Distortion-Free Image Mosaicing for Tunnel Inspection Based on Robust Cylindrical Surface Estimation through Structure from Motion

Krisada Chaiyasarn[1]; Tae-Kyun Kim[2]; Fabio Viola[3]; Roberto Cipolla[4]; and Kenichi Soga[5]

Abstract: Visual inspection, although labor-intensive, costly, and inaccurate, is a common practice used in the condition assessment of underground tunnels to ensure safety and serviceability. This paper presents a system that- can construct a mosaic image of a tunnel surface with little distortion, allowing a large area of tunnels to be visualized, and enabling tunnel inspection to be carried out off-line. The system removes distortion by a robust estimation of a tunnel surface through structure from motion (SFM), which can create a 3D point cloud of the tunnel surface from uncalibrated images. SFM enables the mosaicing system to cope with images with a general camera motion, in contrast to standard mosaicing software that can cope only with a strict camera motion. The estimation of the tunnel surface is further improved by support vector machine (SVM), which is used to remove noise in the point cloud. Some curvatures are observed in the mosaics when an inaccurate surface is used for mosaicing, whereas the mosaics from a surface estimated using the proposed method are almost distortion-free, preserving all physical attributes, e.g., line parallelism and straightness, which is important for tunnel inspection. **DOI: 10.1061/(ASCE)CP.1943-5487.0000516.**

Author keywords: Visual inspection; Mosaicing; Structure from motion; Support vector machine; Robust surface estimation.

Introduction

Much of the underground infrastructure in many major cities around the world shows signs of deterioration due to aging and may later cause problems in structural integrity. Maintenance works are regularly carried out and visual inspection is a common practice, which is used in detecting and monitoring anomalies (e.g., cracks, spalling, and staining) in the infrastructure. A number of systems have been proposed to facilitate the laborious process of visual inspection using cost-effective digital photographs and videos. These systems can be categorized into the following themes: (1) automation system for fast data acquisition, (2) automatic crack detection systems, and (3) organization of a large amount of photographic data. This paper presents a system that focuses on Theme 3 so that a thorough examination of the growing number of photographs can be achieved more effectively. A typical solution is to apply image mosaicing, a technique that stitches individual images together to form a larger image, hence reducing the number of images required for inspection. Additionally, mosaicing also enhances visualization for inspectors by providing a larger field of view in greater detail than if a single image is used to capture an entire scene. As exemplified in Zhu et al. (2010), commercial software was applied to stitch the images of columns under bridges to enhance data visualization, and Lee et al. (2013) applied image rectification from a known tunnel geometry before applying image stitching.

Mosaicing typically requires manually identified control points, especially in remote sensing, in which mosaicing is applied to detect landscape changes from satellite images (Zitová and Flusser 2003). With improvement in feature detection and matching algorithms (Szeliski 2010) such as scale invariant feature transform (SIFT) (Lowe 1999), mosaicing can now be carried out automatically, and the manual control points are no longer required. Nevertheless, the application of automatic mosaicing software is still limited because mosaicing systems are also dependent upon motion models, which can cause distortion in a mosaic if an unsuitable motion model is used (Szeliski 2006).

One motion model for home and industrial applications, widely adopted owing to its simplicity [M. Brown and R. Szeliski, "Multi-image feature matching using multi-scale oriented patches," U.S. Patent No. 10/833 (2004)], is the homography-based model, which relates images of the same planar surface by homography, assuming a pinhole camera model. This model works well when either an interested scene is far away (i.e., a plane at infinity) or a camera undergoes pure rotation (because the homography is further simplified and becomes more stable). A mosaic from this model typically suffers from perspective distortion, especially in the region of the mosaic that is further away from a central projection (Szeliski 2006). Fig. 1(a), which was created from the tunnel data set presented in this paper, shows that the mosaic suffers from distortion and misalignment as the image data set violates the assumptions of the homography-based model.

The motion models, which are specific for mosaicing a quadric surface, may be applied to the images of a tunnel. For example, cylindrical projection mosaicing works by transforming image coordinates from Euclidean space to cylindrical or spherical coordinates. This allows the warped images to be related by a translation

[1]Ph.D. Student, Dept. of Engineering, Univ. of Cambridge, Cambridge CP2 1PZ, U.K. (corresponding author). E-mail: k.chaiyasarn@gmail.com

[2]Lecturer in Computer Vision and Learning, Dept. of Electrical and Electronic Engineering, Imperial College London, London SW7 2AZ, U.K.

[3]Lecturer, Faculty of Engineering, Thammasat Univ., Pathumthani 12120, Thailand; formerly, Ph.D. Student, Dept. of Engineering, Univ. of Cambridge, Cambridge CP2 1PZ, U.K.

[4]Professor of Information Engineering, Dept. of Engineering, Univ. of Cambridge, Cambridge CP2 1PZ, U.K.

[5]Professor of Civil Engineering, Dept. of Engineering, Univ. of Cambridge, Cambridge CP2 1PZ, U.K.

Note. This manuscript was submitted on October 29, 2014; approved on May 14, 2015; published online on August 18, 2015. Discussion period open until January 18, 2016; separate discussions must be submitted for individual papers. This paper is part of the ***Journal of Computing in Civil Engineering***, © ASCE, ISSN 0887-3801/04015045(13)/$25.00.

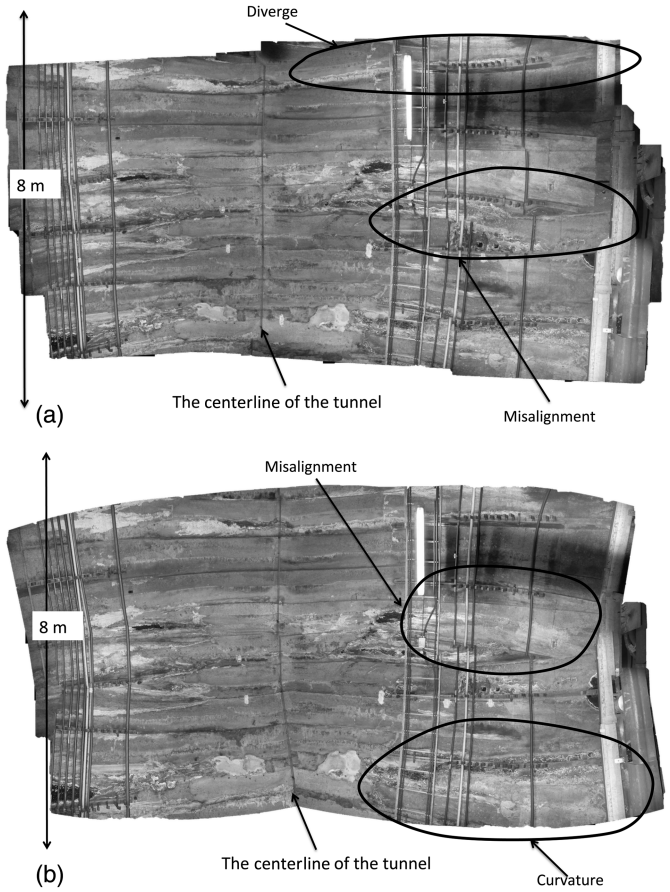

Fig. 1. Distortions observed in the mosaics created from typical motion models from the Prague data set: (a) homography-based mosaicing; (b) cylindrical projection mosaicing

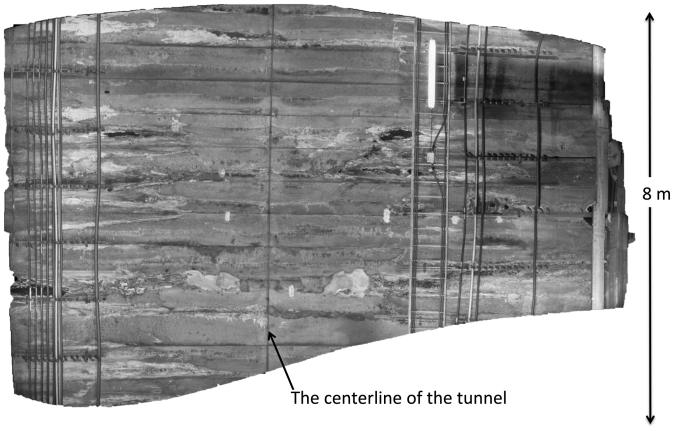

Fig. 2. Result created from the proposed mosaicing system that preserves all physical attributes, e.g., line parallelism and straightness, which are important for tunnel inspection; the mosaic was constructed from 173 images

motion model, which is simpler and more stable. However, this method only works if the images are all taken with a level camera or with a known tilt angle (Szeliski 2010). Fig. 1(b) shows various distortions introduced by this method. The hierarchical estimation of the motion models, such as Shum and Szeliski (1998) and Can et al. (2002), are also specific for mosaicing the quadric surface, although these models still impose constraints on a camera motion and the type of the scene, which must be a true quadric.

The data sets presented in this paper are obtained from a camera with an arbitrary motion; also, the tunnel surface is not a true quadric, which violates the assumptions of the existing models and causes unwanted image distortion. The mosaicing system proposed in this paper is independent of camera motion and can create an almost distortion-free mosaic. This is achieved by exploiting the properties of developable surfaces (e.g., underground tunnels), which can be flattened onto a plane without distortion (Fig. 2). This mosaic, created from 173 images, preserves all physical attributes, including line parallelism, collinearity, line straightness, and angles.

The mosaicing system presented in this paper removes distortion by recovering the true cylindrical surface of a tunnel through structure from motion (SFM). The SFM system creates a three-dimensional (3D) point cloud of a tunnel from uncalibrated images and then a cylindrical proxy is fitted onto the point cloud by a nonlinear least-squares method. The proxy is used to warp individual images, which are then input into stitching software to create a final distortion-free mosaic. Additionally, support vector machine (SVM) is applied to remove some 3D points from the point cloud because these 3D points are those that lie on protuberant regions such as pipes, pans, and tunnel ridges, and will cause inaccuracy in the cylindrical surface estimation. The mosaics created without SVM still contained curvature distortion, whereas distortion was completely removed when SVM was applied.

Background

Photographic Technology in Visual Inspection

Visual inspection must be carried out on a regular basis by trained engineers to obtain an initial assessment of the structural components. In the United Kingdom, for example, the frequency of inspections ranges between 2 and 6 years (Messervey 2008). For tunnel inspection, the guideline from Federal Highway and Transit Administration (2005) suggests that tunnel owners should establish a frequency for up-close inspections based on the age and condition of the tunnels; this period could be as great as 5 years and, for older tunnels, a much more frequent inspection period may be required, possibly every 2 years. Visual inspection heavily relies on the experience of the inspectors; often, inspection is required to be carried out in many inaccessible areas, such as narrow ventilation shafts, which is impractical. Visual inspection usually results in an inspection report, which contains sketches of the defects found in problematic areas (Fig. 3).

Photographic technologies, such as images and videos, are commonly used to aid visual inspection because they provide rich information, such as texture, color, and 3D cues, which are useful when assessing the conditions of structures. As an example, leakage, which is identified as a major problem for tunnels (Delatte et al. 2003; Mallett 1994), can only be detected through photographic techniques using color information. In recent years, visual inspection incorporates techniques to allow fast data acquisition, such as the use of a self-navigated robot to acquire videos of sites under inspection, for example, FATA automation (FATA 2011) that mounts infrared cameras onto a vehicle to allow the acquisition of data through remote control; sewer inspection systems (Makar 1999; Costello et al. 2007), and Lawson and Pretlove (1998) applied a stereo vision system and augmented reality to navigate a robot. Idoux (2005) presents a system called ATLAS 70, which is a complete sensor and software package to acquire data and post-process remotely by a trained inspector. Yu et al. (2007) presents an

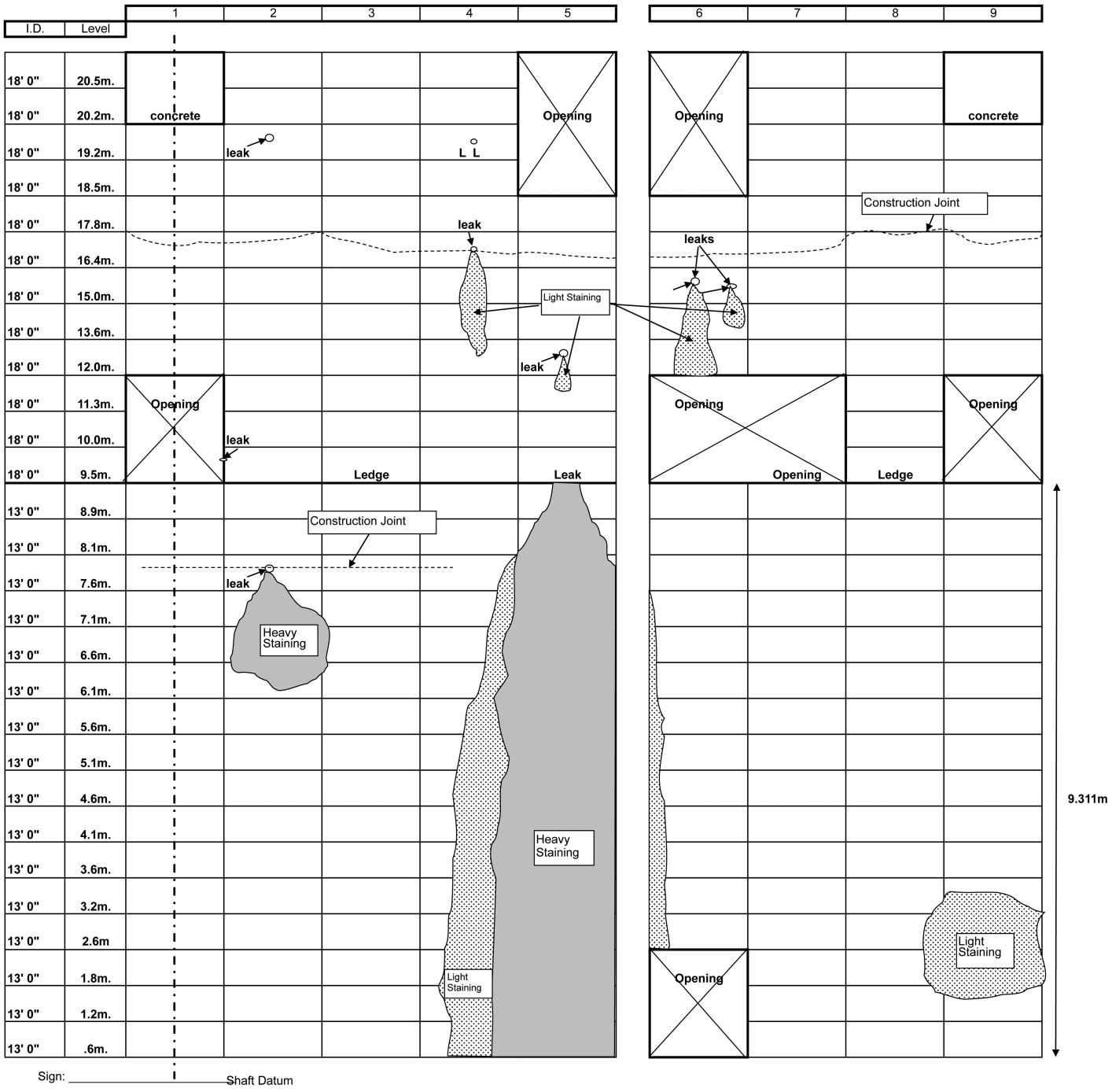

Fig. 3. Example of an inspection report containing sketches of anomalies on tunnel surface

integrated system for automatic tunnel inspection, which includes both image acquisition from a mobile robot and a crack detection algorithm.

The captured videos and photographs can then be automatically processed to detect defects (Sinha and Fieguth 2006). To this end, many crack detection systems have been developed. Algorithms for crack detection generally involve a preprocessing step and a crack identification step (Guo et al. 2009). The preprocessing step applies image processing techniques to extract potential crack features, such as edges. Miyamoto et al. (2007) computed the difference in intensity between each pixel and the average intensity of each row in an image matrix; a pixel that differs considerably from the average is deemed to belong to a crack. Fujita et al. (2006) applied a line filter using the Hessian matrix and a threshold is applied to extract the crack regions. Ukai (2000) developed a system to model cracks, which can be characterized by eight quantities, including area and Ferets occupancy rate. Yamaguchi and Hashimoto (2010) modeled cracks based on the percolation model, which is a physical model based on liquid permeation. Paar et al. (2006) proposed a crack detection algorithm based on the line tracing algorithm. The identification step usually applies crack modeling and/or pattern recognition techniques to determine whether the extracted features belong to crack regions. Zhu and Brilakis (2010) proposed an algorithm for detecting concrete columns based on texture using artificial neural networks. Liu et al. (2002) applied a support vector machine classifier to classify if crack features

appear in an image patch, which is preprocessed to extract potential crack features based on intensity. Abdelqader et al. (2006) applied a principal component principles (PCA) algorithm, which was used to reduce the dimensions of feature vectors based on eigenvalues, to extract cracks from concrete bridge decks.

Once cracks have been identified, more semantic information can be obtained so that a subsequent step, such as a repair regime, can be planned. To this end, algorithms that classify cracks and indicate the states of cracks have been proposed. Kaseko and Ritchie (1993) proposed an artificial neural network system that classifies pavement surface cracking by the type, severity, and extent of cracks detected in video images. Sinha and Fieguth (2006) applied a new neuro-fuzzy classifier to classify extracted features in buried pipe images into different types of cracks and other defects, such as holes on the pipe surface. Kim et al. (2007) proposed a computer-assisted crack diagnosis system for reinforced concrete structures that aids the nonexpert to diagnose the cause of cracks at the level of an expert in the general inspection of structures. Lim et al. (2005) proposed a system to monitor the change of cracks from multitemporal images. The system is based on a two-dimensional (2D) projective transformation to accurately extract the size of cracks, which is then monitored in the images as the cracks propagate. Chen and Hutchinson (2010) proposed a framework for concrete surface crack monitoring and quantification.

As the image or video data become large, it is essential to organize or, more importantly, be able to visualize the data for a thorough examination. In this area of research, many systems have been proposed to improve visualization of a large amount of data. A typical solution is to use image mosaicing, which stitches images together to form a larger image to provide a larger field of view. Zhu et al. (2010) applied commercial software to stitch the images of columns under bridges so that an entire column can be visualized in greater detail than if a single image is used. Song et al. (2005) created a stitched image of construction sites from a series of images taken by a robotic camera, and the stitched image is used to record the progress of the construction. Jahanshahi et al. (2011) created stitched images of structural systems from a tilt-and-pan camera to detect missing parts in time-lapse photos. Image registration techniques were applied to rectify images so that all images are in the same coordinate frame and can then be compared. Stent et al. (2013) and Chaiyasarn et al. (2009) created mosaicing of tunnels to form the basis for change detection for tunnel images.

Mosaicing Technology

The visual presentation and quality of a mosaic depends largely on an accurate motion model. The motion models can be broadly grouped into single viewpoint and multi-viewpoint. A single-viewpoint panorama is the most common type in the commercial stitching software. This type of panorama is created with the assumption that all images share the same center of projection (Haenselmann et al. 2009). This can be achieved by rotating a camera around its optical center (Agarwala et al. 2006). One model is based on planar projective transformation, in which all images are aligned using homographies. The final panorama is created by warping other images based on a homography to a chosen reference image. One drawback of this method is that the panorama appears to diverge toward the edge, and only a small number of images can be combined without causing severe distortion (Peleg et al. 2000). To avoid this problem, first warp each image to cylindrical or spherical coordinates and then relate each image using a translational model (Chen and Klette 1999). This model, however, distorts a panorama by making all the straight lines appear curved. This model also requires a level camera for the cylindrical projection or a level camera with known tilt angles for the spherical projection. The drawback of the single-viewpoint panorama is that it does not work well with images with general camera motion.

A multi-viewpoint panoramic image consists of patches that do not have a common projection center, but are taken from changing viewpoints (Haenselmann et al. 2009). Strip panoramas, which are created from a translating camera, are one of the methods used for creating a multi-viewpoint panorama. There are many variants of strip panoramas, such as pushbroom panoramas (Gupta and Hartley 1997) and adaptive manifolds and x-slits images (Zomet et al. 2003). The strip panoramas are suitable for visualizing images or videos with long sequences, such as long street scenes. A common method used with strip panoramas is to sample vertical strips from each image or video frame and then warp the strips according to the chosen transformation model to stitch the strips onto a final image plane. The strip panoramas exhibit some degree of distortion. Only objects at a certain depth from the camera plane are shown with a correct aspect ratio; objects that are further away may appear horizontally stretched, and closer objects appear squashed. These panoramas are usually created from videos, which are generally of lower quality than images. In addition, acquiring suitable video data can be challenging.

Carroll and Seitz (2009) create multi-viewpoint panoramas using developable surfaces, which can be unwrapped onto a plane without distortion. The result is a mosaic without any perspective distortion. The system estimates a camera pose for each frame and then, with known camera poses, an inverse projection of each frame can be performed onto a developable surface. This system, however, requires tracking pixels in the pose estimation algorithm; hence, the camera path has to be smoothed. Therefore, it is only suitable for cameras with forward motion. Forward motion is unsuitable for scenes taken inside tunnels owing to the large size of the tunnels because only a small portion of the tunnel surface for each panel is visible in the images as shown in Fig. 4. This is inadequate for inspection purposes.

Agarwala et al. (2006) developed a system that creates multi-viewpoint panoramas from digital images taken from a standard single-lens reflex (SLR) or a fish-eye lens camera. The system uses structure from motion, which explicitly recovers a sparse 3D point cloud of a scene and camera poses. The texture from each camera is projected onto a dominant plane, which corresponds to the facade of the buildings along the streets, to create the final panoramas. In this work, a dominant plane is chosen automatically by principal component analysis (PCA). PCA is able to find a direction with the largest variance of data. In this work, PCA is applied to a 3D point cloud, and the direction of the largest variance in the point cloud is used to form the parameters for a plane. The PCA method works when a street scene is not curved. If it is slightly curved, a dominant plane is selected manually. For a scene with a large curvature (such as a tunnel), a geometry proxy (such as a cylindrical surface) can be used as a dominant plane, as demonstrated in this paper. In Agarwala et al. (2006), when a distance between a scene and a camera is large, only a slight distortion in the mosaic is observed for objects that do not lie on a dominant plane, such as cars; and therefore, standard stitching and blending algorithms are sufficient to cope with these objects.

The method developed in this study is similar to Agarwala et al. (2006), which relied on an SFM system to create multi-viewpoint panoramas. The SFM system allows stitching for a general camera motion, which eases the process of image acquisition and also enables images to be stitched in both the radial and longitudinal directions for tunnels. Because the tunnel surface is a developable surface, the panoramas can be distortion free. The primary challenge of the current work, also faced by Agarwala et al. (2006),

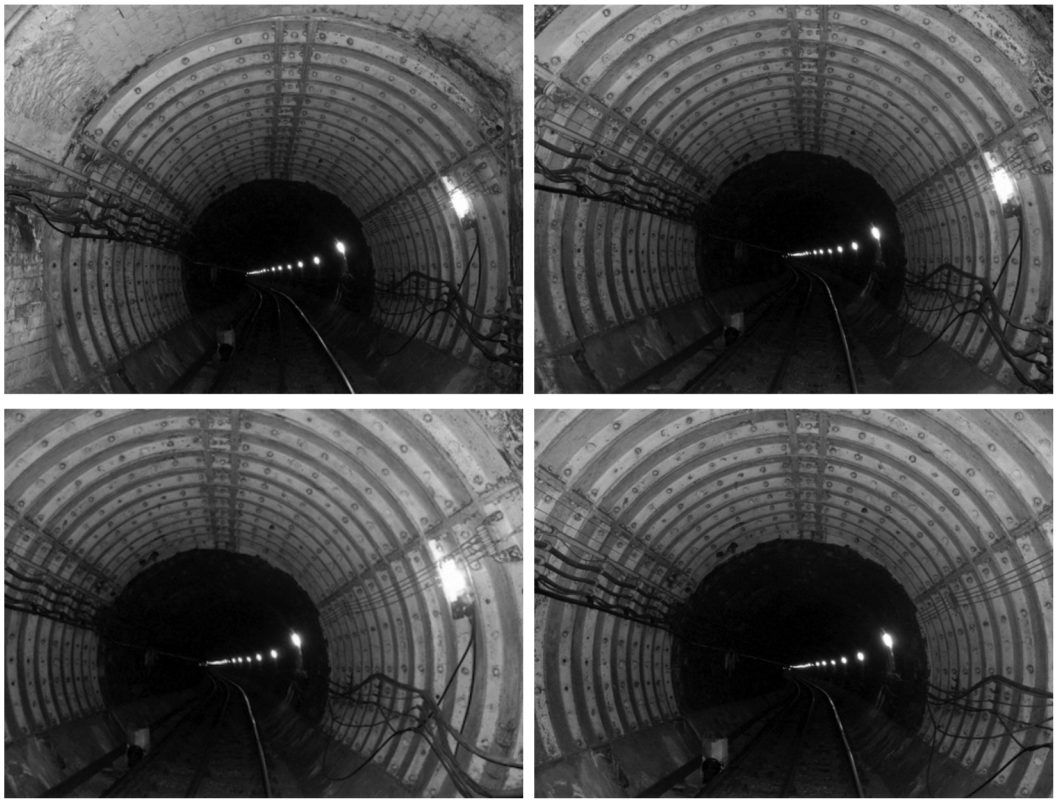

Fig. 4. Pictures of tunnels from the Aldwych site obtained by forward motion; only small parts of the tunnel surface are visible in these images, making them unsuitable for inspection

is how to extract a geometry proxy for mosaicing. Agarwala et al. (2006) identifed a dominant plane automatically by the use of PCA. In this paper, the estimation of the tunnel proxy is done manually by initializing parameters for the cylindrical surface, and then the parameters are optimized by the least-squares method. Additionally, support vector machine classification is applied to remove the noise from the 3D point cloud, i.e., points lying on the protuberant regions causing inaccuracy in the surface estimation step.

Methodology

Fig. 5 shows an outline of the proposed system. The first component is image acquisition, in which the images of underground scenes are systematically obtained based on the method described previously. To mosaic images in the proposed system, tunnel images are first input into the structure from motion (SFM) module to recover the 3D point cloud of a tunnel model and camera poses. SFM is the process of finding the 3D structure of an object from multiple uncalibrated images taken at various locations. The SFM program used in this paper is called Bundler, implemented by Snavely et al. (2006) as explained previously.

Next, a surface proxy is fitted to the point cloud in the surface estimation module by nonlinear least-squares optimization. In this system, the tunnel surface is estimated using either the manual input obtained from users or the input obtained from SVM classifiers. SVM is applied to classify 3D points into those that either lie or do not lie on an actual tunnel surface, because nonsurface points can corrupt the optimizer to converge to an incorrect surface. The SVM code from the statistical pattern recognition toolbox (SPRT)

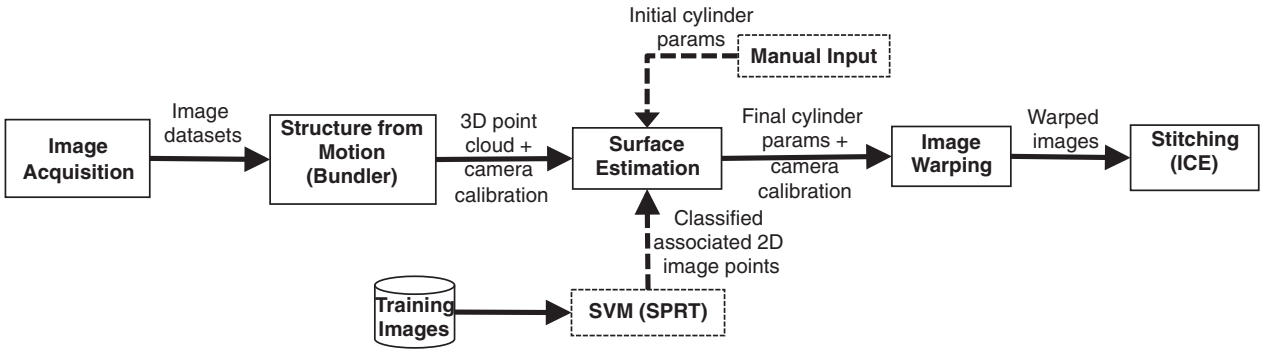

Fig. 5. System outline

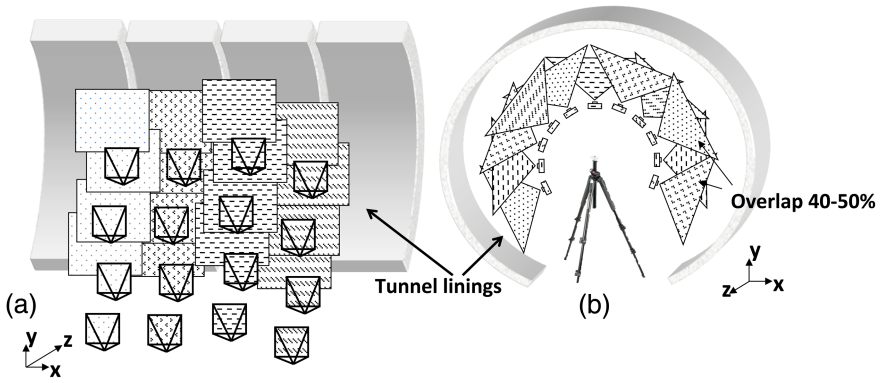

Fig. 6. Configuration of image acquisition system: (a) side view where the tunnel linings are in front of the camera; (b) view looking into the tunnel

implemented by Vojtěch and Václav (2004) is used. The nonsurface points can badly skew an estimated surface and create distortion in a final mosaic. The estimated surface is then used to warp input images in the image warping module; and finally, the warped images are composited at the stitching stage by a commercial program called *ICE* (Microsoft 2011).

Image Acquisition

It is important to ensure that sufficient 3D information is captured to make the SFM module successful. The image acquisition procedure is designed to ensure that the image data sets have (1) sufficient 3D information; (2) the full coverage of the tunnel linings; and (3) sufficient overlaps between images. There are three main pieces of equipment: a standard digital camera, a tripod, and a portable spotlight.

As a rule of thumb, there should be some degree of overlapping for every three images; therefore, an overlap of more than 50% for every two images was chosen (Fig. 6). The procedure is as follows. For each tunnel ring, a camera mounted on a tripod is rotated to cover approximately 270° [Fig. 6(b)], and the overlap between the images should be at least 50%. The camera is then moved along the tunnel by approximately half the width of the ring [Fig. 6(a)]. The amount of distance moved along the tunnel is flexible, depending on the width of the tunnel lining. The camera is rotated to cover the entire ring of a tunnel lining, but the rail tracks at the bottom section of the ring are omitted to allow a more accurate 3D model of the tunnel to be reconstructed.

Each image is taken using a self-timer function (e.g., 3–10 s can be used) to release the camera shutter to avoid blurring. The F-stop and shutter speed were set to the P mode when the flash was not used, and the Auto mode was used when the flash was used. These modes are standard in modern digital cameras. The P mode lets the camera calculate both the shutter speed and aperture to obtain automatic image exposure, whereas the Auto mode allows the camera to alter all of the settings (e.g., flash, exposure compensation). An external light source is used when the flash is turned off (i.e., in the P mode). A camera is locked into position on a tripod before images are taken to avoid vibration, which can cause image blurring.

Table 1. Summary of Data Sets

Data sets	Number of images	Length (m)	Number of images/length (m^{-1})
Aldwych	23	3	8
Prague	173	8	21
Bond Street	123	7	17

Data Sets

There are three data sets presented in this paper, referred to as the Aldwych data set (taken at Aldwych tunnels, London), the Prague data set (taken at Prague Metro, Czech Republic), and the Bond Street data set (taken at Bond Street tunnels, London). Table 1 summarizes the number of images and the length of a tunnel covered in each data set (the number of images/length is rounded to the nearest integer).

Structure from Motion

Structure from motion refers to a process of recovering a 3D point cloud and camera poses from uncalibrated or calibrated cameras. Fig. 7 shows an example of the typical pipeline of a structure from motion system; the reader is referred to Snavely et al. (2006) for full details. The algorithm starts with the SIFT matching algorithm to extract features and their associated descriptor vectors for each image (Lowe 1999). This feature is invariant to scale and robust to affine transformation. The 128-dimensional descriptor vectors are then pairwise matched by the *k*-nearest neighbor method for each pair of images. Incorrect matches are filtered out by the geometrical consistency constraint by the RANdom sampling consensus (RANSAC matching) algorithm. The fundamental matrix $\mathbf{F}$ obtained by the RANSAC is used to obtain the essential matrix, $\mathbf{E} = \mathbf{K}_2^T \mathbf{F} \mathbf{K}_1$, where $\mathbf{K}$ is the camera calibration matrix. The essential matrix is decomposed by singular value decomposition (SVD) to produce the relative camera rotation and translation matrixes, hence the projection matrixes, between each camera pair. The ambiguity of the projection matrixes is solved by the cheirality constraint (Nister 2004). To register all cameras in a global scale, the scale ambiguity is solved in a tripletwise fashion. The scale ratio of each triplet is found by the SVD method. Once all camera projection matrixes in the global frame are found, 3D points can be recovered from any pair of the projection matrixes by triangulation in the triangulation step. The DLT algorithm is used in the triangulation step (Hartley and Zisserman 2000). All 3D points, the camera rotation and translation matrixes, and the calibration matrix are then used to initialize the bundle adjustment (BA) algorithm. This algorithm can improve the reconstruction accuracy by global registration in which all parameters (i.e., 3D points and all camera

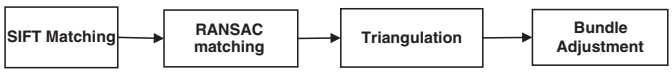

Fig. 7. Example of the pipeline of a structure from motion system

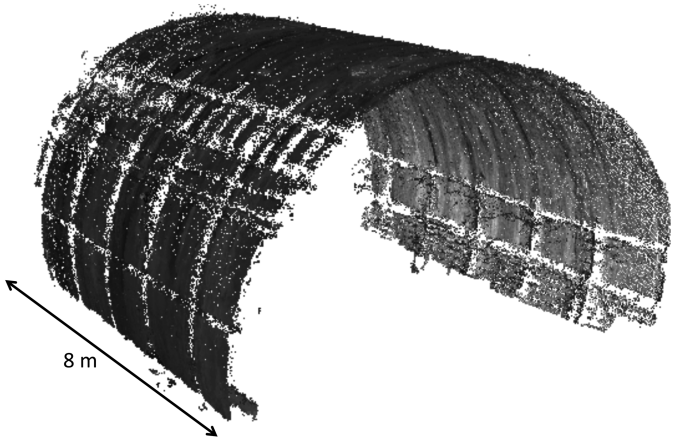

Fig. 8. Example of a 3D point cloud obtained from the Prague data set

projection matrixes) are optimized together by minimizing a suitable cost function.

Bundler, which is a noncommercial state-of-the-art SFM software package developed by Snavely et al. (2006), was applied to obtain the 3D point clouds for each of the data sets. Notably, Bundler estimates camera calibration parameters **K** from the EXIF tags of images, and then all camera parameters (i.e., rotation, translation, and **K**) together with initialized 3D points are optimized in the sparse bundle adjustment algorithm (Lourakis and Argyros 2004). Hence, the method described in this paper relies on Bundler to give a 3D point cloud, **K**, **R**, and **T** from uncalibrated input images. Fig. 8 shows an example of a 3D point cloud obtained from the Prague data set. The software successfully reconstructed the model and created 139,665 3D points; all 173 images are registered in the model. For the spatial accuracy of the SFM system, the 3D point cloud was compared with the laser imaging detection and ranging (LiDAR) data by an iterative closest point (ICP) algorithm. The point cloud from the SFM system can give the accuracy between 120 and 200 mm compared with 4.5–8 mm for the LiDAR in the 5 m range for the Prague data set. However, the SFM system had an accuracy of 25–40 mm in the 50 m range as shown in Harwin and Lucieer (2012). The spatial accuracy of the SFM system can be improved further with higher image resolution. The experiment is performed on an Intel Core i3 3.06 GHz with 4 GB of memory.

Cylindrical Surface Estimation

After a sparse 3D point cloud is created from the SFM system, the point cloud is fed into the surface estimation module to fit a geometric proxy onto the point cloud. The cylindrical surface is estimated using the nonlinear least-square optimization of a cylindrical surface as explained subsequently.

Let C be a cylinder defined by a center $(x_0,y_0,z_0)^T$, a unit directional vector $(a,b,c)^T$, and a radius r. A point $\mathbf{p} \in \mathcal{R}^3$ that lies on the cylinder is then defined as

$$\left\| \mathbf{p} - \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} \right\|^2 = \left\{ \left(\mathbf{p} - \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} \right) \cdot \begin{bmatrix} a \\ b \\ c \end{bmatrix} \right\}^2 + r^2 \qquad (1)$$

Let $\mathbf{x}_i = (x_i,y_i,z_i)^T$ be a 3D coordinate; and the minimum distance (i.e., the shortest distance between a point to a surface) between the point $\mathbf{x}$ and the cylinder C is then defined as $d_i = \min\|\mathbf{x}_i - \mathbf{p}\|$, where $\mathbf{p} \in C$. To fit a cylinder to a point cloud $\mathbf{X} = \{\mathbf{x}_i, \ldots, \mathbf{x}_n\}$, the following cost function is minimized:

$$\min \sum_{i=1}^{n} w_i d_i^2 \qquad (2)$$

where $w_i \in [0,1]$ = weight of a point $\mathbf{x}_i$. The preceding cost function is minimized using the nonlinear least-squares method by the Gauss-Newton algorithm, in which the implementation from Eurometros (2011) is used.

It is assumed that the geometry of a tunnel can be modeled by a cylinder. This assumption is valid because the deformation of a tunnel is negligible in comparison with the size of a tunnel.

The surface estimation requires an initial input for r_0, a directional vector $\mathbf{a}_0$, and a cylinder center $\mathbf{x}_0$; and these inputs are required from the users.

As shown in Eq. (2), w is the weight given to each point in the point cloud. Generally, not all points lie on a cylindrical surface because some points are found on tunnel ridges, cables, and other protuberant regions. These points can cause inaccuracy in the surface estimation if all points are equally weighted, especial in cast iron linings, where many protuberant regions are found, especially in the Aldwych data set. The next section explains how the weights for each point using an SVM classifier are obtained to improve accuracy in the surface estimation for the Aldwych data set. For the Prague and Bond Street data sets, the SVM is not required because they are concrete linings, and the effect caused by the protuberant regions is negligible; hence, w is assigned as 1 for all points in these two data sets.

Estimation by Support Vector Machine

In this section, the estimation of the weight w for each 3D point using SVM is explained. Because not all of the 3D points are on the cylinder surface, the weight is proposed as

$$w_i = \Sigma_j p(\mathbf{x}_{ji}) \qquad (3)$$

The weight w implies the probability that the 3D point belongs to the surface class. In addition, $\mathbf{x}_{ji}$ is the scale invariant feature transform (SIFT) vector of the jth image point of ith 3D point, which is reprojected by the projection matrix as denoted by

$$\mathbf{x}_{ji} = \mathcal{G}(\mathbf{P}_{ji}\mathbf{X}_i) \qquad (4)$$

where $\mathcal{G}$ and $\mathbf{P}$ denote the SIFT function and the camera projection matrix, respectively. To retrieve the weights for each $\mathbf{X}_i$, the probabilities are summed over the associated image points $\mathbf{x}_{ji}$ as shown in Eq. (3). The probability p is given as

$$p(\mathbf{x}_{ji}) = 1/\{1 + \exp[-f(\mathbf{x}_{ji})]\} \qquad (5)$$

where the function f is the SVM classifier in Eq. (6). Fig. 9 illustrates the conceptual diagram of how SVM is used in obtaining the weights for each 3D point.

Learning Support Vector Machine

The SVM classifier is applied to discriminate the tunnel surface points from the nonsurface points. The image patches of the two classes exhibit quite distinctive appearances (Fig. 10); hence, they are separable. The interest points detected by the SIFT algorithm on or near the protuberant regions are collected as the nonsurface class and others as the surface class [Figs. 11(a and b)]. The image patches of the two classes exhibit quite distinctive appearances (Fig. 10); hence, they are separable. The training data set is composed of $\{\mathbf{x}_{ji},y_k\}$, where $\mathbf{x}_{ji}$ is the 128-dimension SIFT descriptor vectors described in Eq. (4), and $y_k \in \{-1,+1\}$ is the class label of the point $\mathbf{x}_{ji}$. The SVM classifier (Bishop 2006) that optimally separates the positive class from the negative class is learned as

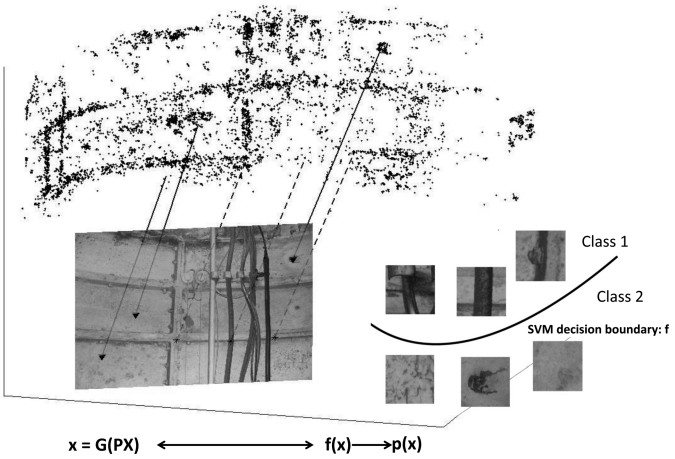

Fig. 9. Weights obtained from the SVM classifier are placed on 3D points for accurate surface estimation; 3D points are projected onto images by projection matrixes and then classified by the SVM classifier by their associated image points

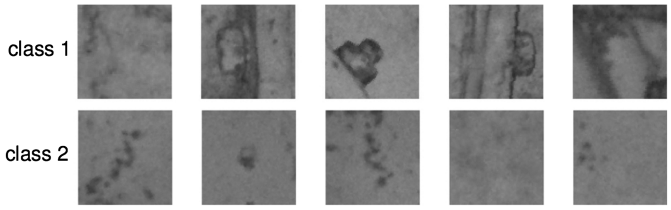

Fig. 10. Example of two classes: interest points on the protuberant regions are Class 1; points on the surface are Class 2

$$f(\mathbf{x}_{ji}) = \sum_{k=1}^{N_s} \alpha_k y_k K(\mathbf{s}_k, \mathbf{x}_{ji}) + b \tag{6}$$

where $\{\mathbf{s}_1, \ldots, \mathbf{s}_{N_s}\}$ = set of support vectors; α_k = weight for the kth support vector; and b = bias. The kernel function used in the system is $K(\mathbf{z}_i, \mathbf{z}_j) = e^{-\|\mathbf{z}_i - \mathbf{z}_j\|^2/2\sigma^2}$, where σ is the kernel variance. The variance σ and the penalty constants are set using a validation set.

For the SVM classification, the data set contains 1,369 training points (i.e., training set) and 635 test points (i.e., validation set) from 19 labeled images [e.g., Fig. 11(a)]. The ROC curves for the different variance and the constants are shown in Fig. 12. The best performance is achieved when the kernel variance and the penalty constants are set as $\sigma = 0.5$ and $[C_1, C_2] = [2,1]$, respectively, and these values are used to classify all other points in the data set.

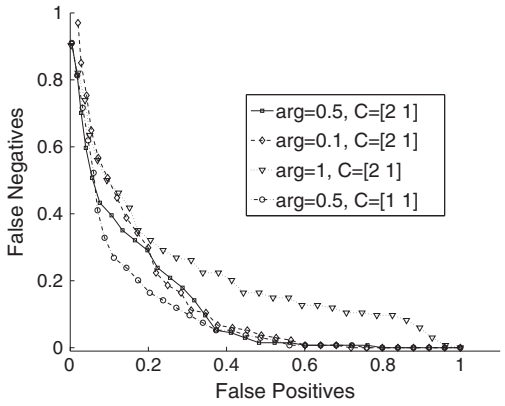

Fig. 12. ROC curves for the different kernel parameters and the penalty constants

Image Warping and Stitching

Warping

A surface proxy is estimated to fit onto a 3D reconstructed point cloud. This proxy is a developable surface whose properties allow the surface to be flattened onto a plane without distortion, such as stretching or compressing. These special properties enable a mosaic from tunnel images to be created with little or no distortion. A developable surface is considered as a special case of a ruled surface—a surface that can be swept out by moving a line in space. The ruled surface R carries a one-parameter family of straight lines L (Peternell 2004), known as generators. If all points on each generator, $\mathbf{x} \in L$, have the same tangent plane, this surface is said to be a developable surface. Some examples of ruled surfaces are a plane, a cylinder, and a cone.

Given the constraints on the image collection process, cameras are located inside a developable surface, and each ray intersects the surface only at a single visible point. The intersection defines, for each image, a one-to-one mapping between image samples and

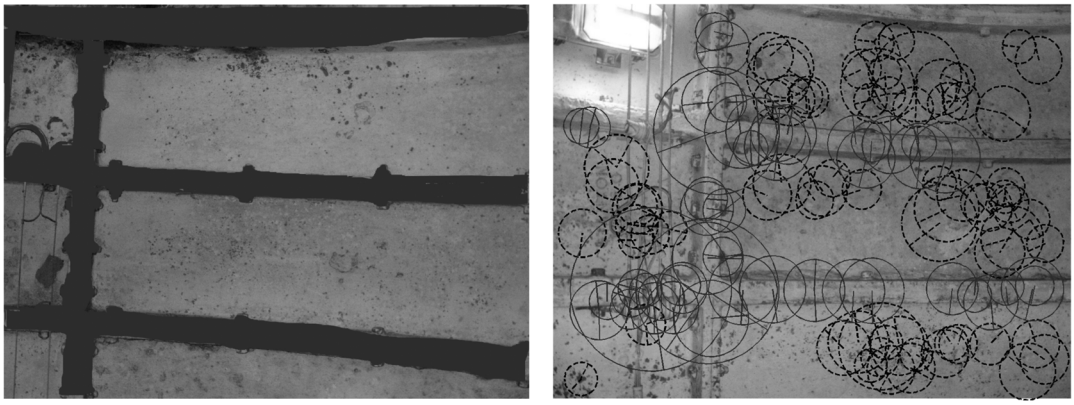

Fig. 11. (a) Example of a labeled image for SVM training and testing, the nonsurface region is colored in dark grey; (b) the points in solid grey lines and in dotted black lines are estimated as the nonsurface class and the surface class, respectively

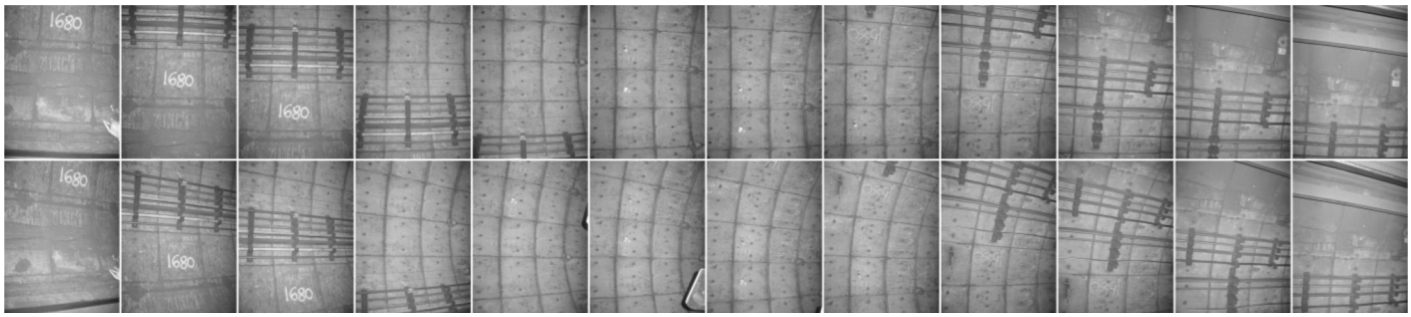

Fig. 13. Examples of the input images from the Bond Street data set before warping

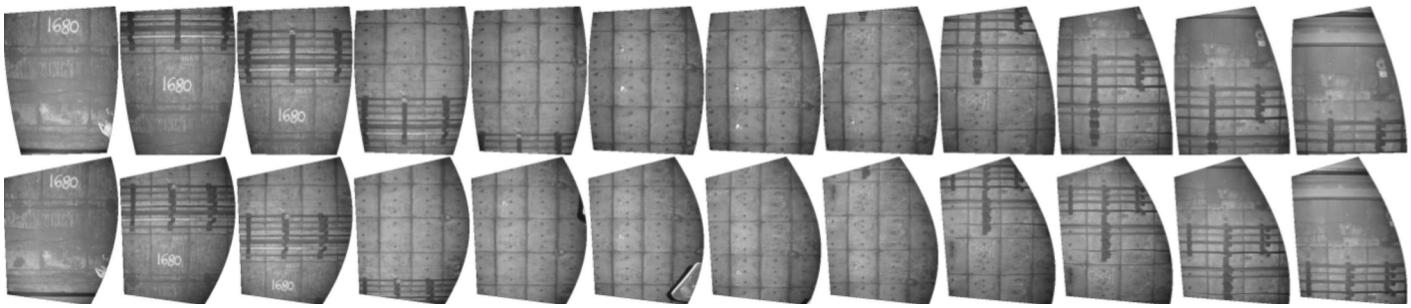

Fig. 14. Input images from the Bond Street data set after warping

points on the surface. In other words, each image is projected onto the surface. These facts allow us to define a warping that produces the flattened versions of the input images as shown in Fig. 13 (i.e., before warping) and Fig. 14 (i.e., after warping).

Compositing

The final stage of mosaicing involves different processes, including selecting a compositing surface, selecting pixel contributions to the final composite, and blending these pixels to minimize visible seams, blur, and ghosting. Most commercial stitching software packages already contain the aforementioned algorithms to perform a final composite. However, these packages only work well with images related by homographies, and they do not work well with the data sets presented in this paper. In the proposed system, input images can be warped more accurately because better camera calibration parameters and a more accurate estimated tunnel surface are applied to remove distortion before feeding into stitching software. This allows a translational model, which is a simpler and more stable model, to be applied to relate the warped images in the stitching software to obtain a final result. The stitching software by Microsoft *Image Composite Editor* (Microsoft 2011) is applied in this paper.

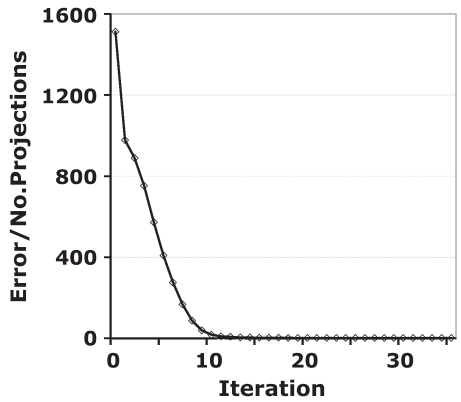

Fig. 15. Reprojection error of the initialized point cloud is converged when the bundle adjustment is applied

Results and Discussion

Surface Estimation Results

For the Aldwych data set, SVM is applied to improve the results in the surface estimation step. The convergence graph from the BA algorithm (Fig. 15) quantitatively shows a significant improvement in global registration in 3D points as shown in Figs. 16(a and b) as the cost function converges. Figs. 16(a and b) show the 3D point cloud of the Aldwych tunnel with and without applying the BA algorithm, respectively. After the BA algorithm is applied, the tunnel linings can be observed more clearly, and camera poses are also improved. The convergence graph from the BA algorithm (Fig. 15) quantitatively shows a significant improvement in global registration as the cost function converges. Fig. 16(c) shows the result of the 3D point cloud that is classified by SVM as rectangular markers for the surface points and the nonsurface points as dot markers. From Fig. 16(d), the estimated surface shown in light grey is without SVM (all points are used for estimation, i.e., the weight w_i for each point set to 1), whereas the dark grey surface is estimated by using the weights from the SVM classifier. As shown next, the light grey surface creates a curvature in the final mosaic.

Final Mosaic Results

In Fig. 17(a), the mosaic is created from the point cloud before the BA algorithm is applied. The mosaic shows a number of misalignments as expected because the camera poses were not yet optimized

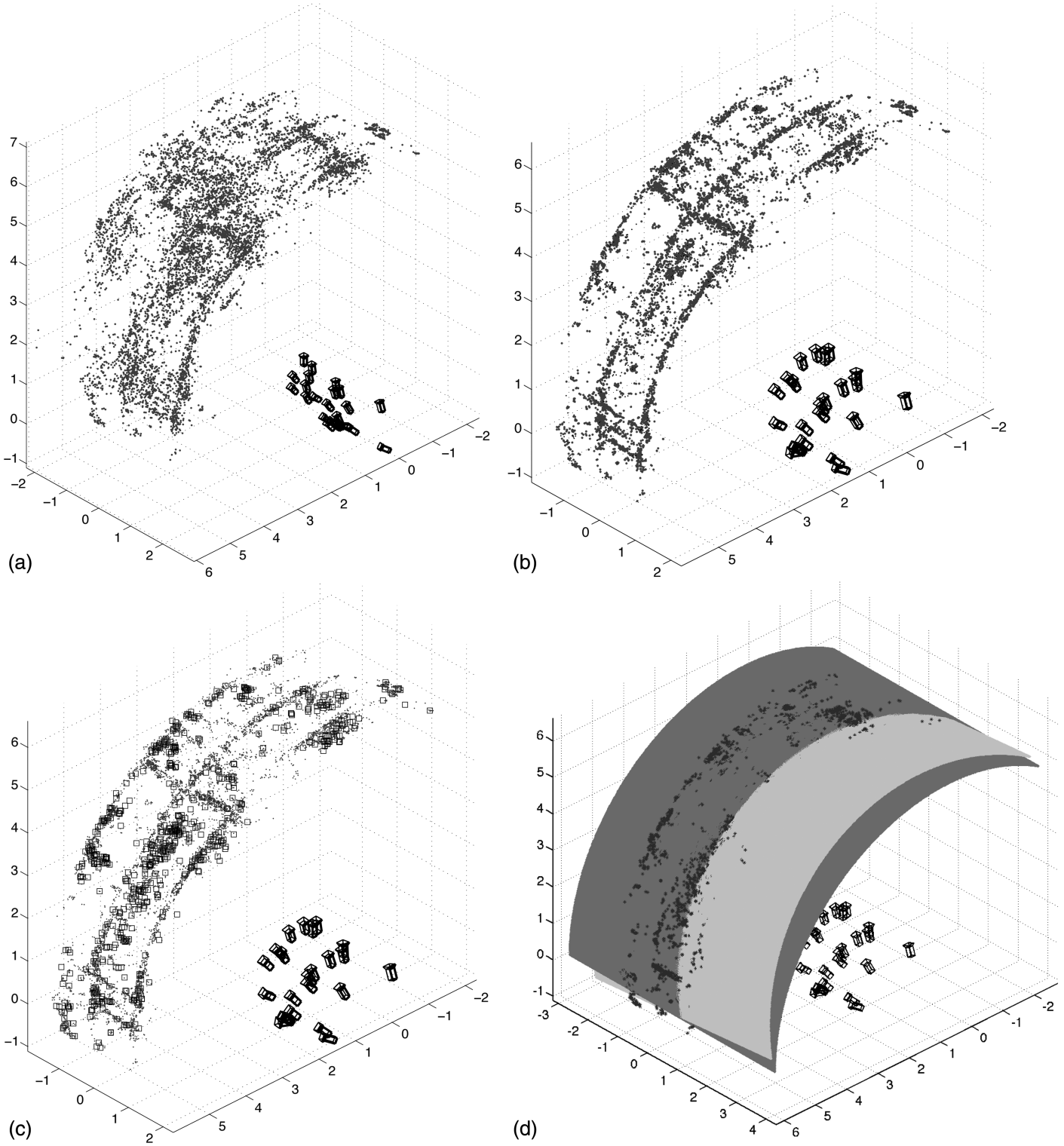

Fig. 16. Process of robust surface estimation: (a) reconstructed model before applying the bundle adjustment algorithm; (b) model after applying the BA algorithm; (c) classification of 3D points by SVM, surface points as rectangular markers and nonsurface points as dot markers; (d) estimation of surface, the light grey surface represents estimation with all 3D points having the weights equal to 1, and the dark grey surface is a more accurate estimation by using the weights obtained from SVM

by the BA algorithm. Fig. 17(b) shows a typical result obtained from commercial stitching software. The mosaic shows a significant degree of perspective distortion because an unsuitable camera motion is used by the software. From Fig. 17(c), the mosaic is created from the surface without SVM [i.e., the light grey surface from Fig. 16(d)], whereas Fig. 17(d) is constructed from a more accurate surface [i.e., the light grey surface from Fig. 16(d)]. The quality of the final mosaics are shown to depend primarily on the accuracy of estimated surfaces. From Fig. 17(c), the mosaic results in curvature, whereas the mosaic from Fig. 17(d) preserves all physical

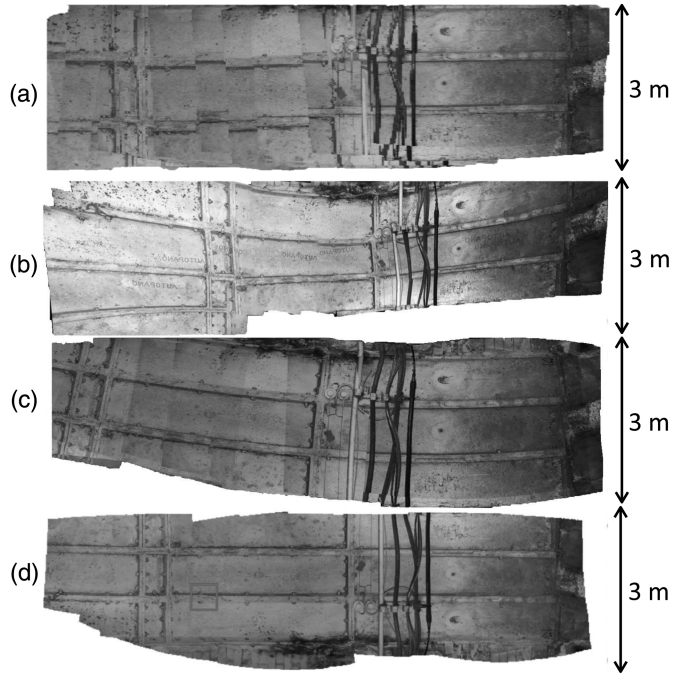

Fig. 17. Results obtained from the Aldwych data set: (a) mosaic before the BA algorithm is applied; (b) typical result from commercial stitching software; (c) result obtained from the experiment when an incorrect surface proxy is used for mosaicing; (d) final result obtained from an accurate surface proxy

geometries, such as line straightness, parallelism, and a 90° angle between horizon and vertical lines. The curvature is caused by the skewness of the estimated surface, which is induced by the nonsurface points. In contrast to the mosaic produced from the commercial software shown in Fig. 17(b), which exhibits strong perspective distortion, the mosaic generated from this study is almost distortion free; this improvement in the mosaic quality is desirable and suitable for a tunnel inspection report.

Fig. 18 shows the result from the Bond Street data set. This result demonstrates that the proposed system allows images to be stitched in the longitudinal direction for a longer distance. This means that the mosaic can be created to cover a much larger section of a tunnel. The ability to stitch in the longitudinal direction is not possible in standard stitching software. Notably, misalignment can be observed toward the bottom right corner of the mosaic, but not on the left corner. This is caused by the the rail track section of the tunnel lying further away from the true tunnel surface, which violates the assumption made in the proposed system. The proposed system works well for the areas that are close to the tunnel geometry. The further away the real scene is from the estimated surface, the more violated the assumption becomes. As a result, greater inaccuracy can be observed in the bottom right corner as induced by the significant part of the rail track section of the tunnel.

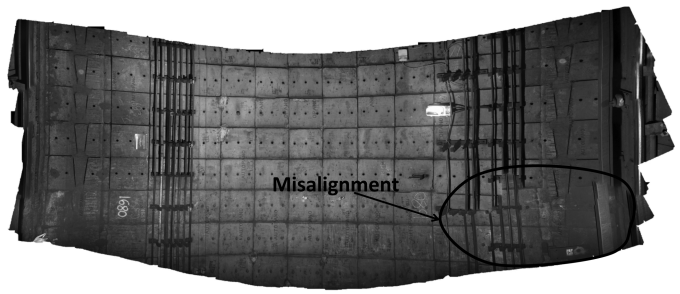

Fig. 18. Result obtained from Bond Street Station is created from 123 images and covers 7 m in the longitudinal direction

The result from the Prague data set is shown in Fig. 2. This result covers almost 8 m of the tunnel surface in the longitudinal direction and approximately 270° in the radial direction. A small amount of misalignment can be observed in this result as labeled in Fig. 2 for the same reason explained previously for the Bond street data set. Nevertheless, this result preserves all physical geometries and provides a larger field of view for the tunnel surface, which is desirable for tunnel inspection.

The method presented in this paper is suitable for cylindrical tunnels. For other noncylindrical tunnels, such as an arch shape, a different geometry is required in the surface estimation step. As an example, an arch tunnel, which is semicircular in shape, can be estimated as a combination of planes and cylinders. This is a nontrivial problem, and further study is required to automatically extract primitive geometries from a point cloud. Nevertheless, cylindrical tunnels are commonly found in most manmade tunnels; hence, the proposed system is still applicable in a majority of tunnels.

Conclusion and Future Work

A system for stitching images of a tunnel surface with little distortion is presented. The quality of the mosaics depends primarily on the accuracy of the surface estimation of a tunnel. With the proposed mosaicing system, SFM allows images to be stitched in both the radial and longitudinal directions by relaxing the constraints on camera motion; also, the point cloud allows a tunnel surface to be estimated more accurately. The system exploits the properties of a developable surface to create a distortion-free mosaic for cylindrical tunnels. A support vector machine is also applied to remove the protuberant points, which are noise, from the point cloud in the Aldwych data set. SVM helps to remove the curvature by excluding the 3D points that do not belong to the true tunnel surface.

The mosaic images from the presented results have a number of advantages. One direct impact is the ease of creating an inspection report. The mosaics allow the inspectors to have a wider field of view of a tunnel surface, making it easier for them to carry out inspections and analyses from the mosaics. The inspectors can use mosaic images to guide the localization of anomalies when creating an inspection report. The automatic localization of cracks or anomalies can subsequently be performed.

Future plans are to conduct further validation on more data sets, with the hope of testing the system in a real scenario with human participants to gain further insights in the effectiveness of the system in identifying anomalies in real structures. Furthermore, the prototype of the system, including the software and the apparatus for acquiring images (e.g., 20–30 m of the tunnel length), is currently being developed so that it can be practically adopted in the tunnel inspection procedure.

Acknowledgments

The research is funded by the Engineering and Physical Sciences Research Council [EP/E003338/1 Micro-Measurement and Monitoring System for Ageing Underground Infrastructures (Underground M3)].

References

Abdelqader, I., Pashaierad, S., Abudayyeh, O., and Yehia, S. (2006). "PCA-based algorithm for unsupervised bridge crack detection." *Adv. Eng. Software*, 37(12), 771–778.

Agarwala, A., Agrawala, M., Cohen, M., Salesin, D., and Szeliski, R. (2006). "Photographing long scenes with multi-viewpoint panoramas." *ACM Trans. Graphics*, 25(3), 853–861.

Bishop, C. (2006). *Pattern recognition and machine learning*, Vol. 4, Springer, New York.

Can, A., Stewart, C., Roysam, B., and Tanenbaum, H. (2002). "A feature-based, robust, hierarchical algorithm for registering pairs of images of the curved human." *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(3), 347–364.

Carroll, R., and Seitz, S. (2009). "Rectified surface mosaics." *Int. J. Comput. Vision*, 85(3), 307–315.

Chaiyasarn, K., Kim, T., Viola, F., Cipolla, R., and Soga, K. (2009). "Image mosaicing via quadric surface estimation with priors for tunnel inspection." *16th Institute of Electrical and Electronics Engineers (IEEE) Int. Conf. on Image Processing*.

Chen, C., and Klette, R. (1999). "Image stitching—Comparisons and new techniques." *CAIP '99: Proc., 8th Int. Conf. on Computer Analysis and Images and Patterns*, Springer, Berlin, 615–622.

Chen, Z., and Hutchinson, T. (2010). "Image-based framework for concrete surface crack monitoring and quantification." *Adv. Civ. Eng.*, 2010, 18.

Costello, S., Chapman, D., Rogers, C., and Metje, N. (2007). "Underground asset location and condition assessment technologies." *Tunnelling Underground Space Technol.*, 22(5-6), 524–542.

Delatte, N., et al. (2003). "The application of nondestructive evaluation to subway tunnel systems." *Transportation Research Record 1845*, Transportation Research Board, Washington, DC, 127–135.

Eurometros. (2011). "The least squares geometric elements library." ⟨http://www.eurometros.org/gen_report.php?category=implementations&pkey=28&subform=yes⟩ (Aug. 22, 2009).

FATA. (2011). "Tunnel inspection vehicles." http://www.fataautomation.co.uk/case-studies/tunnel-inspection-vehicle.html (Aug. 19, 2011).

Federal Highway and Transit Administration. (2005). "Highway and rail transit tunnel inspection manual." U.S. Dept. of Transportation, Washington, DC.

Fujita, Y., Mitani, Y., and Hamamoto, Y. (2006). "A method for crack detection on a concrete structure." *ICPR'2006: IEEE 18th Int. Conf. on Pattern Recognition*, Vol. 3, New York, 901–904.

Guo, W., Soibelman, L., and Garrett, J. (2009). "Automated defect detection for sewer pipeline inspection and condition assessment." *Autom. Constr.*, 18(5), 587–596.

Gupta, R., and Hartley, R. (1997). "Linear pushbroom cameras." *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(9), 963–975.

Haenselmann, T., Busse, M., Kopf, S., King, T., and Effelsberg, W. (2009). "Multi-perspective panoramic imaging." *Image Vision Comput.*, 27(4), 391–401.

Hartley, R., and Zisserman, A. (2000). *Multiple view geometry in computer vision*, Cambridge University Press, Cambridge, U.K.

Harwin, S., and Lucieer, A. (2012). "Assessing the accuracy of georeferenced point clouds produced via multi-view stereopsis from unmanned aerial vehicle (UAV) imagery." *Remote Sens.*, 4(6), 1573–1599.

Idoux, M. (2005). "Multisensor system for tunnel inspection." *Proc., Society of Photo-Optical Instrumentation Engineers (SPIE)*, Vol. 5640, SPIE, Bellingham, WA, 303.

Jahanshahi, M., Masri, S., and Sukhatme, G. (2011). "Multi-image stitching and scene reconstruction for evaluating defect evolution in structures." *Struct. Health Monit.*, 10(6), 643–657.

Kaseko, M., and Ritchie, S. (1993). "A neural network-based methodology for pavement crack detection and classification." *Transp. Res. Part C Emerging Technol.*, 1(4), 275–291.

Kim, Y., Kim, C., and Hong, G. (2007). "Fuzzy set based crack diagnosis system for reinforced concrete structures." *Comput. Struct.*, 85(23–24), 1828–1844.

Lawson, S. W., and Pretlove, J. R. G. (1998). "Augmented reality for underground pipe inspection and maintenance." *Proc., SPIE Int. Symp. on Intelligent Systems and Advanced Manufacturing, Telema- Nipulator and Telepresence Technologies V*, Boston.

Lee, C.-H., Chiu, Y.-C., Wang, T.-T., and Huang, T.-H. (2013). "Application and validation of simple image-mosaic technology for interpreting cracks on tunnel lining." *Tunnelling Underground Space Technol.*, 34, 61–72.

Lim, Y., Kim, G., Yun, K., and Sohn, H. (2005). "Monitoring crack changes in concrete structures." *Comput-Aided Civ. Infrastruct. Eng.*, 20(1), 52–61.

Liu, Z., Azmin, S., Ohashi, T., and Ejima, T. (2002). "A tunnel crack detection and classification systems based on image processing." *Society of Photo-Optical Instrumentation Engineers (SPIE) Conf. Series*, Vol. 4664, Machine Vision Applications in Industrial Inspection X, San Jose, CA, 145–152.

Lourakis, M., and Argyros, A. (2004). "The design and implementation of a generic sparse bundle adjustment software." ⟨http://citeseer.ist.psu.edu⟩ (Aug. 1, 2009).

Lowe, D. G. (1999). "Object recognition from local scale-invariant features." *Proc., 7th IEEE Int. Conf. on Computer Vision, 1999*, Vol. 2, New York, 1150–1157.

Makar, J. (1999). "Diagnostic techniques for sewer systems." *J. Infrastruct. Syst.*, 10.1061/(ASCE)1076-0342(1999)5:2(69), 69–78.

Mallett, G. (1994). *Repair of concrete bridges*, Thomas Telford, London.

Messervey, T. (2008). "Integration of structural health monitoring into the design, assessment, and management of civil infrastructure." Ph.D. thesis, Lehigh Univ., Bethlehem, PA.

Microsoft. (2011). "Microsoft image composite editor." ⟨http://research.microsoft.com/en-us/um/redmond/groups/ivm/ICE/⟩ (Aug. 1, 2009).

Miyamoto, A., Konno, M., and Bruhwiler, E. (2007). "Automatic crack recognition system for concrete structures using image processing approach." *Asian J. Inf. Technol.*, 6(5), 553–561.

Nister, D. (2004). "An efficient solution to the five-point relative pose problem." *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(6), 756–770.

Paar, G., Kontrus, H., and Sidla, O. (2006). "Optical crack following on tunnel surfaces." *Proc., SPIE Int. Society for Optical Engineering*, Vol. 6382, Boston.

Peleg, S., Rousso, B., Rav-Acha, A., and Zomet, A. (2000). "Mosaicing on adaptive manifolds." *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(10), 1144–1154.

Peternell, M. (2004). "Developable surface fitting to point clouds." *Comput. Aided Geom. Des.*, 21(8), 785–803.

Shum, H.-Y., and Szeliski, R. (1998). "Construction and refinement of panoramic mosaics with global and local alignment." *6th Int. Conf. on Computer Vision, 1998*, IEEE, New York, 953–956.

Sinha, S., and Fieguth, P. (2006). "Automated detection of cracks in buried concrete pipe images." *Autom. Constr.*, 15(1), 58–72.

Snavely, N., Seitz, S., and Szeliski, R. (2006). "Photo tourism: exploring photo collections in 3d." *ACM Trans. Graphics*, 25(3), 835–846.

Song, D., Hu, Q., Qin, N., and Goldberg, K. (2005). "Automating inspection and documentation of remote building construction using a robotic camera." *IEEE Int. Conf. on Automation Science and Engineering*, New York, 172–177.

Stent, S., Gherardi, R., Stenger, B., Soga, K., and Cipolla, R. (2013). "An image-based system for change detection on tunnel linings." *International Association for Pattern Recognition (IAPR) Int. Conf. on Machine Vision Applications*, Springer, London.

Szeliski, R. (2006). "Image alignment and stitching: A tutorial." *Found. Trend. Comput. Graphics Vision*, 2(1), 1–104.

Szeliski, R. (2010). *Computer vision: Algorithms and applications*, Springer, New York.

Ukai, M. (2000). "Development of image processing technique for detection of tunnel wall deformation using continuously scanned image." *Q. Rep. RTRI*, 41(3), 120–126.

Vojtěch, F., and Václav, H. (2004). "Statistical pattern recognition toolbox for Matlab." *Res. Rep. CTU–CMP–2004–08*, Center for Machine Perception, Czech Technical Univ., Prague, Czech Republic.

Yamaguchi, T., and Hashimoto, S. (2010). "Fast crack detection method for large-size concrete surface images using percolation-based image processing." *Mach. Vision Appl.*, 21(5), 797–809.

Yu, S., Jang, J., and Han, C. (2007). "Auto inspection system using a mobile robot for detecting concrete cracks in a tunnel." *Autom. Constr.*, 16(3), 255–261.

Zhu, Z., and Brilakis, I. (2010). "Machine vision-based concrete surface quality assessment." *J. Constr. Eng. Manage.*, 10.1061/(ASCE)CO.1943-7862.0000126, 210–218.

Zhu, Z., German, S., and Brilakis, I. (2010). "Detection of large-scale concrete columns for automated bridge inspection." *Autom. Constr.*, 19(8), 1047–1055.

Zitová, B., and Flusser, J. (2003). "Image registration methods: A survey." *Image Vision Comput.*, 21(11), 977–1000.

Zomet, A., Feldman, D., Peleg, S., and Weinshall, D. (2003). "Mosaicing new views: The crossed-slits projection." *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(6), 741–754.