown several experimental results, in which the proposed method is compared with MVS-based methods, which use MVS to reconstruct the scene structures and differentiate two reconstructions to detect changes. The experimental results show that the proposed method outperforms the MVS-based ones.

It should be noted that our method estimates scene changes independently at each image pixel; no prior on the smoothness or continuity of scene structures is used. This is contrasted with MVS, which always uses some prior about them. Such priors, which have been confirmed to be very effective in dense reconstruction, are in reality a double-edged sword. We may say that the reason why MVS needs such priors is because it has to deterministically determine scene structures even if only insufficient observations are

available. Considering that our method achieves better results (even) without such priors, it could be possible that such priors do more harm than good as far as change detection is concerned.

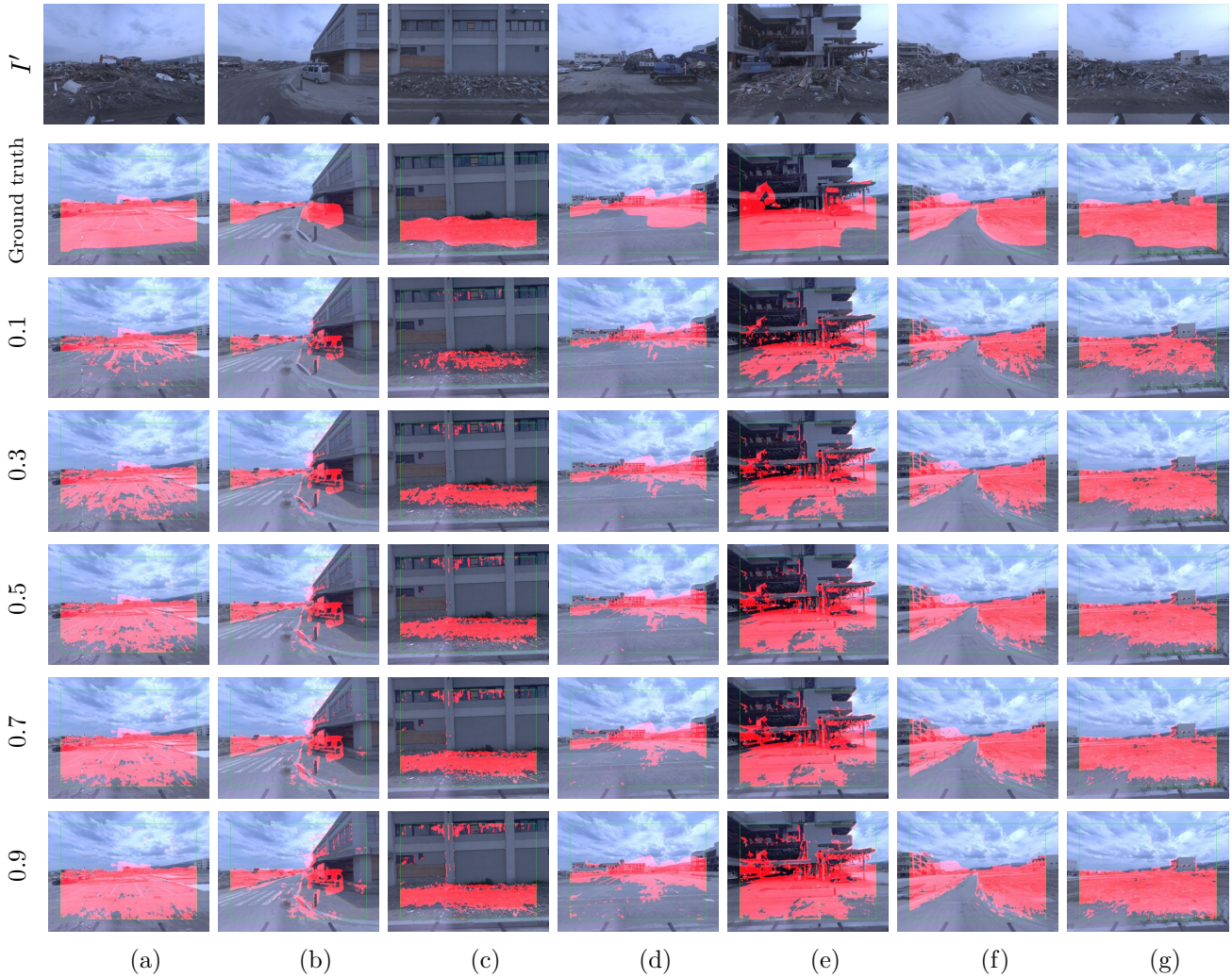


Figure 5.13: Results of the proposed method for different $p(c = 1)$ values.

Table 5.3: F_1 scores of the proposed method for different $p(c = 1)$ values for the scene shown in Fig. 5.13.

$p(c = 1)$	(a)	(b)	(c)	(d)	(e)	(f)	(g)	Average
0.1	0.59	0.65	0.44	0.73	0.70	0.80	0.80	0.67
0.2	0.75	0.67	0.65	0.80	0.78	0.87	0.88	0.77
0.3	0.82	0.68	0.72	0.82	0.80	0.89	0.91	0.81
0.4	0.86	0.67	0.75	0.84	0.81	0.90	0.91	0.82
0.5	0.88	0.67	0.77	0.85	0.82	0.91	0.92	0.83
0.6	0.89	0.66	0.79	0.85	0.82	0.91	0.93	0.84
0.7	0.90	0.65	0.80	0.85	0.82	0.91	0.93	0.84
0.8	0.91	0.64	0.81	0.85	0.82	0.92	0.93	0.84
0.9	0.92	0.62	0.81	0.84	0.82	0.92	0.93	0.84

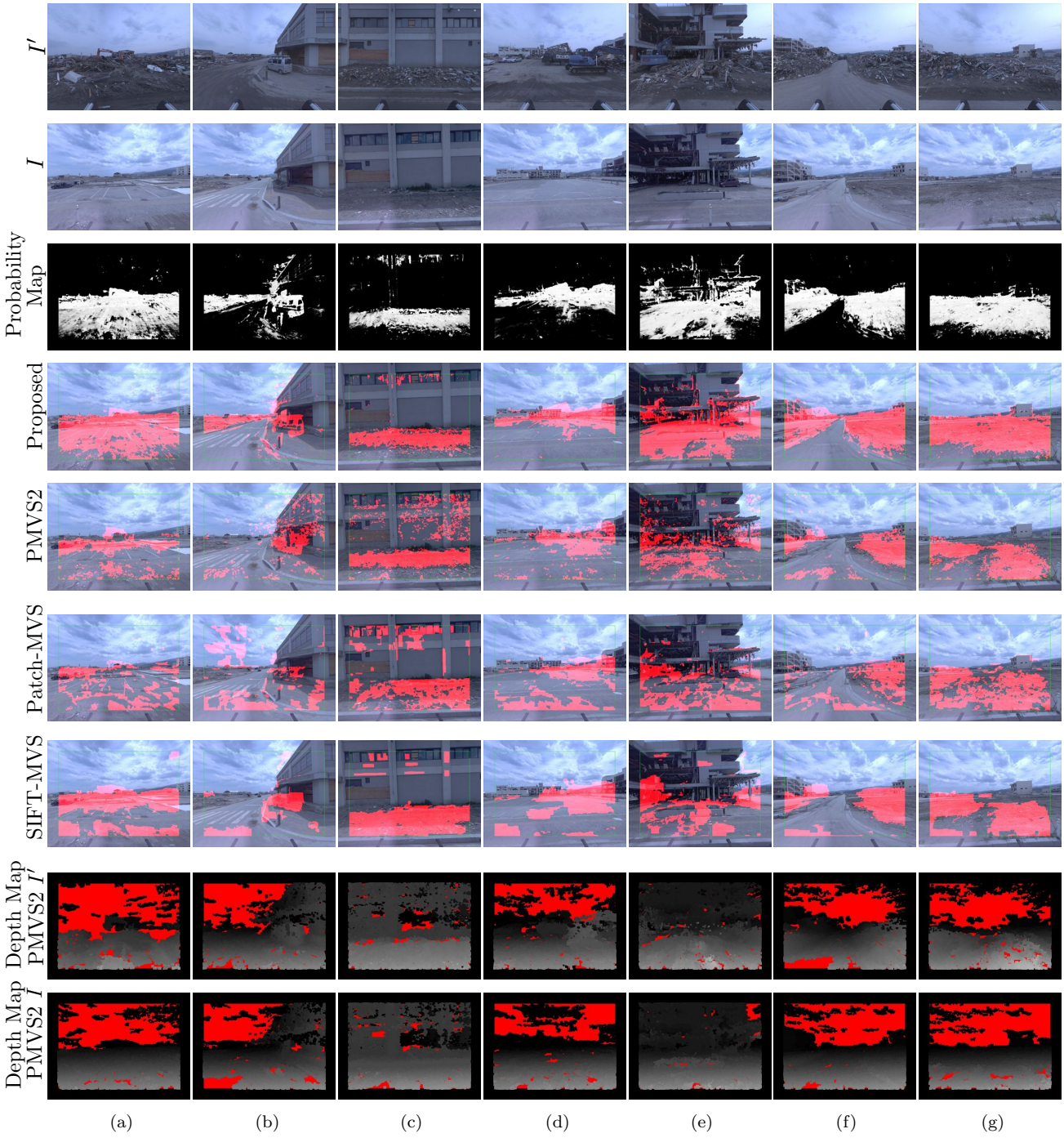


Figure 5.14: Extended results for the scene of Fig.5.12. From top to bottom rows, I', I, the change probability maps, the results of the proposed method, those of PMVS2, Patch-MVS, SIFT-MVS, and the depth maps obtained by PMVS2, respectively.

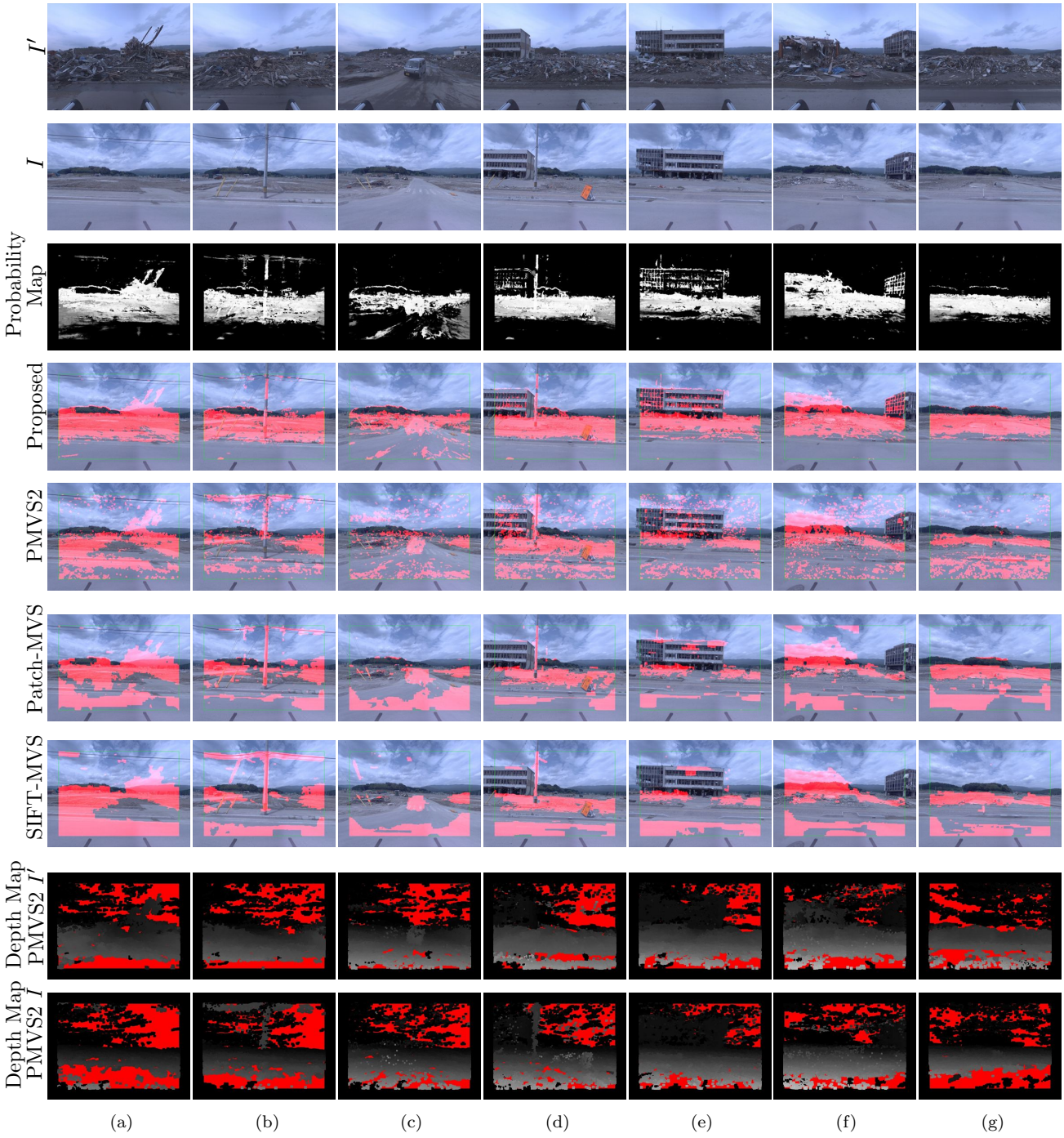


Figure 5.15: Results for a different scene. From top to bottom rows, I', I, the change probability maps, the results of the proposed method, those of PMVS2, Patch-MVS, SIFT-MVS, and the depth maps obtained by PMVS2, respectively.

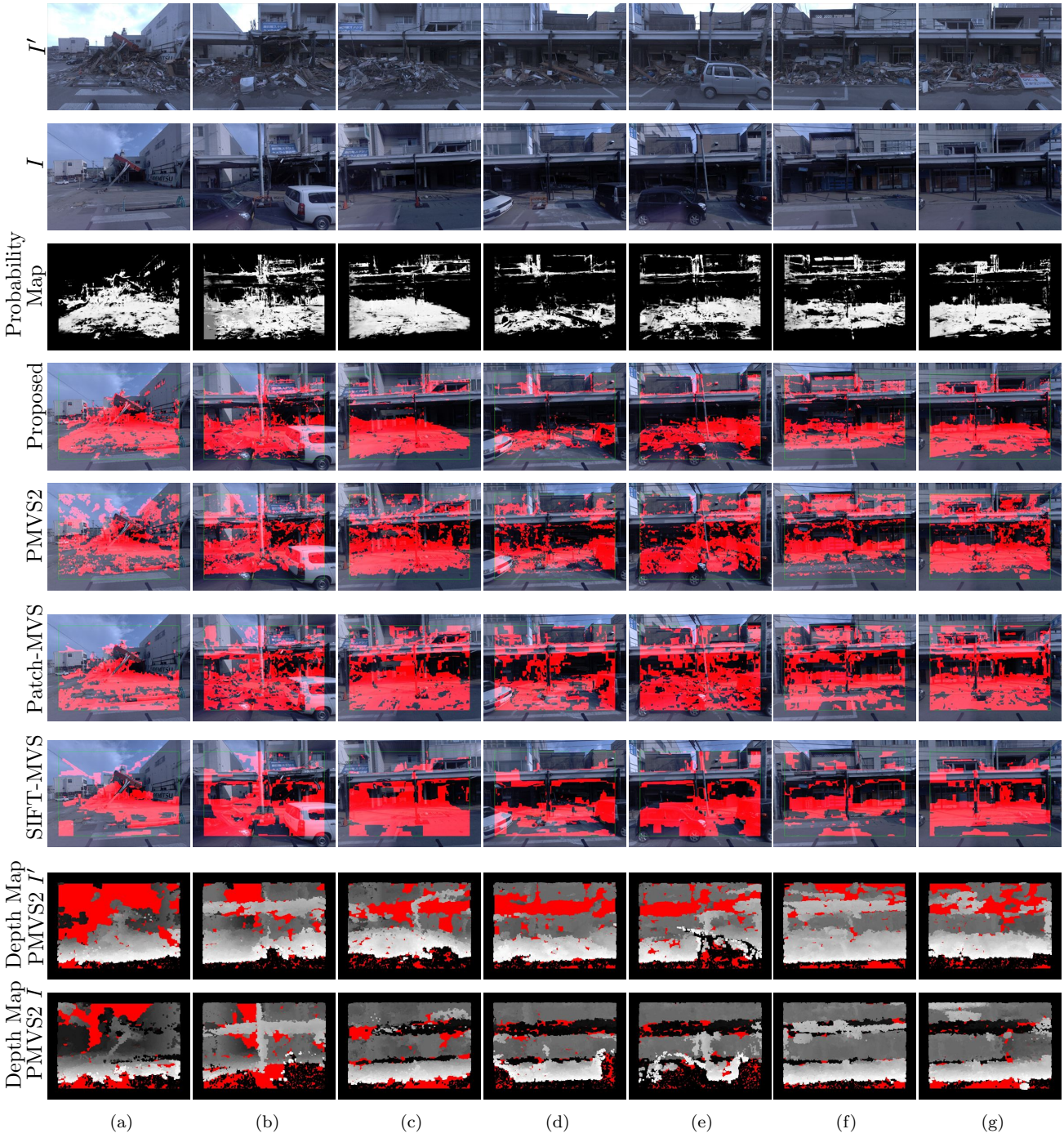


Figure 5.16: Results for a different scene. From top to bottom rows, I', I, the change probability maps, the results of the proposed method, those of PMVS2, Patch-MVS, SIFT-MVS, and the depth maps obtained by PMVS2, respectively.

Chapter 6

Land Surface Condition Analysis

This chapter presents a unified framework for robustly integrating image data taken at vastly different viewpoints to generate large-scale estimates of land surface conditions. The previous sections proposed the 2D and 3D change detection methods. The method proposed in this chapter estimates change of debris distribution in a city based on object recognition. For recovery operation in tsunami-damaged area, it is essential to make it possible to understand debris distribution in a city.

Automated visual analysis is an effective method for understanding changes in natural phenomena over massive city-scale landscapes. However, the view-point spectrum across which image data can be acquired is extremely wide, ranging from macro-level overhead (aerial) images spanning several kilometers to micro-level front-parallel (street-view) images that might only span a few meters. To validate the proposed approach this study attempt to estimate the amount of post-tsunami damage over the entire city of Kamaishi, Japan (over 4 million square-meters). The results show that the proposed approach can efficiently integrate both micro and macro-level images, along with other forms of meta-data, to efficiently estimate city-scale phenomena. Experiments evaluate the proposed approach on two modes of land condition analysis, namely, city-scale debris and greenery estimation, to show the ability of the proposed method to generalize to a diverse set of estimation tasks.

6.1 Motivation

We address the task of estimating large-scale land surface conditions using overhead aerial (macro-level) images and street view (micro-level) images. These two types of images are captured from orthogonal viewpoints and have different resolutions, thus conveying very different types of information that can be used in a complementary way. Moreover, their integration is necessary to make it possible to accurately understand changes in natural

Figure 6.1: Aerial images affected by weather condition (Left: March 11, 2011, Right: March 31, 2011). The land surface might be covered by clouds and illumination conditions change drastically in aerial image.

phenomena over massive city-scale landscapes.

Aerial images are an excellent source for collecting wide-area information of land surface conditions. However, it may come at the cost of a lower resolution (i.e., number of pixels per meter) and visiiblity may drastically change depending on the weather. For example, clouds may obscure the visibility of the land surface (Fig. 6.1). A more important limitation of aerial images is that they are limited to a vertical (top-down) perspective of the ground surface, such that areas occluded by a roof or highway overpass are not visible to the camera (first and second row of Fig. 6.2) making it difficult to estimate land conditions in covered areas.

Street-view images, on the other hand, captured from the ground-level can obtain higher resolution images of vertical structures and have better access to information about covered areas. They are also less affected by weather conditions. In the same token however, street view images are constrained to the ground plane and a single image has limited physical range. It is also labor intensive to acquire street-level images of large land surface areas (i.e., millions of square meters).

The key technical challenge is devising a method to integrate these two disparate types of image data in an effective manner, while leveraging the wide coverage capabilities of macro-level images and detailed resolution of micro-level images. The strategy proposed in the work uses macro-level imaging to learn land condition correspondences between land regions that share similar visual characteristics (e.g, mountains, streets, buildings, rivers), while micro-level images are used to acquire high resolution statistics of land conditions (e.g., the amount of debris on the ground). By combining the macro and micro level information about region correspondences and surface conditions, our proposed method generates detailed estimates of land surface conditions over the entire city.

The technical contribution of this paper is a novel procedure for generalizing from a

Figure 6.2: Example aerial and street-view images. There are many cases in which aerial images and street-view images give complementary information about the land surface condition. For example, the areas covered by the building roof (the top and second row), stacked objects (the bottom row) are best viewed from the street.

sparse set of visual recognition results to a large-scale land condition regression estimate. The proposed system carefully brings together the state-of-the-art algorithms for semantic scene understanding, structure-from-motion and non-parametric regression to generate a massive city-scale land condition probability map (Fig.6.3). To the best of our knowledge, this is the first work of its kind to use sparse image-based street-level object recognition results to extrapolate the surface conditions of an entire city (over 4 million square meters).

Although our method can generalize to different types of large-scale phenomena, we ground our proposed approach in a real-world application of post-Tsunami city-scale damage estimation. In regions affected by such disasters, it is extremely hard to efficiently assess the large-scale impact of a natural disaster. Technologies that enable fast and efficient city-scale estimates of damage can be extremely helpful for expediting aid to seriously damages areas. The approach describe in this paper can also be used for long-term analysis by monitoring and tracking recovery efforts.

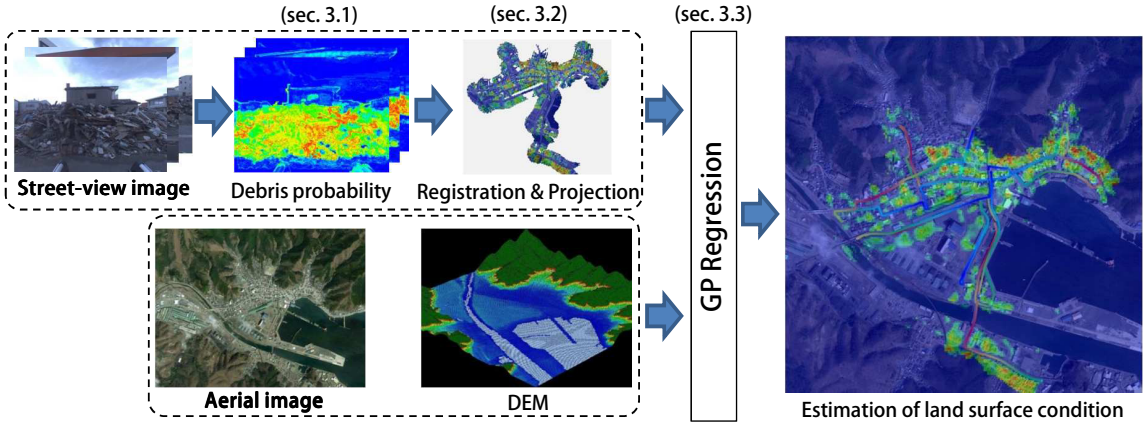


Figure 6.3: Data flow diagram of city-scale estimation of land surface condition. Our approach efficiently integrates both micro (street-view) and macro-level (aerial) images along with other forms of meta-data to estimate city-scale land surface condition.

6.2 Large-scale estimation of land surface condition

Our framework integrates aerial and street-view images to estimate land surface conditions. In this section, we explain the details of the proposed method contextualized for post-Tsunami debris detection. Although the following explanation takes debris as an example, the method is generally applicable to other types of land surface conditions. The proposed method consists of the following three steps;

(i) Debris detection on perspective street-view image. (sec.6.2.1)

(ii) Projection of debris probabilities on street-view images to the ground using building contours. (sec.6.2.2)

(iii) Estimation of debris over an entire city by integrating the projection result with all other data (e.g. aerial image, DEM) using a Gaussian process.(sec.6.2.3)

In the first step, the probability map of debris is calculated for each street-view image. Then, using the camera parameters for the street-view image, the probability map is projected onto the ground plane registered to a corresponding part of the aerial image.This projection method takes the existence of building walls into consideration. Finally in order to complement the estimation results obtained from street-view images, the projected probability map is integrated with the information obtained from aerial images and DEM using Gaussian process regression model.

6.2.1 Debris detection

We developed a method to calculate the probability map of debris (Fig. 6.4). The debris model is learned from a hand-labeled training image.The debris in the images are irregular, complicated in shape and appearance. Therefore, we exploit Geometric Context [60] as geometric feature and pixel-wise object probability [76] as an appearance feature. Geometric Context estimates the probabilities that a super-pixel belongs to seven classes. We chose four of the seven classes, "ground plane", "sky", "porous non-planar" and "solid non-planar", and used the probabilities of them as debris features. The pixel-wise object probability p_{object} is calculated using [76], Lab, HOG[77], BRIEF[78] and ORB[79]. The feature vector of debris is as follows.

$$\mathbf{x} = \left(p_{\mathrm{ground}}, p_{\mathrm{sky}}, p_{\mathrm{porous}}, p_{\mathrm{solid}}, p_{\mathrm{object}}, m_{\mathrm{patch}}, v_{\mathrm{patch}}\right)^{\mathrm{T}}, \tag{6.1}$$

where p_{ground}, p_{sky}, p_{porous} and p_{solid} are the probabilities of "ground plane", "sky", "porous non-planar" and "solid non-planar", respectively. In addition to these probabilities, mean m_{patch} and variance v_{patch} of grayscale patch (5×5) are added to the features.

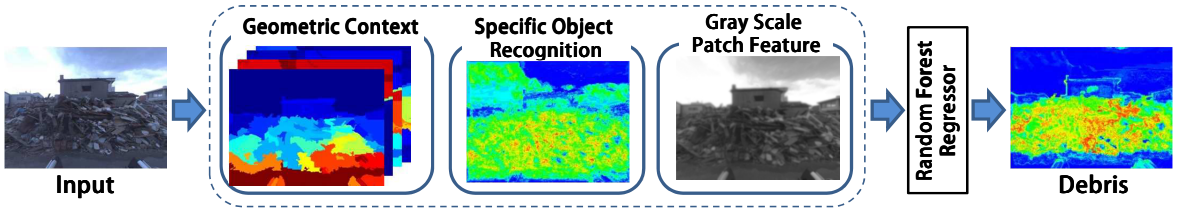


Figure 6.4: Data flow diagram of debris detection. As features of debris, the probabilities of geometric context, specific object recognition and patch features are employed.

Table 6.1: Feature importance of debris detector using random forest.

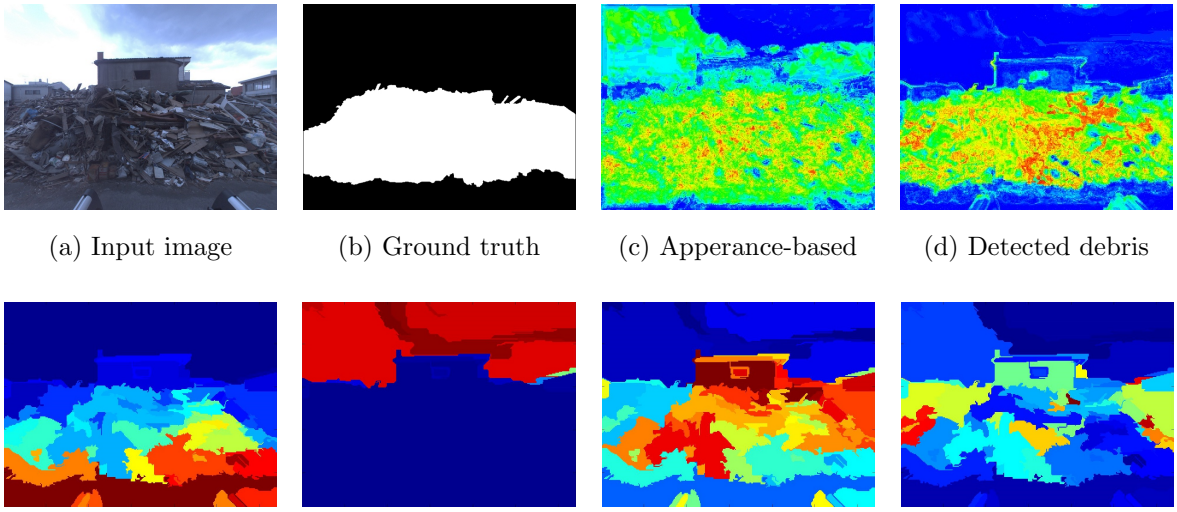
Geometric Context				Apperance-based	Local-patch	
Ground	Sky	Porous	Solid		Average	Variance
0.046	0.070	0.032	0.032	0.338	0.201	0.280
0.180				0.338	**0.481**	

We evaluated the accuracy of our debris detector. We made two datasets for the evaluation. Figure 6.5 shows an example of the datasets and detection results. Each dataset consists of fifty images of debris. The images in two data set were taken in different date and time. We compared random forest [80], logistic regression [81] and support vector machine [82]. Figure 6.6 shows the F_1-scores of the debris detections. We chose the random forest as our debris detector for all experiments because the score of random forest regressor is the best.

Furthermore, we evaluated the feature importance of the debris detector. Table 6.1 shows the evaluation result of the feature importances of the debris detector. This result indicates that average and variance of local-patch intensities are discriminative features for debris. This result can help to define what debris is.

6.2.2 Projection of debris probabilities onto the ground

The debris probability explained in the previous section is the probability map on the street-view image. In order to integrate this probability map with the aerial image, the debris probability is projected onto the ground plane. Figure 6.7 shows the data flow diagram of projection of street-view image to the coordinate of the aerial image. The projection requires camera parameters of each street-view image. First, we performed Structure from Motion (SfM) to acquire the camera trajectories. We employ a standard



(a) Input image (b) Ground truth (c) Apperance-based (d) Detected debris

(e) Ground (f) Sky (g) Solid (h) Porous

Figure 6.5: Inputs and outputs of debris detection. First rows: (a) input image. (b) hand-labeled ground truth of debris. (c) result of specific object recognition. (d) final result of debris detection. Second rows: probability of geometric context (e) ground plane, (f) sky, (g) solid non-planar, (h) porous non-planar. Color denotes probability of each class, with blue corresponding to 0 and red to 1.

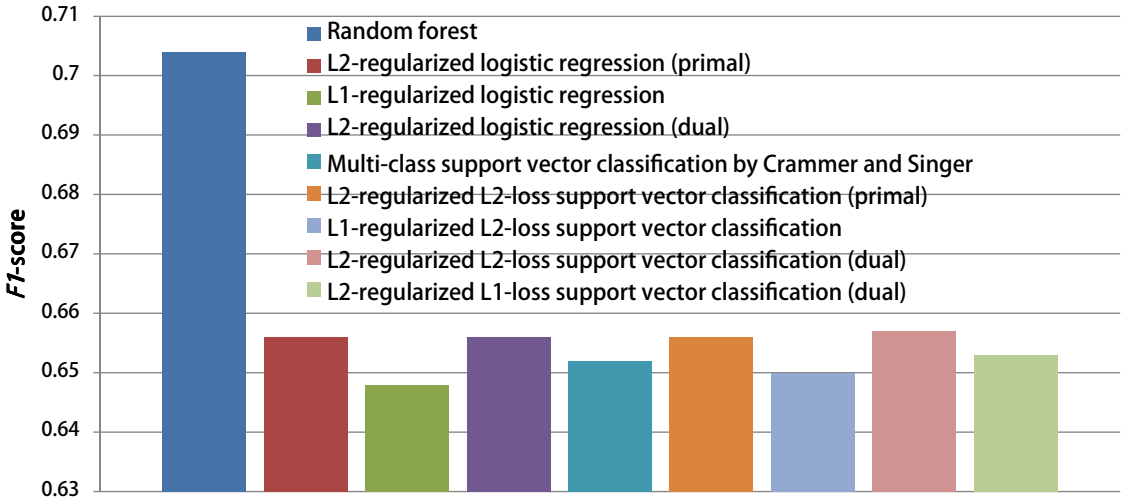


Figure 6.6: F_1-score of debris detection.

SfM method [38, 61, 71] with extensions to deal with omni-directional images [31]. The estimated camera trajectories are fitted to the GPS trajectory by similarity transformations in a least squares sense.

Dividing the ground plane into a grid, we project the debris probability to the grid

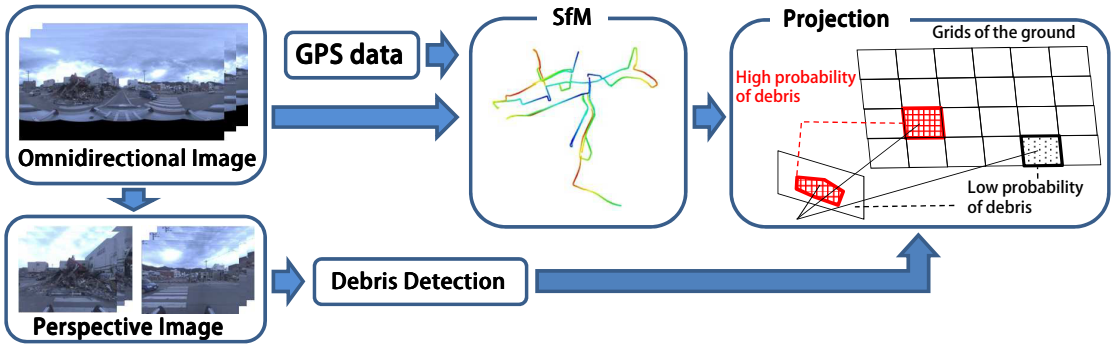
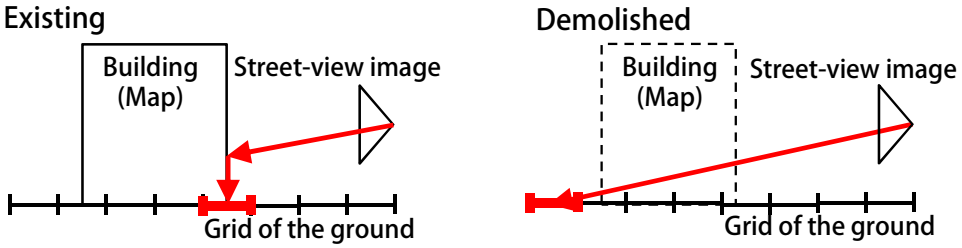


Figure 6.7: Data flow diagram of the projection onto the ground plane. SfM is performed using omnidirectional street-view images. The street-view camera poses are registered to a common coordinate with aerial images and other forms of meta-data using the GPS data. After debris detection, the debris probabilities are projected to the ground plane.

using projection matrix of each image. In this projection, we use the 3D models of the buildings that are generated from a 2D map of the city (Sec. 6.3.2). To be specific, the debris probability is projected to a building wall if the wall is on the projection path, and otherwise it is directly projected to the ground, as shown in Fig. 6.8.



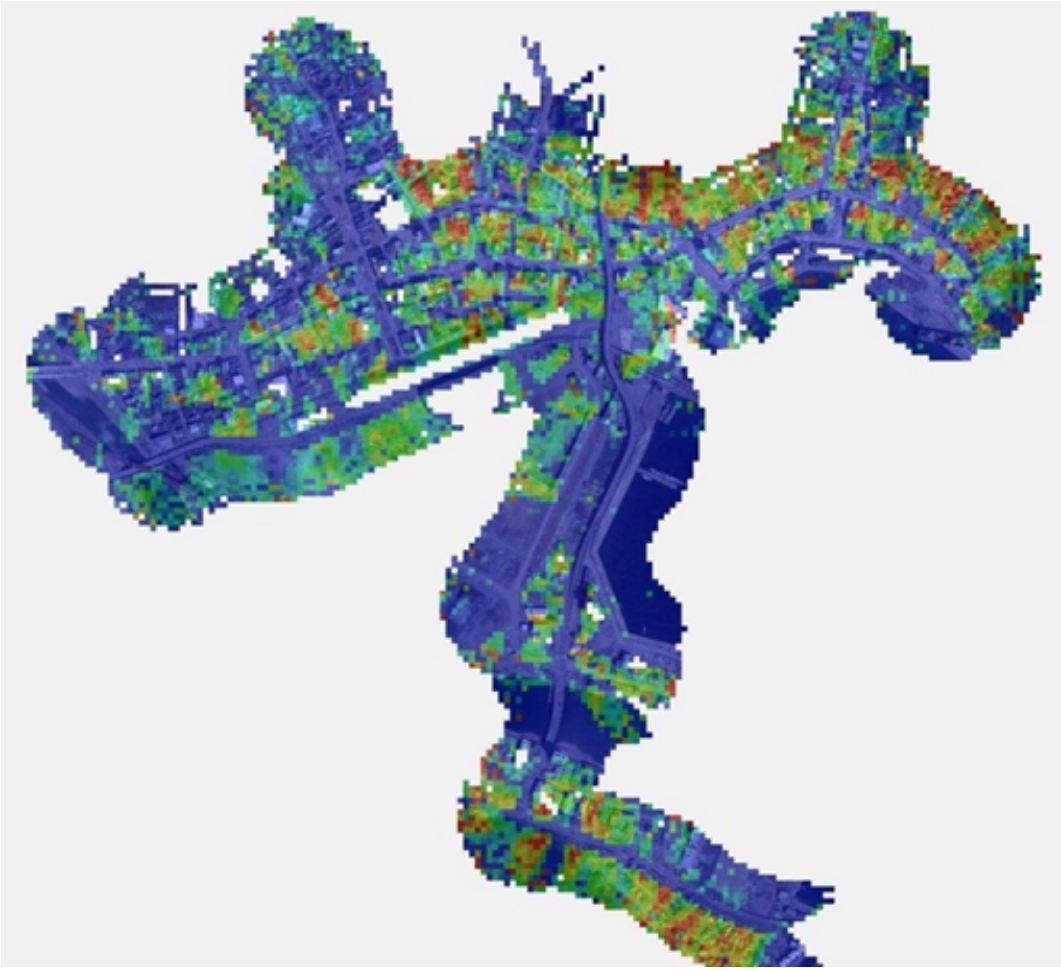


Figure 6.8: Projection of probabilities on street-view images to the grids of the ground plane using building information. Top : The probabilities on a street-view image are projected to a building wall if the building is on the projection path, otherwise it is projected to the ground directly. Bottom : Example of projection results (top-view). The area unobserved from street-view images is shown in white.

6.2.3 Integration using Gaussian Process regression

The projected debris probability map obtained up to now has no information for some areas because of occlusions or the lack of street-level images, as shown in Fig. 6.8. Estimating debris probability map from only an aerial image is difficult due to its low-resolution, occlusion or weather conditions. To mutually complement the street-view images and the aerial image, we used a Gaussian process regression model[83]. The main idea here is that similar geographical location tend to have similar debris probability. In the case of Tsunami-disaster, Tsunami continuously spreads from seashore to hill side, which means the damage caused by Tsunami has strong correlation with the location, especially with the elevation.

As described in the previous section, the debris probability of each grid $p_{\mathrm{s},i}$ $(i = 1, ..., n)$ is estimated from the street-view images. For each grid, its feature vector $\mathbf{x}_i$ is defined as follows.

$$\mathbf{x}_i = (x_i, y_i, z_i, p_{\mathrm{a},i})^{\mathrm{T}} \tag{6.2}$$

where (x_i, y_i) is a center position of each grid, z_i is a elevation of each grid calculated from DEM and $p_{\mathrm{a},i}$ is debris probability of each grid estimated from aerial image using pixel-wise object recognition[76]. The column vector $\mathbf{x}_i$ for all n grid are aggregated in the $4 \times n$ training inputs matrix X, and the training outputs $p_{\mathrm{s},i}$ are collected in the vector $\mathbf{y}$.

$\mathbf{x}_i$ contains $p_{\mathrm{a},i}$ as the visual feature of the ith grid. Although $p_{\mathrm{a},i}$ is a scalar, due to the pixel-wise object recognition[76], it summarizes the visual information of the ith grid. Compared to using general visual feature descriptors, such as SIFT[61], directly in the feature vector $\mathbf{x}_i$, $p_{\mathrm{a},i}$ saves computational resources required in the following calculation of covariance function. The covariance function for Gaussian process regression in the proposed method is as follows.

$$k(\mathbf{x}_p, \mathbf{x}_q) = {\sigma_f}^2 \exp\left(-\frac{1}{2l^2}|W\left(\mathbf{x}_p - \mathbf{x}_q\right)|^2\right) + {\sigma_n}^2 \delta_{pq} \tag{6.3}$$

where W is the weight diagonal matrix, l is the length-scale, ${\sigma_f}^2$ is the signal variance, ${\sigma_n}^2$ is the noise variance and δ_{pq} is a Kronecker delta which is one if $p = q$ and zero otherwise. Test input $\mathbf{x}_\star$ is each grid feature vector and test output is debris probability of each grid $\overline{f}_\star$.

The relationship between test input $\mathbf{x}_*$ (feature vector of each grid) and test output $\overline{f}_*$ (debris probability of each grid) is as follows [83]

$$\bar{f}_* \;=\; {\mathbf{k}_*}^{\mathrm{T}}(K + {\sigma_n}^2 I)^{-1}\mathbf{y}, \tag{6.4}$$

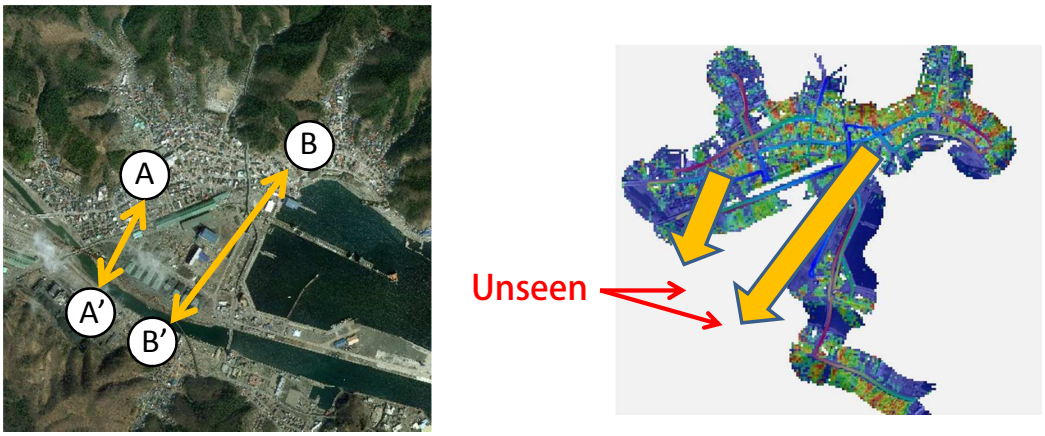


Figure 6.9: Integration using visual similarity. Resolution of aerial image is not enough to detect debris. However, it is possible to estimate whether the area conditions are similar or not. For example, intuitively, if A and B are in debris conditions similar to A' and B', respectively, the debris probabilities of A' and B' can be estimated from the observed areas A and B.

where K is covariance matrix, $k(\mathbf{x}_p, \mathbf{x}_q)$ is the element of K in row p, column q, $\mathbf{k}(\mathbf{x}_*) = \mathbf{k}_*$ is the vector of covariances between the test point and the training points, the vector $\mathbf{y}$ contains the training outputs $p_{\mathrm{s},i}$, I is identity matrix. We solve Eq.(6.4) using cholesky decomposition.

The key insight to note here is that the output of the aerial image regressor $p_{\mathrm{a},i}$ enforces a correlation between parts of the scene that look similar. If two parts of the scene belong to an open field, the per-pixel response of the aerial object detection regressor will produce a similar response. The DEM also works in a similar manner to draw correlations between regions with similar elevation. The location feature enforces local smoothness over the final estimate of debris over the city. When the feature vectors $\mathbf{x}_i$ are used to compute the covariance function, regions that are similar in appearance and elevation will be constrained to have similar target values (debris estimates generated by high resolution debris regressor computed on the street images). In this way, the Gaussian process regression model is able to propagate local estimates of debris to the entire map. This regression mechanism is what allows our model to effectively estimate debris over the entire city from only a sparse set of street view debris estimation results.

6.3 Experimental results

In order to evaluate the effectiveness of our proposed approach for estimating large-scale land conditions, we perform two experiments. Our first experiment is a comprehensive ablative analysis to examine the benefit of integrating micro and macro-level imagery for city-scale land condition estimation. In addition to color imaging, we also evaluate the contributions of two other modes of data, namely, a digital elevation map (DEM) and building occupancy maps (BOM). In our second experiment, we focus on estimating the amount of greenery and vegetation across the entire city of Kamaishi. We use the exact same approach as the debris estimation described in this paper and apply it to greenery estimation. Our results show that our approach is not limited to post-disaster analysis but can easily be applied to other modes of land condition analysis.

We created the ground truth labels used for the following evaluation by many hours of manual labeling of regions on the aerial images. Ground truth data of debris and greenery were generated by visual inspection by comparing the aerial image against the street-view images available on Google Earth. Many hours of ground truth labeling confirms that the manual inspection of large-scale land conditions is not a practical solution for real-world applications.

6.3.1 Our data

Our experiment includes two image-based input modalities and two sources of city-scale meta-data, which are described below.

Street images. We have been creating image archives of urban and residential areas damaged by Great East Japan Earthquake in 2011.The target area is 500 kilometers long along the northern-east coastal line in Japan. The images were captured every three to four months by a vehicle having an omni-directional camera (Ladybug 3 and 5 of Point Grey Research Inc.) on its roof. The image data accumulated so far amount to about 20 terabytes. The target of this experiment is the entire city of Kamaishi, Japan (over 4 million square-meters). For the experiments, we chose the two image sequences captured on April 26th, 2011 (one month after the Tsunami) and August 17th, 2013 (two years and five months after the Tsunami). The debris can often be seen in the earlier images, while they tend to disappear in the later images as the recovery operation proceeds.

The street images are used for appearance-based recognition of 'stuff' [84] described in Section 6.2.1. The results of pixel-wise regression are then projected onto the ground plane as an input feature for our city-scale GP regressor.

Aerial images We downloaded aerial images from Google Earth for March 31st, 2011 and May 13th, 2012. We chose these dates to match up the timestamp of the street

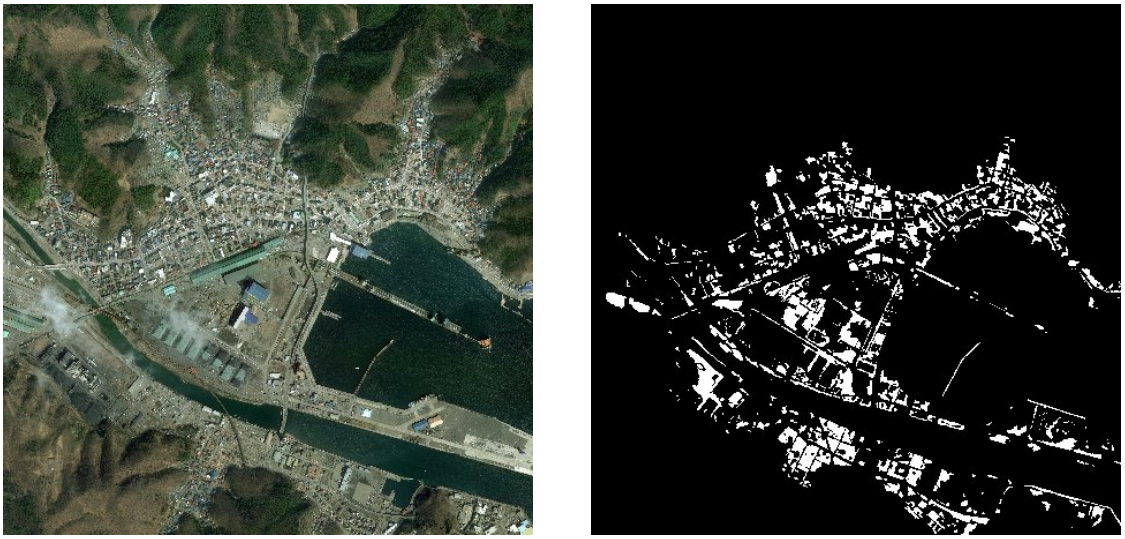


Figure 6.10: Estimation target area in Kamaishi on March 31st, 2011 (left) and its hand-labeled ground truth of debris area (right). White area shows debris area.

images.

We used the aerial images for appearance-based recognition of 'stuff' categories using the same method describe in Section 6.2.1 but applied to the entire aerial image as a comparative baseline. We used the aerial images of May 13th, 2012 as the labeled training data and test on the March 31st, 2011 aerial image. Figure 6.10 shows an example of the hand-labeled ground truth of the debris area on the aerial images.

Digital Elevation Map (DEM). We obtained the DEM information freely available from the Geospatial Information Authority, under the Ministry of Land, Infrastructure, Transportation and Tourism in Japan. The mesh resolution of the DEM is 5×5 square-meters and contains the elevation level for each grid location. The elevation is used directly as a feature for the city-scale GP regression.

Building Occupancy Map (BOM) The BOM provides building contours. We obtained the data from Zenrin Company. The building contour data used for this experiment was made before the earthquake. We used the BOM to prevent 'stuff' from being projected onto the ground over building location.

6.3.2 Ablative Analysis

We examine the effects of each input data type on the overall performance of our proposed approach. Figure 6.21 shows the estimation results of the debris amounts in the entire city on April 26th, 2011 and August 17th, 2013, respectively. The lines on the aerial images are the camera trajectories. Figure 6.13 shows the performance of our debris detection by PR-plot and $F1$-score using different combination of input data. The results indicate that using aerial images alone yields low performance because the appearance of land conditions can change significantly over time due to changes in imaging conditions. When compared to the independent use of aerial images, our results indicate that street images are more accurate for estimating city-scale debris. Furthermore, when both aerial and street images are combined we obtain better performance as the aerial information helps the city-scale GP regression to generalize to across similar looking city regions.

Additionally, we evaluated the effects of each input data type and different number of street-view images in three different streets. Figures 6.14, 6.15 and 6.16 show (a) input aerial image, (b) the hand-labeled ground truth which we made from aerial and street-view images, (c) projection results of debris probabilities of street-view images, (d) final debris estimation results integrating both street-view and aerial images along with digital elevation map (DEM) and building occupancy map (BOM) of each street.

Figure 6.17, 6.18 and 6.19 show precision-recall curve (Left) and $F1$-scores (Right, recall=0.5) of the debris area detection. We examined the detection performance for a specific area. The effect of different input type is different depending on the area-condition. For example, aerial image could cause errors in occluded areas we mentioned in the introduction, and DEM information could cause errors because its elevation includes height of building. However, street-view images basically provide detailed and accurate information of land-surface condition.

We also tested the effect of street coverage. Figure 6.20 shows the $F1$-score of different number of street-view images. In this experiment, we randomly sampled street-view images. The accuracy improves as we add more images, but it quickly saturates. This indicates our algorithm needs only sparse street-view images. The sparse sampling requirement of our algorithm is beneficial in many other applications, for example, large scale citizen science or journalism in which images captured at the scene are sent to cloud computers to analyze city scale condition.

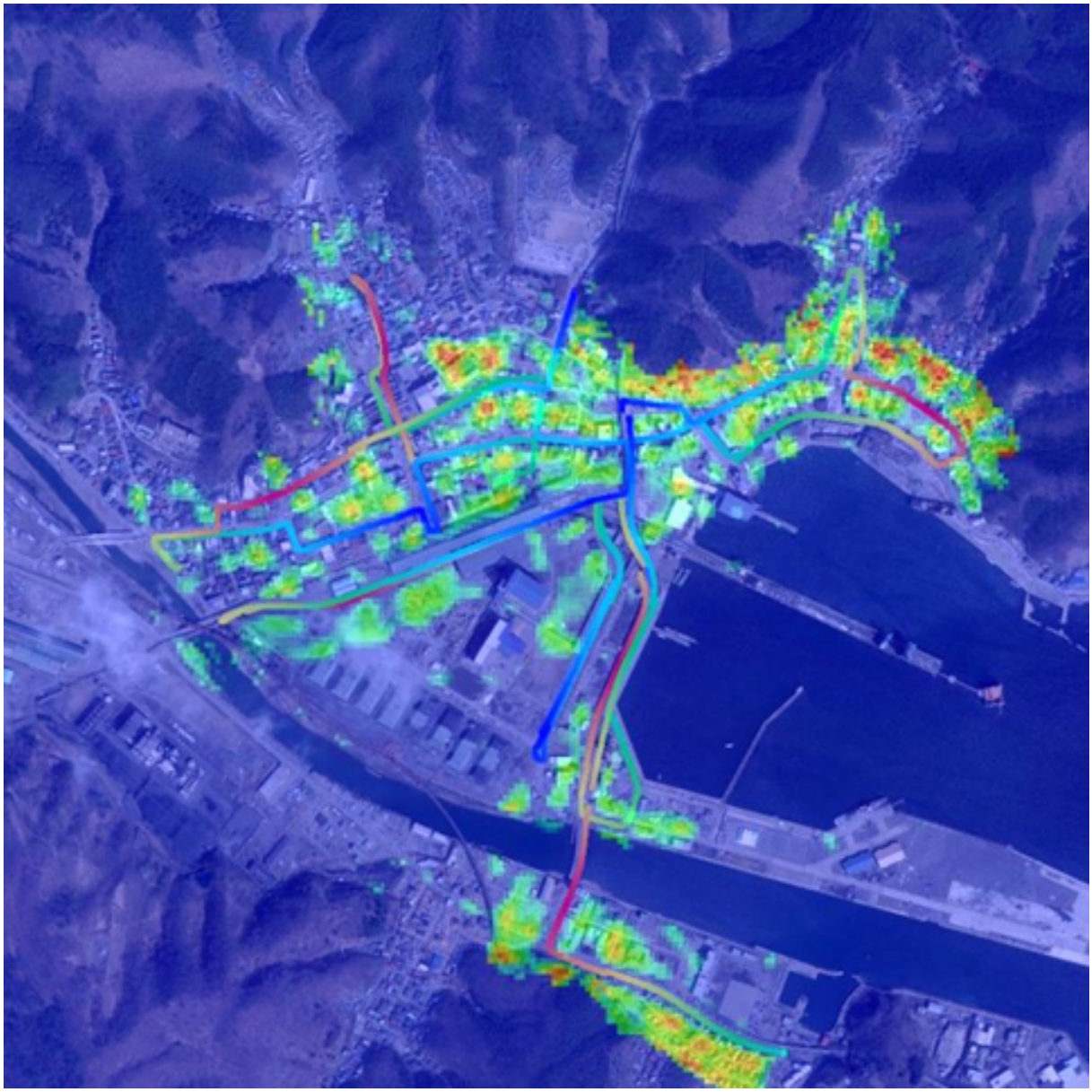


Figure 6.11: City-scale **Debris** Probability in Kamaishi before the recovery operation (April 26th, 2011). Color denotes probability of debris, with blue corresponding to 0 and red to 1.

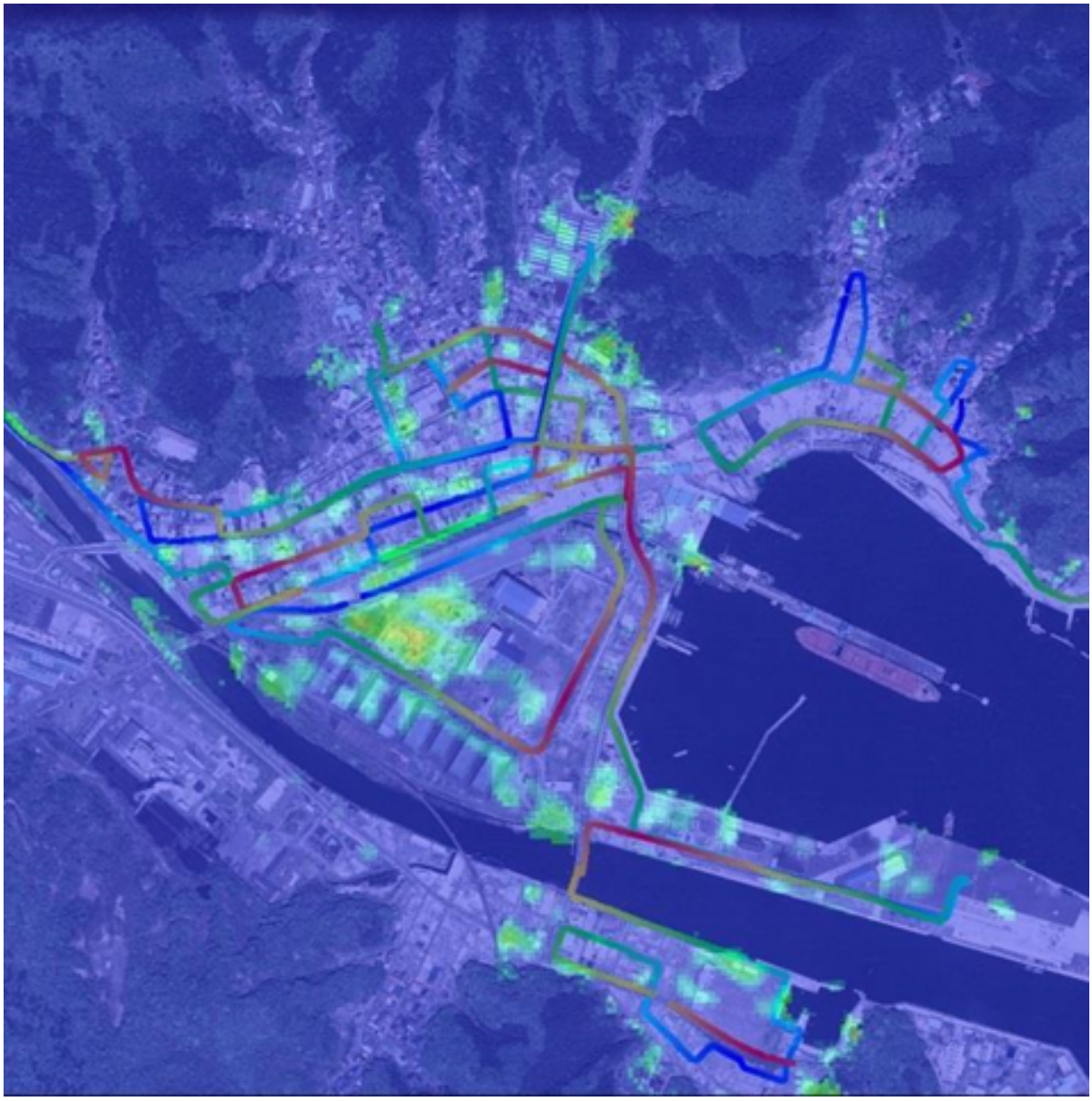


Figure 6.12: City-scale **Debris** Probability in Kamaishi after the recovery operation (August 17th, 2013). Color denotes probability of debris, with blue corresponding to 0 and red to 1.

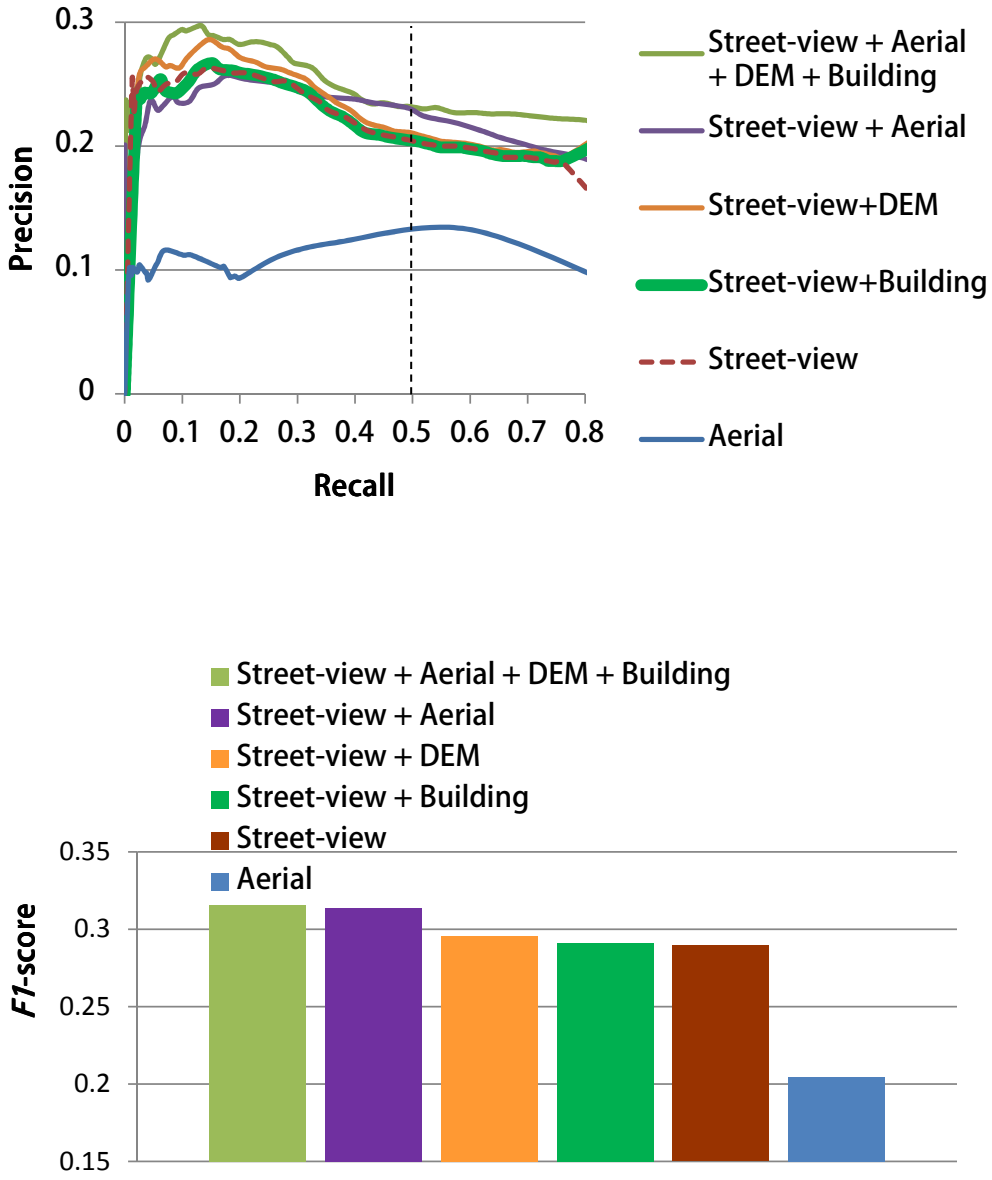
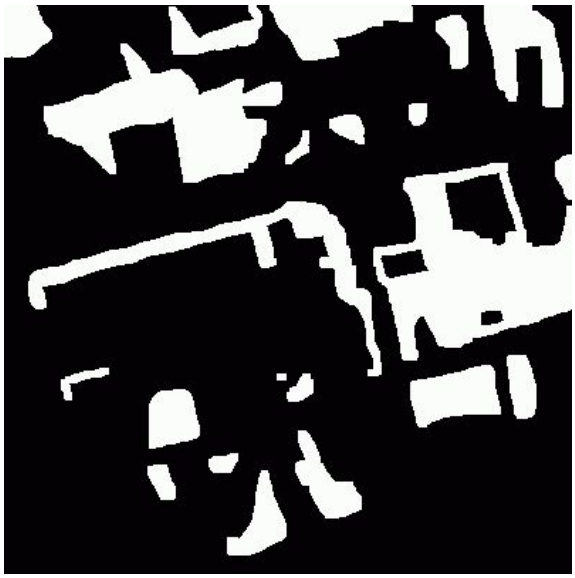
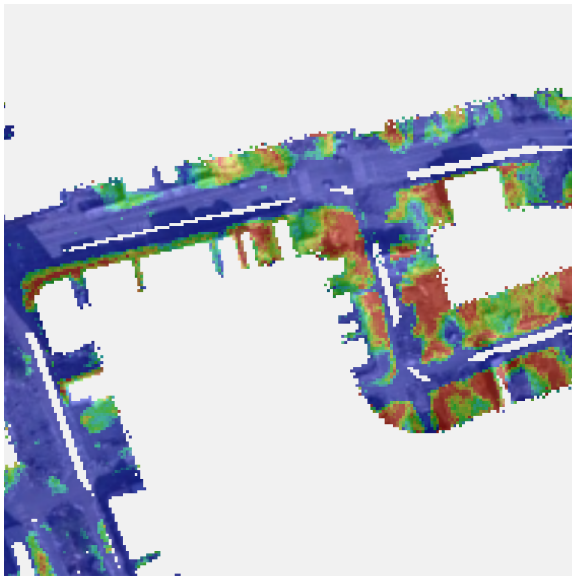


Figure 6.13: Precision-recall curve of the debris area detection whose ground truth is Fig. 6.10. These figures show that the integration of street-view image with aerial image is efficient to estimate city-scale land surface condition.

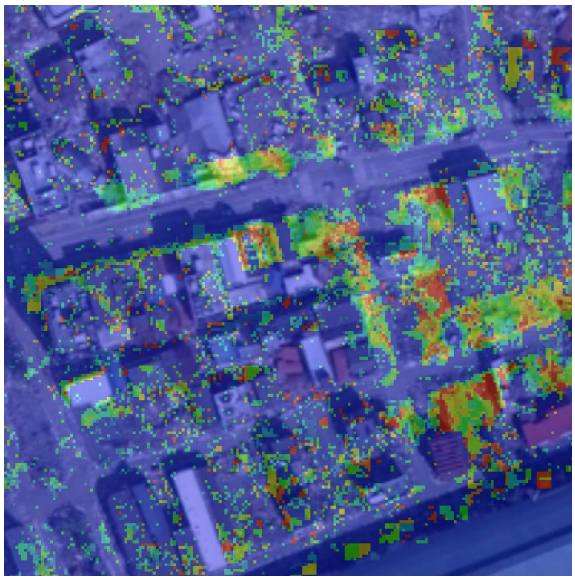
(a) Input aerial image



(b) Hand-labeled ground truth



(c) Projection using street-view images

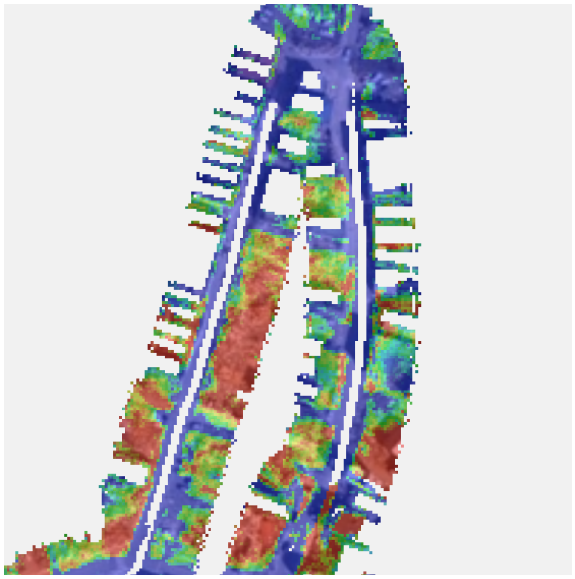


(d) Final estimation result

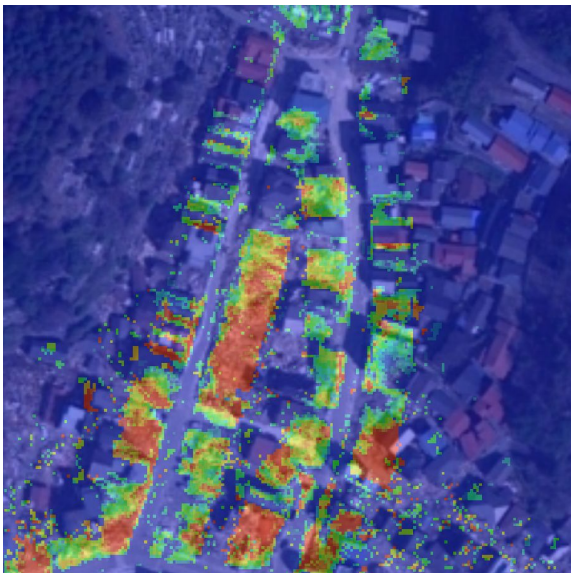
Figure 6.14: Debris probability in area 1 (200×200 m^2, grid size: 1 m). (a) Input aerial image. (b) Hand-labeled ground truth. (c) Projection result using street-view images (top-view). The area unobserved from street-view images is shown in white. (d) Final debris estimation result integrating both street-view and aerial images along with DEM and BOM.

(a) Input aerial image

(b) Hand-labeled ground truth



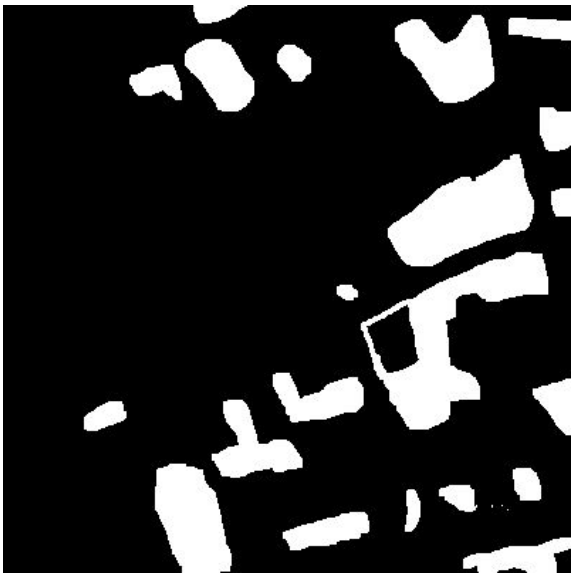
(c) Projection using street-view images



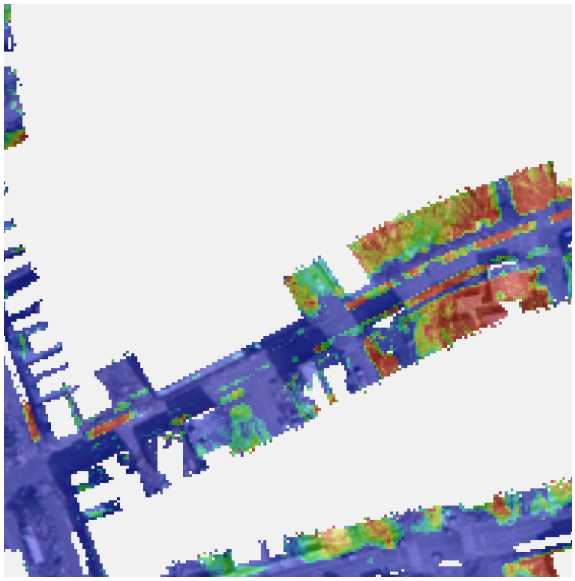
(d) Final estimation result

Figure 6.15: Debris probability in area 2 (200×200 m^2, grid size: 1 m). (a) Input aerial image. (b) Hand-labeled ground truth. (c) Projection result using street-view images (top-view). The area unobserved from street-view images is shown in white. (d) Final debris estimation result integrating both street-view and aerial images along with DEM and BOM.

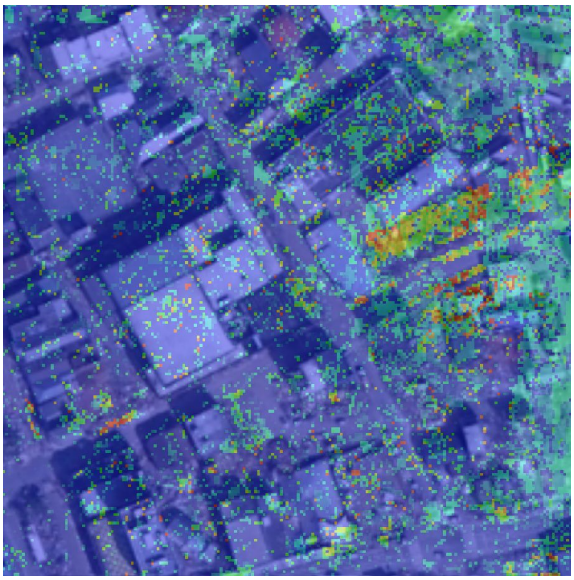
(a) Input aerial image



(b) Hand-labeled ground truth

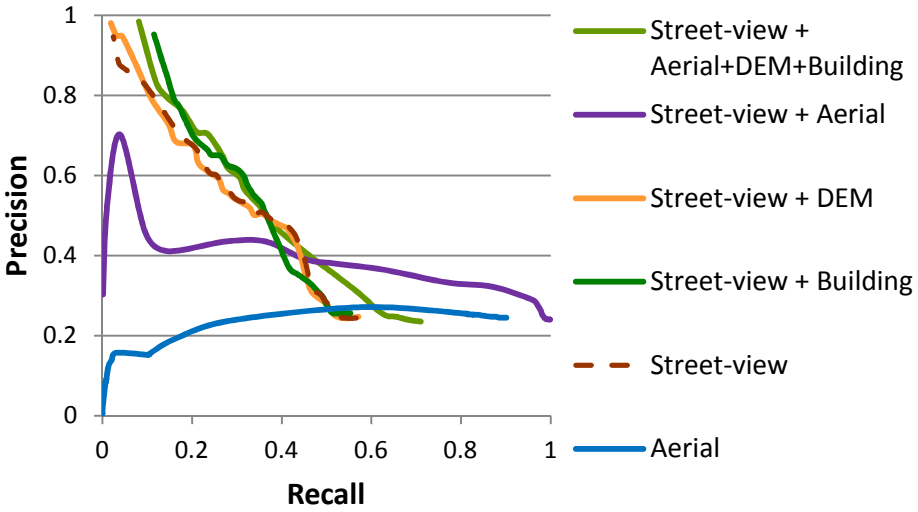


(c) Projection using street-view images



(d) Final estimation result

Figure 6.16: Debris probability in area 3 (200×200 m^2, grid size: 1 m). (a) Input aerial image. (b) Hand-labeled ground truth. (c) Projection result using street-view images (top-view). The area unobserved from street-view images is shown in white. (d) Final debris estimation result integrating both street-view and aerial images along with DEM and BOM.



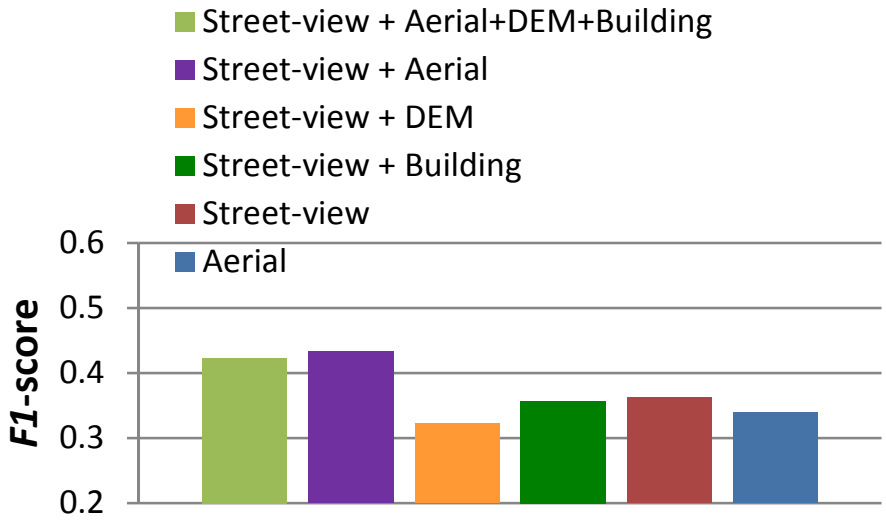


Figure 6.17: Area1: Precision-recall curve (Top) and $F1$-scores (Bottom, recall=0.5) of the debris-area detection. These plots show that the integration of street-view image with aerial image is effective to estimate the condition of land surface.

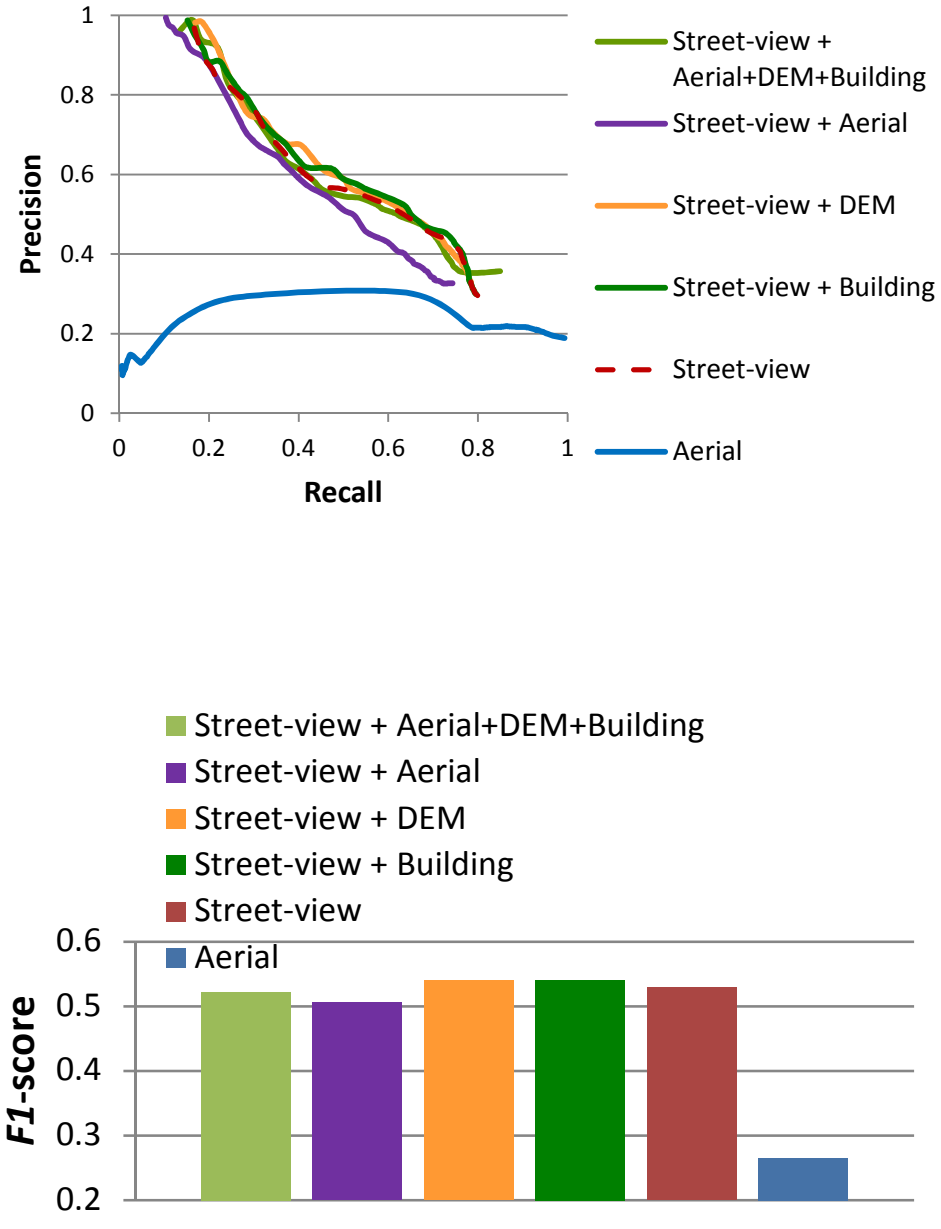


Figure 6.18: Area2: Precision-recall curve (Top) and $F1$-scores (Bottom, recall=0.5) of the debris-area detection. These plots show that the integration of street-view image with aerial image is effective to estimate the condition of land surface.

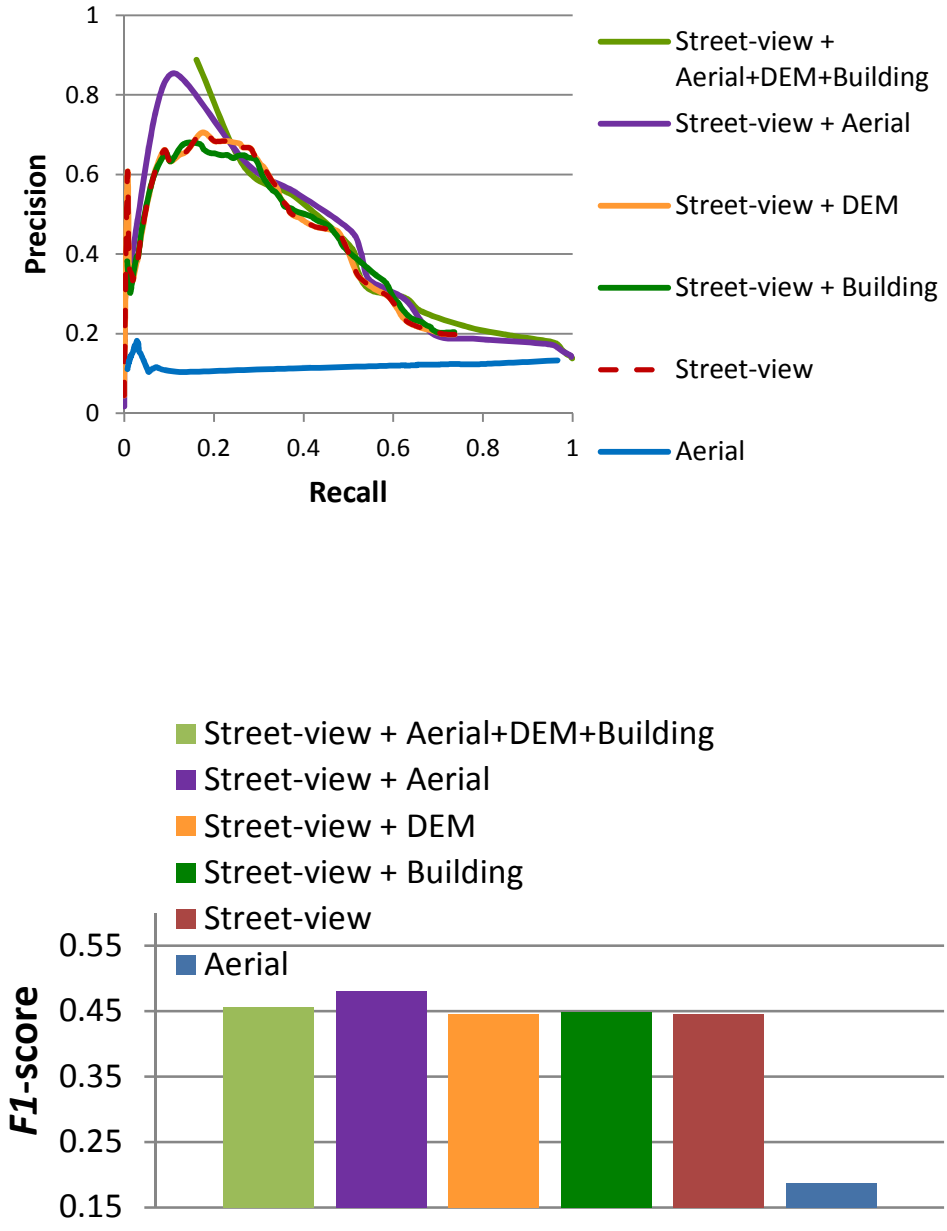
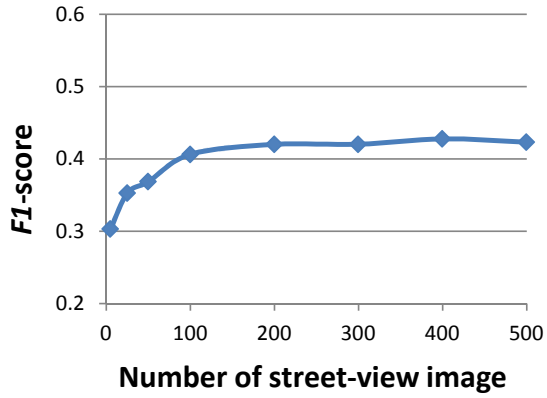
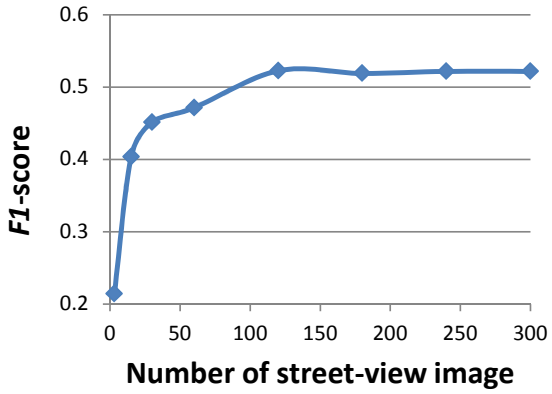


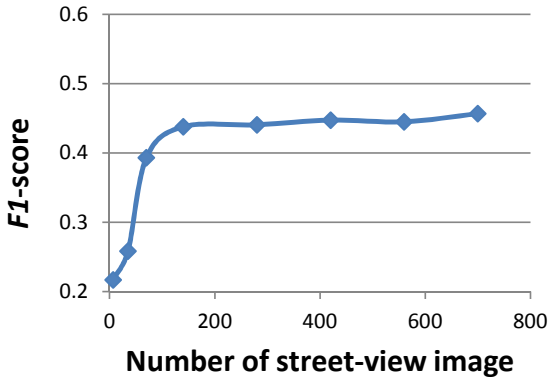
Figure 6.19: Area3: Precision-recall curve (Top) and $F1$-scores (Bottom, recall=0.5) of the debris-area detection. These plots show that the integration of street-view image with aerial image is effective to estimate the condition of land surface.



(a) Area 1



(b) Area 2



(c) Area 3

Figure 6.20: $F1$-scores of different number of street-view images. Our algorithm does not require large number of images.

6.3.3 Qualitative Results for City-scale Debris Estimation

Figure 6.21 shows the estimation results of the debris amounts in the entire city on April 26th, 2011 and August 17th, 2013, respectively. The lines on the aerial images are the camera trajectories. The locations A-E correspond to A'-E', respectively. The heat map color shows probability of debris (red means more likely debris and blue less likely debris).

As mentioned in introduction of the main paper, there are certain types of debris which cannot be observed using aerial image. The locations A and B in Fig.6.21 show areas where debris is covered by the roof of the building and are not observed from the aerial image. However, since our method integrates both street and aerial images this region has been estimated as a high debris area (red and yellow green). In the same region observed on August 17th, 2013 (A' and B'), all the debris (including the roof structure!) has been removed.

At location C, there are stacked mounds of debris and the area is estimated as a moderate debris area (yellow) but is completely restored C' by 2013.

Location D is estimated as having a moderate level of debris (yellow) which is confirmed by the street image. By 2013 region D' has been restored. The debris, building and car have been removed and a field of weeds has grown there.

Location E shows a failure case of our approach, where this type of debris was not encountered in our debris training data and therefore not detected in the street image. The post-restoration image shows that the area is now under construction.

6.3.4 Extensions to City-Scale Vegetation Estimation

We applied our method to vegetation detection, to show how our approach can generalize to other modes of land condition estimation. Figure 6.22 shows an example of vegetation estimation in street-level images. The green vegetation detected in the street-view images is estimated using the same pixel-wise object recognition method [76].

Figure 6.25 shows the results of vegetation estimation for the entire city similar to Fig.6.21. (The location A-E, A'-E' in Fig.6.25 correspond to A-E, A'-E' in Fig.6.21, respectively.) By observing the vegetation heat map for the entire city, it is clear that most of the vegetation has been washed away by the Tsunami. There is also a sharp contrast between the wide spread distribution of debris and the lack of vegetation in the time period directly after the Tsunami. By 2013 however, we can see a large increase in the number of regions covered by vegetation. Our successful vegetation detection indicates that our proposed method can indeed generalize to different types of targeted estimation of city-scale land conditions.

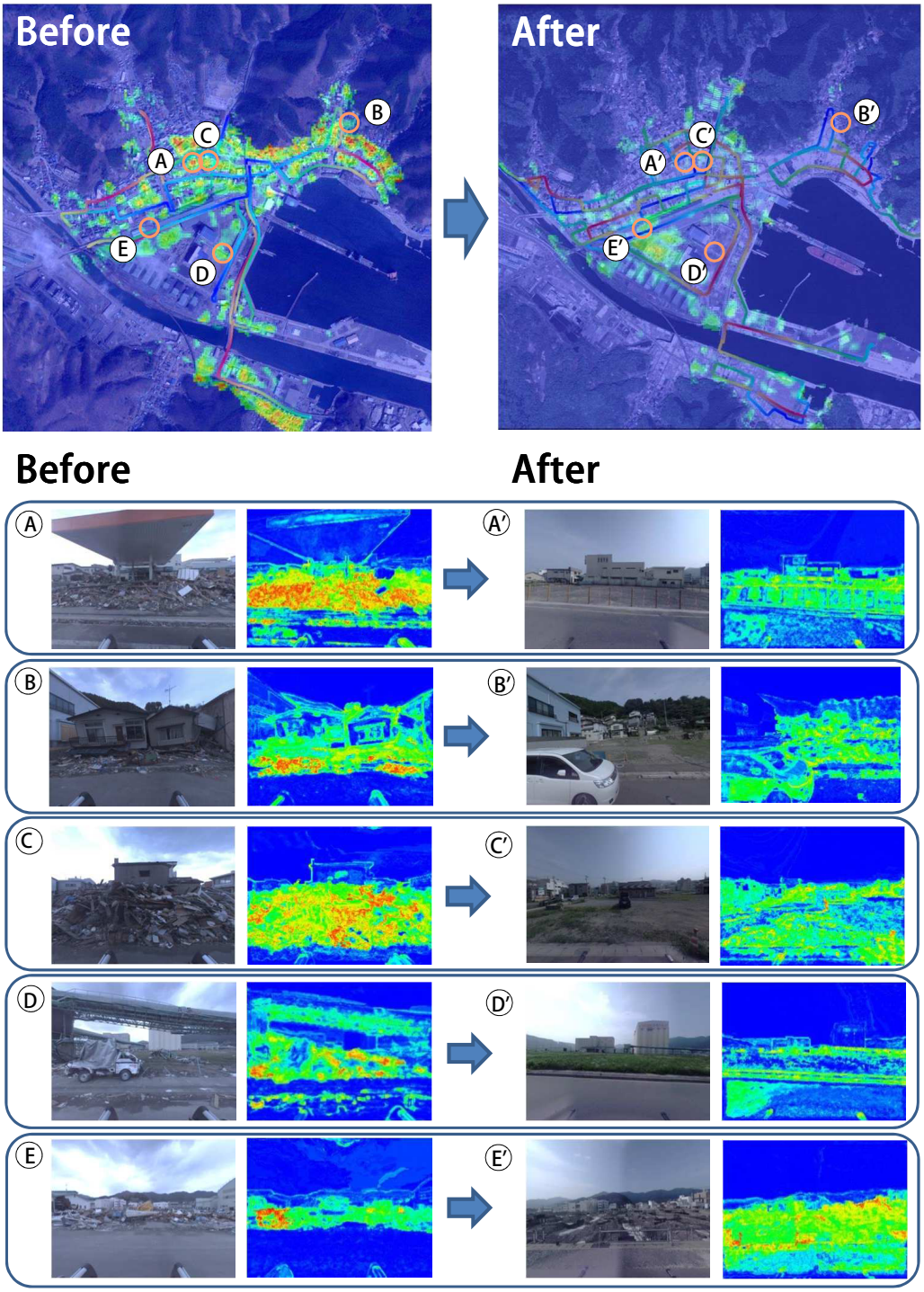


Figure 6.21: City-scale **Debris** probability in Kamaishi before and after the recovery operation (Left: April 26th, 2011, Right: August 17th, 2013). In the earlier images, there are much debris in the entire city, however, most of them have been cleaned up in later images. The city-scale temporal change is estimated and visualized accurately by our approach. Color denotes probability of debris, with blue corresponding to 0 and red to 1.

(a) Input image (b) Ground truth (c) Detected Vegetation

Figure 6.22: Green vegetation detection. (a) input image. (b) hand-labeled ground truth of green vegetation. (c) probability of green vegetation. Color denotes probability of green vegetation, with blue corresponding to 0 and red to 1.

Figure 6.23: City-scale **Vegetation** Probability in Kamaishi before the recovery operation (April 26th, 2011). Color denotes probability of green vegetation, with blue corresponding to 0 and red to 1.

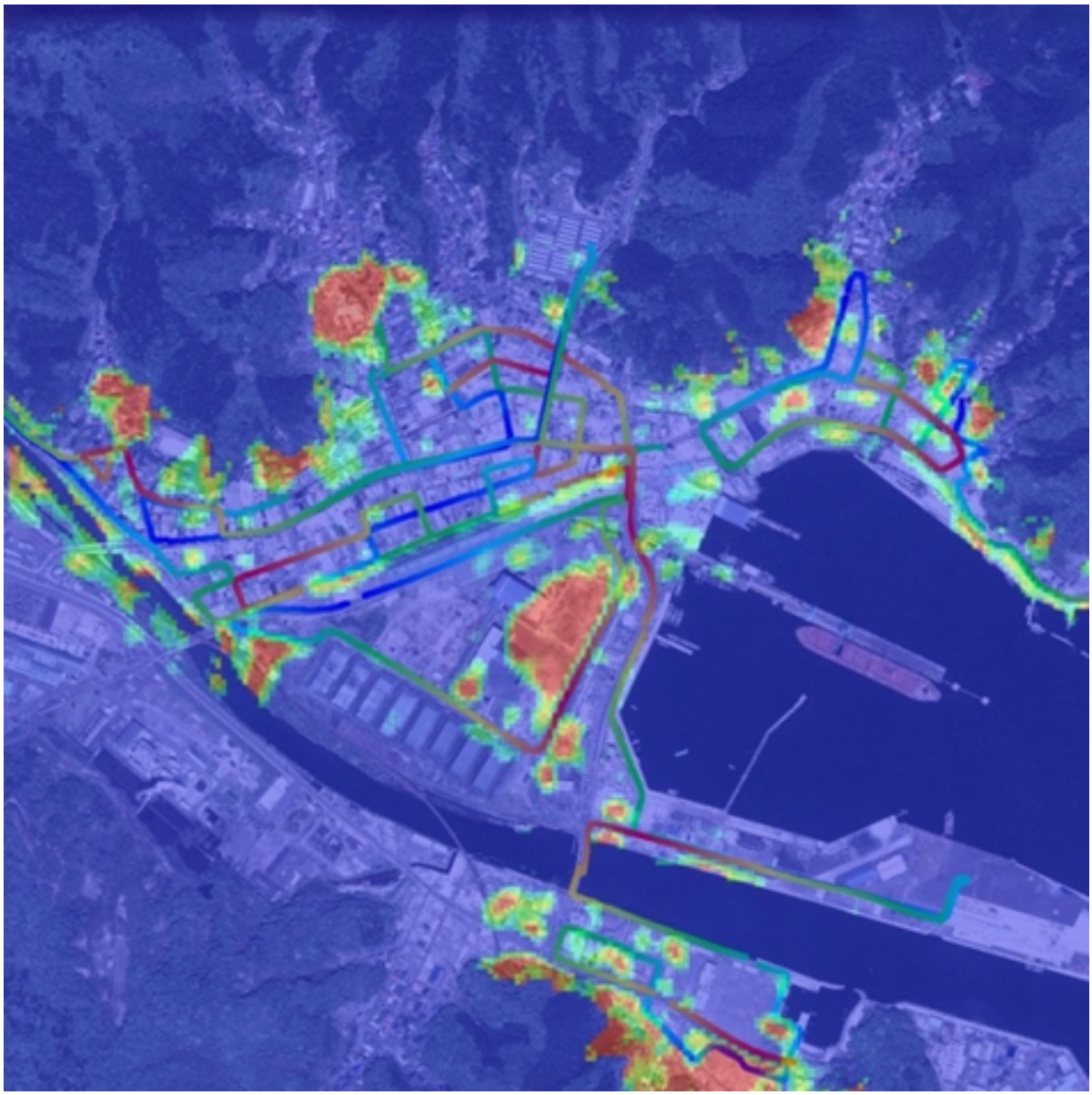


Figure 6.24: City-scale **Vegetation** Probability in Kamaishi after the recovery operation (August 17th, 2013). Color denotes probability of green vegetation, with blue corresponding to 0 and red to 1.

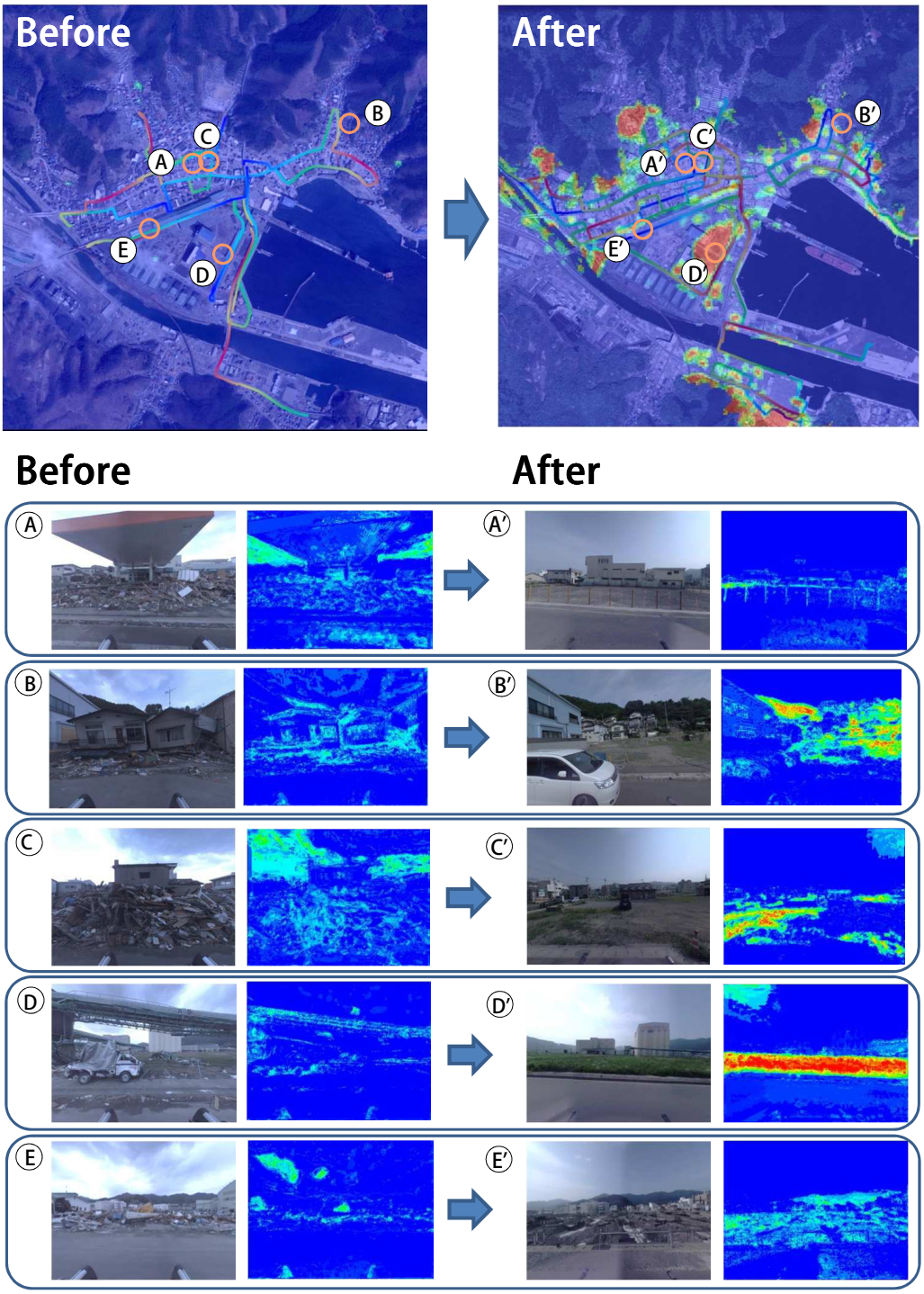


Figure 6.25: City-Scale **Vegetation** probability in Kamaishi before and after the recovery operation (Left: April 26, 2011, Right: August 17, 2013). In contradiction to the debris (Fig. 6.21), there was no green vegetation due to Tsunami-damage in April, 2011, however, the vegetation in entire city has grown and recovered until August, 2013. Our approach can estimate and visualize such changes in natural phenomena over massive city-scale landscapes. Color denotes probability of green vegetation, with blue corresponding to 0 and red to 1

6.4 Summary

We presented a unified framework for robustly integrating image data taken at vastly different viewpoints to generate large-scale estimates of land surface conditions. The proposed strategy uses macro-level imaging to learn land condition correspondences between land regions that share similar visual characteristics, while micro-level images are used to acquire high resolution statistics of land conditions. For the validation of our approach, we conducted experiments to estimate the amount of post-Tsunami damage over the entire city of Kamaishi, Japan. The experimental results show that our approach can effectively integrate both macro (aerial) and micro-level (street view) images, along with other forms of meta-data, to estimate city-scale phenomena.

Furthermore, we showed that our detection method can be successfully applied to vegetation estimation. The results indicate our method can generalize well to many kinds of applications to estimate city-scale phenomena by replacing the detector target, for example, human flow, real-estate and dirt quality. These types of image data are available from many kinds of data sources, such as camera equipped mobile devices, surveillance cameras and car-mounted video recorders, or aerial-vehicle-mounted cameras. Our approach provides an effective and robust method for integrating different kinds of data to estimate city-scale phenomena.

For future work, we plan to improve the estimation accuracy of our approach. Our method has relatively low absolute precision because (i) the grid size is too large due to limitation of computational resources and (ii) the estimated camera poses have errors due to GPS errors. We believe that we can solve the first problem using large-scale Gaussian process [83]. The GPS issue can be addressed with [35, 36, 85] while taking temporal changes into account. Furthermore, in the case of extreme calamities, methods will be developed to take into consideration the complete disappearance of the buildings due to disasters.

Chapter 7

Conclusion

This dissertation proposed the novel and practical methods for four-dimensional city modeling using vehicular imagery. The estimation target is the tsunami-damaged areas across the three prefectures whose total length is almost 400 kilometers. To estimate and visualize temporal change of the the areas, the image archive activity started one month after Great East Japan Earthquake which caused the giant Tsunami. The Tsunami gave serious damages to the Pacific coast area of the Tohoku. The images periodically recorded the the scenes of the tsunami-damaged areas.

From the periodic images, this dissertation visualized the tsunami-damage and recovery of the tsunami-damaged area. First, the 2D change detection method using grid feature roughly but quickly estimates scene change of entire areas. Next, the structural change detection method estimates more accurate scene change even if there is ambiguity in estimated scene depth. Finally, the method of land surface condition analysis estimates city-scale temporal change integrating aerial and vehicular imagery.

The 2D method detects scene change using grid feature from an image pair without 3D model and pixel-level registration. The experimental results show the effectiveness of the proposed method integrating high discrimination of convolutional neural network (CNN) feature with accurate segmentation of superpixel in 2D change detection. As a by-product, the method can reduce the computational time.

The structural change detection method detects temporal changes of the three dimensional structure of an outdoor scene from its multi-view images captured at two separate times. The method estimates scene structures probabilistically, not deterministically to maximize the accuracy of change detection. The proposed method is compared with the methods that use multi-view stereo (MVS) to reconstruct the scene structures of the two time points and then differentiate them to detect changes. The experimental results show that the proposed method outperforms such MVS-based methods. Unlike MVS-based methods, the proposed method can estimate accurate shape of the scene change

(e.g. debris) because the proposed method utilizes no prior on the smoothness of scene structure.

The method of land surface condition analysis is a unified framework for robustly integrating image data taken at vastly different viewpoints to generate large-scale estimates of land surface conditions. The method uses macro-level imaging to learn land condition correspondences between land regions that share similar visual characteristics, while micro-level images are used to acquire high resolution statistics of land conditions. The experimental results show that the proposed approach can effectively integrate both macro (aerial) and micro-level (vehicular) images, along with other forms of meta-data, to estimate city-scale phenomena. Furthermore, the proposed method can be successfully applied to vegetation estimation. The results indicate the method can generalize well to many kinds of applications to estimate city-scale phenomena by replacing the detector target (e.g. human flow, real-estate and dirt quality).

This dissertation achieved the objective of developing the methods for 4D city modeling in tsunami-damaged area using vehicular imagery. As mentioned in section 1, to estimate temporal change of regional-scale area using vehicular imagery, there are three challenges to overcome as follows, (i) limited camera viewpoint, (ii) limited physical range, (iii) large computation. The 3D change detection method makes it possible to detect structural change even if there is depth ambiguity due to the limited camera viewpoint. The method of land surface condition analysis integrates aerial and vehicular imagery and estimates change of debris distribution for entire city. Furthermore, the 2D change detection method can reduce the computational time and makes it possible to process the entire tsunami-damaged areas with a single workstation.

For future work, the three methods mentioned above can be integrated into a system which estimates temporal changes of vastly wide area, for example, the entire tsunami-damaged areas of the Tohoku. It is possible for all the methods to process multiple areas in parallel. If multiple computers are available, the temporal change of the entire tsunami-damaged areas can be estimated in a day or a few days.

If the number of sensors increases in the future (e.g. came mounted on self-driving car), scene images of cities will be available in real-time. The real-time sensor networks can generate real-time 3D map [86] and apply statistical analysis. The proposed 4D modeling approach is fast enough to be applied to such on-line sensory information. Combined with the real-time big data, the proposed method can extend to real-time monitoring of the city.

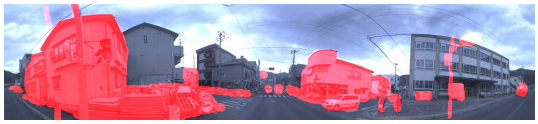
Appendix A

Other Results of 2D Change Detection

Figures A.1 - A.18 show all the other results of 2D change detection in Panoramic Change Detection Dataset (Chap. 4).

Input image (query)

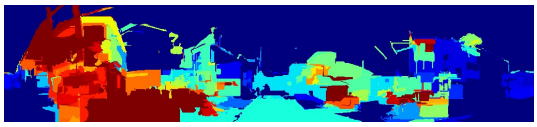
Input image (database)



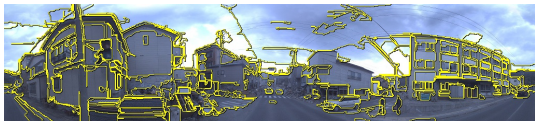
Ground-truth (superimposed)

Ground-truth (mask)

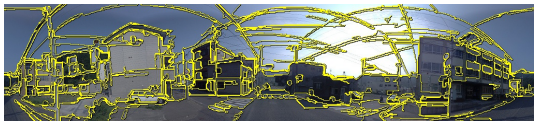
Change estimation (binarized)



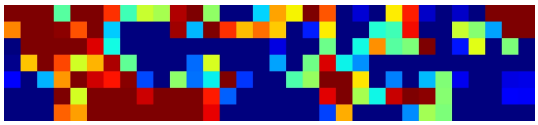
Change estimation (distance)



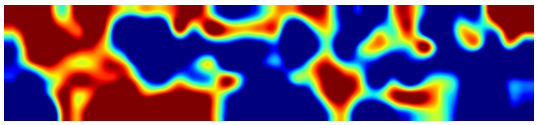
Superpixel segmentation (query)



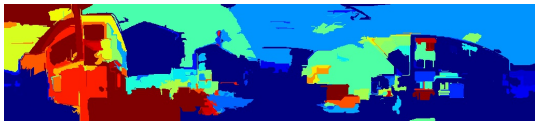
Superpixel segmentation (database)



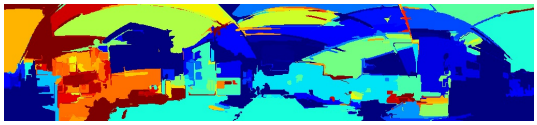
Feature distance (each grid)



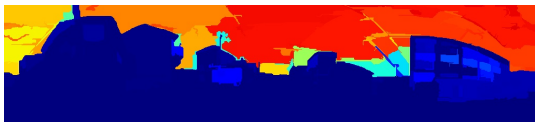
Feature distance (interpolation)



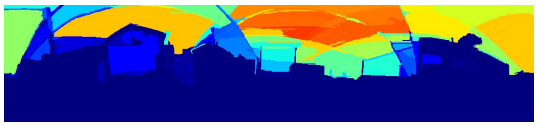
Feature distance in superpixel (query)



Feature distance in superpixel (database)

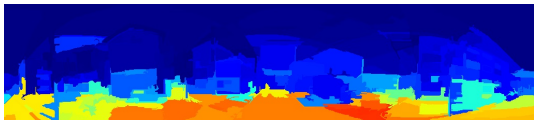


Sky probabilities (query)



Sky probabilities (database)

Ground probabilities (query)

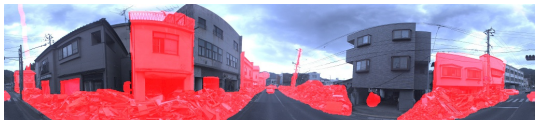


Ground probabilities (database)

Figure A.1: Results of change detection using pool-5 feature of CNN (Frame No. 2)

Input image (query)

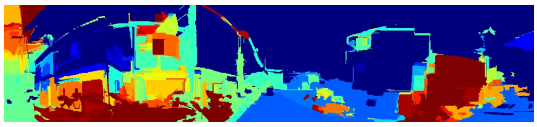
Input image (database)



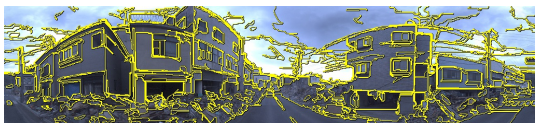
Ground-truth (superimposed)

Ground-truth (mask)

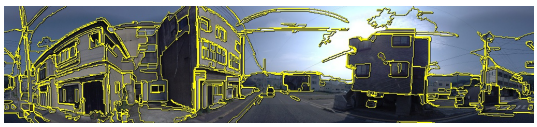
Change estimation (binarized)



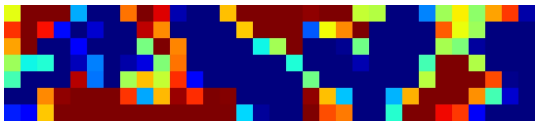
Change estimation (distance)



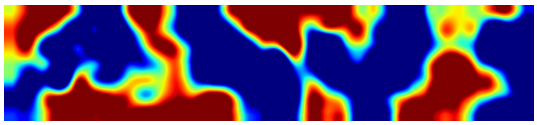
Superpixel segmentation (query)



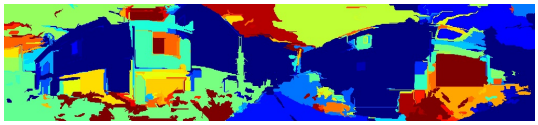
Superpixel segmentation (database)



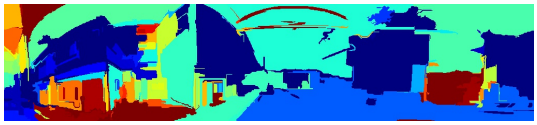
Feature distance (each grid)



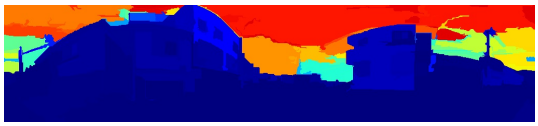
Feature distance (interpolation)



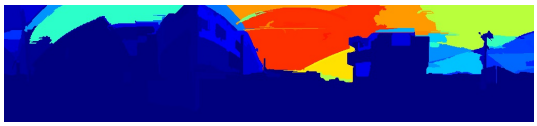
Feature distance in superpixel (query)



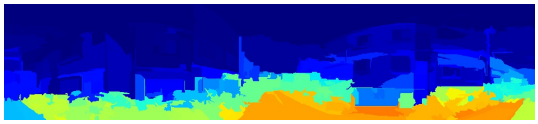
Feature distance in superpixel (database)



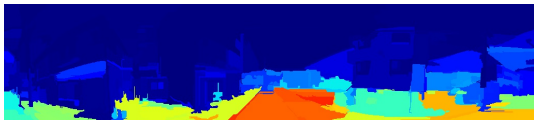
Sky probabilities (query)



Sky probabilities (database)



Ground probabilities (query)



Ground probabilities (database)

Figure A.2: Results of change detection using pool-5 feature of CNN (Frame No. 3)

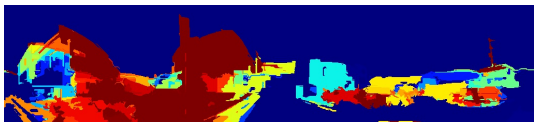
Input image (query)

Input image (database)

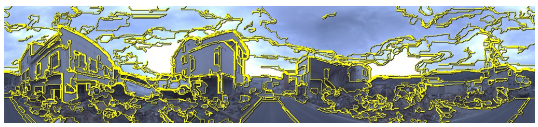
Ground-truth (superimposed)

Ground-truth (mask)

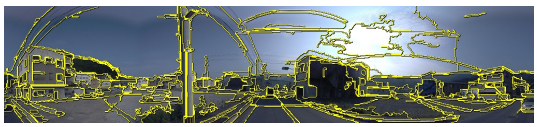
Change estimation (binarized)



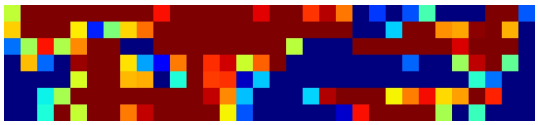
Change estimation (distance)



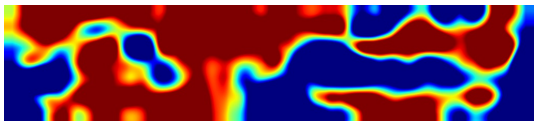
Superpixel segmentation (query)



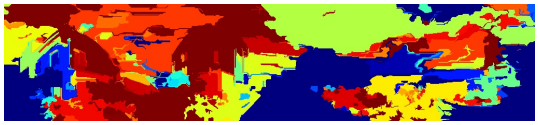
Superpixel segmentation (database)



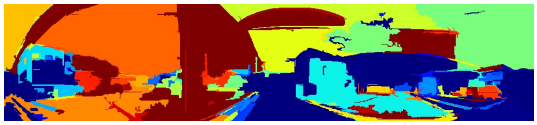
Feature distance (each grid)



Feature distance (interpolation)

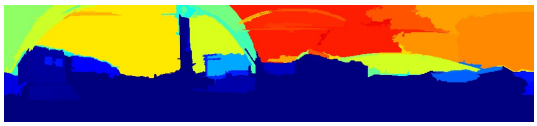


Feature distance in superpixel (query)

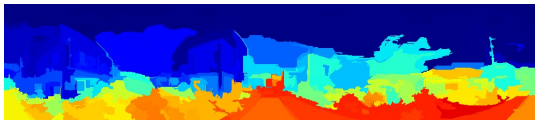


Feature distance in superpixel (database)

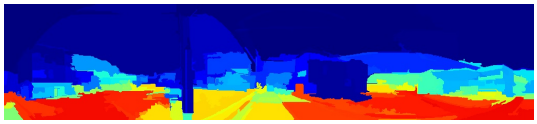
Sky probabilities (query)



Sky probabilities (database)



Ground probabilities (query)

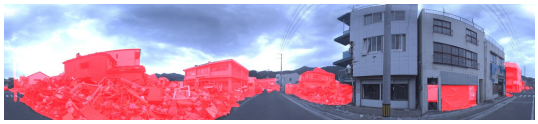


Ground probabilities (database)

Figure A.3: Results of change detection using pool-5 feature of CNN (Frame No. 4)

Input image (query)

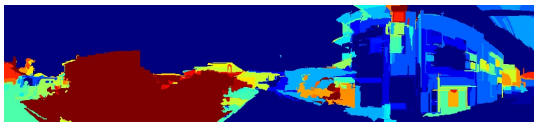
Input image (database)



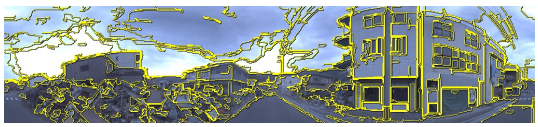
Ground-truth (superimposed)

Ground-truth (mask)

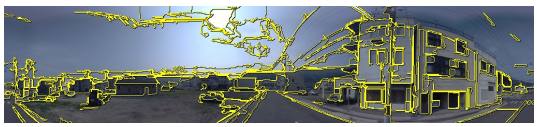
Change estimation (binarized)



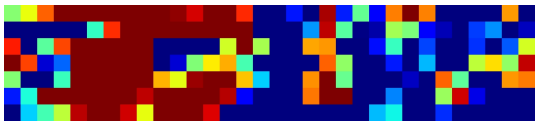
Change estimation (distance)



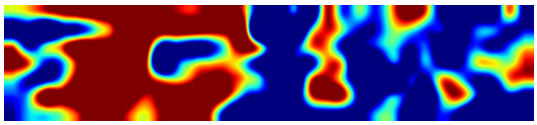
Superpixel segmentation (query)



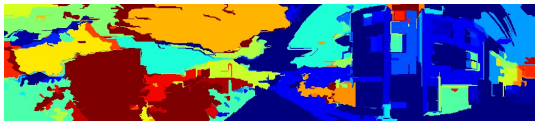
Superpixel segmentation (database)



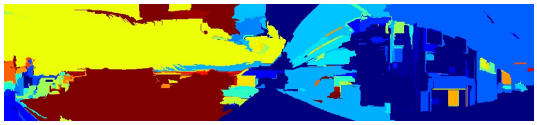
Feature distance (each grid)



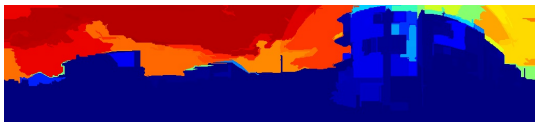
Feature distance (interpolation)



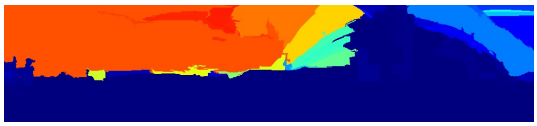
Feature distance in superpixel (query)



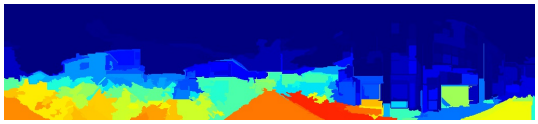
Feature distance in superpixel (database)



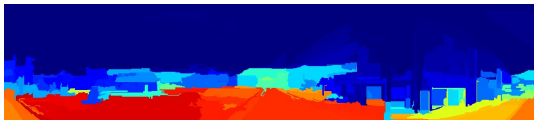
Sky probabilities (query)



Sky probabilities (database)



Ground probabilities (query)



Ground probabilities (database)

Figure A.4: Results of change detection using pool-5 feature of CNN (Frame No. 5)

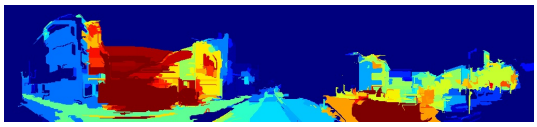
Input image (query)

Input image (database)

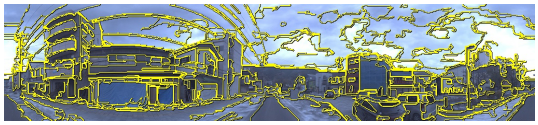
Ground-truth (superimposed)

Ground-truth (mask)

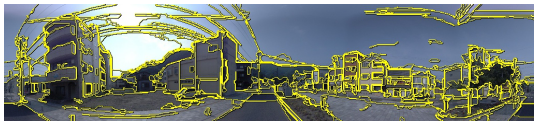
Change estimation (binarized)



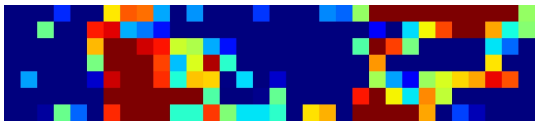
Change estimation (distance)



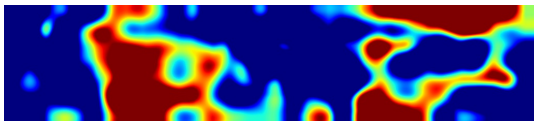
Superpixel segmentation (query)



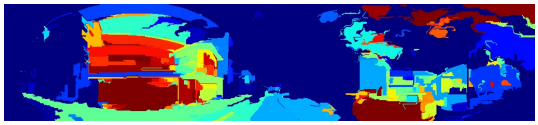
Superpixel segmentation (database)



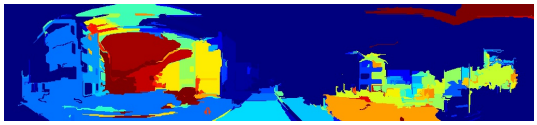
Feature distance (each grid)



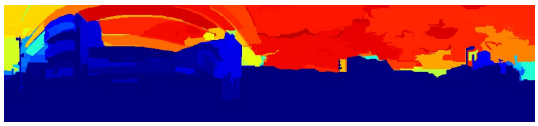
Feature distance (interpolation)



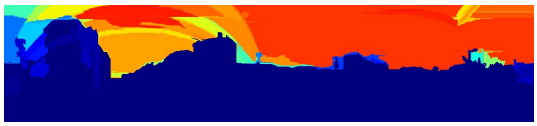
Feature distance in superpixel (query)



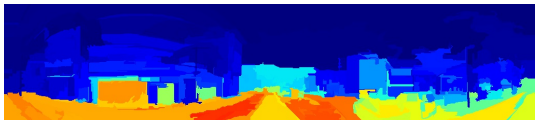
Feature distance in superpixel (database)



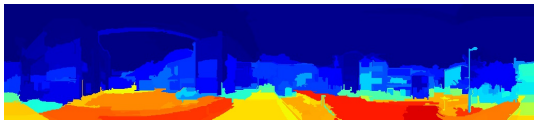
Sky probabilities (query)



Sky probabilities (database)



Ground probabilities (query)

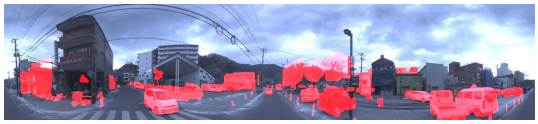


Ground probabilities (database)

Figure A.5: Results of change detection using pool-5 feature of CNN (Frame No. 6)

Input image (query)

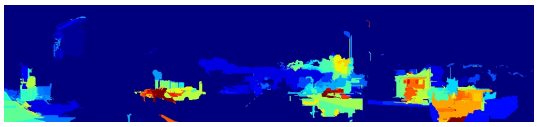
Input image (database)



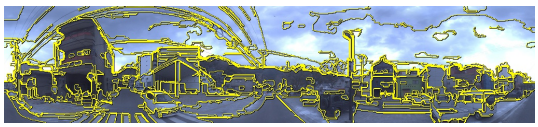
Ground-truth (superimposed)

Ground-truth (mask)

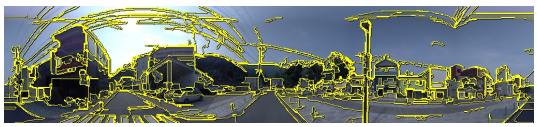
Change estimation (binarized)



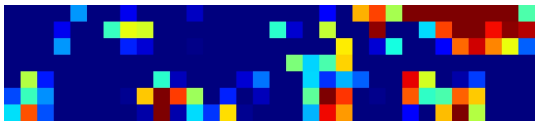
Change estimation (distance)



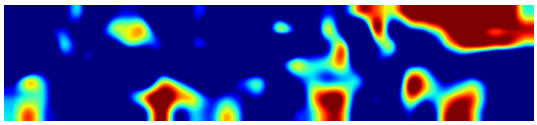
Superpixel segmentation (query)



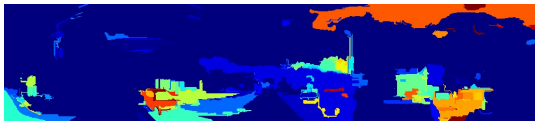
Superpixel segmentation (database)



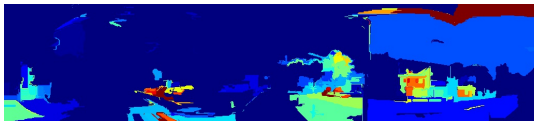
Feature distance (each grid)



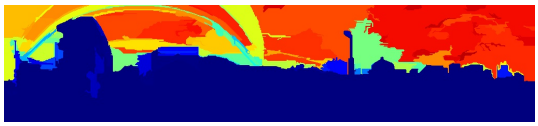
Feature distance (interpolation)



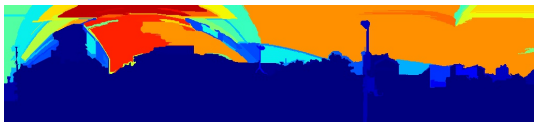
Feature distance in superpixel (query)



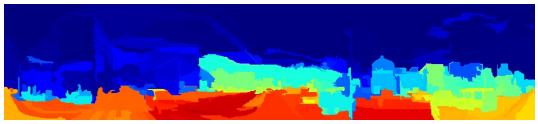
Feature distance in superpixel (database)



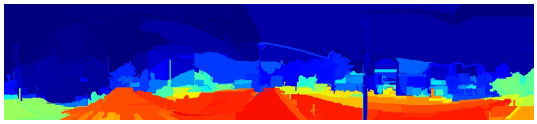
Sky probabilities (query)



Sky probabilities (database)



Ground probabilities (query)



Ground probabilities (database)

Figure A.6: Results of change detection using pool-5 feature of CNN (Frame No. 7)

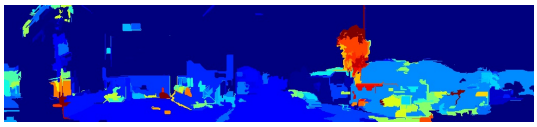
Input image (query)

Input image (database)

Ground-truth (superimposed)

Ground-truth (mask)

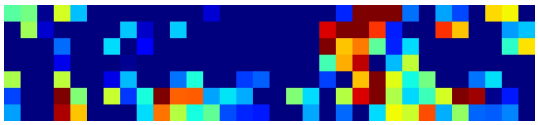
Change estimation (binarized)



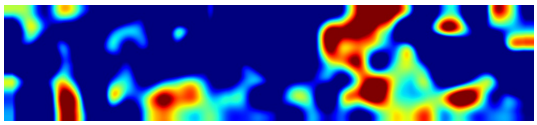
Change estimation (distance)

Superpixel segmentation (query)

Superpixel segmentation (database)

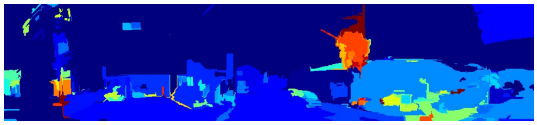


Feature distance (each grid)

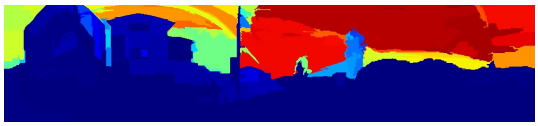


Feature distance (interpolation)

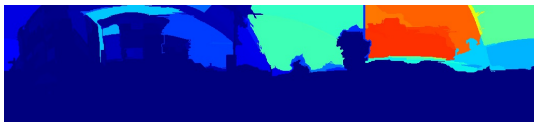
Feature distance in superpixel (query)



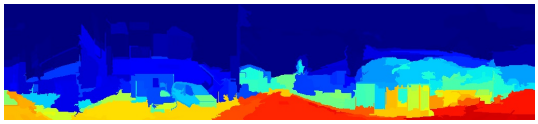
Feature distance in superpixel (database)



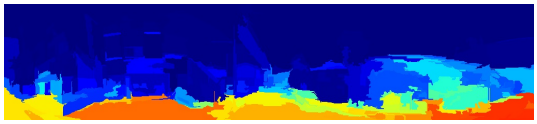
Sky probabilities (query)



Sky probabilities (database)



Ground probabilities (query)

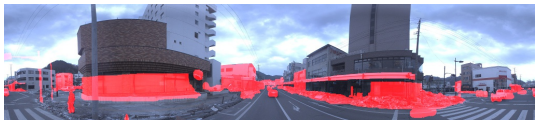


Ground probabilities (database)

Figure A.7: Results of change detection using pool-5 feature of CNN (Frame No. 8)

Input image (query)

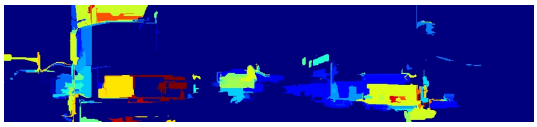
Input image (database)



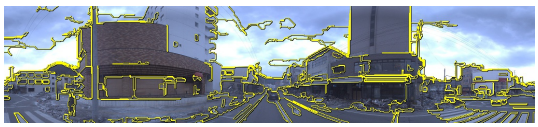
Ground-truth (superimposed)

Ground-truth (mask)

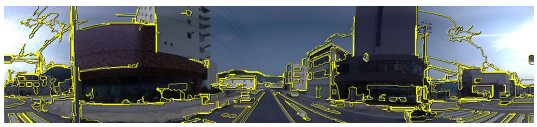
Change estimation (binarized)



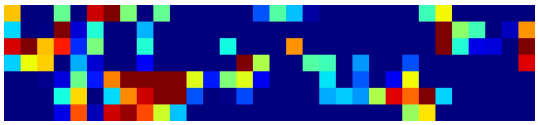
Change estimation (distance)



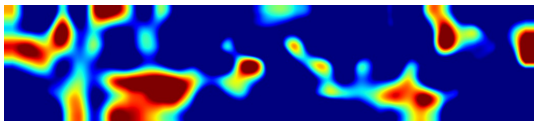
Superpixel segmentation (query)



Superpixel segmentation (database)

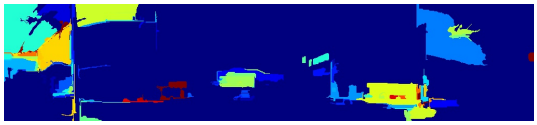


Feature distance (each grid)



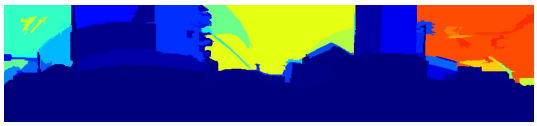
Feature distance (interpolation)

Feature distance in superpixel (query)

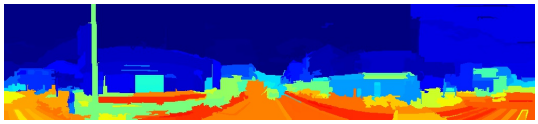


Feature distance in superpixel (database)

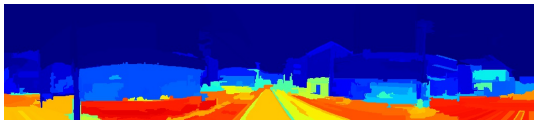
Sky probabilities (query)



Sky probabilities (database)



Ground probabilities (query)



Ground probabilities (database)

Figure A.8: Results of change detection using pool-5 feature of CNN (Frame No. 9)

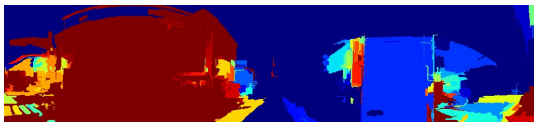
Input image (query)

Input image (database)

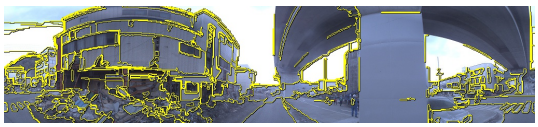
Ground-truth (superimposed)

Ground-truth (mask)

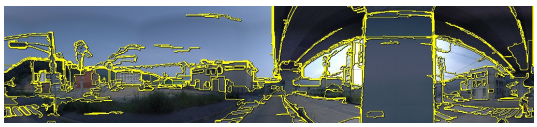
Change estimation (binarized)



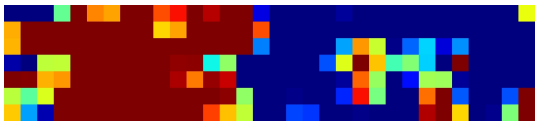
Change estimation (distance)



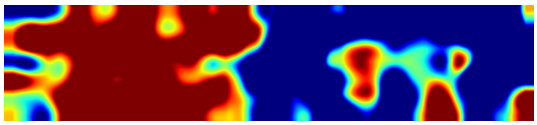
Superpixel segmentation (query)



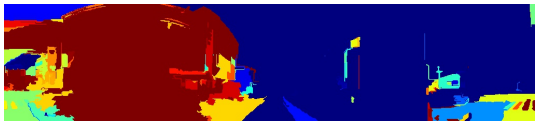
Superpixel segmentation (database)



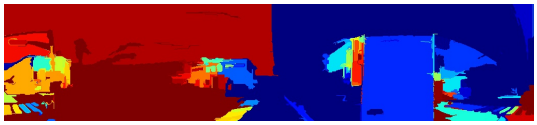
Feature distance (each grid)



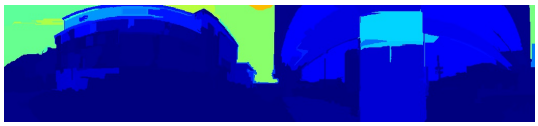
Feature distance (interpolation)



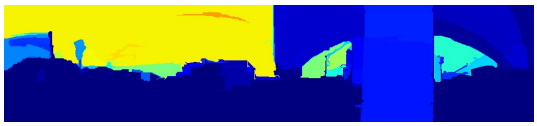
Feature distance in superpixel (query)



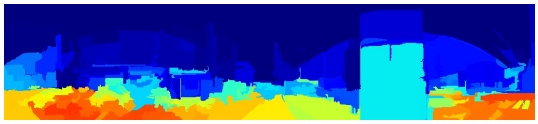
Feature distance in superpixel (database)



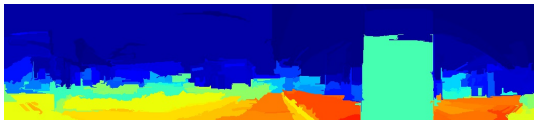
Sky probabilities (query)



Sky probabilities (database)



Ground probabilities (query)



Ground probabilities (database)

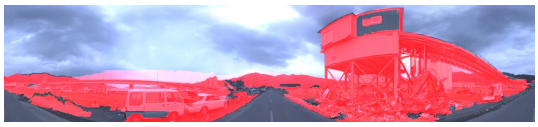
Figure A.9: Results of change detection using pool-5 feature of CNN (Frame No. 10)

Input image (query)

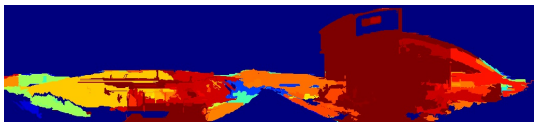
Input image (database)

Ground-truth (superimposed)

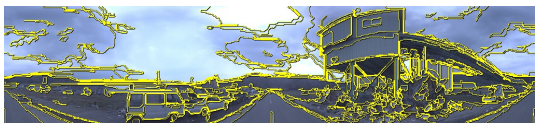
Ground-truth (mask)



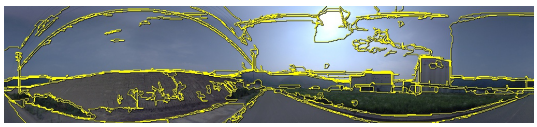
Change estimation (binarized)



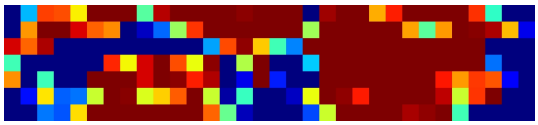
Change estimation (distance)



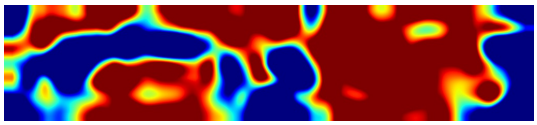
Superpixel segmentation (query)



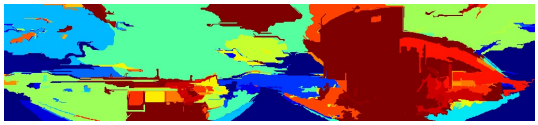
Superpixel segmentation (database)



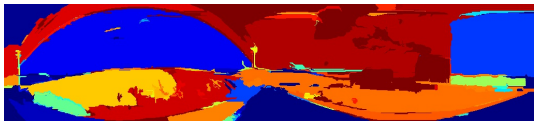
Feature distance (each grid)



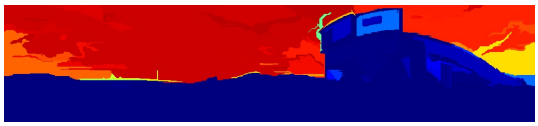
Feature distance (interpolation)



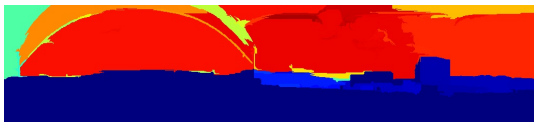
Feature distance in superpixel (query)



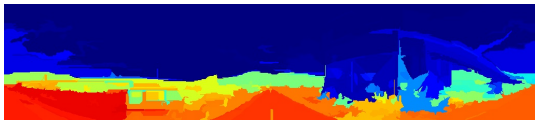
Feature distance in superpixel (database)



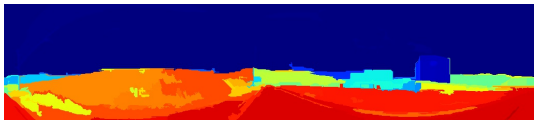
Sky probabilities (query)



Sky probabilities (database)



Ground probabilities (query)



Ground probabilities (database)

Figure A.10: Results of change detection using pool-5 feature of CNN (Frame No. 11)

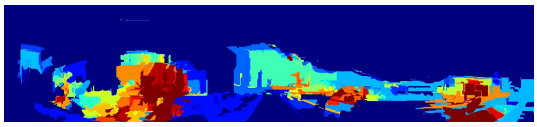
Input image (query)

Input image (database)

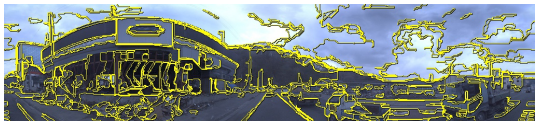
Ground-truth (superimposed)

Ground-truth (mask)

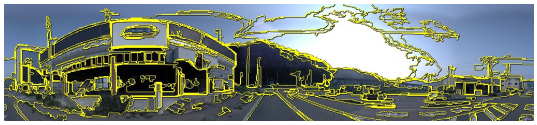
Change estimation (binarized)



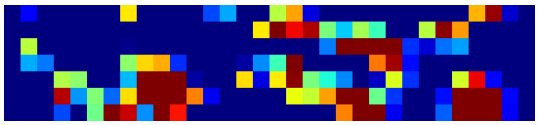
Change estimation (distance)



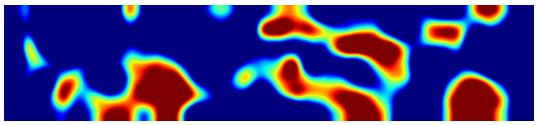
Superpixel segmentation (query)



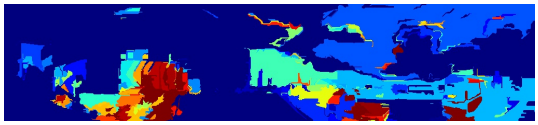
Superpixel segmentation (database)



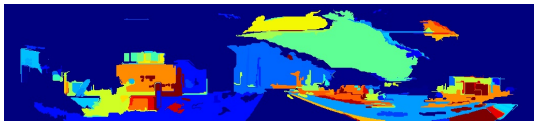
Feature distance (each grid)



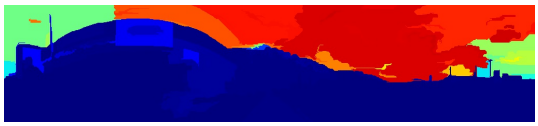
Feature distance (interpolation)



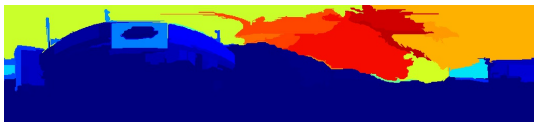
Feature distance in superpixel (query)



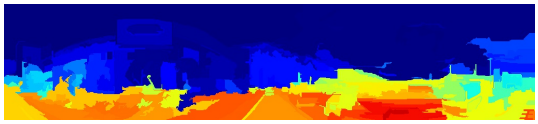
Feature distance in superpixel (database)



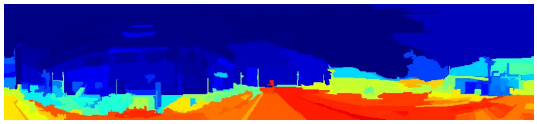
Sky probabilities (query)



Sky probabilities (database)



Ground probabilities (query)



Ground probabilities (database)

Figure A.11: Results of change detection using pool-5 feature of CNN (Frame No. 12)

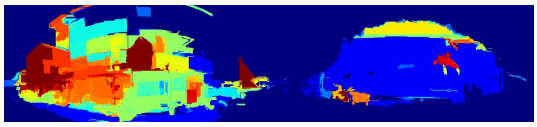
Input image (query)

Input image (database)

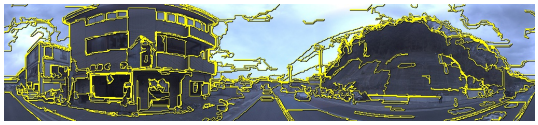
Ground-truth (superimposed)

Ground-truth (mask)

Change estimation (binarized)

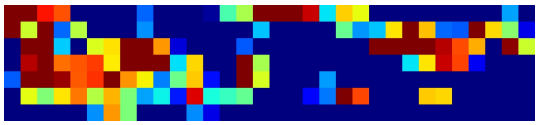


Change estimation (distance)

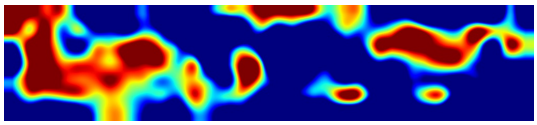


Superpixel segmentation (query)

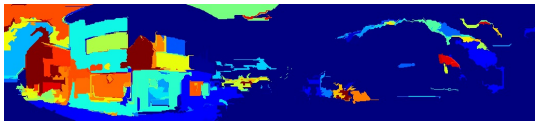
Superpixel segmentation (database)



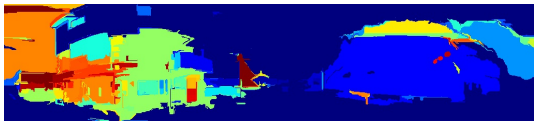
Feature distance (each grid)



Feature distance (interpolation)



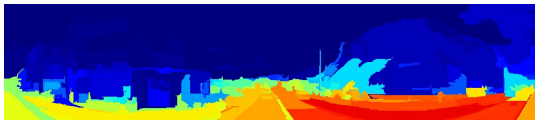
Feature distance in superpixel (query)



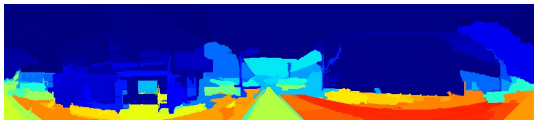
Feature distance in superpixel (database)

Sky probabilities (query)

Sky probabilities (database)



Ground probabilities (query)



Ground probabilities (database)

Figure A.12: Results of change detection using pool-5 feature of CNN (Frame No. 13)

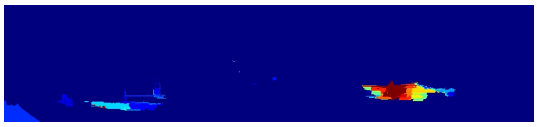
Input image (query)

Input image (database)

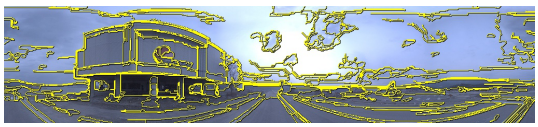
Ground-truth (superimposed)

Ground-truth (mask)

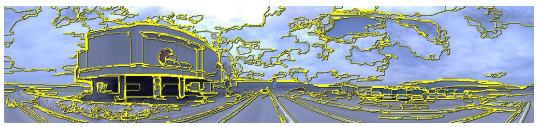
Change estimation (binarized)



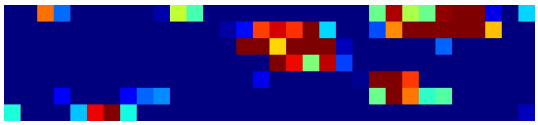
Change estimation (distance)



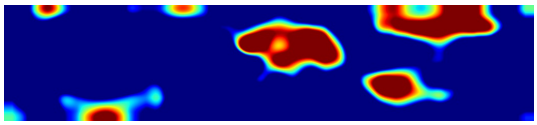
Superpixel segmentation (query)



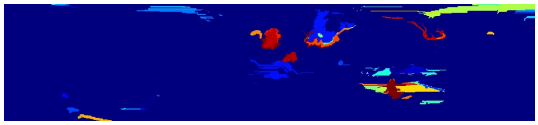
Superpixel segmentation (database)



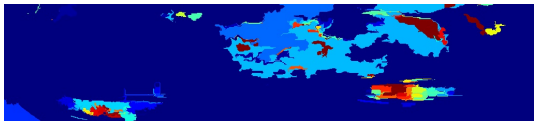
Feature distance (each grid)



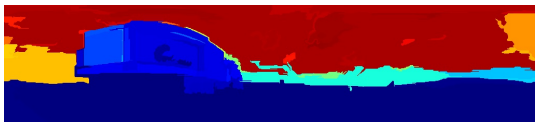
Feature distance (interpolation)



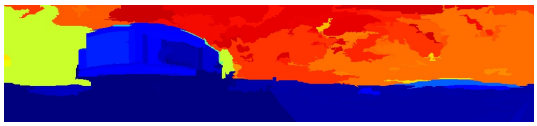
Feature distance in superpixel (query)



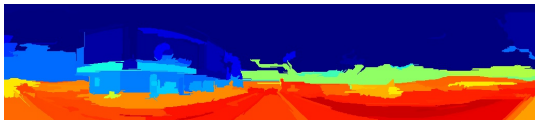
Feature distance in superpixel (database)



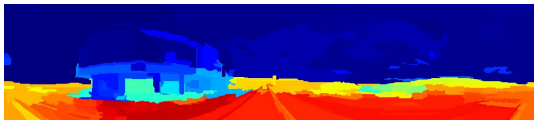
Sky probabilities (query)



Sky probabilities (database)



Ground probabilities (query)



Ground probabilities (database)

Figure A.13: Results of change detection using pool-5 feature of CNN (Frame No. 14)

Input image (query)

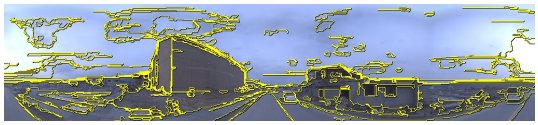
Input image (database)

Ground-truth (superimposed)

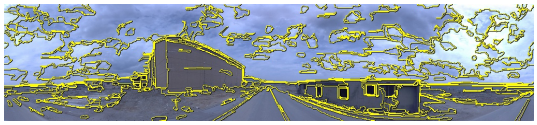
Ground-truth (mask)

Change estimation (binarized)

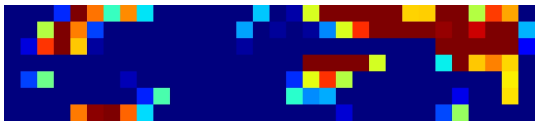
Change estimation (distance)



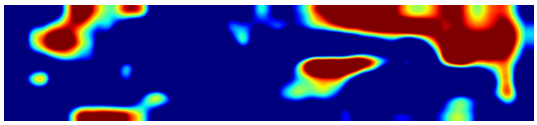
Superpixel segmentation (query)



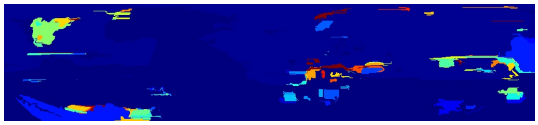
Superpixel segmentation (database)



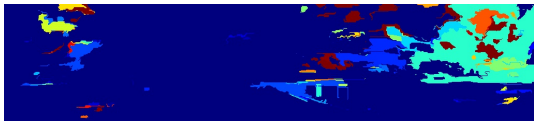
Feature distance (each grid)



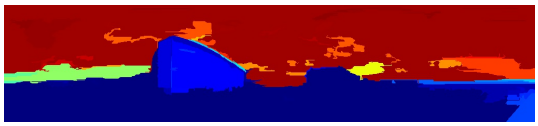
Feature distance (interpolation)



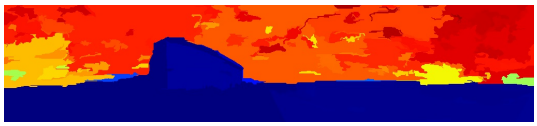
Feature distance in superpixel (query)



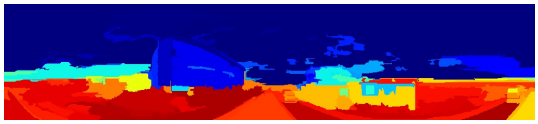
Feature distance in superpixel (database)



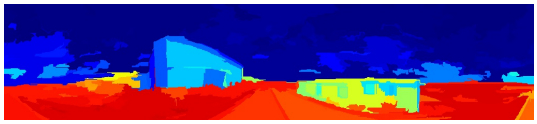
Sky probabilities (query)



Sky probabilities (database)



Ground probabilities (query)

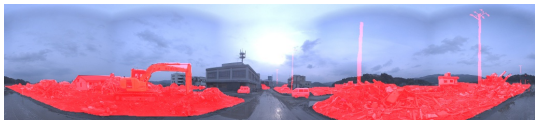


Ground probabilities (database)

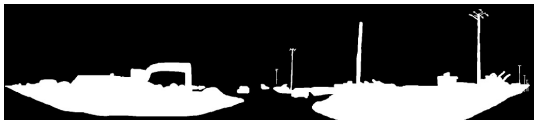
Figure A.14: Results of change detection using pool-5 feature of CNN (Frame No. 15)

Input image (query)

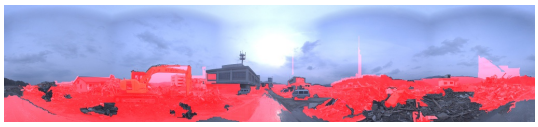
Input image (database)



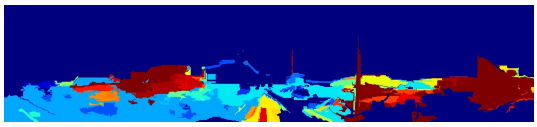
Ground-truth (superimposed)



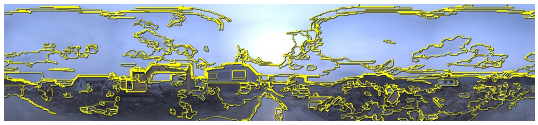
Ground-truth (mask)



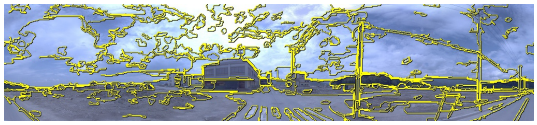
Change estimation (binarized)



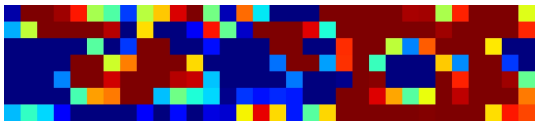
Change estimation (distance)



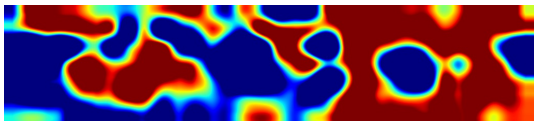
Superpixel segmentation (query)



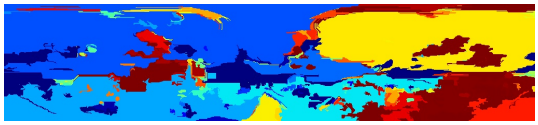
Superpixel segmentation (database)



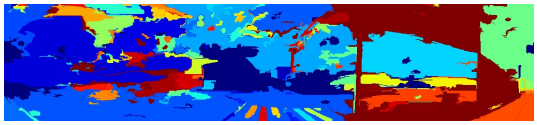
Feature distance (each grid)



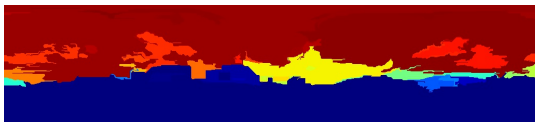
Feature distance (interpolation)



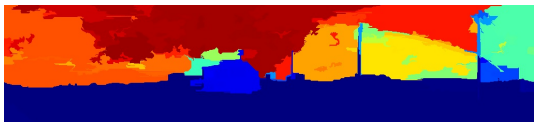
Feature distance in superpixel (query)



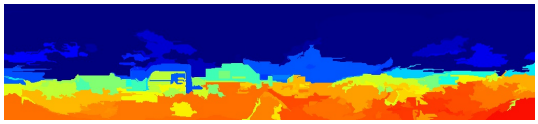
Feature distance in superpixel (database)



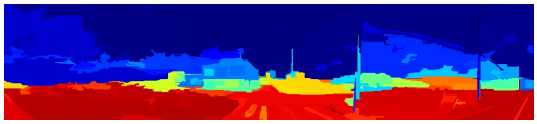
Sky probabilities (query)



Sky probabilities (database)



Ground probabilities (query)

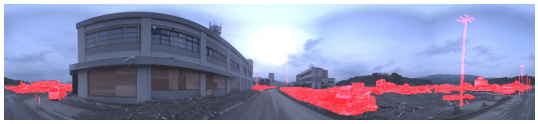


Ground probabilities (database)

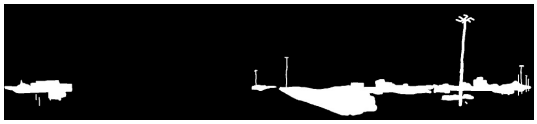
Figure A.15: Results of change detection using pool-5 feature of CNN (Frame No. 16)

Input image (query)

Input image (database)



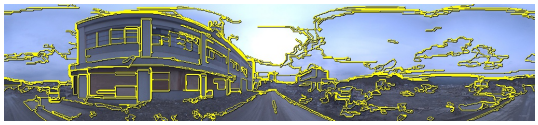
Ground-truth (superimposed)



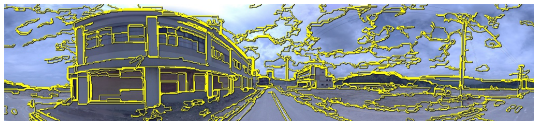
Ground-truth (mask)

Change estimation (binarized)

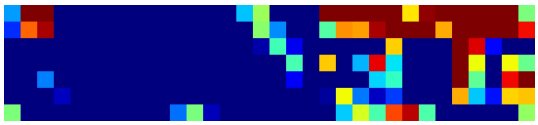
Change estimation (distance)



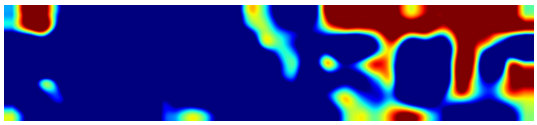
Superpixel segmentation (query)



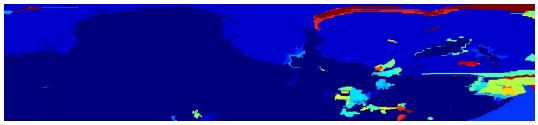
Superpixel segmentation (database)



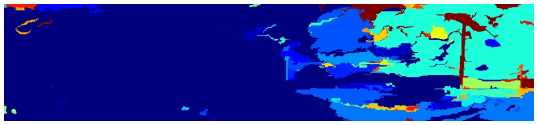
Feature distance (each grid)



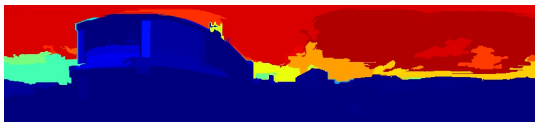
Feature distance (interpolation)



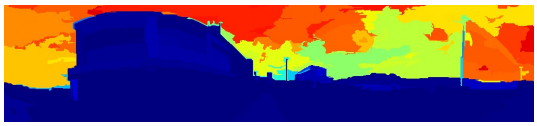
Feature distance in superpixel (query)



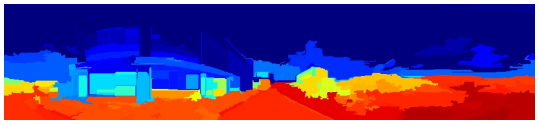
Feature distance in superpixel (database)



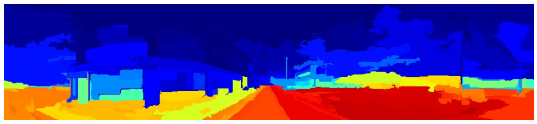
Sky probabilities (query)



Sky probabilities (database)



Ground probabilities (query)

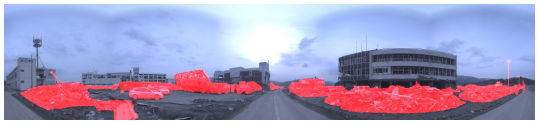


Ground probabilities (database)

Figure A.16: Results of change detection using pool-5 feature of CNN (Frame No. 17)

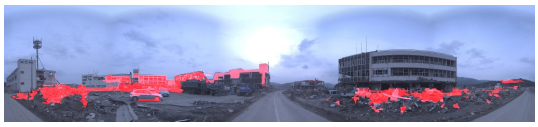
Input image (query)

Input image (database)

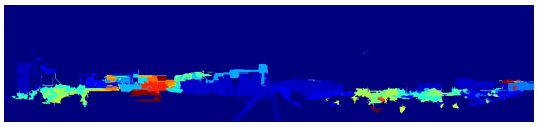


Ground-truth (superimposed)

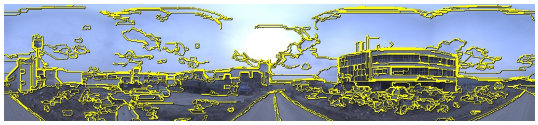
Ground-truth (mask)



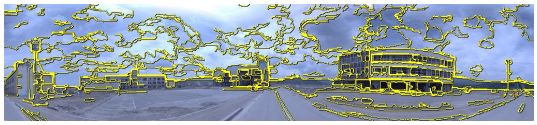
Change estimation (binarized)



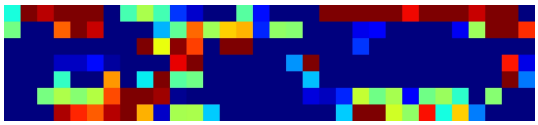
Change estimation (distance)



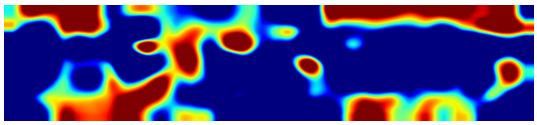
Superpixel segmentation (query)



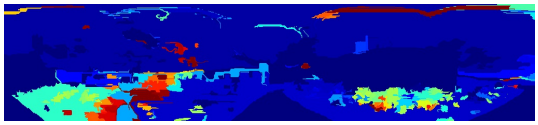
Superpixel segmentation (database)



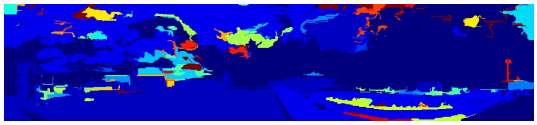
Feature distance (each grid)



Feature distance (interpolation)



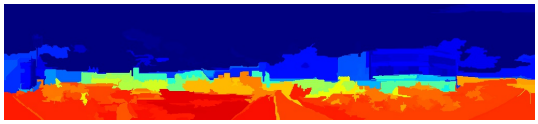
Feature distance in superpixel (query)



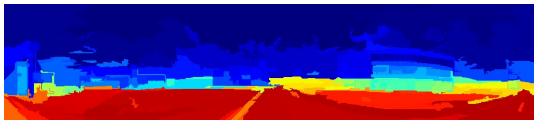
Feature distance in superpixel (database)

Sky probabilities (query)

Sky probabilities (database)



Ground probabilities (query)

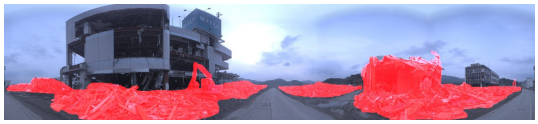


Ground probabilities (database)

Figure A.17: Results of change detection using pool-5 feature of CNN (Frame No. 18)

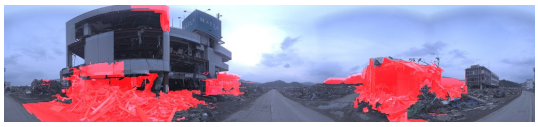
Input image (query)

Input image (database)

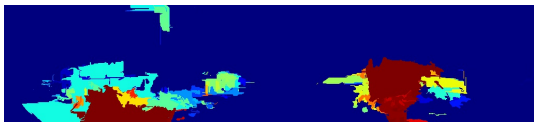


Ground-truth (superimposed)

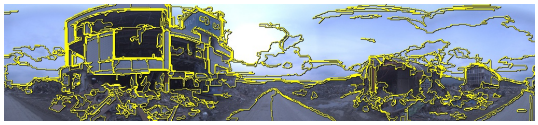
Ground-truth (mask)



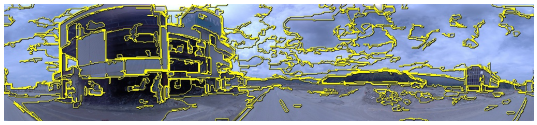
Change estimation (binarized)



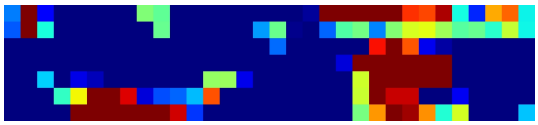
Change estimation (distance)



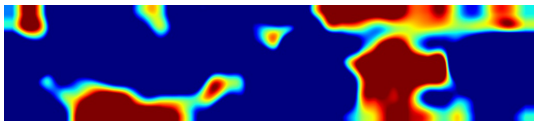
Superpixel segmentation (query)



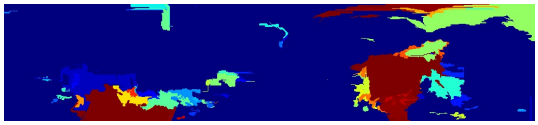
Superpixel segmentation (database)



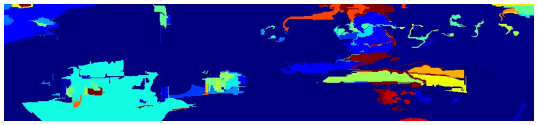
Feature distance (each grid)



Feature distance (interpolation)



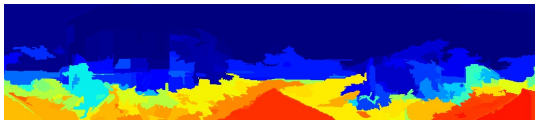
Feature distance in superpixel (query)



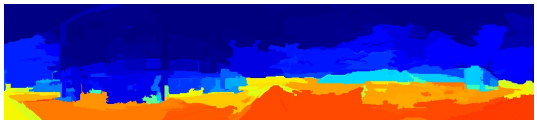
Feature distance in superpixel (database)

Sky probabilities (query)

Sky probabilities (database)



Ground probabilities (query)



Ground probabilities (database)

Figure A.18: Results of change detection using pool-5 feature of CNN (Frame No. 19)

Bibliography

[1] Yasutaka Furukawa and Jean Ponce. Accurate, Dense, and Robust Multi-View Stereopsis. *PAMI*, Vol. 32, No. 8, pp. 1362–1376, 2010.

[2] Jinhui Hu, Suya You, and Ulrich Neumann. Approaches to large-scale urban modeling. *Computer Graphics and Applications, IEEE*, Vol. 23, No. 6, pp. 62–69, 2003.

[3] Norbert Haala and Martin Kada. An update on automatic 3d building reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 65, No. 6, pp. 570–580, 2010.

[4] Franz Rottensteiner, Gunho Sohn, Jaewook Jung, M Gerke, Caroline Baillard, Sebastien Benitez, and Uwe Breitkopf. The isprs benchmark on urban object classification and 3d building reconstruction. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences I-3*, pp. 293–298, 2012.

[5] Franz Rottensteiner, Gunho Sohn, Markus Gerke, and Jan Dirk Wegner. Isprs test project on urban classification and 3d building reconstruction. *Commission III-Photogrammetric Computer Vision and Image Analysis, Working Group III/4-3D Scene Analysis*, pp. 1–17, 2013.

[6] James B. Campbell and Randolph H. Wynne. *Introduction to Remote Sensing (5th edition)*. Guilford Press, 2011.

[7] Przemyslaw Musialski, Peter Wonka, Daniel G. Aliaga, Michael Wimmer, Luc van Gool, and Werner Purgathofer. A Survey of Urban Reconstruction. *Computer Graphics Forum*, Vol. 32, No. 6, pp. 146–177, sep 2013.

[8] Sameer Agarwal, Noah Snavely, Ian Simon, Steven M Seitz, and Richard Szeliski. Building Rome in a day. In *ICCV*, pp. 72–79, 2009.

[9] Jan-Michael Frahm, Pierre Fite-Georgel, David Gallup, Tim Johnson, Rahul Raguram, Changchang Wu, Yi-Hung Jen, Enrique Dunn, Brian Clipp, Svetlana Lazebnik, et al. Building rome on a cloudless day. In *Computer Vision–ECCV 2010*, pp. 368–381. Springer, 2010.

[10] M. Pollefeys, D. Nistér, J.-M. Frahm, A. Akbarzadeh, P. Mordohai, B. Clipp, C. Engels, D. Gallup, S.-J. Kim, P. Merrell, C. Salmi, S. Sinha, B. Talton, L. Wang, Q. Yang, H. Stewénius, R. Yang, G. Welch, and H. Towles. Detailed Real-Time Urban 3D Reconstruction from Video. *IJCV*, Vol. 78, No. 2-3, pp. 143–167, 2008.

[11] Charalambos Poullis and Suya You. 3d reconstruction of urban areas. In *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2011 International Conference on*, pp. 33–40. IEEE, 2011.

[12] Charalambos Poullis and Suya You. Automatic creation of massive virtual cities. In *Virtual Reality Conference, 2009. VR 2009. IEEE*, pp. 199–202. IEEE, 2009.

[13] Charalambos Poullis and Suya You. Automatic reconstruction of cities from remote sensor data. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 2775–2782. IEEE, 2009.

[14] Charalambos Poullis and Suya You. Photorealistic large-scale urban city model reconstruction. *Visualization and Computer Graphics, IEEE Transactions on*, Vol. 15, No. 4, pp. 654–669, 2009.

[15] Qian-Yi Zhou and Ulrich Neumann. A streaming framework for seamless building reconstruction from large-scale aerial lidar data. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 2759–2766. IEEE, 2009.

[16] Qian-Yi Zhou and Ulrich Neumann. 2.5 d building modeling with topology control. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 2489–2496. IEEE, 2011.

[17] Florent Lafarge and Clement Mallet. Building large urban environments from unstructured point data. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 1068–1075. IEEE, 2011.

[18] Florent Lafarge and Clément Mallet. Creating large-scale city models from 3d-point clouds: a robust approach with hybrid representation. *International journal of computer vision*, Vol. 99, No. 1, pp. 69–85, 2012.

[19] Randi Cabezas, Oren Freifeld, Guy Rosman, and John W. Fisher III. Aerial reconstructions via probabilistic data fusion. In *IEEE Computer Vision and Pattern Recognition Conference on Computer Vision*, June 2014.

[20] Thomas Pollard and Joseph L. Mundy. Change Detection in a 3-d World. In *CVPR*, pp. 1–6, 2007.

[21] Richard J Radke, Srinivas Andra, Omar Al-Kofahi, and Badrinath Roysam. Image Change Detection Algorithms: A Systematic Survey. *Transactions on Image Processing*, Vol. 14, No. 3, pp. 294–307, 2005.

[22] Daniel Crispell, Joseph Mundy, and Gabriel Taubin. A Variable-Resolution Probabilistic Three-Dimensional Model for Change Detection. *Geoscience and Remote Sensing*, Vol. 50, No. 2, pp. 489–500, 2012.

[23] David Cooper Ibrahim Eden. Using 3D Line Segments for Robust and Efficient Change Detection from Multiple Noisy Images. In *ECCV*, pp. 172–185, 2008.

[24] Andres Huertas and Ramakant Nevatia. Detecting Changes in Aerial Views of Man-Made Structures. In *ICCV*, pp. 73–80, 1998.

[25] Aparna Taneja, Luca Ballan, and Marc Pollefeys. Image based detection of geometric changes in urban environments. In *ICCV*, pp. 2336–2343, 2011.

[26] David Crandall, Andrew Owens, Noah Snavely, and Daniel Huttenlocher. Discrete-Continuous Optimization for Large-Scale Structure from Motion. In *CVPR*, pp. 3001–3008, 2011.

[27] Noah Snavely, Steven M. Seitz, and Richard Szeliski. Modeling the World from Internet Photo Collections. *IJCV*, Vol. 80, No. 2, pp. 189–210, 2007.

[28] Chenxi Zhang, Liang Wang, and Ruigang Yang. Semantic Segmentation of Urban Scenes Using Dense Depth Maps. In *ECCV*, pp. 708–721, 2010.

[29] Guofeng Zhang, Jiaya Jia, Wei Xiong, Tien-Tsin Wong, Pheng-Ann Heng, and Hujun Bao. Moving Object Extraction with a Hand-held Camera. In *ICCV*, pp. 1–8, 2007.

[30] Grant Schindler and Frank Dellaert. Probabilistic temporal inference on reconstructed 3D scenes. In *CVPR*, pp. 1410–1417, 2010.

[31] Akihiko Torii, Michal Havlena, and Tomas Pajdla. From Google Street View to 3D City Models. In *ICCV Workshops*, pp. 2188–2195, 2009.

[32] Chungan Lin and Ramakant Nevatia. Building Detection and Description from a Single Intensity Image. *Computer Vision and Image Understanding*, Vol. 72, No. 2, pp. 101–121, 1998.

[33] Ildiko Suveg and George Vosselman. Reconstruction of 3D building models from aerial images and maps. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 58, No. 3-4, pp. 202–224, 2004.

[34] Lukas Zebedin, Andreas Klaus, Barbara Gruber-Geymayer, and Konrad Karner. Towards 3D map generation from digital aerial images. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 60, No. 6, pp. 413–427, 2006.

[35] R.S. Kaminsky, N. Snavely, S.M. Seitz, and R. Szeliski. Alignment of 3D Point Clouds to Overhead Images. *CVPR Workshops*, pp. 63–70, 2009.

[36] Christoph Strecha, Timo Pylvanainen, and Pascal Fua. Dynamic and Scalable Large Scale Image Reconstruction. In *CVPR*, pp. 406–413, 2010.

[37] Aparna Taneja, Luca Ballan, and Marc Pollefeys. City-Scale Change Detection in Cadastral 3D Models Using Images. In *CVPR*, pp. 113–120, 2013.

[38] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision Second Edition*. Cambridge University Press, 2004.

[39] F Li, Thomas J. Jacksona, William P. Kustasa, Thomas J. Schmuggea, Andrew N. Frenchb, Michael H. Cosha, and Rajat Bindlish. Deriving land surface temperature from Landsat 5 and 7 during SMEX02/SMACEX. *Remote Sensing of Environment*, Vol. 92, No. 4, pp. 521–534, 2004.

[40] Qihao Weng, Dengsheng Lu, and Jacquelyn Schubring. Estimation of land surface temperature–vegetation abundance relationship for urban heat island studies. *Remote Sensing of Environment*, Vol. 89, No. 4, pp. 467–483, 2004.

[41] Robert A. Schowengerdt. *Remote Sensing: Models and Methods for Image Processing*. 2006.

[42] J Martinez and T Letoan. Mapping of flood dynamics and spatial distribution of vegetation in the Amazon floodplain using multitemporal SAR data. *Remote Sensing of Environment*, Vol. 108, No. 3, pp. 209–223, 2007.

[43] Qihao Weng. *Remote Sensing of Impervious Surfaces*. CRC Press, 2010.

[44] Dengsheng Lu, Scott Hetrick, and Emilio Moran. Impervious surface mapping with quickbird imagery. *International journal of remote sensing*, Vol. 32, No. 9, pp. 2519–2533, 2011.

[45] Andrew Hall, John Louis, and David Lamb. Characterising and mapping vineyard canopy using high-spatial-resolution aerial multispectral images. *Computers & Geosciences*, Vol. 29, No. 7, pp. 813–822, 2003.

[46] Jose A J Berni, Student Member, Pablo J Zarco-tejada, Lola Suárez, and Elias Fereres. Thermal and Narrowband Multispectral Remote Sensing for Vegetation Monitoring From an Unmanned Aerial Vehicle. *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 47, No. 3, pp. 722–738, 2009.

[47] Carole Delenne, Sylvie Durrieu, Gilles Rabatel, and Michel Deshayes. From pixel to vine parcel: A complete methodology for vineyard delineation and characterization using remote-sensing data. *Computers and Electronics in Agriculture*, Vol. 70, No. 1, pp. 78–83, 2010.

[48] C.J. van der Sande, S.M. de Jong, and a.P.J. de Roo. A segmentation and classification approach of IKONOS-2 imagery for land cover mapping to assist flood risk and flood damage assessment. *International Journal of Applied Earth Observation and Geoinformation*, Vol. 4, No. 3, pp. 217–229, 2003.

[49] Martin Herold, XiaoHang Liu, and Keith C. Clarke. Spatial Metrics and Image Texture for Mapping Urban Land Use. *Photogrammetric Engineering & Remote Sensing*, No. 9, pp. 991–1001, 2003.

[50] Peng Gong, R Pu, and J Chen. Mapping Ecological Land Systems and Classification Uncertainties from Digital Elevation and Forest-Cover Data Using Neural Networks. *Photogrammetric Engineering & Remote Sensing*, Vol. 62, No. 11, pp. 1249–1260, 1996.

[51] Anne H Schistad Solberg. Contextual Data Fusion Applied to Forest Map Revision. *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 37, No. 3, pp. 1234–1243, 1999.

[52] M. C. Hansen, P. V. Potapov, R Moore, M. Hancher, S. A. Turubanova, A. Tyukavina, D. Thau, S. V. Stehman, S. J. Goetz, T. R. Loveland, A. Kommareddy, A. Egorov, L. Chini, C. O. Justice, and J. R. G. Townshend. High-Resolution Global Maps of 21st-Century Forest Cover Change. *Science*, Vol. 342, pp. 850–853, 2013.

[53] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *ECCV*, pp. 404–417. Springer, 2006.

[54] David Nistér. An efficient solution to the five-point relative pose problem. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, Vol. 26, No. 6, pp. 756–770, 2004.

[55] Martin A Fischler and Robert C Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, Vol. 24, No. 6, pp. 381–395, 1981.

[56] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, Vol. 24, No. 6, pp. 381–395, 1981.

[57] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pp. 1097–1105. Curran Associates, Inc., 2012.

[58] Yoshua Bengio. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, Vol. 2, No. 1, pp. 1–127, 2009.

[59] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *ECCV*, pp. 818–833. Springer, 2014.

[60] Derek Hoiem, Alexei A. Efros, and Martial Hebert. Geometric context from a single image. In *ICCV*, pp. 654–661, 2005.

[61] David G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *IJCV*, Vol. 60, No. 2, pp. 91–110, 2004.

[62] Ce Liu, Jenny Yuen, Antonio Torralba, Josef Sivic, and William T Freeman. Sift flow: Dense correspondence across different scenes. In *ECCV*, pp. 28–42. Springer, 2008.

[63] Andrea Vedaldi and Brian Fulkerson. Vlfeat: An open and portable library of computer vision algorithms. In *International Conference on Multimedia*, MM '10, pp. 1469–1472. ACM, 2010.

[64] Jun Yang, Yu-Gang Jiang, Alexander G. Hauptmann, and Chong-Wah Ngo. Evaluating bag-of-visual-words representations in scene classification. In *International Workshop on Workshop on Multimedia Information Retrieval*, pp. 197–206. ACM, 2007.

[65] Florent Perronnin and Christopher Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, pp. 1–8. IEEE, 2007.

[66] Florent Perronnin, Jorge Sánchez, and Thomas Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*, pp. 143–156. Springer, 2010.

[67] Daniel LK Yamins, Ha Hong, Charles F Cadieu, Ethan A Solomon, Darren Seibert, and James J DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, p. 201403112, 2014.

[68] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, Vol. abs/1409.1556, , 2014.

[69] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, Vol. 59, No. 2, pp. 167–181, 2004.

[70] S.M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms. In *CVPR*, pp. 519–528, 2006.

[71] Bill Triggs, Philip McLauchlan, R.ichard Hartley, and A.ndrew Fitzgibbon. Bundle Adjustment - Modern Synthesis. In *ICCV*, pp. 298–372, 1999.

[72] Sing Bing Kang, Richard Szeliski, and Jinxiang Chai. Handling Occlusions in Dense Multi-view Stereo. In *CVPR*, pp. I–103–I–110, 2001.

[73] Guofeng Zhang, Jiaya Jia, Tien-tsin Wong, and Hujun Bao. Recovering Consistent Video Depth Maps via Bundle Optimization. In *CVPR*, pp. 1–8, 2008.

[74] Engin Tola, Vincent Lepetit, and Pascal Fua. DAISY: An Efficient Dense Descriptor Applied to Wide-Baseline Stereo. *PAMI*, Vol. 32, No. 5, pp. 815–830, 2010.

[75] Vladimir Kolmogorov and Ramin Zabih. What energy functions can be minimized via graph cuts? *PAMI*, Vol. 26, No. 2, pp. 147–59, 2004.

[76] Li, Cheng and Kitani, Kris M. Pixel-level Hand Detection in Ego-Centric Videos. In *CVPR*, pp. 3570–3577, 2013.

[77] Navneet Dalal and Bill Triggs. Histograms of Oriented Gradients for Human Detection. In *CVPR*, pp. 886–893, 2005.

[78] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. BRIEF : Binary Robust Independent Elementary Features . In *ECCV*, 2010.

[79] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. ORB : an efficient alternative to SIFT or SURF. In *ICCV*, pp. 2564–2571, 2011.

[80] LEO BREIMAN. Random Forests. *Machine Learning*, Vol. 45, pp. 5–32, 2001.

[81] Tom M Mitchell. *Machine Learning*. 1997.

[82] Vladimir Naumovich Vapnik. *Estimation of Dependences Based on Empirical Data*. 2006.

[83] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[84] Geremy Heitz and Daphne Koller. Learning spatial context: Using stuff to find things. In *Computer Vision–ECCV 2008*, pp. 30–43. Springer, 2008.

[85] Tat-Jen Cham, Arridhana Ciptadi, Wei-Chian Tan, Minh-Tri Pham, and Liang-Tien Chia. Estimating Camera Pose from a Single Urban Ground-View Omnidirectional Image and a 2D Building Outline Map. In *CVPR*, pp. 366–373.

[86] Kuan-Hui Lee, Jenq-Neng Hwang, Greg Okapal, and James Pitton. Driving recorder based on-road pedestrian tracking using visual slam and constrained multiple-kernel. In *International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2629–2635. IEEE, 2014.

Acknowledgement

First and foremost, I would like to thank my advisor Prof. Takayuki Okatani for supporting me all the time towards the completion of the dissertation research. I was a beginner of computer vision since I changed my major when I started PhD course. I guess that it was not easy to advise me, however, Takayuki always tackled many research problems together. Takayuki motivated his students with his enthusiasm and unique ideas. I am really happy to spend my student life with him for discussion, fieldworks.

I would like to thank Prof. Koichiro Deguchi for co-advising me towards the degree. Koichiro helped me a lot for studying at Carnegie Mellon University. Furthermore, I spend a long time with Koichiro for recording images in tsunami-damaged area and its data reduction. His backup data saved us when our data broke. His most impressive words is "You should answer to a question with 'Yes', 'No' or 'I don't know'.". This is very essential for all things.

I would like to thank Prof. Kota Yamaguchi for co-advising me towards the degree. Kota advised me from a point of view different from Takayuki and Koichiro. His advice is very clear and practical, and he motivated me with his interesting and unique sense.

I would like to thank Prof. Kris M. Kitani for collaborating with me. Kris always gave me concrete advises and motivated me with his infinite ideas. I never thought that I could have such a interesting collaboration. The collaboration with him drastically widened my research area.

Lastly, I would like to thank all my fellow students at Tohoku University and Carnegie Mellon University. Eisuke Ito and Masaki Saito always helped me with their unique ideas and knowledge about research. Jun Yanagisawa and Daiki Tetsuka worked with me for disaster projects. Yasuhiro Akashi, Hirokazu Omokawa, Yuta Shirakawa, Makoto Ozeki, Xing Liu and Sumadianto Eka Putra discussed with me from their view points and helped me to make ground-truth data.

List of Authorial Publications

[1] Ken Sakurada, Takayuki Okatani, Kris M. Kitani, "Massive City-scale Surface Condition Analysis using Ground and Aerial Imagery", ACCV2014 (Oral, Acceptance Rate: Less than 4%), "Best Application Paper Honorable Mention Award"

[2] Ken Sakurada, Takayuki Okatani, Koichiro Deguchi, "Detecting Changes in 3D Structure of a Scene from Multi-view Images Captured by a Vehicle-mounted Camera", CVPR2013 (Poster, Acceptance Rate: 25.2%)

[3] Takayuki Okatani, Ken Sakurada, Jun Yanagisawa, Daiki Tetsuka, Koichiro Deguchi, "Creating Multi-Viewpoint Panoramas of Streets with Sparsely Located Buildings", FSR2012

[4] Ken Sakurada, Takayuki Okatani, Kris M. Kitani, " Hybrid Macro-Micro Visual Analysis for City-Scale State Estimation ", MIRU2014 (Oral, Acceptance Rate: 29.4%), July 2014

[5] Ken Sakurada, Takayuki Okatani, Koichiro Deguchi, " Detecting Changes in 3D Structure of a Scene from Multi-view Images Captured by a Vehicle-mounted Camera ", MIRU2013 (Invited talk), July 2013

[6] Ken Sakurada, Takayuki Okatani, Koichiro Deguchi, "Three-dimensional Change Detection of a Scene from a Few Multi-view Images", MIRU2012, July 2012 (In Japanese)

[7] Takayuki Okatani, Ken Sakurada, Jun Yanagisawa, Daiki Tetsuka, Shinya Sato, Koichiro Deguchi, "Spatiotemporal City Modeling of Areas Damaged by the Great East Japan Earthquake", MIRU2012, July 2012 (In Japanese)

[8] Koichiro Deguchi, Takayuki Okatani, Satoshi Saga, Ken Sakurada, Jun Yanagisawa, Daiki Tetsuka, Shinya Sato, Katsushi Ikeuchi, Shintaro Ono, Takeshi Oishi, Kagesawa Masataka, Yoshihiro Sato, Atsuhiko Banno, Tetsuya Kakuta, Wang Zhipeng, "Spatiotemporal Image Archive of Damage and Recovery Process in Areas Damaged by the Great East Japan Earthquake", MIRU2012, July 2012 (In Japanese)

[9] Ken Sakurada, Jun Yanagisawa, Takayuki Okatani, Koichiro Deguchi, "Detecting Temporal Changes of a Large-scale Space from Images Acquired by a Car-mounted Camera", CVIM Research Seminar, March 2012 (In Japanese)

[10] Jun Yanagisawa, Daiki Tetsuka, Ken Sakurada, Takayuki Okatani, Koichiro Deguchi, "Creation of Multi-Viewpoint Panoramas for Sparse Streets with a Small Number of Buildings", CVIM Research Seminar, March 2012 (In Japanese)