ROBUST REGISTRATION FOR VIDEO MOSAICING USING CAMERA MOTION PROPERTIES

Thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science (by Research)
in
Computer Science

by

Pulkit Parikh
200405017
pulkitparikh@research.iiit.ac.in

International Institute of Information Technology
Hyderabad, India
January 2006

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled "ROBUST REGISTRATION FOR VIDEO MOSAICING USING CAMERA MOTION PROPERTIES" by Pulkit Parikh, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date	Adviser: Dr. C. V. Jawahar

Dedicated to Jyoti - my beloved mother

Acknowledgements

The first person that comes to mind, as I begin to write this section, is my advisor Dr C. V. Jawahar, who assisted me right throughout my stint with computer vision. A great deal of my advancement from a novice who had been exposed to precious little beyond rudimentary image processing to a battle-hardened vision researcher ought to be credited to him. He has also been kind enough to watch over me, career-wise as well as personally. I also thank Dr P. J. Narayanan and Dr Anoop Namboodiri for helping me widen my horizons in my domain. While these professors and my research center CVIT introduced me to computer vision at large, it was through my RAship commitments with ADE, DRDO, Bangalore that I got acquainted with video mosaicing, the topic around which my thesis revolves.

I have always stressed on discussions and brainstorming sessions as much as reading. To that end and with technical help, my lab mates Paresh and Visesh made my life much easier by sharing their rich know-how. I would like to single out Visesh for his uncanny talent to be able to remark (usefully) on any topic under the sun. It would be unjust if I did not mention the names of Ranjith, Sunil Mohan Ranta, Piyush Bhargava, Praveen, Balu, Santosh, Anurag, Pradhee, Vandana, Gopal, Jagmohan and other CVITians whose names I am unable to retrieve at the minute.

As much as I acknowledge the role played by CVIT in shaping up my career, life outside work, at times, can have an equally big impact on one's work. I was blessed to have friends like Lini, Satya and Satish, ensuring that my tenure in IIIT was more or less smooth sailing. Another mortal whose support I simply could not have done without is Renjini. The striking quality of these individuals was that you only had to drop one hint and they would be right by your side, right away. Thanks aplenty to each one of them. This section would be incomplete without a mention to my M. Tech. friends Saurabh Khanna and Mainak.

Ironically, the mention of my family, whom I treasure the most, comes as late as in the concluding paragraph. Foremost comes my mother who I maximally owe what I have achieved, for the way she has always put me ahead of herself. My family also comprises my father who has stood by me unconditionally all these years and my little sister Ruchi who manages to maximize the likelihood of me being cheerful, often with her sheer presence. I also take this opportunity to express my gratitude towards my uncle cum childhood role model Mr N. T. Shah, primarily responsible for my early inclination towards science and technology. To wind it up, thanks a heap to Archit and Parth for being there round the clock, putting up merrily with those discussions, which ran wild more times than not, ranging from cricket and career to life and philosophy.

Abstract

In recent years, video mosaicing has emerged as an important problem in the domain of computer vision and computer graphics. Video mosaics find applications in many popular arenas including video compression, virtual environments and panoramic photography. Mosaicing is the process of generating a single, large, integrated image by combining the visual clues from multiple images. The composite image called mosaic provides high field of view without compromising the image resolution. When the input images are the frames of a video, the process is called video mosaicing. The process of mosaicing primarily consists of two key components: image registration and image compositing. The focus of this work is on image registration - the process of estimating a transformation that relates the frames of the video. In addition to mosaicing, registration has a wide range of other applications in ortho-rectification, scene transfer, video in-painting, etc. We employ homography, the general 2D transformation, for image registration. Typically, homography estimation is done from a set of matched feature points, extracted from the frames of the input video.

Video mosaicing has been viewed traditionally as a problem of registering (and stitching) only successive video frames. While some of the recent global alignment approaches make use of the information stemming from non-consecutive pairs of video frames, the registration of each frame pair is typically done independently. No real emphasis has been laid on how the imaging process and the camera motion relate the pair-wise homographies. Therefore, accurate registration and in turn, mosaicing, especially, in face of poor feature point correspondence, still remains a challenging task. For example, mosaicing a desert video wherein the frames have minimal texture to provide feature points (e.g. corners) or mosaicing in presence of repetitive texture leading to several mismatched points is highly error-prone.

It is known that the camera that captures the video frames to be mosaiced, does not undergo arbitrary motion. The trajectory of the camera is almost always smooth. Some examples where such motion is seen are aerial videos taken by the camera mounted on an aircraft, videos captured by robots, automated vehicles, etc. We propose an approach which exploits this smoothness constraint to refine outlier homographies i.e., homographies that are detected to be erroneous. In many scenarios, especially in machine-controlled environments, the camera motion, apart from being continuous, also follows a model (say a linear motion model), giving us much more concrete and precise information. In this thesis, we derive relationships between homographies in a video sequence, under many practical scenarios, for various camera motion models. In other words, we *translate the camera motion model into a global homography model whose parameters can characterize homographies between every pair of frames. We present a generic methodology which uses this global homography model for accurate registration.*

Above mentioned derivations and algorithms have been implemented, verified and tested, on various datasets. The analysis and comparative results demonstrate significant improvement over the existing approaches, in terms of accuracy and robustness. Superior quality mosaics have been developed using our algorithms, in presence of many commonly observed issues like texture-less frames and frames containing repetitive texture, with applicability to indoor (e.g., mosaicing a map) as well as outdoor (e.g., representing an aerial video as a mosaic) settings. For quantitative evaluation of mosaics, we have devised a novel, computationally efficient quality measure. The quantitative results are completely in tune with the visual perception, further testifying the superiority of our approach. In a nutshell, the premise that the properties of the camera motion can serve as an important additional cue for improved video mosaicing, has been empirically validated.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The field of view (FOV) of the commercial camera is much smaller than that of humans. Thus, large objects can not be seen completely in one image. FOV is directly related to the size of the CCD array and inversely related to the focal length. Thus, one way of increasing the FOV is to adapt the hardware. However, very often, that is neither possible nor desirable as it may degrade the picture quality or resolution. This is where image mosaicing comes in. Mosaicing is a computer-vision-based approach to attain high field of view without compromising the image quality (Fig. 1.1). It is the process of generating a single, large, integrated image by combining the visual clues from multiple images [4, 5].

1.1 Mosaicing: An Overview

Mosaicing of images have been in practice since long before the age of digital computers. Shortly after the photographic process was developed in 1839, the use of photographs was demonstrated on topographical mapping [6]. Images acquired from hill-tops or balloons were manually pieced together. After the development of airplane technology (1903), aerophotography became an exciting new field. The limited flying heights of the early airplanes and the need for large photo-maps, forced imaging experts to construct mosaic images from overlapping photographs. This was initially done by manually mosaicing images which were acquired by calibrated equipment. The need for mosaicing continued to increase later in history as satellites started sending pictures back to earth for further processing and visualization. Improvements in computer technology became a natural motivation to develop computational techniques and to solve related problems.

The construction of mosaic images and the use of such images on several computer vision/graphics applications have been active areas of research in recent years. There have been a variety of new additions to the classic applications mentioned above that primarily aim to enhance image resolution and field of view. Image-based rendering [7] has become a major focus of attention combining two complementary fields: computer vision and computer graphics [8]. Mosaicing images on smooth

Figure 1.1: Mosaicing - achieving high FOV, keeping picture quality in tact: Three input frames (top), containing parts of the castle, are registered and stitched into a mosaic (bottom), allowing us to view the entire castle in a single image, whose resolution is the same as that of the input images.

surfaces (e.g. cylindirical [9, 10] or spherical [11, 12, 13]) avoides discontinuities that can result from images that are acquired separately. Such immersive environments (with or without synthetic objects) provide the users an improved sense of presence in a virtual scene. A combination of such scenes used as nodes [9, 13] allows the users to navigate through a remote environment. Computer vision methods can be used to generate intermediate views [10, 14] between the nodes. As a reverse problem the 3D structure of scenes can be reconstructed from multiple nodes [13, 15, 16, 17, 18]. Among other major applications of image mosaicing in computer vision are image stabilization [19, 20], resolution enhancement [21, 22], video processing [22] (e.g. video compression [23]. video indexing [24]).

The problem of image mosaicing (Fig. 1.3) is mainly a combination of the following two problems:

1. Image registration: the process of estimating a transformation that maps the points from one image to the other.

2. Image compositing: the process of combining the registered images into a composite image, in such a way that no seams are visible in mosaic, at the image boundaries.

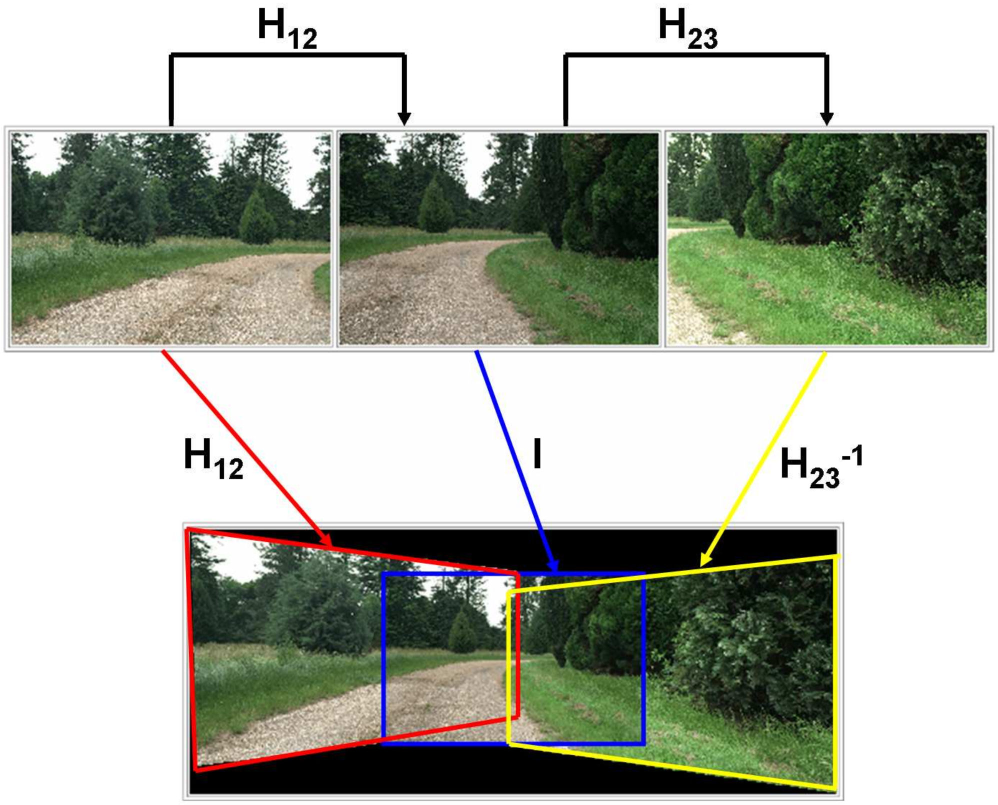

Figure 1.2: Process of mosaicing: Registering and stitching all frames of a video sequence towards the coordinate system of the reference frame (the middle one, in this example)

1.1.1 Types of Mosaicing

Although the fundamental steps - image registration and blending are used in any type of mosaicing, it is worthwhile to discuss some popular kinds of mosaicing. The most commonly used forms are planar and cylindrical mosaicing.

When it is known that the image sequences are related by a (planar) homography, all the images can be *warped* onto a plane and the mosaic can be constructed. For every image, the transformation required to map it to the reference plane is estimated. After the registration is done, images are blended into a seam-less mosaic. An example, taken from [25], is shown in Fig. 1.4 (Sample frames in can be seen in Fig. 1.3).

In cylindrical mosaicing (Fig. 1.5), the translational motion across images is estimated in terms of cylindrical coordinates. In other words, the input image are warped onto a cylinder by mapping the world point P from Cartesian coordinates $[X\ Y\ Z]$ to 2D cylindrical coordinates (θ, r). In Spherical mosaicing (Fig. 1.7), the images are mapped to a sphere as opposed to a cylinder (e.g., [1]).

Another important category of mosaics is formed by the single view-point constraint. Single

Figure 1.3: Sample frames of an aerial video

Figure 1.4: Mosaic of an aerial video, some of whose frames are shown in Fig. 1.3.

Figure 1.5: An example of a cylindrical mosaic

Figure 1.6: An example of a manifold mosaic

view point is a desirable property for a mosaic. It can generate a large FOV image as if seen from a single point (in a perspective projection). One of the input frames is chosen as the reference frame and the whole mosaic appears as though it is a single image captured from the camera position and orientation from which the reference frame was imaged. The mosaic shown in Fig. 1.1 follows the single view-point constraint.

Other types of mosaicing include manifold mosaicing (Fig. 1.6) and stereo mosaicing (Fig. 1.8). Restrictions on the motion of camera used for mosaicing, can be removed using manifold mosaicing. Stereo mosaicing is a harder form of mosaicing wherein one panorama is generated for the left eye, and another panorama is generated for the right eye (e.g., [1]).

In this thesis, we focus on generating mosaics of planar scenes using the single view-point constraint.

1.1.2 Applications of Mosaicing

The most common applications of mosaicing include constructing high resolution images that cover an unlimited field of view using inexpensive equipment, creating immersive environments for effective information exchange through the internet. These applications have been extended towards the creation of completely navigatable virtual environments by creating arbitrary views

Figure 1.7: An example of a spherical mosaic, created by warping images onto a sphere. Courtesy [1]

from a limited number of nodes [10, 14]. The reconstruction of 3D scene structure from multiple nodes [13, 15, 16, 17, 18] has also been another active area of research.

The use of image mosaicing is beginning to make a significant impact in video processing [26]. The complete representation of static scenes resulting from mosaicing video frames in conjunction with an efficient representation for dynamic changes provide a versatile environment for visualizing, efficiently coding, accessing, analyzing information. Besides video compression [23] and indexing [24, 27], this environment is shown to be useful for image stabilization [19, 20] and building high quality images using low-cost imaging equipments [21, 22].

1.1.3 Video Mosaicing

In recent years, mosaicing has gone beyond stitching a set of uncorrelated images and become an integral part of many video applications. Video mosaics find applications in video compression, virtual environments, panoramic photography etc. Video capture technology is now available at low prices, but often with poor field of view. Video mosaicing is emerging as a convenient way to capture images without limitations on field of view. Conventional video mosaicing algorithms extend the image mosaicing procedure to video by registering (computing homographies) and stitching successive frames [4, 5]. Successive frames are employed because they are most likely to be the maximally overlapping ones. However, a video has more useful information than a set of overlapping frames.

Figure 1.8: A stereo (top - left eye, bottom - right eye) mosaic corresponding to the mosaic shown in Fig. 1.7. Courtesy [1]

1.2 Image Registration

Accurate registration of constituent images is the key in obtaining geometrically correct mosaics. A comprehensive survey of conventional registration techniques, for a pair of images, can be found in [28, 29, 30]. The problem of image registration can be defined as finding the (spatial) transformation that relates the pixels of the two input images. The majority of the registration methods consists of the following three steps:

1.2.1 Feature detection

Salient and distinctive objects (closed-boundary regions, edges, contours, line intersections, corners, etc.) are manually or, preferably, automatically detected. For further processing, these features can be represented by their point representatives (centers of gravity, line endings, distinctive points), which are called control points (CPs) in the literature. Formerly, the features were objects manually selected by an expert. During an automation of this registration step, two main approaches to feature detection have been formed:

Area-based methods:

Area-based methods put emphasis more on the feature matching step than on their detection. No features are detected in these approaches so the first step of image registration is almost omitted. More on the area-based methods is discussed in section 1.2.2 where correlation-like methods, Fourier methods and mutual information method are described.

Feature-based methods:

The second approach is based on the extraction of salient structures or features in the images. They should be distinct, spread all over the image and efficiently detectable in both images. In contrast to the area-based methods, the feature-based ones do not work directly with image intensity values. The features represent information on higher level. This property makes feature-based methods suitable for situations when illumination changes are expected or multi-sensor analysis is demanded.

1.2.2 Feature matching

The detected features in the reference and sensed images can be matched by means of the image intensity values in their close neighborhoods or the description of the neighborhood around each feature. Some methods, while looking for the feature correspondence, simultaneously estimate the parameters of mapping functions and thus merge the second and third registration steps. Some of the most popular techniques are the KLT (Kanade-Lucas-Tomasi) feature tracker [31] and SIFT (Scale-invariant feature transform) [32].

Correlation-based methods: Correlation is a measure of similarity, computed for window pairs from the sensed and reference images. Although correlation-based registration was initially used to align mutually translated images only, it can also be successfully applied when limited rotation and scaling are present. There are generalized versions for images having undergone more involved geometric deformations. They compute the correlation for each assumed geometric transformation and pick the transformation which maximizes it. However, this searching can be computationally intensive, given that the gradient behavior of correlation-based similarity measures (e.g., Normalized Cross Correlation) is usually not smooth.

Fourier methods: If an acceleration of the computational speed is needed or if the images were acquired under varying conditions or they are corrupted by frequency-dependent noise, then Fourier methods are preferred rather than the correlation like methods. They exploit the Fourier representation of the images in the frequency domain (e.g., [33]).

Mutual Information (MI) methods: The MI, originating from the information theory, is a measure of statistical dependency between two data sets and it is particularly suitable for registration of images from different modalities. This class of methods is based on the maximization of MI (e.g., [34]). Often the speed up of the registration is implemented, exploiting the coarse-to-fine resolution strategy (the pyramidal approach).

Methods using invariant descriptors: The correspondence of features can be estimated using their description, preferably invariant to the expected image deformation . The description should fulfill several conditions. The most important ones are invariance (the descriptions of the corresponding features from the reference and sensed image have to be the same), uniqueness (two different features should have different descriptions), stability (the description of a feature which is slightly deformed in an unknown manner should be close to the description of the original feature), and independence (if the feature description is a vector, its elements should be functionally independent). Among all the feature descriptors, SIFT [32] has emerged to be the popular, especially in recent years.

1.2.3 Transformation estimation

The final step is computation of the transformation relating the images. In this step, matched feature points are used to find the transformation which best minimizes the reprojection error. Least squared fitting is very popular for that. An over-constrained system of equations can be formed from the correspondences and then can be solved using a linear technique. Note that this transformation estimation step may not always be required explicitly. A good example of that is the Fourier methods. In this thesis, the focus is mainly on this crucial aspect of image registration,

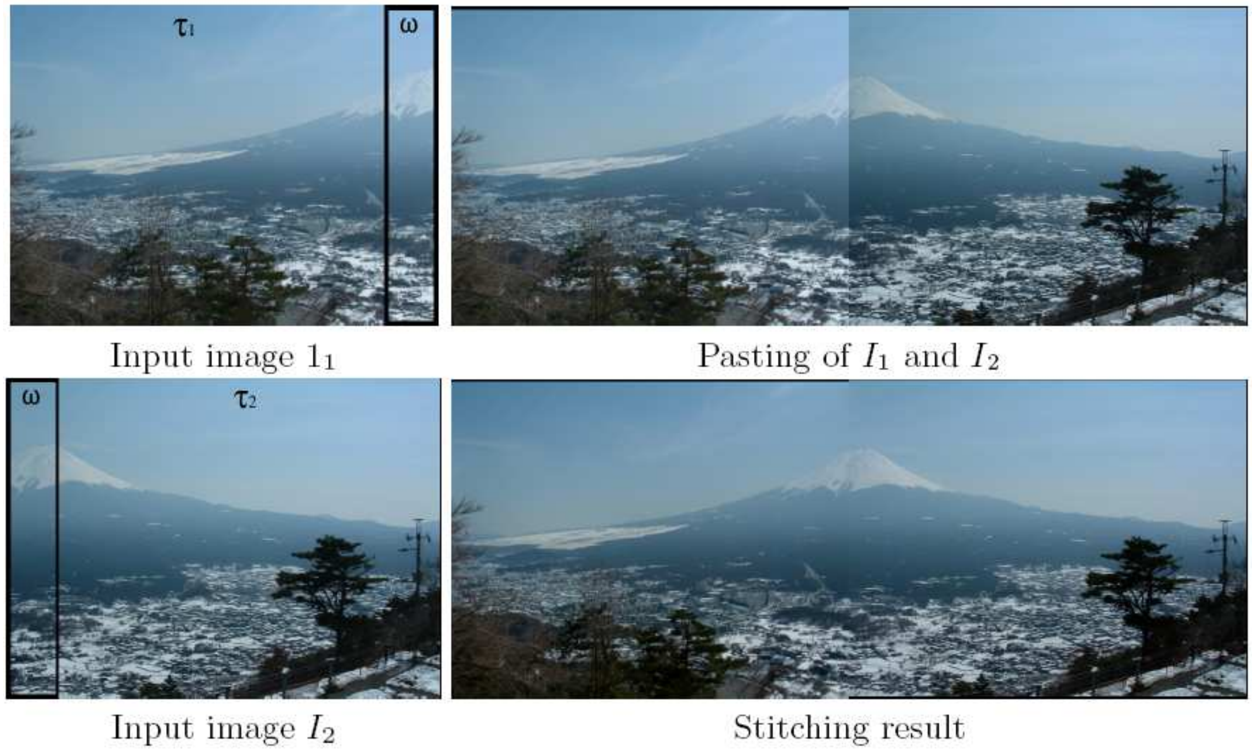

Figure 1.9: Image stitching (courtesy [2]). On the left are the input images. ω is the overlap region. On top right is a simple pasting of the input images. On the bottom right is the result after seam-less stitching.

not on feature extraction or matching. More details on transformation estimation can be found in the next chapter.

1.3 Image Compositing

The registered images generally require further processing to eliminate remaining distortions and discontinuities. This is taken care of while stitching or compositing them. In addition to mosaicing, stitching is common in many applications such as object insertion, super resolution and texture synthesis. The aim of a stitching algorithm is to produce a visually plausible mosaic with two desirable properties: First, the mosaic should be as similar as possible to the input images, both geometrically and photometrically. Second, the seam between the stitched images should be invisible. An example of seam-less stitching is shown in Fig. 1.9.

Alignment of images may be imperfect due to registration errors resulting from various factors. Some of the common factors are incompatible model assumptions, dynamic scenes, changing lighting conditions while capturing images, automatic controls of cameras and different scanning devices.

The effects of the above factors can be alleviated during the compositing process by appropriately blending the (registered) images. The main challenge in image compositing is the problem of determining how the pixels in an overlapping area should be represented. If the alignment is extremely accurate, the compositing process becomes trivial. In real life, most of the artifacts, which are visually detectable, are due to the misalignment of certain frames and change in intensity across frames. While the latter can be handled by performing histogram manipulation [35], in general, a smooth blending function is required to produce a seam-less mosaic. In Feathering [36] or alpha blending, the mosaic image I is a weighted combination of the input images I1, I2. The weighting coefficients (alpha mask) vary as a function of the distance from the seam. In pyramid blending [37], different frequency bands are combined with different alpha masks. Lower frequencies are mixed over a wide region, and fine details are mixed in a narrow region. This produces gradual transition in lower frequencies, while reducing edge duplications in textured regions. In [2], a gradient-domain stitching approach is presented to minimize the disturbing edges along the seam. Since the focus of this work is on image registration, we employ the computationally light yet effective alpha blending for mosaic generation.

1.4 Challenges in Video Mosaicing

Although considerable work has gone into the area of video mosaicing, under many practical scenarios, mosaicing still remains a challenging problem. Some of the most commonly faced issues are highlighted below.

1. The video may have a series of texture-less frames, yielding no features to track. For example, an aerial video over a desert has minimal information for registration (See Fig. 1.10(a)).

2. There are errors in extracting feature points from the images. These errors typically stem from poor localization and sensitivity to image degradation, can significantly affect the mosaic quality.

3. The correspondence computation scheme fails in presence of repetitive texture, due to several mis-matched points. An example of this is shown in Fig. 1.10(b). The points get matched incorrectly because of the presence of very similar regions throughout the images. This causes the feature representation to become non-unique.

4. The video often consists of some (isolated) poor quality frames, due to motion blur or sensor error. Such outlier frames need to be detected and their role in the mosaicing process needs to be suppressed. Note that these frames may be needed in the mosaic for visual continuity. Therefore, they can not always be discarded from the scheme.

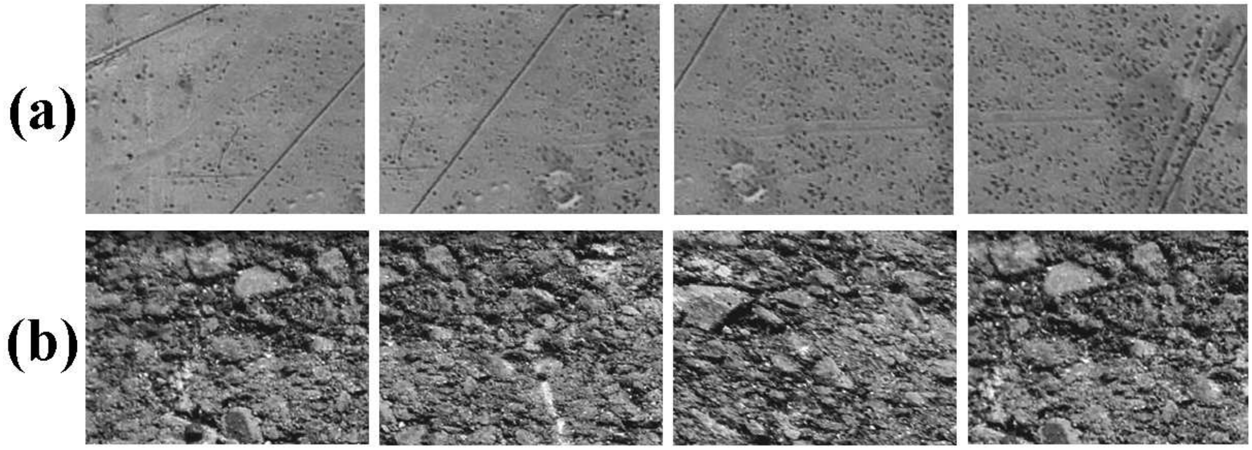

Figure 1.10: (a) Sample frames from an aerial video over a desert region, containing very less visual information; the video frames in (b) Frames with repetitive texture which leads to false correspondences. Mosaicing these video frames is a very challenging problem.

5. When homographies are concatenated, the registration errors accumulate. Errors in registering the initial pairs of frames get carried forward to the subsequent ones. This is known as the error propagation phenomenon. Sometimes, it is reflected in the first and the last video frames being poorly registered even if they have a good amount of overlap and visual similarity.

In addition to the above, in certain types of mosaicing situations, the type of transformation model chosen plays a crucial role in the registration accuracy. If no *apriori* information is available, the model should be flexible and general enough to handle all geometric transformations and other degradations which might appear. However, if prior knowledge is available, its usage can lead to simpler, more stable and faster numerical estimation of the transformation. Another issue affecting the quality of the mosaic is the selection of reference frame, in case of a single view-point mosaic. It can have a significant effect on the appearance of the final mosaic as it determines the fixed view-point from which the registered images are to be viewed in the mosaic. Generally, the choice of the reference frame is made in such a way that the error propagation is suppressed maximally.

1.5 Scope of the Thesis

While the existing approaches alleviate issues like error accumulation and noisy correspondence, to some extent, they are rendered powerless to a great extent when video frames yield features that fall in one or more of the following categories:

- Inadequate in number

- Erroneous features
- Mis-matched features.

The primary reason behind the failure of the existing approaches is the fact that mosaicing has been worked upon primarily in the image or 2D domain. Not much attention has been put on role that the imaging process can have for robust registration for mosaicing. Thus, registration of one pair of video frames is typically done independent of the other frames. Until recently, only consecutive frames had been utilized for video mosaicing because they were likely to have maximal overlap. Thus, there remains scope for more robustness and more accuracy in video mosaicing. This motivates us to explore a different cue - properties of the camera motion - for enhancing video mosaicing.

It is observed that the motion of the camera, that captures the video to be mosaiced, is not arbitrary. The trajectory of the camera is often smooth. We propose a technique which utilizes this important piece of information for improved registration and thus, improved mosaicing. Moreover, in many scenarios, camera motion that we encounter is not only smooth, but also follows a specific known motion model. This is especially seen in machine-controlled environments. In such cases, the model followed by the camera motion relates the transformations required to register different pairs of video frames. We exploit these relationships to estimate all transformations together, by combining information from multiple frames. We also propose a computationally inexpensive yet effective quality measure for mosaics. It is aimed at quantifying fine differences between mosaics.

1.6 Organization of the Thesis

- In this chapter, we have introduced the problem of mosaicing, briefly explained various types of mosaicing and the applications of video mosaicing. Image registration and image stitching, the major steps of mosaicing, have also been outlined. Then, we discussed that despite significant work having gone into the areas of mosaicing and registration, video mosaicing remains a challenging problem, motivating us to explore a new cue - properties of the camera motion - for robust, automatic generation of mosaics from videos.
- We begin with the details of homography-based registration, in Chapter 2. Homography is introduced and popular approaches of homography estimation are outlined. We provide brief analysis and comparison of some of these approaches. Video mosaics built using pair-wise, independent registrations are shown. The suboptimal quality of mosaics points at the need for exploring the relationships between homographies.
- In Chapter 3, we show that if the camera follows a specific known motion model, the ensemble of homographies required for video mosaicing can be shown to be related by a well-defined

model. We derive algebraic relationships for various camera models, for various types of homographies. To put it differently, the camera motion models are translated into global homography models.

- Chapter 4 presents an algorithm that exploits the derived relationships for robust estimation of the homographies and thereby obtaining visually pleasing mosaics. Homography estimation is performed by first estimating the parameters of this model and expressing the required homographies in terms of these parameters.

- In Chapter 5, we utilize the smoothness of the camera trajectory to refine homographies for improved mosaicing. A simple but effective technique is proposed, which utilizes the decomposition of homography into *pose* (rotation and translation) and extrapolation in the pose space to refine the outlier homography estimates.

- In chapter 6, a new quality measure is devised for quantitative evaluation of mosaics. It is aimed at discriminating between mosaics with subpixel alignment. The low-cost, single-pass measure is shown to be smooth and devoid of local maxima. An exhaustive set of visual results on various scenarios is also demonstrated. Comparison is made with mosaics obtained using existing approaches.

Chapter 2

Robust Homography Estimation for a Pair of Images

In the previous chaper, we discussed the major steps of registration: feature extraction, feature matching and transformation estimation. In this section, we place emphasis on the estimation of the transformation that relates the given pair of images. Primitive image registration algorithms assume the images to be related by a similarity transformation (i.e., translation, rotation and scaling). They estimate the transformation parameters by spatial correlation or transform domain techniques [28, 29, 30]. Subsequently, a six-parameter affine transformation model became popular. In recent years, researchers employ a full projective transformation (*homography*) with 8 free parameters for image alignment. A comprehensive survey of registration techniques can be found in [28, 29, 30]. Since homography is the most generic transformation in the projective space, in this thesis, we exclusively use it as the transformation model.

2.1 Homography

Homography is a non-singular linear relationship. Images of points on a plane are related to corresponding image point in a second view by a (planar) homography. This is a projective relation since it only depends on the intersection of planes with lines. The homography transfers points from one view to the other as if they were images of points on the plane. If x and x' are images of the same world point, represented in the homogeneous coordinate system,

$$\mathbf{x}' = \mathbf{H}\mathbf{x} \tag{2.1}$$

The homography induced by the plane is unique, meaning that every plane can contribute one homography. The homography depends on the intrinsic and extrinsic parameters of the camera, as well as on the 3D plane parameters. Homography also exists for non-planar scenes when the camera (strictly) rotates about its optical center. More details about homography can be found in [38].

Several methods have been reported for estimating the planar homography. A homography is uniquely specified from two projective bases, which are two sets of four points such that none of three points lie on a line. The homography is defined only up to an unknown non-zero scale factor as the points are in homogeneous coordinates. Each pair of matching points gives two equations – for x and the other for y – in the 8 real unknowns of $\mathbf{H}$. Direct linear methods try to estimate these unknowns using 4 point matches [39]. Accuracy of estimation can be improved in the presence of noise if more point matches are available using a least-square error framework. Homography can be estimated by identifying other matching features also. Matching lines provide exactly the same information as matching points since lines and points are duals of each other. Conics such as circles and ellipses have also been used to compute the homography due to the plane on which they lie. They use either algebraic curves or use invariant fitting for this purpose.

Computing the homography induced by a plane between two views is an important step toward calibration, metric rectification, reconstruction, etc. They can also be applied to problems like image registration, image mosaics and other applications using perspective geometry.

2.1.1 Homography induced by a plane

When a planar object is imaged from multiple view-points, the images are related by a unique homography. A plane π in the scene defines a mapping between points in two views by the following procedure: the ray corresponding to a point $\mathbf{x}$ is extended to meet the plane π in a point $\mathbf{X}_\pi$; this point is projected to a point $\mathbf{x}'$ in the other image (See Fig. 2.1).

Given the projection matrices for the two views

$$\mathbf{P} = [\mathbf{I} \mid \mathbf{0}] \text{ and } \mathbf{P}' = [\mathbf{A} \mid \mathbf{a}],$$

and a plane defined by $\pi^{\mathbf{T}}\mathbf{X} = \mathbf{0}$ with $\pi = (\mathbf{v}^{\mathbf{T}}, 1)^T$, then the homography induced by the plane is given by [39]:

$$\mathbf{x}' = \mathbf{H}\mathbf{x} \tag{2.2}$$

with

$$\mathbf{H} = \mathbf{A} - \mathbf{a}\mathbf{v}^{\mathbf{T}} \tag{2.3}$$

Considering the intrinsic camera matrices K' and K respectively, the planar homography is given by

$$\mathbf{H} = \mathbf{K}'\big(\mathbf{A} - \mathbf{a}\mathbf{v}^{\mathbf{T}}\big)\mathbf{K}^{-1} \tag{2.4}$$

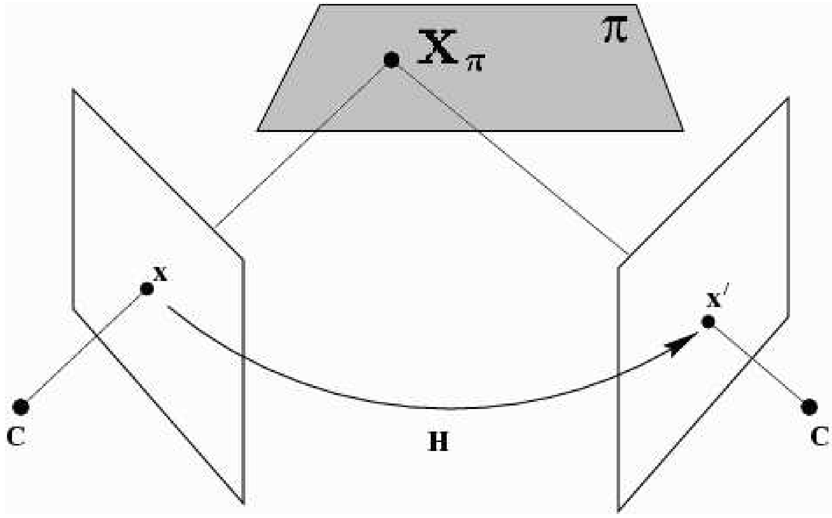

Figure 2.1: Plane π provides a mapping between the images $\mathbf{x}$ and $\mathbf{x}'$ of point $\mathbf{X}_\pi$, captured by cameras C and C'.

2.2 Homography Estimation from Point Correspondences

Points are the simplest and the most primitive features that can be used for estimating homography. They have also been the main focus of the researchers for homography estimation. The main point-based methods that can been employed for estimating homography are discussed below.

2.2.1 Direct Linear Transformation

We begin with a simple linear algorithm for determining $\mathbf{H}$ given a set of four 2D to 2D point correspondences, $\mathbf{x_i} \leftrightarrow \mathbf{x_i'}$. The equation $\mathbf{x}' = \mathbf{H}\mathbf{x}$ may be expressed in terms of the vector cross product as $\mathbf{x_i'} \times \mathbf{H}\mathbf{x_i} = \mathbf{0}$.

If the j-th row of the matrix $\mathbf{H}$ is denoted by $\mathbf{h}^{jT}$. Then

$$\mathbf{H}\mathbf{x_i} = \begin{pmatrix} \mathbf{h^{1T}}\mathbf{x_i} \\ \mathbf{h^{2T}}\mathbf{x_i} \\ \mathbf{h^{3T}}\mathbf{x_i} \end{pmatrix}.$$

Writing $\mathbf{x_i'} = (x_i', y_i', w_i')^T$, the cross product may then be given explicitly as

$$\mathbf{x_i'} \times \mathbf{H}\mathbf{x_i} = \begin{pmatrix} y_i'\mathbf{h^{3T}}\mathbf{x_i} - w_i'\mathbf{h^{2T}}\mathbf{x_i} \\ w_i'\mathbf{h^{1T}}\mathbf{x_i} - x_i'\mathbf{h^{3T}}\mathbf{x_i} \\ x_i'\mathbf{h^{2T}}\mathbf{x_i} - y_i'\mathbf{h^{1T}}\mathbf{x_i} \end{pmatrix}.$$

The equation may be rewritten in the form

$$\begin{bmatrix} \mathbf{0^T} & -w_i'\mathbf{x_i^T} & y_i'\mathbf{x_i^T} \\ w_i'\mathbf{x_i^T} & \mathbf{0^T} & -x_i'\mathbf{x_i^T} \\ -y_i\mathbf{x_i^T} & x_i'\mathbf{x_i^T} & \mathbf{0^T} \end{bmatrix} \begin{pmatrix} \mathbf{h^1} \\ \mathbf{h^2} \\ \mathbf{h^3} \end{pmatrix} = \mathbf{0}. \tag{2.5}$$

These equations have the form $\mathbf{A_i h} = \mathbf{0}$ where $\mathbf{A_i}$ is a 3×9 matrix, and let $\mathbf{h} = [\mathbf{h}^{1T}\ \mathbf{h}^{2T}\ \mathbf{h}^{3T}]^T$. Although there are three equations in (2.5), only two of them are linearly independent. Thus each point correspondence gives two equations in the entries of $\mathbf{H}$. Then (for future reference) the set of equations becomes

$$\begin{bmatrix} \mathbf{0^T} & -w_i'\mathbf{x_i^T} & y_i'\mathbf{x_i^T} \\ w_i'\mathbf{x_i^T} & \mathbf{0^T} & -x_i'\mathbf{x_i^T} \end{bmatrix} \begin{pmatrix} \mathbf{h^1} \\ \mathbf{h^2} \\ \mathbf{h^3} \end{pmatrix} = \mathbf{0}.$$

This will be written

$$\mathbf{L_i h} = \mathbf{0}.$$

Given a set of four point correspondences from the plane, a set of equations $\mathbf{Lh} = \mathbf{0}$ is obtained, where $\mathbf{L}$ is the matrix of equation coefficients built from the matrix rows $\mathbf{L_i}$ contributed from each correspondence, and $\mathbf{h}$ is the vector of unknown entries of $\mathbf{H}$.

In practice, the extracted image points do not satisfy the relation $\mathbf{x}' = \mathbf{Hx}$ because of noise in the extracted image points. Lets assume that $\mathbf{x_i'}$ is corrupted by Gaussian noise with mean $\mathbf{0}$ and covariance matrix $\mathbf{\Sigma_{x'}}$. Then, the maximum likelihood estimation of $\mathbf{H}$ is obtained by minimizing the following function:

$$\sum \left(\mathbf{x_i'} - \hat{\mathbf{x}}_\mathbf{i}'\right)^\mathbf{T} \mathbf{\Sigma}_{\mathbf{x}'}^{-1} \left(\mathbf{x_i'} - \hat{\mathbf{x}}_\mathbf{i}'\right),$$

where

$$\hat{\mathbf{x}}_\mathbf{i}' = \frac{1}{\mathbf{h^{3T} x_i}} \begin{bmatrix} \mathbf{h^{1T} x_i} \\ \mathbf{h^{2T} x_i} \end{bmatrix},$$

where $\mathbf{h^i}$ is the ith row of $\mathbf{H}$.

In practice, simply assume $\lambda \mathbf{x_i'} = \sigma^2 \mathbf{I}$ for all i. The above problem becomes a nonlinear least-squares estimation problem, i.e., finding $\mathbf{H}$ such that $\|\mathbf{x_i'} - \hat{\mathbf{x}}_\mathbf{i}'\|^2$ is minimum. Then

$$\begin{bmatrix} \mathbf{x}^T & \mathbf{0} & -u\mathbf{x}^T \\ \mathbf{0} & \mathbf{x}^T & -v\mathbf{x}^T \end{bmatrix} \mathbf{h} = 0.$$

This can be written as

$$\mathbf{L_i h} = \mathbf{0}.$$

Given n points, they can be written in matrix equation as $\mathbf{Lh} = \mathbf{0}$, where $\mathbf{L}$ is a $2n \times 9$. We seek a non-zero solution $\mathbf{h}$, since the obvious solution $\mathbf{h} = \mathbf{0}$ is of no interest to us. However, there will not be an exact solution to the over-determined system of the $\mathbf{Lh} = \mathbf{0}$ apart from the zero solution. Instead of demanding an exact solution, a solution that minimizes a suitable cost function is sought. To avoid the solution $\mathbf{h} = \mathbf{0}$, a condition on the norm is used, such as $\|\mathbf{h}\| = 1$. This is identical to the problem of finding the minimum of the quotient $\|\mathbf{Lh}\|/\|\mathbf{h}\|$. The solution is the (unit) eigenvector of $\mathbf{L^T L}$ with least eigenvalue. Equivalently, the solution is the right singular vector associated with the smallest singular value of $\mathbf{L}$.

Algorithm 1 Direct Linear Transformation [39]

Objective:

Given $n \geq 4$ 2D to 2D point correspondences $\mathbf{x_i} \leftrightarrow \mathbf{x'_i}$, determine the 2D homography matrix $\mathbf{H}$ such that $\mathbf{x'_i} = \mathbf{H}\mathbf{x_i}$.

Algorithm

1. For each correspondence $\mathbf{x_i} \leftrightarrow \mathbf{x'_i}$ compute $\mathbf{L_i}$. Usually only two first rows needed.
2. Assemble n 2×9 matrices $\mathbf{L_i}$ into a single $2n \times 9$ matrix $\mathbf{L}$.
3. Obtain SVD of $\mathbf{L}$. Specifically, if $\mathbf{L} = \mathbf{UDV^T}$ with $\mathbf{D}$ diagonal with positive diagonal entries, arranged in descending order down the diagonal, then $\mathbf{h}$ is last column of $\mathbf{V}$.
4. Determine $\mathbf{H}$ from $\mathbf{h}$.

In $\mathbf{L}$, some elements are constant 1, some are in image coordinates, some are in world coordinates, and some are multiplications of both. This makes $\mathbf{L}$ poorly conditioned numerically. Much better results can be obtained by performing a simple data normalization, prior to running the above procedure. The procedure is suggested as follows.

1. Transform the image coordinates according to the transformations $\mathbf{\tilde{x}_i} = \mathbf{T}\mathbf{x_i}$ and $\mathbf{\tilde{x}'_i} = \mathbf{T}'\mathbf{x'_i}$.
2. Find the transformation $\mathbf{\tilde{H}}$ from the correspondences $\mathbf{\tilde{x}_i} \leftrightarrow \mathbf{\tilde{x}'_i}$.
3. Set $\mathbf{H} = \mathbf{T}'\mathbf{\tilde{H}}\mathbf{T}$.

One of the commonly used transformation is to translate the points so that their centroid is at the origin and the points are scaled such that the average distance from the origin is equal to $\sqrt{2}$. Hartley shows that data normalization gives dramatically better results and hence should be considered as an essential step in the algorithm [39].

2.2.2 RANSAC

So far it has been assumed that in the set of correspondences, $\mathbf{x_i} \leftrightarrow \mathbf{x'_i}$, presented the only source of error is in the measurement of the point's position, which follows a Gaussian distribution. In many practical situations this assumption is not valid because correspondences are computed automatically and points are often mismatched. The mismatched points can be considered as outliers to the

Gaussian error distribution. These outliers can severely disturb the estimated homography, and consequently should be identified. The goal then is to determine a set of inliers from the presented correspondences so that the homography can be estimated in an optimal manner from these inliers using the algorithm described in the previous section. This is robust estimation since the estimation is robust or tolerant to outliers, i.e. measurements following a different, and possibly, un-modeled error distribution.

The RANSAC algorithm [40] can be applied to the putative correspondences to estimate the homography and the (inlier) correspondences which are consistent with this estimate. The sample size is four, since four correspondences determine a homography. The number of samples is set adaptively as the proportion of outliers is determined from each consensus state.

Algorithm 2 RANSAC-based Homography estimation [39]

Objective

Compute the homography between the two images given a set of candidate matches.

Algorithm

1. Select four points from the set of candidate matches, and compute homography.
2. Select all the pairs which agree with the homography. A pair $(\mathbf{x}; \mathbf{x}')$, is considered to agree with a homography H, if $dist(\mathbf{Hx}; \mathbf{x}') < \epsilon$, for some threshold ϵ and $dist()$ is the Euclidean distance between two points.
3. Repeat steps (1) and (2) until a sufficient number of pairs are consistent with the computed homography.
4. Re-compute the homography using all consistent correspondences.

Three questions immediately arise [39].

1. *What is the distance threshold?* The distance threshold t should be chosen, such that the point is an inlier with a probability α. This calculation requires known probability distribution for the distance of an inlier from the model. In practice the distance threshold is chosen empirically.
2. *How many samples?* It is often computationally infeasible and unnecessary to try every possible sample. Instead the number of samples N is chosen sufficiently high to ensure with a probability p, that at least one of the random samples of 4 points is free from outliers. Usually p is chosen at 0.99.
3. *How large is consensus set?* A rule of thumb is to terminate if the size of the consensus set is similar to the number of the number of inliers believed to be in the data set, given the

assumed proportion of outliers, i.e. for n data points $T = (1-\epsilon)n$.

2.2.3 Kanatani's statistical optimization

The least-squares method is widely used because of its computational simplicity. However the least-squares method produces statistical bias, because homogeneous coordinates are treated as if they were physical quantities without distinguishing measurement data from artificial constants. Since noise occurs in image coordinates subject to Euclidean geometry, it is preferable to work in the Euclidean framework.

Kanatani [41] proposes one such method to compute the homography in the Euclidean framework. The uncertainty of data points (x, y) and (x', y') is described by their covariance matrices $\mathbf{\Sigma}$ and $\mathbf{\Sigma}'$. It follows that the vectors $\mathbf{x}$ and $\mathbf{x}'$ have the following singular covariance matrices.

$$\mathbf{V}\left[\mathbf{x}_\alpha\right] = \frac{1}{f^2}\begin{bmatrix} \mathbf{\Sigma} & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} \text{ and } \mathbf{V}\left[\mathbf{x}'_\alpha\right] = \frac{1}{f^2}\begin{bmatrix} \mathbf{\Sigma}' & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix}.$$

It can be assumed that the covariance matrices are known up to a scale i.e. $\mathbf{V}[\mathbf{x}_\alpha] = \varepsilon^2 \mathbf{V_0}[\mathbf{x}_\alpha]$ and $\mathbf{V}[\mathbf{x}'_\alpha] = \varepsilon^2 \mathbf{V_0}[\mathbf{x}'_\alpha]$. The unknown magnitude ϵ can be considered to be the noise level. The normalized covariance matrices $\mathbf{V_0}[\mathbf{x}_\alpha]$ and $\mathbf{V_0}[\mathbf{x}'_\alpha]$ specify the relative dependence of noise occurrence on positions and orientations. If no prior knowledge is available for them, simply assume isotropy and homogeneity and use the default values $\mathbf{V_0}[\mathbf{x}_\alpha] = \mathbf{V_0}[\mathbf{x}'_\alpha] = diag(1,1,0)$. Let $\mathbf{H}'$ be an estimate of the homography, and $\mathbf{H}$ its true value. The uncertainty of the estimate $\mathbf{H}'$ is measured by its covariance tensor

$$V[\mathbf{H}'] = E[P((\mathbf{H}' - \mathbf{H}) \otimes (\mathbf{H}' - \mathbf{H}))P^T],$$

where $E[.]$ denotes expectation. The symbol $\otimes$ denotes tensor product. The $(ijkl)$ element of the tensor P is given by $P_{ijkl} = \delta_{ik}\delta_{jl} - H_{ij}H_{kl}$, where δ_{ij} is the Kronecker delta. Let

$$\mathbf{W}_\alpha = \mathbf{x}'_\alpha \times \mathbf{H}\mathbf{V_0}[\mathbf{x}]\mathbf{H}^\mathbf{T} \times \mathbf{x}'_\alpha + (\mathbf{H}\mathbf{x}_\alpha) \times \mathbf{V_0}[\mathbf{x}_\alpha] \times (\mathbf{H}\mathbf{x}_\alpha).$$

The theoretical accuracy bound can be attained by minimizing the squared Mahalanobis distance. Using Langrage multipliers and first order approximation we get

$$\mathbf{J} = \sum \left(\mathbf{x}'_\alpha \times \mathbf{H}\mathbf{x}, \mathbf{W}_\alpha \left(\mathbf{x}'_\alpha \times \mathbf{H}\mathbf{x}_\alpha\right)\right).$$

The eigenmatrix $\mathbf{U}$ of the resulting covariance tensor $V[\mathbf{H}']$ for the maximum eigenvector λ indicates the orientation in the 9-dimensional space along which error is most likely to occur. The solution $\mathbf{H}$ is perturbed along that orientation in both directions as $\mathbf{H}^{(+)} = N[\mathbf{H}' + \sqrt{\lambda}\mathbf{U}]$ and $\mathbf{H}^{(-)} = N[\mathbf{H}' - \sqrt{\lambda}\mathbf{U}]$ The operator $N[.]$ denotes normalization to unit norm.

The homography can be estimated by renormalization which is described as follows [41]

1. Let $c = 0$ and $\mathbf{W}_\alpha = \mathbf{I}, \alpha = 1, ..., N$.

2. Define the following tensor M

$$M = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{3}\mathbf{W}_\alpha^{(\mathbf{kl})}(\mathbf{e}^{(\mathbf{k})} \times \mathbf{x}'_\alpha) \otimes \mathbf{x}_\alpha \otimes (\mathbf{e}^{(}\mathbf{l}) \times \mathbf{x}'_\alpha) \otimes \mathbf{x}_\alpha. \tag{2.6}$$

3. Compute the following tensor $N = (N_{ijkl})$

$$N_{ijkl} = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{m,n,,p,q=1}^{3}\epsilon_{imp}\epsilon_{knq}\mathbf{W}_\alpha^{\mathbf{mn}}(\mathbf{V_0}[\mathbf{x}_\alpha]_{jl}x'_{\alpha(p)}x'_{\alpha(q)} + \mathbf{V_0}[\mathbf{x}'_\alpha]_{jl}x_{\alpha(j)}x_{\alpha(l)}). \tag{2.7}$$

4. Compute the nine eigenvalues $\lambda_1 \geq ... \geq \lambda_9$ of tensor

$$\hat{M} = M - cN, \tag{2.8}$$

and the corresponding orthonormal system of eigenmatrices $\mathbf{H_1}$, ...,$\mathbf{H_9}$ of unit norm. If $\lambda_9 \approx 0$, stop. Else update c and $\mathbf{W}_\alpha$ in the following way and go back to step 2:

$$c \leftarrow c + \frac{\lambda_9}{(\mathbf{H_9}; N\mathbf{H_9})}. \tag{2.9}$$

$$\mathbf{W}_\alpha \leftarrow \mathbf{x}'_\alpha \times \mathbf{H_9}\mathbf{V_0}[\mathbf{x}_\alpha]\mathbf{H_9^T} \times \mathbf{x}'_\alpha + (\mathbf{H_9}\mathbf{x}_\alpha) \times \mathbf{V_0}[\mathbf{x}'_\alpha] \times \mathbf{V_0}[\mathbf{x}'_\alpha] \times (\mathbf{H_9}\mathbf{x}_\alpha). \tag{2.10}$$

Algorithm 3 Kanatani's statistical optimization [41]

Objective

Compute the homography between the two images given the correspondences between the noisy points $\mathbf{x_i}$ and $\mathbf{x'_i}$.

Algorithm

1. Initialise the weight matrix $\mathbf{W}_\alpha$ to an identity matrix.
2. Compute the tensors M and N using Equations 2.6 and 2.7.
3. Compute the nine eigenvalues of tensor $\hat{M}$ from Equation 2.8.
4. If the last eigenvalue is approximately zero then stop, else update c and $\mathbf{W}_\alpha$ using Equations 2.9 and 2.10 and repeat the above procedure.
5. The homography is the eigenmatrix $\mathbf{H_9}$ of tensor $\hat{M}$

2.3 Analysis and Results

Table 2.1 gives the performance of these algorithms in the presence of Gaussian white noise, with zero mean and varying standard deviation (σ). As the noise level increases, the performance of all

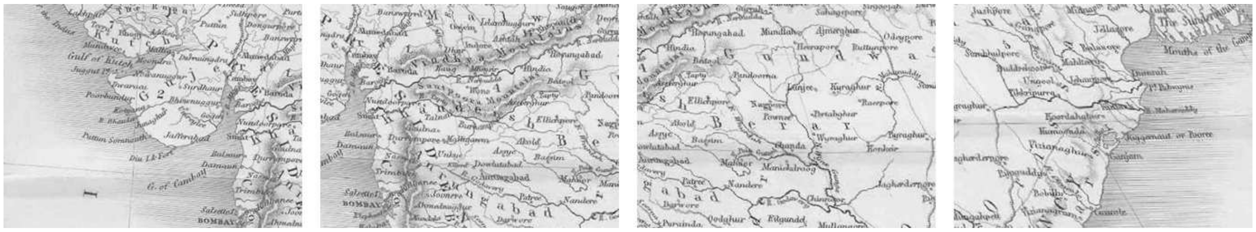

Figure 2.2: Sample frames of a map video

Figure 2.3: Mosaic of the map video built using pair-wise RANSAC

σ	DLT	RANSAC	Kanatani
0	0.0000	0.0000	0.0000
0.5	0.703	0.766	0.680
1	1.404	1.386	1.384
2	3.671	3.618	2.712
Quantization	0.540	0.5694	0.520

Table 2.1: Performance of homography estimation algorithms under different (Gaussian) noise levels and quantization

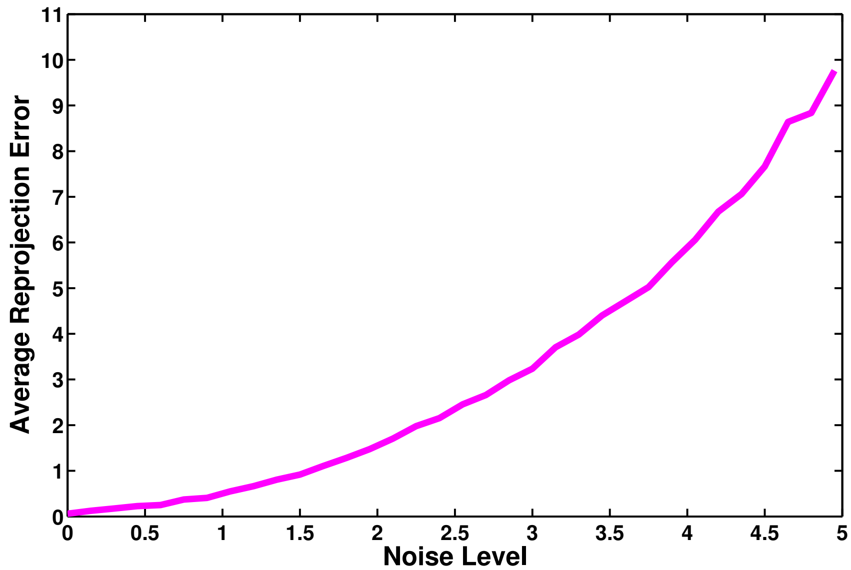

Figure 2.4: Cumulative Average Reprojection Error Vs Uniform Noise

the algorithms degrades (as it should). However, RANSAC and Kanatani's statistical technique handle noise better. RANSAC-based technique has the important advantage of being able to detect and remove outliers. This makes RANSAC the most widely used homography estimation technique. More detailed comparative study of these algorithms can be found in [38].

We used this RANSAC-based approach for homography estimation to build video mosaics. We conducted several studies to know its limitations. First, we analyzed the sensitivity of this approach to noise in the image measurements. Uniform noise was added to the set of point correspondences. Fig. 2.4 shows that the cumulative average reprojection error increases alarmingly with the level of noise, which is not desirable. The cumulative average reprojection error is calculated as

$$E = \frac{1}{m * (N-1)} \sum_{i,j} \|\Pi(X_{i+1}^{j}) - \Pi(H_{i,i+1}X_{i}^{j})\|^2$$

where, $i \in \{0, 1, \ldots, N-1\}$ denotes the frame number and $j \in \{0, 1, \ldots, M-1\}$ is the feature point index and Π is the imaging function.

A common problem, in videos, is the presence of some poor quality frames. These frames worsen the quality of homography estimates and consequently, the mosaic. We conducted an experiment wherein the quality of one of the frames was degraded. The reprojection error was as high as 5.7898.

In Fig. 2.3, we present the mosaic of a map of India, in an in-door setting. The video frames (See samples in Fig. 2.2) have been registered with reasonable precision into the mosaic. For viewing at lower resolutions, the mosaic given by pair-wise RANSAC is adequately accurate. However, it

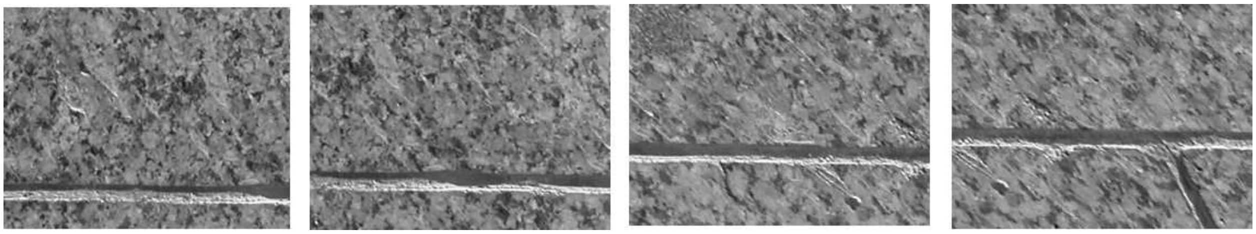

Figure 2.5: Sample frames of a video containing repetitive rexture

should be noted that more accuracy can be obtained. A better-quality mosaic of the same video is shown in chapter 6, with parts of it zoomed to show the increased clarity.

2.4 Summary and Further Challenges

The conventional video mosaicing approach of registering successive pairs of video frames presents a neat closed form solution scheme in theory. It also works quite well when sufficient, reasonably noise-free points are available for transformation estimation. However, when the noise level is high and/or the features are few and/or mis-matched, registration and consequently, mosaicing fails.

Fig. 2.6 depicts the performance of the RANSAC-based mosaicing approach in presence of the frequently encountered repetitive texture problem. The blurred patches in the mosaic indicate that the pair-wise RANSAC approach was severely affected by the mis-matched points and was not able to yield accurate homographies and consequently, an acceptable mosaic.

The above analysis and visual results confirm our argument that there remains scope for improvement in video mosaicing. Under many practical problems, video mosaicing needs to go beyond every pair of frames being registered individually. This motivates us to explore a new cue for robust registration for video mosaicing. The cue is the properties of the camera motion.

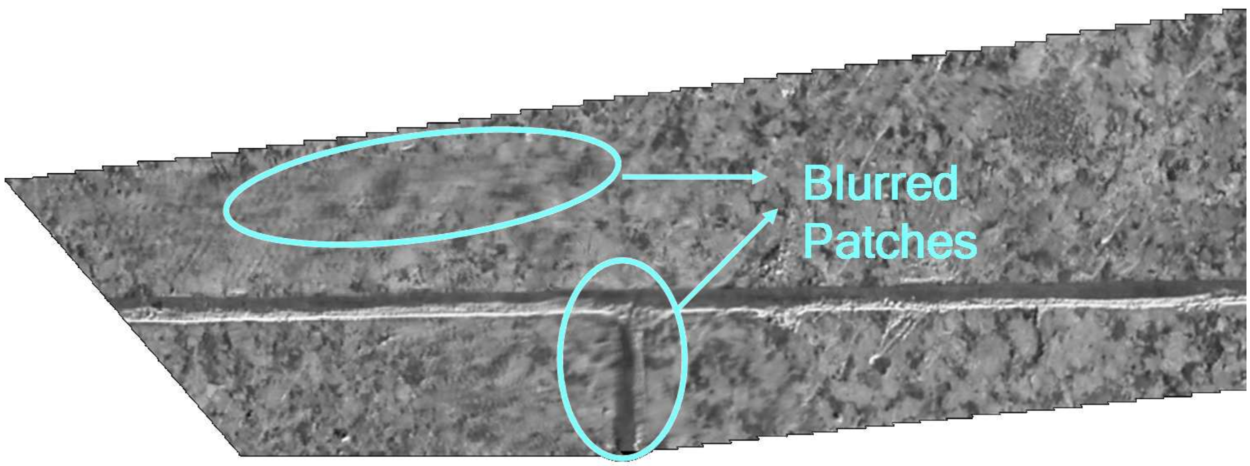

Figure 2.6: Mosaic of the video whose sample frames are shown in Fig. 2.5. The repetitive texture phenomenon leads to a lot of false feature matches, resulting in poor registration and a poor mosaic.

Chapter 3

Motion Constraints for Homography Estimation in a Video Sequence

The drawbacks of independent pair-wise registrations of video frames, described in Chapter 2, motivate us to explore relationships between homographies pertaining to different frame pairs. In this chapter, we derive a set of constraints on homographies relating frames in a video, by capitalizing the motion information of the camera that captures the video. In other words, the constraints on the camera motion or the relationships between the camera's positions and orientations, corresponding to different video frames, are translated into their counterparts on the induced homographies. These relationships, for two kinds of homographies, widely used in the literature, are explored under various types of camera motion models. The most salient point to have come out of this exploration is that for a well-defined camera motion model, the corresponding global homography model (the parameters that relate homographies across frames) can be derived. This homography model, in the next chapter, is shown to be an important clue in building mosaics (even when computation of homography is unreliable for a specific pair).

3.1 Homographies in a Video

Images of corresponding points on a planar object, taken by two different cameras, are proven to be related by a homography. To state it formally, let H_{12} be the homography relating two different frames 1 and 2. From the fundamentals of multiple view geometry [39], it is well known that

$$X_2 = H_{12}X_1$$

where X_1 and X_2 are the corresponding points in frames 1 and 2, expressed in homogeneous coordinates as $[x, y, 1]^T$ and the homography H_{12} is a 3×3 non-singular matrix. Two frames (images) are related by a homography only under certain conditions. Two of the most popular ones are the world being planar and the camera motion restricted to rotation about its optical axis. For this thesis, we assume planarity of the world.

If there are N frames with indices $0, 1, \ldots, i, i+1, \ldots, N-1$, then the mosaic could be built using the following types of homographies:

Reference Homographies:

All the video frames can be mapped to a reference frame (say r) with the help of a set of reference homographies $H_{i,r}$. Reference frame could be the first, last or any of the intermediate ones.

Incremental or Frame-to-next-frame Homographies:

The homographies between the current frame and the next frame ($H_{i,i+1}$) can be for video mosaicing, by imposing a single view point constraint in an appropriate manner.

The quality of mosaics heavily depends on the accuracy in the registration (more specifically, the accuracy of the estimation of the transformation) of images. This often depends on: (i) Accuracy with which one can estimate the features (say interest points) from the images. (ii) Performance of the correspondence computation scheme. (iii) Accuracy of the numerical procedure for robust computation of the homography from the (noisy) correspondences.

It is worth noting that in most situations, only consecutive frames are registered for mosaicing. If one of the frames is of poor quality, the entire mosaic can become highly inaccurate. This inaccuracy can also get propagated for all frames succeeding this frame. In this chapter, we argue that, there exists relationships connecting homographies. This can help in robustly estimating the ensemble of homographies even when a specific homography, estimated in isolation, is erroneous. In order to derive these relationships, first, we derive an expression for the homography relating two arbitrary views of a planar object.

3.1.1 Homography Relating Two Arbitrary Views of a Planar Object

We know from [39] that if $K[I|0]$ and $K'[R|T]$ are the first and second camera matrices and $\pi = (n^T, d)^T$ represents the world plane (such that, for points P_i on the plane, $n^T P + d = 0$), the homography induced by the plane is given by:

$$H = K\big(R - \frac{Tn^T}{d}\big)K'^{-1} \tag{3.1}$$

However, for the video mosaicing applications, where the two frames are captured from arbitrary poses, we need a generalized relationship. Let $K[R_0|T_0]$ and $K[R|T]$ be the first and second camera matrices and $\pi = (n^T, d)^T$ represent the plane (In case of videos, the camera remains the same, leading to $K = K'$). We apply a rigid transformation on the world coordinate system such that the first camera pose changes from $[R_0|T_0]$ to $[I|0]$. After the coordinate transformation, if $[R_n|T_n]$

is the pose of the second camera, from the fundamentals of rigid transformation, R_n is given by RR_0^{-1} and T_n is given by $T - RR_0^{-1}T_0$.

Similarly, the plane parameters also change after the coordinate transformation. Let $(n_n^T, d_n)^T$ be the new parameters. Using them, we have $n_n^T(R_0P+T_0)+d1 = 0$ or $(n_n^T R_0)P+(n_n^T T_0+d1) = 0$. Equating this with the original plane equation $n^T P+d = 0$, after simplification, we get $n_n^T = n^T R_0^{-1}$ and $d_n = d - n^T R_0^{-1} T_0$. Substituting R_n,T_n,n_n and d_n in Equation 5.2, we arrive at

$$H = K\left[RR_0^{-1} - (T - RR_0^{-1}T_0)\frac{n^T R_0^{-1}}{d - n^T R_0^{-1} T_0}\right]K^{-1} \tag{3.2}$$

It may be seen that when the first camera is at $[I|0]$, the above equation reduces to Equation 5.2.

If the camera motion is only translational and both rotation matrices R and R_0 are I (identity), from Equation 3.2,

$$H = I - K\left[(T - T_0)\frac{n^T}{d - n^T T_0}\right]K^{-1} \tag{3.3}$$

3.2 Space of Homographies

Almost all the mosaicing techniques [3, 4, 5, 42, 43, 44, 45] in the literature consider the homographies between different pairs of images to be independent of each other. However, in a video, the homographies between different pairs of frames may often be tightly related, depending on the camera motion. The seminal work analyzing the relationship between homographies was done by Shashua and Avidan [46]. They considered a scenario wherein all possible planes are viewed from two fixed cameras, leading to a family of homographies. These homographies are shown to be constrained in the following way.

The space of all homographies (induced by any plane) between two fixed views is embedded in a 4 dimensional linear subspace.

In other words, if we consider homography matrices $A_1, A_2, \ldots, A_k$ each as a column vector in a $9 \times k$ matrix, the rank of this matrix is 4. This implies that given any four *base* homographies (corresponding to four *base* planes), any new homography (corresponding to a new plane) can be expressed as their linear combination. This result, in the present from, is not directly relevant for the mosaicing problem. In mosaicing, the problem on hand is the dual of the one described above. The plane is fixed while the camera views are changing. We derive a rank constraint, similar to the one given by Shashua and Avidan [46], for this dual problem when the rotation is constant. While it is not powerful enough to be used directly for homography estimation, it motivates the thought that there may exist tighter and useful relationships between homographies if the video capturing induces some constraints, stemming from the camera motion.

Theorem 1 *For a fixed plane, the space of all homographies, induced by any pair of cameras, with fixed relative rotation and fixed sets of internal calibration parameters across all pairs, is embedded in a 4 dimensional linear subspace.*

Proof: We know from [39] that if $K[I|0]$ and $K[R|T]$ are the first and second camera matrices and $\pi = (n^T, d)^T$ represents the world plane (such that, for points P_i on the plane, $n^T P + d = 0$), the homography induced by the plane is given by:

$$H = K\big(R - \frac{Tn^T}{d}\big)K^{-1} \tag{3.4}$$

For the *ith* camera pair, let $[R|T_i]$ be the relative pose. The homography H_i is given by $H_i = M - \tilde{T}_i m^T$ where, $M = K'RK^{-1}$, $\tilde{T}_i = K'T_i$ and $m = \big[(\frac{n^T}{d})K^{-1}\big]^T$.

We need to prove that given four base homographies H_1, H_2, H_3 and H_4, any new homography H_j can expressed as their linear combination. In other words, we need to prove that there exist unique a_1, a_2, a_3 and a_4 such that, $\sum_{i=1}^{4} a_i H_i = H_j$ or $\sum_{i=1}^{4} a_i M - \big(\sum_{i=1}^{4} a_i \tilde{T}_i\big) m^T = M - \tilde{T}_j m^T$.

If we equate the cofficients of M and m^T, we get the following four equations in a_1, a_2, a_3 and a_4.

$$\sum_{i=1}^{4} a_i \tilde{T}_i^k = \tilde{T}_j^k, k \in 1, 2, 3 \quad and \quad \sum_{i=1}^{4} a_i = 1 \tag{3.5}$$

This set of linear equations has 4 unknowns (a_1, a_2, a_3 and a_4) and 4 equations. This implies that we can always find a unique set of a_1, a_2, a_3 and a_4 such that $\sum_{i=1}^{4} a_i H_i = H_j$. In other words, the homography space is contained in a four dimensional linear subspace.

Note that if we consider only three base homographies, the system of equations analogous to Equation Set 3.5 will have 4 equations and 3 unknowns. Thus, if all the equations are linearly independent, there is no solution i.e. a_1, a_2 and a_3, the coefficients that were presumed to express the new homography as a linear combination of the base homographies.

The constraint given by Theorem 1, while giving useful insight, can not be directly used for homography estimation except for elimination purposes. However, many videos inherently possess information that can lead to more meaningful, tighter constraints, which can be exploited for improving homography estimation. The key point in videos is that the frames are not arbitrary images, they may be tightly related. The camera that captures the video, often, undergoes a motion that has a model. This type of model-following motion is especially found in machine-controlled environments like auto-mobile factories, in-door robots and so on. We show how this model of the camera motion can be translated into a model that can characterize the homography between any pair of video frames. We call this a global homography model. The estimation of homographies no longer needs to be done separately for each pair. We can combine information from multiple frames to perform homography estimation in two passes: first, estimate the parameters of the global homography model and directly, compute the homographies from the model.

For now, we limit our focus to the linear motion model. In this section, the relationships or the models for the homographies are derived for the first two motion derivatives i.e. constant velocity and constant acceleration motions. However, they can be extended, with trivial additional work, to any number of motion derivatives. Now, we present a more generic result, which forms the backbone of the lemmas and observations to be presented subsequently.

Theorem 2 *Under constant velocity motion, comprising rotational and translational velocity, the homography between any pair of frames can be computed from a* 21 *parameter model. In presence of constant acceleration, the number of model parameters increases to* 27*.*

Rewriting Equation 3.2, the homography relating frames I_i and I_j is given by

$$H_{i,j} = K\left[R_j R_i^{-1} - (T_j - R_j R_i^{-1} T_i)\frac{n^T R_i^{-1}}{d - n^T R_i^{-1} T_i}\right]K^{-1}$$

where $K[R_i|T_i]$ and $K[R_j|T_j]$ are the camera matrices for the two frames.

Looking carefully at the components of the above equation, calibration matrix K, leaving aside radial and other complex distortions, is characterized by five parameters given by aspect ratio, focal length, principal point and the skew angle. The two poses $[R_i|T_i]$ and $[R_j|T_j]$ are characterized by twelve parameters: rotational velocity (three parameters for the three rotation angles), translational velocity (three parameters, one for each of the three axis) and initial rotational and translational positions (three parameters each). Finallly, the plane $\pi = (n^T, d)^T$ adds 4 parameters, resulting in a total of 21 parameters. Given these 21 parameters and the two indices i and j, the homography between any pair of video frames I_i and I_j can be computed, under constant velocity motion. In case of constant acceleration, we have six more parameters for rotational and translation acceleration, making the total 27.

Note that the above-mentioned relationship is non-linear. In other words, the homography can not be linearly expressed in terms of the 21 homography model parameters. If we notice carefully, it can be seen that the non-linearity arises from the rotational velocities. In the next section, we show that if we assume rotation to be constant, linear relationships can be derived for the two important types of homographies: incremental and reference. The underlying motivation behind exploring a purely translational motion is two-fold. It yields linear relationships which are more preferred as they can be exploited more robustly for video mosaicing. Secondly, we often consider the motion model to be present locally (*for a small window around each frame*), not globally (*for the whole video*). The rotation-less linear model is a valid and well-suited approximation in this approach.

It should be noted that the above relationships are much tighter than the rank 4 constraint derived by Shashua and Avidan [46]. They show that a new homography is characterized by four parameters (associated with four *base* homographies) when the two views are fixed and the plane parameters vary. Thus, for a video sequence consisting n frames, considering only successive frames for alignment, we have to estimate $4(n-1)$ parameters, four for the homography for each pair. On the contrary, in our case wherein the plane remains fixed and the camera moves according to a motion model, we have to estimate only the global homography parameters, parametrzing all the $(n-1)$ homographies. For instance, for a constant velocity translational motion, we have to estimate only 10 parameters. Even for a short sequence of 50 frames, the ratio of the number of parameters to be estimated using the constraint in [46] and that for the relationship derived above is 196 : 10, which is nearly 20. Note that the cumulated number of correspondences (i.e., samples for parameter estimation) is the same for both the cases. This complies with the intuition that if we have more precise information in form of the type of the camera motion model, we should be able to derive tighter or more robust relationships on homographies.

3.3 Homographies under Uniform Translational Motion

We derive constraints on incremental homographies $H_{i,i+1}$ and reference homographies $H_{i,r}$, for a purely translational motion. We restrict ourselves to constant velocity and constant acceleration motion models.

3.3.1 Incremental Homographies

In a video, because of the continuity of the camera motion, consecutive pairs of frames are likely to have maximum overlap. High degree of overlap is typically related to a good number of features that can be extracted. This makes incremental homography the most popular kind of homographies, used for mosaicing. Although mosaicing is done through reference homographies, which map all the frames to the reference frame, most approaches (e.g. [4, 5, 45]), first, compute incremental homographies (from the feature point correspondence). Each reference homography is then computed by multiplying a series of incremental homographies. We, now, derive the relationships between incremental homographies under constant velocity motion.

Lemma 1 *In presence of uniform translational motion, all the incremental homographies $H_{i,i+1}$ are related by a 10 parameter model:*

$$H_{i,i+1} = \begin{bmatrix} c_1 + i.c_{10} & c_2 & c_3 \\ c_4 & c_5 + i.c_10 & c_6 \\ c_7 & c_8 & c_9 + i.c_{10} \end{bmatrix} \qquad (3.6)$$

In presence of uniform acceleration, the number of parameters increases to 19*.*

Proof: For a purely translational motion, the camera moving with velocity V (and acceleration A) is equivalent to its translational pose T moving with velocity $-V$ (and acceleration $-A$).
If T_i is the translational pose of the camera for frame I_i, assuming that the time elapsed between the capturing of successive frames is constant (t), for uniform velocity motion, we have

$$T_i = T_0 - i.t.V \quad and \quad T_{i+1} - T_i = -t.V \tag{3.7}$$

Substituting the above in Equation 3.3,

$$H_{i,i+1} = I - K\left[(-t.V)\frac{n^T}{d - n^T(T_0 - i.t.V)}\right]K^{-1}$$

Substituting $Q = t.KVn^TK^{-1}$, $q_1 = d - n^TT_0$ and $q_2 = t.n^TV$ in the above equation, after rearrangement, results in

$$H_{i,i+1} = I + \frac{Q}{q_1 + i.q_2} \tag{3.8}$$

Since homography is only defined upto scale, multiplying with $q_1 + i.q_2$, $H_{i,i+1} = (q_1 + i.q_2)I + Q$. Concatenating matrices $(q_1 + i.q_2)I$ and Q, after simple rearrangement and renaming, we arrive at the desired ten parameter model. The result for uniform acceleration motion can also be proven along similar lines.

While the number of parameters of this homography model is 10, when we use it in the two linearly independent equations coming from $X' = HX$ (X and X' are corresponding (homogeneous) points in two successive frames), we get a *homogeneous* system of equations in the 10 model parameters. Thus, nine or more equations (i.e. five or more corresponding points) suffice for the homography model estimation.

It is interesting to note that when the camera motion is parallel to the world plane, all the incremental homographies are the same. This results from the fact that in presence of linear motion parallel to the world plane, $n^TV = 0$.

3.3.2 Reference Homographies

Reference homographies are used in situations wherein there exists enough overlap between every frame and the reference frame, ensuring adequately many matched feature points. The factors that make this possible are the the number of video frames considered (or the size of the video), the position of the reference frame and the nature of the camera motion. If the video size is small and reference frame is located somewhere in the middle, there is a good probability of the required overlap to be present. Also, this probability increases when the camera motion is bidirectional wherein it comes back to a viewed object. This scenario is where the graph-based approaches [5, 42, 43, 3] are applicable. They actually consider both incremental and reference homographies. In this section, we derive the global homography model for the inverse matrices of the reference homographies, which we call inverse reference homographies.

Lemma 2 *In presence of uniform velocity translational motion, all the inverse reference homographies $H_{0,i}$ are related by a 9 parameter model: $H_{0,i} = I + iK_c$ where, K_c is a 3×3 matrix. In presence of uniform acceleration, the number of parameters increases to* 18*.*

Proof: Substituting $T_i = T_0 - i.t.V$ in Equation 3.3, we directly get

$$H_{0,i} = I - K\left[- i.t.V\frac{n^T}{d^*}\right]K^{-1} \tag{3.9}$$

where, $d^* = d - n^T T_0$. Simple rearrangement of this leads to the desired equation.

$$H_{0,i} = I + i\left[\underbrace{t.KV\frac{n^T}{d^*}K^{-1}}_{K_c}\right] \tag{3.10}$$

A similar procedure can be used to show the relationship for uniform acceleration motion.

As a corollary, if the camera moves horizontally, imaging objects lying on a horizontal plane, the first order difference between the normalized inverse reference homographies is constant. In normalization, H(3,3) is set to 1.

As the camera moves horizontally, the z component of V is zero. Therefore, V is of the form $[x \;\; x \;\; 0]^T$, where x is an arbitrary unknown. Moreover, the world plane is horizontal so vector n is of the form $[0 \;\; 0 \;\; x]^T$. We also know that the calibration matrix K is affine and so is its inverse.

Now, from Equation A.2, K_c is a multiplication of three matrices - K, Vn^T and K^{-1} and two scalars. From that, it directly follows that $K_c(3,3)$ is zero. Thus, from Equation 3.10, H(3,3) - $1 + i.K_c(3,3)$ becomes one and the normalized homography is the same as what is given by Equation 3.10 ($I + iK_c$). Therefore, the first order difference $\Delta H_i = H_{0,i+1} - H_{0,i} = K_c =$ constant.

Note that when the camera moves parallel to an arbitrary plane and images objects on that plane, this corollary holds only when the camera's Z axis (which is usually the viewing axis) is perpendicular to the plane.

More detailed derivations can be found in the appendix.

3.4 Homography Model Parameters versus Number of Motion derivatives

The derivations of homography models, from Section 3.3, for constant velocity and constant acceleration camera motion, for incremental and reference homographies, for translational and general motion are summarized in Tabel 3.1.

Type of Motion	Incremental Homographies	Inverse Reference Homographies
Constant Velocity Translational	10 Parameters (Linear)	9 Parameters (Linear)
Constant Velocity General	21 Parameters (Non-Linear)	21 Parameters (Non-Linear)
Constant Acceleration Translational	19 Parameters (Linear)	18 Parameters (Linear)
Constant Acceleration General	27 Parameters (Non-Linear)	27 Parameters (Non-Linear)

Table 3.1: The number of global homography model parameters under various motion models, for incremental and inverse reference homographies. Here, the term linear (or non-linear) means that the homographies are linearly (or non-linearly) expressed in terms of the global homography model parameters.

These derivations for the first and second order motion derivatives (i.e. velocity and acceleration, respectively) can be readily extended to higher order motion derivatives. In Table 3.2, we list the number of parameters requited to characterize homographies as a function of the number of motion derivatives (k).

	Translational Motion	General Motion
Incremental Homographies	$9k + 1$	$6k + 15$
Reference Homographies	$9k$	$6k + 15$

Table 3.2: The number of global homography model parameters as a function of the number of motion derivatives (k)

Table 3.2 shows that for both translational and general motions, the number of model parameters required to represent all the homographies (incremental and reference) is linear in the number of motion derivatives. This implies that the ensemble of homographies for any complex motion (with arbitrarily high but finite motion derivatives) can be compactly expressed using the framework that we have proposed.

3.5 Role of Homography relationships in Mosaicing

As shown in Chapter 2, mosaicing, especially in case of videos, is error-prone in many situations if the homographies are estimated independently for each pair of video frames. Thus, the homography relationships or the homography model, derived in this section from the camera motion model, can be capitalized to estimate all homographies jointly. Summarizing from the above sections, we have derived the following theorem.

Theorem 3 *The homographies induced by a fixed plane, between pairs of video-frames are related by a fixed set of parameters, depending on the camera motion model.*

The above results imply that a large number of homographies can be computed from a fixed, (usually) small number of parameters. Therefore, rather than using pair-wise information to compute the homographies individually, the information from all the frames can be pooled in to estimate the parameters of the global homography model. The desired homographies can then be directly computed from these parameters. These homographies can be used to register and blend the video frames to generate the mosaic. Mosaics generated this way are unlikely to suffer from registration errors as registration is done simultaneously wherein a large number of samples, coming from many frames (not just one pair of frames), contribute to it.

Chapter 4

Video Mosaicing with Motion Constraints

In the previous chapter, the motion model of the camera was utilized to relate homographies. we argued that these relationships can be exploited for robust registration, giving better quality mosaics. Before presenting an algorithm utilizing the homography relationships, we look at the types of videos in which mosaicing finds applications, in the next section. In the remainder of this chapter and the next chapter, we present mosaicing algorithms which, put together, span the whole breadth of video types.

4.1 Types of Videos for Mosaicing

We consider two types of classifications which are relevant to the work. The first type of classification of videos deals with the motion of the camera corresponding to the video. The major categories based on the camera motion are listed below. Most of them have been discussed before.

1. The trajectory of the camera is smooth. This is the trait of almost real videos (e.g., videos taken by cameras mounted on aircrafts and hand-held cameras). Chapter 5 presents an algorithm which utilizes the smoothness of the camera trajectory.

2. The camera follows a motion model. This type of motion model is especially seen in automated environments. Camera mounted on robots and cameras found in automobiles have this property. In this chapter, we use the motion constraints derived in Chapter 3, for robust video mosaicing.

3. The camera motion arbitrary (e.g., video sequences from unknown sources). To mosaic this category for videos, we can use the existing approaches, some of which were detailed in Chapter 2.

Another important category of videos depends on the availability of calibration with the video. Calibration [39] has two parts: intrinsic parameters and extrinsic parameters. Based on that, videos for mosaicing are divided as follows:

1. Full (internal and external) calibration is available (e.g., machine-controlled cameras, as in the case of robots and autonomous vehicles and aerial imaging). However, external parameters are often unreliable.

2. Only internal calibration is available (as in the case of hand-held, personal digital cameras).

3. Calibration is completely unavailable (e.g., video sequences from unknown sources).

Our approach caters to the above imaging configurations. Chapter 5 outlines an approach which uses the internal calibration for refining homographies. if it is unavailable, we have the uncalibrated approach, presented in this chapter.

4.2 Mosaicing Approaches with Global Alignment

Considerable work ([3, 21, 47, 48, 49]) has been done in the past to address some of the mosaicing issues mentioned in Section 1.4. RANSAC attempts to solve the problem of false matches by determining and discarding outlier correspondences. However, its success is greatly undermined when the number of correspondences in the image pairs is small. In such a situation, the size of RANSAC's *minimal set* (for homography, 4 pairs of corresponding points [39]) becomes quite close to the number of data points, resulting in premature termination of the computation. The advances in recent years have, to some extent, countered some of these issues. These approaches can be broadly categorized into the following two groups:

***1. Graph-based Approaches*:** Temporally non-successive frames can also turn out to be better candidates for robustly estimating homographies. The graph-based global alignment approaches [5, 42, 43, 3] extract and utilize this information in an attempt to address some of the above-mentioned issues. Typically, the graph-based approaches can be decomposed into two major steps. First, homographies are computed for all pairs of images with sufficient overlap and a graph-based representation is built from it. In the second step, the problem of global registration is cast as the identification of an optimal structure (Minimum Spanning Tree being the most commonly used) in the graph. The desired homographies are uniquely characterized by this graph structure (e.g., MST). The graph-based approaches are well-suited for bi-directional camera motions where the camera comes back to an object seen before. However, for uni-directional motions, it more or less degenerates to the conventional register-only-successive-video-frames approach because of the lack of overlap between non-consecutive frames. Moreover, optimality in the true sense is not

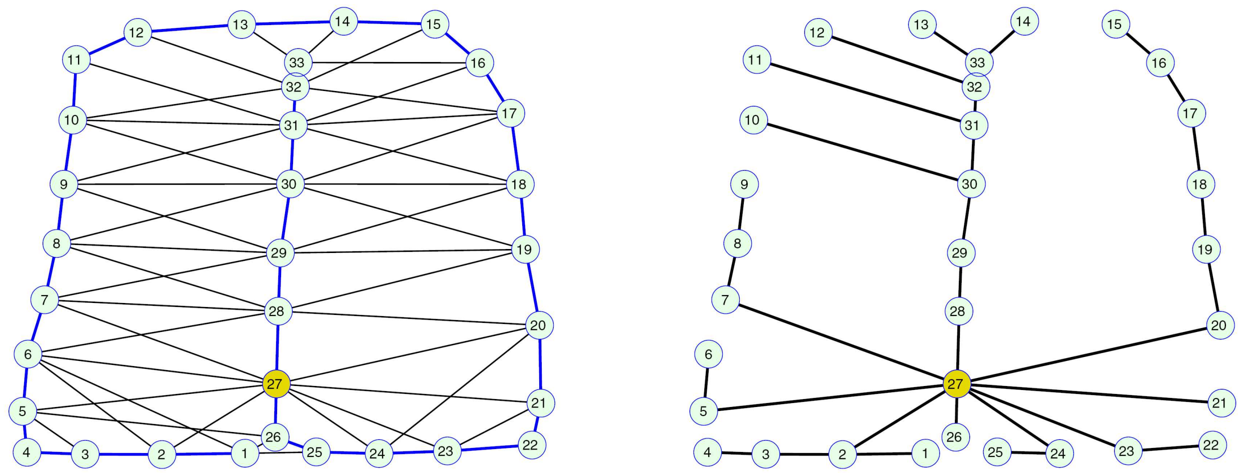

Figure 4.1: Left: Video represented as a graph. The nodes denote the video frames. The blue (bold) edges link consecutive frames in the video sequence. Black (thin) edges are drawn between nodes corresponding to overlapping frames. Right: Spanning tree composed by all shortest paths from each node to the root (27). Courtesy: [3].

obtained in this framework as it is only demanded for the selection of the graph structure (e.g., MST), not demanded for the pair-wise homographies.

***2. Approaches using Bundle Adjustment*:** The principle behind using bundle adjustment is to combine features from all frames and refine all the homography parameters at once. As demonstrated in [21, 3, 50], bundle adjustment helps overcome the phenomenon of error propagation i.e., the errors in registering the initial pairs of frames being carried forward to all the subsequent ones. Typically, the closed form estimate of the homography parameters is given as the initial estimate for bundle adjustment. Sometimes, the large magnitude of parameters to be estimated can become a limiting factor in the bundle adjustment formulation. Also, it is heavily dependent on the starting estimate.

In summary, while the above approaches alleviate issues like error accumulation and to some extent, noisy correspondence, they are rendered somewhat powerless when video frames yeild features that are inadequate in number, erroneous or mis-matched.

4.3 Formulation Exploiting Homography Relationships

In this section, we discuss how the relationships between homographies, derived in the previous chapter, can be used to mosaic a video. The quality of the mosaic is directly related to the quality of homographies between video frames. The conventional techniques use the point correspondence between successive frames to individually compute the incremental homographies. However, we argue that these homographies are not independent of each other and the relationship between them can be exploited to achieve robustness.

Recollecting from the previous section, all the incremental homographies can be expressed in terms of a fixed parameter model. In other words, given these parameters, the series of incremental homographies can be directly computed from them. Thus, the problem of robust estimation of the ensemble of homographies boils down to accurate computation of the model parameters. We propose a method which uses correspondences from all consecutive pairs of frames pooled in, to estimate the model parameters.

Assume the number of correspondences between all consecutive frames to be the same (m). The conventional approach of individually estimating homographies can be thought of as using m samples to estimate a model (homography) of 8 parameters. This is done $N-1$ times. In contrast to this, the proposed approach uses $m(N-1)$ samples to compute a maximum of 21 parameters. Under constant velocity translational motion, this number goes down to 11. Compared to the conventional approach, the ratio of samples available to parameters to be computed is very high for our approach. This ratio is directly proportional to the accuracy of parameter estimation. Thus, the model-based approach clearly estimates the homographies more accurately.

It is especially suitable for situations when the camera motion is controlled. To facilitate better understanding of the model-based approach, we begin with a simple model — the linear, translational camera motion model. We know from the previous chapter that under linear translational motion, all incremental (frame-to-next frame) homographies are related by an 11 parameter model:

$$H_{i,i+1} = I + \frac{C}{c_1 + i.c_2} \tag{4.1}$$

where, C is a 3×3 matrix and c_1 and c_2 are scalars. This directly leads to

$$X_{i+1}^{j} = \left[I + \frac{C}{c_1 + i.c_2} \right] X_i^j \tag{4.2}$$

where, X denotes a feature point, i denotes the frame number and j is the feature point index in the given frame.

It can be seen from the above equation that each corresponding pair of points gives two linearly independent equations in the model parameters C and c_1, c_2. Thus, we use SVD to solve this over-determined linear system of equations in the model parameters. Mostly, the solution given by this linear technique is satisfactory. If not, it may be used as the initial guess to a non-linear, iterative algorithm like the Levenberg and Marquardt algorithm. This simple technique is generalized to cater to any arbitrarily complex homography model. The steps of this generic algorithm are given in Algorithm 4.

Algorithm 4 Build mosaic using a camera motion model

1: Establish correspondences between all pairs of frames.

2: Estimate the homography model parameters M_p such that

$$E = \sum_{i,j} \|(\Pi(X_i) - \Pi(H(M_p, i, j)X_j))\|^2$$

is minimized. (Use a non-linear optimization technique such as the Levenberg-Marquardt algorithm to do so.)

3: Identify samples, for which reprojection error is greater than twice the average reprojection error, as the outliers.

4: Reestimate the model parameters M_p from the inliers alone.

5: Compute the reference homographies from the model and build the mosaic

Algorithm 4 exploits the constraints on the homographies relating pairs of frames (Theorem 1) for error correction and refinement of the homography estimates from the image measurement. First, the global model (M_p) is estimated from the measurements from the individual frames and the homography is computed directly from. i.e., $H_{i,j} = f(i, j, M_p)$, where M_p is the set of parameters of the global homography model. Note that the noisy image measurements will have far lesser influence on the estimation of $f(\cdot)$, compared its influence on the individual homography estimates. This helps us in providing better mosaics, by not worrying about the failure in correspondence in a specific frame pair.

When there is not enough, relevant visual content in certain frames, the correspondence computation and thereby, the homography estimation fails. This often results in the mosaic getting visually distorted. Our algorithms take care of these situations effectively. Note that the global camera motion model, employed in the previous algorithm, can easily be replaced by several local motion models. For instance, it is, very often, safe to assume that the motion is uniform and smooth in a small continuous set of frames.

Note that the proposed approach inherently handles object occlusion as we do not need to track points across frames. The correspondence set for frames I_{i+1} and I_{i+2} can have entirely different points than the correspondence set for frames I_i and I_{i+1} has. Image zooming and rotation are also accounted for, under the general linear (and acceleration) motion model of the camera.

The formulation used in [50] also parameterizes homographies. Homography is represented by three rotation angles and focal length to recognize images that are a part of the panorama and then, to generate the panorama. It is applicable for panoramic mosaicing where the camera rotates about its optical center, enabling them to use a four parameter homography representation. Our work is aimed at planar mosaicing and accounts for general camera motion.

4.3.1 Weighted Least Squares Error

It should be noted that the two linear equations that we obtain from Equation 4.2 contain the frame index number i in the coefficients. Thus, if there is noise in the extraction of feature points, it gets multiplied by this frame index number i. This means that frames farther from the starting frame introduce significantly more noise than the frames near by. Therefore, if we minimize the normal least square error (which is what the SVD-based closed-form technique minimizes), it leads to the a poor fit for the model parameters. To suppress this phenomenon of noise amplification, we weight each equation by the inverse of i and then, perform SVD. This simple and cost-effective strategy yields significantly improved performance. In case a non-linear optimization is applied on top of this, the $\frac{1}{i}$ term is incorporated in the objective function - the average reprojection error (Equation refer changes to the equation below).

$$E = \frac{1}{m * (N-1)} \sum_{i,j} \| \frac{1}{i} (\Pi(X^j_{i+1}) - \Pi(H_{i,i+1} X^j_i)) \|^2 \tag{4.3}$$

4.4 Performance Analysis

Since homographies are characterized by a fixed set of parameters, the information from all the frames is pooled to estimate the parameters of the global homography model. The desired homographies are directly expressed in terms of these parameters. Thus, the problem of homography estimation essentially is reduced to that of (homography) model fitting, given samples from multiple video frames.

Note that estimation of a pair-wise homography from point correspondence can also be viewed as model fitting, wherein an 8-parameter model (i.e. homography) is fitted to a subset of matching points. However, the ratio of the number of samples available to the number of parameters to be estimated, is much lower in this case compared to the previous case where we have samples from all the frames for estimating the parameters of the global homography model. Thus, we claim that our approach is much more accurate than the conventional pair-wise approach. It is more tolerant to noise in the image measurements and the presence of poor-quality frames. In the rest of this section, we validate these claims empirically.

The videos that we have used for performance analysis are synthetically generated. We generate world points on a plane randomly and project them using camera poses that follow each of the constraints discussed.

4.4.1 Sensitivity to noisy image measurements

We analyzed the sensitivity of our uncalibrated algorithm to noise in the image measurements. Uniform noise was added to the set of point correspondences. Fig. 4.2 shows that for our approach, the reprojection error increases at a much slower rate with the level of noise, compared to the conventional approach wherein each pair of consecutive video frames was registered independently using RANSAC-based homography estimation, detailed in Chapter 2. Applying bundle adjustment on the RANSAC-given homographies also could not bring the error down.

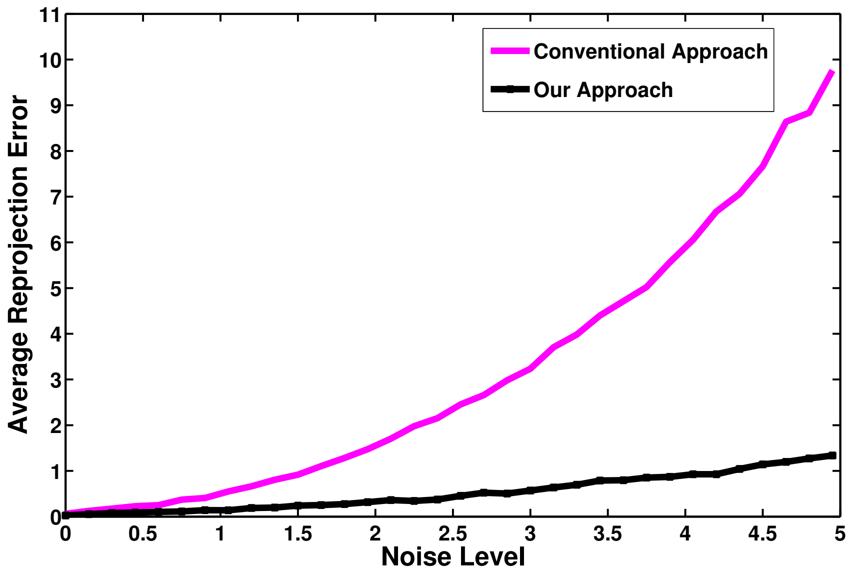

Figure 4.2: Cumulative Average Reprojection Error Vs Noise. The error for our uncalibrated algorithm is always lower than that for the conventional approach.

4.4.2 Sensitivity to poor quality frames

One frequently occurring practical problem, in a long video, is the presence of some poor quality frames. Homographies involving these frames can not be estimated accurately by the conventional pair-wise approach. This happens because of the errors in correspondence for these frames. Since our uncalibrated approach combines information from multiple pairs of frames, it is much more tolerant to this problem. To verify this assertion, we conducted an experiment wherein the quality of one of the frames was degraded. Table 4.1 shows that our solution is much more tolerant to the above problem, compared to the one given by the conventional technique. Note that some poor quality frames can be discarded from the mosaicing process, provided the frames on either side of each of them have adequate overlap for accurate registration.

This approach has also been tested on real videos. The results are shown in Chapter 6. Quantitative as well as qualitative results are shown there. This asserts our claim that exploring relationships between homographies can lead to much more robust registration and therefore, much more accurate mosaics, compared to the mosaics obtained through independent homography estimation of

	Conventional Approach	Our Approach
Average Re-projection Error	5.7898	1.7625

Table 4.1: Average Re-projection Error when one frame is of poor quality. Observe the significantly lower value for our approach

each video pair.

Chapter 5

Mosaicing by Interpolation in Motion Space

The uncalibrated mosaicing approach, presented in Chapter 4, makes use of the fact that the homographies for video mosaicing are related by a global model. Typically, the motion model followed by the camera leads to this global homography model (shown in Chapter 3). This kind of parameterized homographies is not present in all videos, as mentioned before. However, another property of the camera option - smoothness is present in almost all videos. In this chapter, we exploit this property to refine outlier homographies and enhance the mosaic quality.

5.1 Homography Subspaces

In a video, the homographies between different pairs of frames may often be tightly related, depending on the camera motion. In [46], Shashua and Avidan showed that the space of all homographies (induced by any plane) between two fixed views is embedded in a four dimensional linear subspace. This result, in the present from, is not very relevant for the mosaicing problem. In mosaicing, the problem on hand is the dual of the one described above. The plane is fixed while the camera views are changing. A relevant work was done by Zelnik-Manor and Irani [51] wherein a constraint was derived on relative homographies for a pair of planes, over multiple camera views. It was proven that the collection of all relative homographies of a pair of planes (homologies), across multiple views, spans a 4-dimensional linear subspace.

While the above-mentioned constraints provide useful insight, what is of interest for mosaicing is to investigate if there exists a relationship between homographies mapping images of a single plane, captured from multiple views. In Fig. 5.1(a), the homographies between successive frames of three videos are projected onto three of the eight dimensions of homography. The figure indicates the homographies to be independent of each other, in each video. However, when we performed PCA

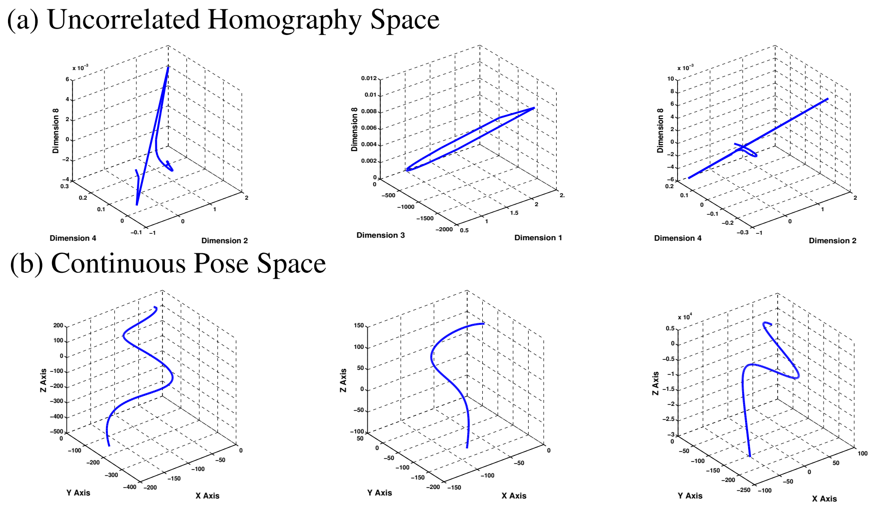

Figure 5.1: (a) Homographies between successive frames projected onto three of the eight dimentions (b) Camera trajectories for the corresponding videos. Even though the homographies seem to have no correlation at first, they are related by the corresponding smooth camera motion.

on all the homography vectors, in the transformed domain, the top two vectors contained nearly all the information. This was a telltale sign that the homographies shared some relationship. Upon further exploration, we found that the relationship between homographies was ascribed to the smooth motion of the video capturing camera (See Fig. 5.1(b)). This smooth camera trajectory, actually, is a trait of almost all real-life videos. Smoothness in the motion can be used to refine outlier homographies, by appropriately extrapolating the neighboring ones. If the camera motion follows a model, this a-priory knowledge can be used to bind the homographies much more tightly and estimate them more robustly.

5.2 Formulation using Smoothness of the Camera Trajectory

We utilize the smoothness property of the camera motion to detect and replace outlier homographies, for robust video mosaicing. The emphasis is laid on replacing inaccurate homographies, and *not* on detecting and removing blurred (outlier) frames which is a popular and well-researched problem, especially in the tracking literature (e.g., Nister's tracker [18]).

The substitute for an outlier homography is computed using the smoothness property of the camera motion. Smoothness in the pose space does not translate into smoothness in the homography space due to the non-linearity in the pose-to-homography relationship. Therefore, in order to refine an outlier homography, we move onto the pose space via homography decomposition, extrapolate in the pose space and finally, reconstruct the desired homography.

The relationship between homography and pose is the following [39]:

$$H = K\big(\tilde{R} - \frac{\tilde{T}\tilde{n}^T}{\tilde{d}}\big)K^{-1}$$

where, the pose $[\tilde{R}|\tilde{T}]$, the plane normal $\tilde{n}$ and the (perpendicular) distance to the plane $\tilde{d}$ are all relative to the first camera. For video mosaicing, we use the basics of rigid transformation to express the current incremental homography $H_{i,i+1}$ relating frame I_i to frame I_{i+1}, in terms of the *current* pose $[R_i|T_i]$, the *next* pose $[R_{i+1}|T_{i+1}]$, n and d, all defined in a general (fixed) world coordinate system.

$$H_{i,i+1} = K\left[R_{i+1}R_i^{-1} - (T_{i+1} - R_{i+1}R_i^{-1}T_i)\frac{n^T R_i^{-1}}{d - n^T R_i^{-1} T_i}\right]K^{-1} \tag{5.1}$$

such that $\tilde{R}_i = R_{i+1}R_i^{-1}$, $\tilde{T}_i = T_{i+1} - R_{i+1}R_i^{-1}T_i$, $\tilde{n}^T = n^T R_i^{-1}$, $\tilde{d} = d - n^T R_i^{-1} T_i$ and K is the internal calibration matrix.

We decompose the each incremental homography, as in [52] (detailed in Section 5.2.1), and use the decomposed parameters to compute the next pose, given the current pose (and d), using the above-mentioned, general homography-pose relationship. Assuming the first camera pose to be $[I|0]$, this allows us to reconstruct the complete camera motion, from only homographies, which are computed from image measurements. Note that no 3D information is assumed to be known other than an approximate estimate of d.

Smooth camera motion implies that the rotation angles and camera position (given by $P_i = -R_i^{-1}T_i$) vary smoothly for successive pairs of frames. For each successive pair of frames, we use this and the reprojection error, to determine if the homography for the pair is an outlier. The cumulative average reprojection error is calculated as

$$E = \frac{1}{m * (N-1)} \sum_{i,j} \|\Pi(X_{i+1}^j) - \Pi(H_{i,i+1}X_i^j)\|^2 \tag{5.2}$$

where, $i \in \{0, 1, \ldots, N-1\}$ denotes the frame number and $j \in \{0, 1, \ldots, M-1\}$ is the feature point index and Π is the imaging function.

If a homography is an outlier, we individually extrapolate in the space of rotation angles and in the position space to obtain a locally smooth estimate of the next pose. From these, along with the previously computed n, a reasonable estimate for the outlier homography is directly computed. The whole procedure is summarized in Algorithm 5.

This approach makes use of internal calibration. It is reasonable to take the internal camera parameters to be known. This is because very often, the cameras used for mosaicing come pre-calibrated (e.g., machine-controlled cameras, as in the case of robots and autonomous vehicles and aerial imaging, hand-held, personal digital cameras, etc.).

Algorithm 5 Compute homographies enforcing smoothness of the camera motion

for $i = 0$ to $N-2$ **do**
  Compute $H_{i,i+1}$ from point-correspondence.
  $[\tilde{R}_i, \frac{\tilde{T}_i}{\tilde{d}_i}, \tilde{n}_i] \leftarrow \text{Decompose}(H_{i,i+1})$
  Compute R_{i+1} and T_{i+1} from the above (decomposed) parameters and $[R_i\ T_i]$ i.e. the pose computed in the previous iteration.
  (Pose in the first iteration is $[I\ \mathbf{0}]$)
  Compute P_{i+1} and [*roll pitch yaw*] from R_{i+1} and T_{i+1}
  Check reliability of $H_{i,i+1}$ using the average projection error and displacement in the camera position
  if $H_{i,i+1}$ is an outlier homography **then**
    Extrapolate in the P-space and space of [*roll pitch yaw*] (in a suitable window) to obtain reliable estimates of R_{i+1} and T_{i+1}.
    Compute the new estimate of $H_{i,i+1}$ using camera internal parameters K, R_{i+1} and T_{i+1}.
  end if
end for

5.2.1 Decomposing Homography into Pose

Once the homography has been estimated, we can extract R, $\frac{t}{d}$ and n [52]:

$$H \Longrightarrow \{R, \frac{t}{d}, n\}$$

Note that the translation is estimated up to a positive scalar factor. For many applications, it is common to assume d to be one and include this factor in the translation vector. For mosaicing, we do not make that assumption. We tackle this problem of relative translation in the formulation itself. However, for the ease of understanding of how the decomposition ambiguity is resolved, in this section, we make the assumption that d is one.

In [52], the homography is decomposed using a numerical method (involving Singular Value Decomposition). In the general case, there exist four solutions, two of them being the opposites of the others:

$$Rtn_a = \{R_a, t_a, n_a\};$$

$$Rtn_{a-} = \{R_a, -t_a, -n_a\};$$

$$Rtn_b = \{R_b, t_b, n_b\};$$

$$Rtn_{b-} = \{R_b, -t_b, -n_b\}.$$

These can be reduced to only two solutions applying the constraint that all the reference points must be visible from the camera (visibility constraint). Without loss of generality, we can assume that the two solutions verifying this constraint are Rtn_a and Rtn_b and that, among them, Rtn_a is the true solution. In practice, in order to determine which one is the good solution, we generally use an approximation of the normal n.

5.3 Verification and Analysis

We have presented an algorithm to compute substitutes for outlier homographies, by going into the pose space using homography decomposition and using the smoothness property in the pose space. The role of the smoothness property, in essence, is to allow us to use pose extrapolation. In this section, we confirm that extrapolation in the pose space can be done successfully, resulting in homographies getting refined. Since this is a calibrated approach and the available (intrinsic) calibration, on some occasions, may be somewhat noisy, we analyze how the deviations from actual calibration (and the actual distance from the plane) affect the performance of the algorithm.

5.3.1 Verification of homography refinement

First, we verify the effectiveness of the proposed (calibrated) technique on a video with smooth camera motion. Frame-to-frame homographies were computed between every pair of successive frames using the conventional technique. It yielded poor homography estimates for some pairs. This may be seen as large errors in the reprojection values at certain frames in Fig. 5.2(a). Camera trajectory is computed from the estimated homographies by their decomposition. As they should, the homography errors propagate into this procedure. In Fig. 5.2(b), one can see the the homography errors reflected in the camera trajectory. Notice that this camera trajectory is not smooth in many places. After applying our approach of extrapolation in the pose space, the camera trajectory becomes smooth (Fig. 5.2(c)). We use these reconstructed poses to compute substitutes for the outlier homographies, and then, perform the mosaicing. The reduction in the reprojection errors (See Fig. 5.2(d)), after applying our calibrated algorithm, may be observed.

5.3.2 Sensitivity to calibration errors

We study the effect of erroneous calibration parameters and the estimate of d on output homographies with an experiment wherein each parameter was perturbed by an increasing percentage of its actual value. Fig. 5.3 shows that the calibration error does not significantly affect the reprojection error and consequently, the mosaic quality. This could be attributed to the fact that we decompose the homography into pose, extrapolate in the pose space and then, reconstruct the homography, meaning that the errors/inaccuracies in calibration (and d) get canceled out in the whole process.

This approach, presented in this chapter, takes care of the commonly encountered problem of outlier homographies. It typically stems from some isolated poor quality frames. These poor quality frames often stem from motion blur. We note again that these poor quality can not always be removed because they may be needed to go into the mosaic to maintain visual continuity. It should also be noted that graph-based approaches can also refine outlier homographies. However, it is subject to there being sufficient overlap between (non-consecutive) frames on either side of the set of frames causing outlier homographies, not always the case.

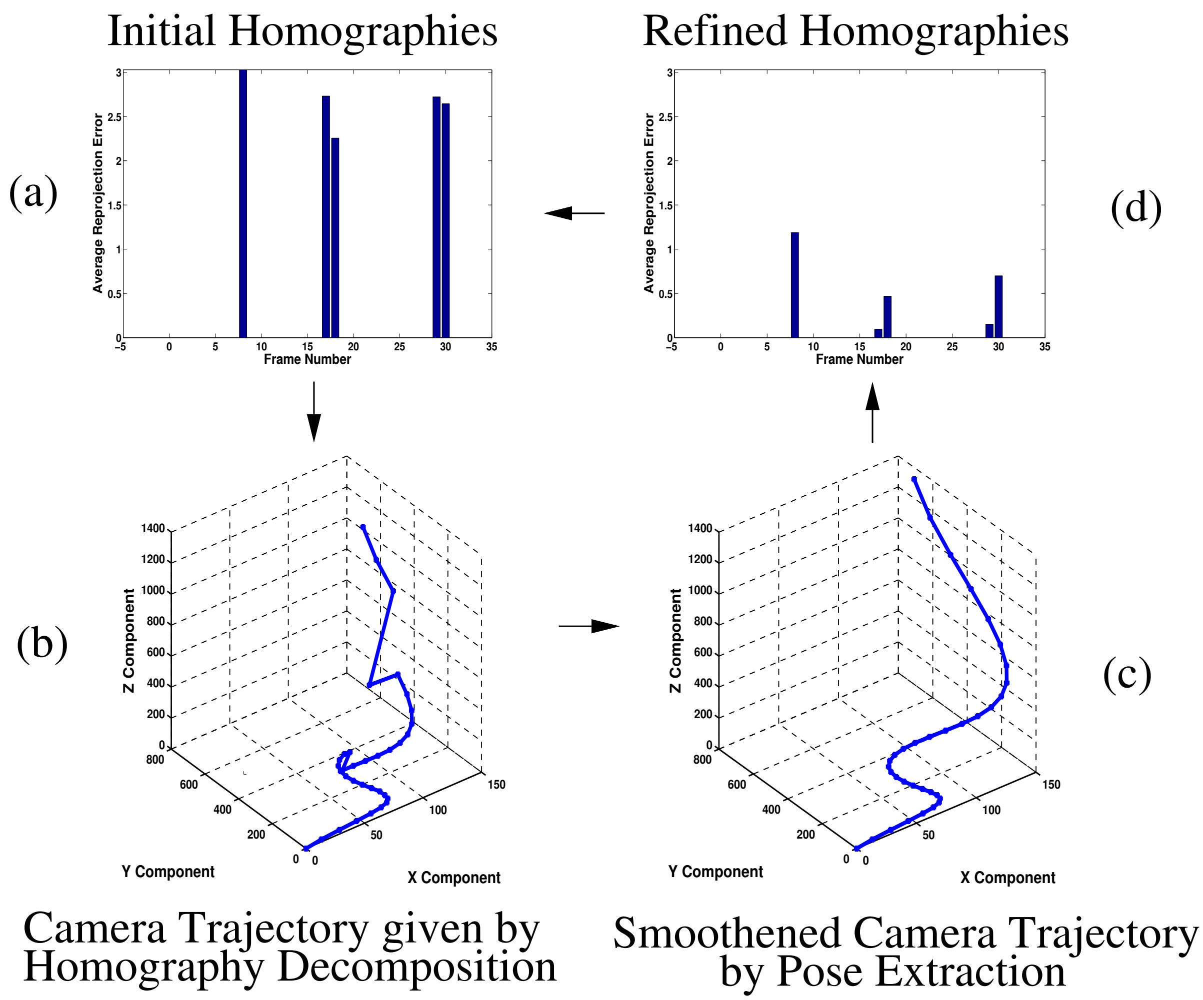

Figure 5.2: (a) and (d) show the reprojection errors for incremental homographies, before and after applying our algorithm exploiting the smoothness property of the camera motion, respectively. (b) is the camera trajectory computed by decomposing homographies. We smoothen it (shown in (c)) and recompute homographies using the refined poses and the known calibration matrix, resulting in low reprojection errors, as seen in (d).

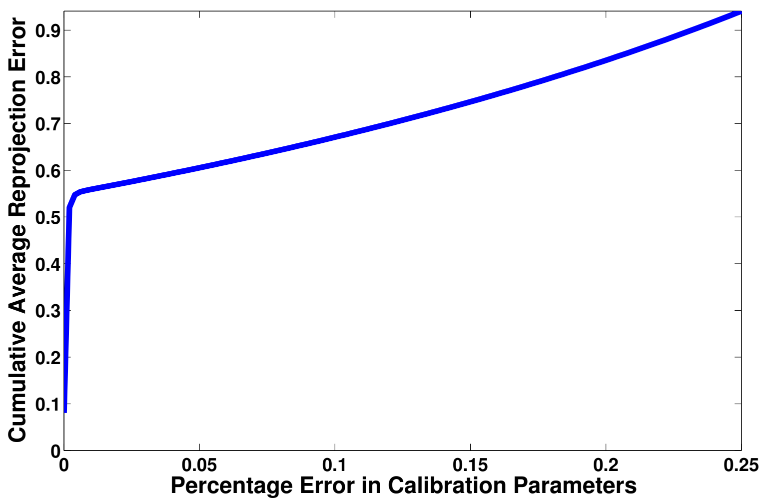

Figure 5.3: Cumulative average reprojection error Vs percentage error in the calibration parameters, for our calibrated algorithm using motion smoothness

Chapter 6

Evaluation of Mosaics: Qualitative and Quantitative

Thus far in the thesis, detailed analysis and verification of the proposed approach has been presented. The effect of the proposed approach on the quality of mosaics is illustrated in this chapter, in full detail. A good number of visual results are presented on various types of videos. However, we do not limit to visual perception which may vary from person to person. We propose a new quality metric for mosaics for quantitative comparison of the mosaics generated by our approach using camera motion properties over the ones generated using existing methods.

6.1 A New Quality Metric

With the advance of mosaicing techniques in the last decade, the average mosaic quality has gone quite high. Thus, a quality measure need not distinguish between a perfectly accurate and a grossly inaccurate mosaic. The desirable trait for a quality measure for mosaics is to discriminate between two reasonably accurate mosaics. In other words, it should be able to detect fine differences between two mosaics which, to the human eye, look similar, from a distance.

To this effect, we have formulated a simple, low-cost and yet, effective measure, aimed at meeting the desirability stated above. This measure is evaluated while stitching each frame onto the current mosaic, bypassing the need for an additional computational step. For each frame, compute the overlapping region between the current mosaic and the current frame in the mosaic coordinate frame. For this overlapping region, compute a measure of similarity between the mosaic-patch and the frame-patch. These similarity values are summed over all frames to compute a global mosaic quality value. In order to make the quality metric more discriminative, we employ Fisher's Z transform on the similarity values computed, before adding them up.

6.1.1 Similarity Measure

Although the algorithm presented here is not strictly dependent on any single similarity measure, Normalized Cross Correlation is our metric of choice as its invariant to illumination changes to a great extent. More conventional measures of similarity between image patches like sum of squared differences (SSD) do not account for changes in illumination. Normalized Cross Correlation is our metric of choice for . For any pair of images $I_2(x)$ and $I_1(x)$, the correlation coefficient r_{ij} between two patches centered at location (x_i, y_j) is defined as

$$r_{ij} = \frac{\sum_{w_x}\sum_{w_y}(\phi_2)(\phi_1)}{\sqrt{\sum_{w_x}\sum_{w_y}(\phi_2)^2\sum_{w_x}\sum_{w_y}(\phi_1)^2}}$$

where,

$$\phi_1 = I_1(x + [w_x \quad w_y]^T) - \mu_1,$$

$$\phi_2 = I_2(x + [w_x \quad w_y]^T) - \mu_2,$$

w_x and w_y are the dimensions of the local patch around (x_i, y_j), and μ_1 and μ_2 are the patch sample means. Correlation is measured between patches around registered frame and the current mosaic and this local correlation is then summed to get a global estimate of quality. However, before such an addition is performed, Normalized Cross Correlation has to be converted into an additive measure. Additivity implies that the average of similarity coefficients in a number of samples represents an average correlation in all those samples. This condition has two consequences. First, that the similarity magnitude rather than the similarity measure is used (hence avoiding a canceling effect on summation of negative and positive measures), and second, that the similarity measure should vary linearly with the magnitude of correlation. This requirement is often ignored while summing correlation measures, resulting in inaccurate assessments of average similarity.

Fisher's Z transform:

The Normalized Cross Correlation coefficient is not a linear function of the relational magnitude between the images, and as a result, correlation coefficients cannot simply be averaged. As a statistic, r has a sampling distribution. (If n sample pairs from two signals were correlated over and over again the resulting distribution of r scores would form a sampling distribution.) This distribution has a negative skew, which is the sampling distribution of r . It can be observed that although no value of r ever exceeds 1.0, there is a bias towards lower values of r . A transformation called Fisher's Z-transformation converts r to a value that is normally distributed and is defined as

$$z_{ij} = \frac{1}{2}(ln(1 + |r_{ij}|) - ln(1 - |r_{ij}|))$$

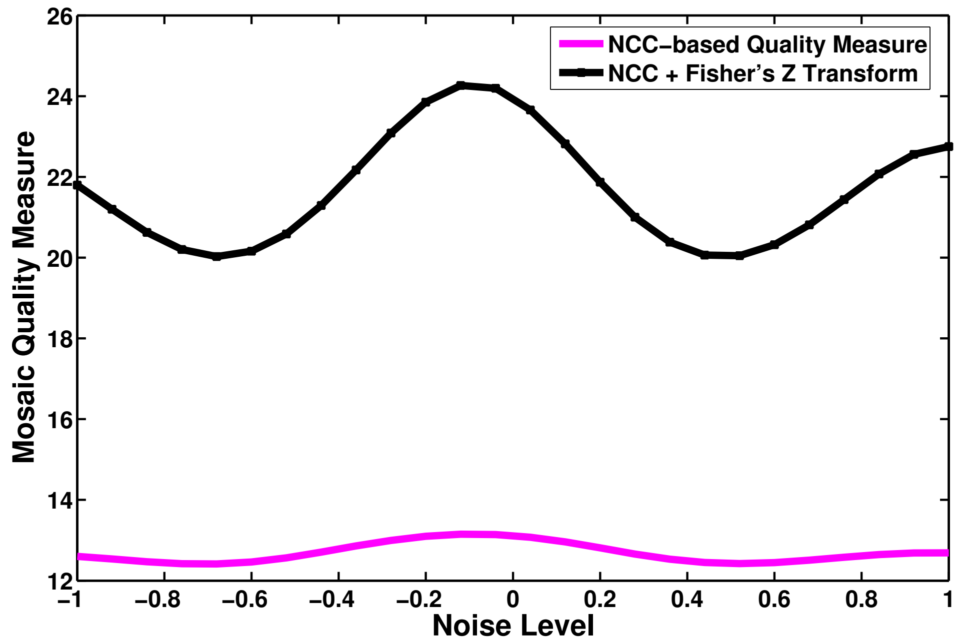

Figure 6.1: Mosaic quality measures versus noise in the homographies. Noise is induced by systematically perturbing the correspondence used for homography estimation. Observe that the usage of Fisher's Z transformation makes the quality measure more discriminative and smoother.

As a result of having a normally distributed sampling distribution, there is an equal probability of detecting any correlation score and hence they can be meaningfully added.

6.1.2 Performance

As can be seen from Fig. 6.1, the measure involving Fisher's Z transformation is more discriminative than the purely NCC-based one. Moreover, it is smooth which is an important property if it is to be used for gradient-based optimization. Another advantageous trait to that effect is that it has no local maxima. We note that there are a number of quality measures which attempt to capture the sharpness or the focus of the mosaic (e.g., [3]). Our approach differs from those in that it tries to capture the quality of alignment, the basic factor affecting the mosaic quality.

6.2 Visual Results

6.2.1 Desert Area

In Fig. 6.4, we present the mosaic of an aerial video of a desert area. There is little texture information in the input frames. Thus, as expected, the feature extractor was not able to find

Figure 6.2: Sample frames of an aerial video over a desert

enough (accurate) points in the frames. We use the KLT tracker to detect and match points for a pair of images. Therefore, the conventional technique, using RANSAC on successive pairs of frames, could not perform accurate registration. Employing the graph-based approach (as in [3]), wherein homographies between all overlapping frame pairs were estimated, yielded an MST wherein almost all the edges were between nodes corresponding to successive frames, thereby, giving virtually no improvement. Using bundle adjustment, with an objective function similar to the one used in [3, 43], also failed to enhance the mosaic quality. However, our model-based approach performed very well since it combines information from multiple frame pairs for computing each homography. A part of the mosaic is zoomed and shown. Note that the graph-based approach does not improve on the pair-wise RANSAC result noticeably because of the uni-directional motion, as mentioned in Section 4.2. One more result is shown in Fig. 6.7.

6.2.2 Repetitive Texture

Fig. 6.10 demonstrates that our approach tackles the frequently encountered repetitive texture problem very well. The existing techniques failed to compute accurate homographies for many frames of this video because of several false matches in the feature matching step. However, since our approach accumulates correspondence from many pairs of video frames, it was less affected by the mis-matched points and was able to yield quite accurate homographies and consequently, a better mosaic. One more result is shown in Fig. 6.13.

6.2.3 Planar Map

In the next experiment, we tested our approach in an in-door setting. We considered a different application domain and attempted to mosaic the video of a map of India. For the existing techniques, the alignment errors render certain characters unreadable, unlike in the mosaic built by our approach (Fig. 6.16).

Figure 6.3: Mosaic built using the graph-based approach

Figure 6.4: Mosaic generated using our approach. Our algorithm is able to produce good quality mosaics even if features to track are rare.

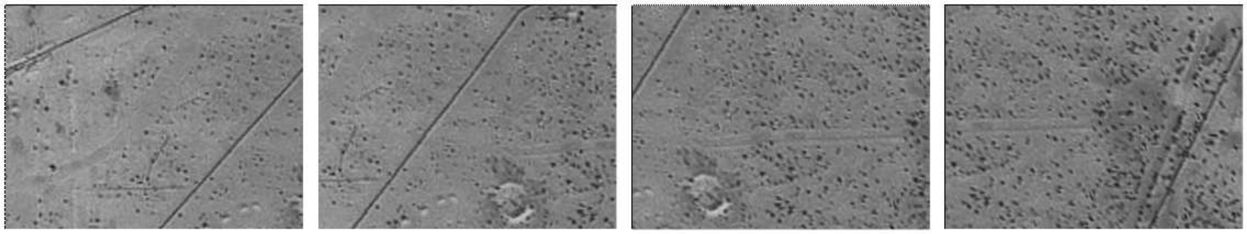

Figure 6.5: Sample frames of an aerial video over a desert

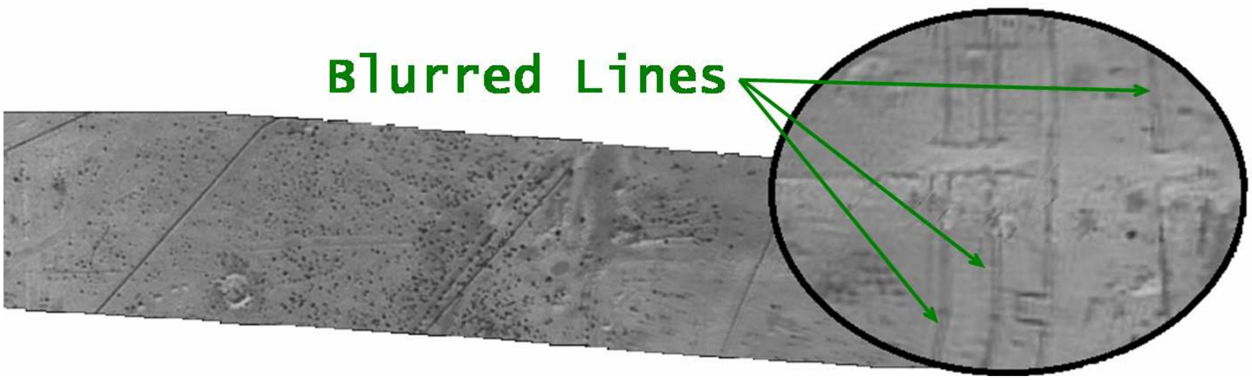

Figure 6.6: Mosaic built using the graph-based approach

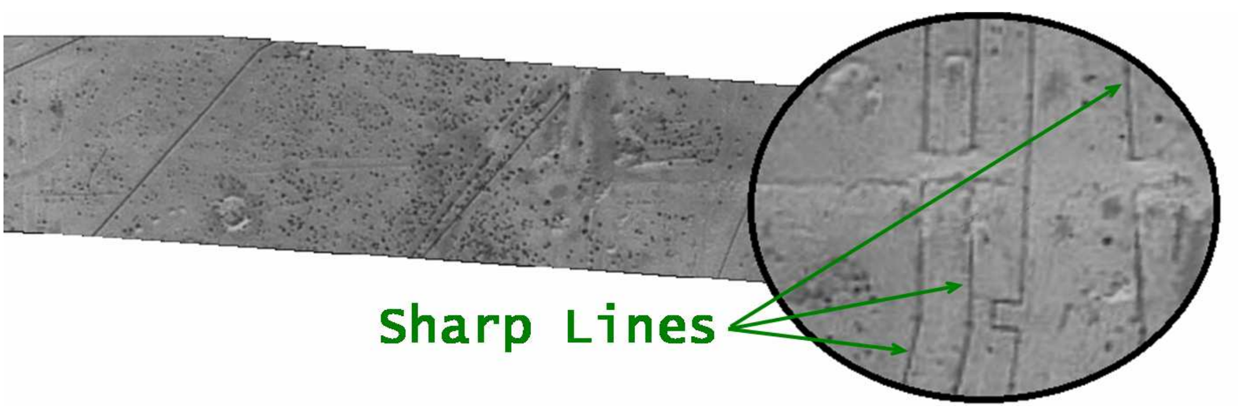

Figure 6.7: Mosaic generated using our approach. Our algorithm is able to produce good quality mosaics even if features to track are rare.

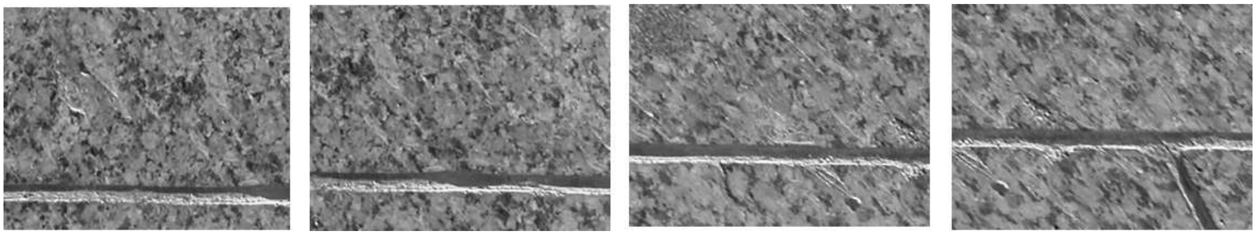

Figure 6.8: Sample frames of a video containing repetitive texture

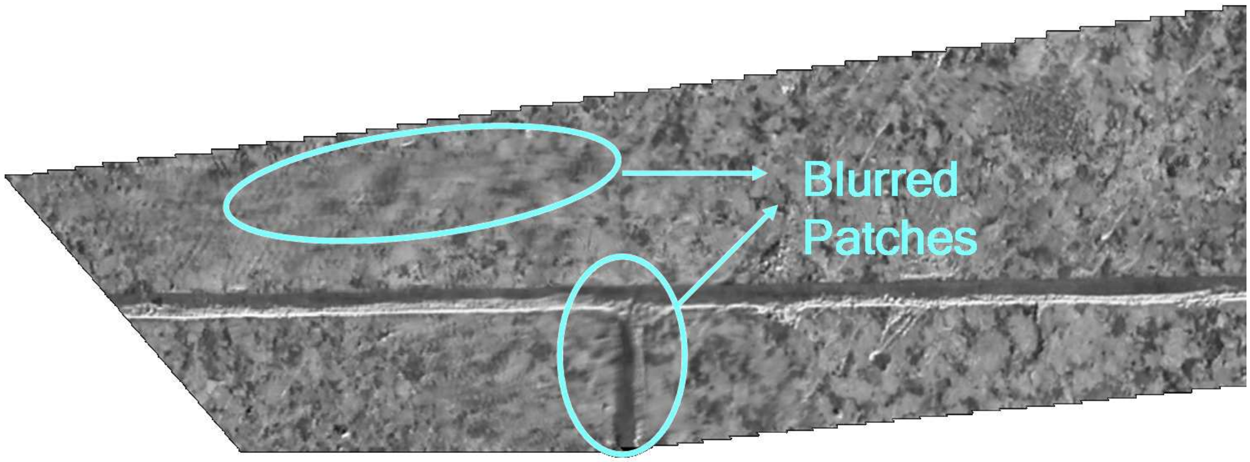

Figure 6.9: Mosaic built using bundle-adjustment

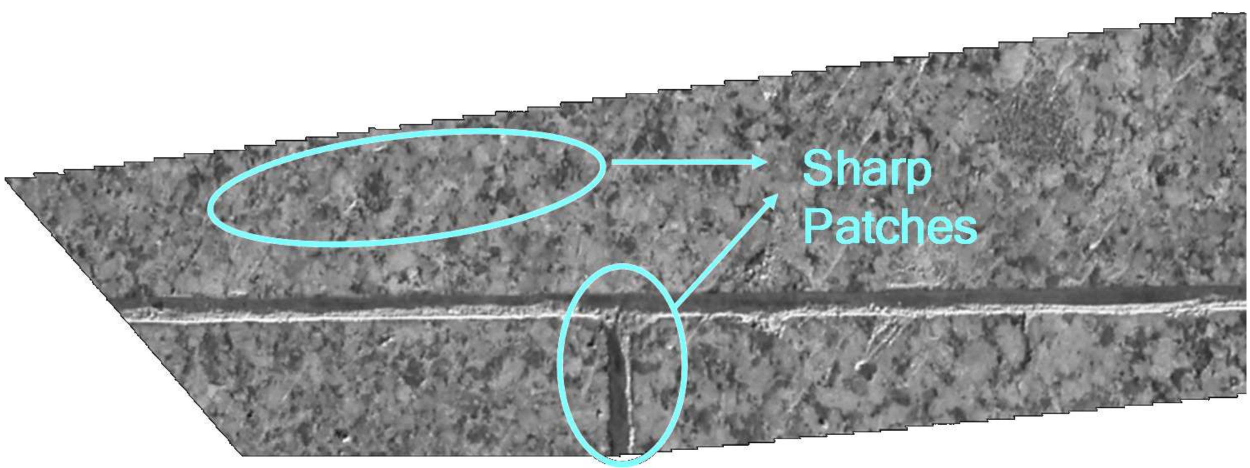

Figure 6.10: Mosaic built using our approach. It shows that our approach stays unaffected under false feature matching.

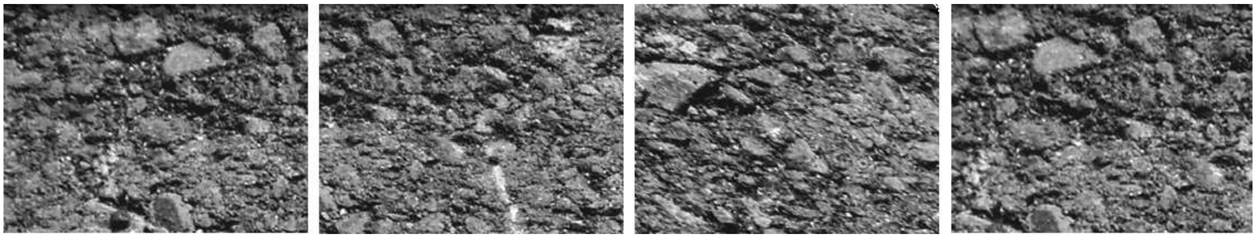

Figure 6.11: Sample frames of a video containing repetitive texture

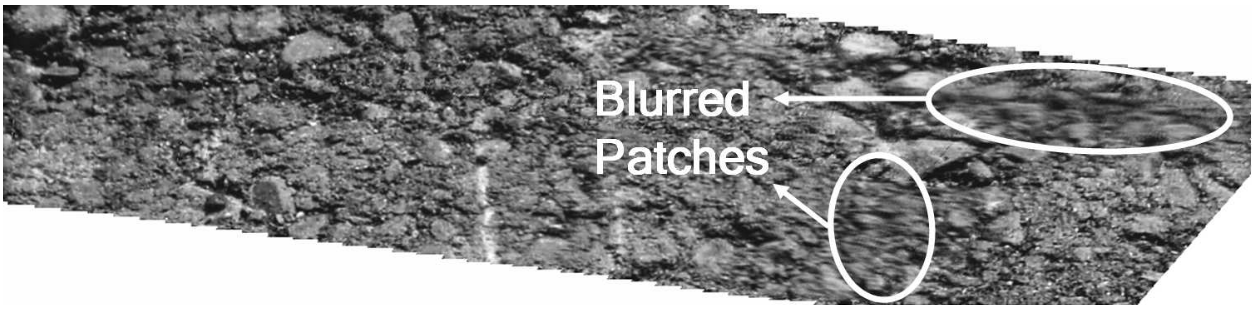

Figure 6.12: Mosaic built using bundle-adjustment

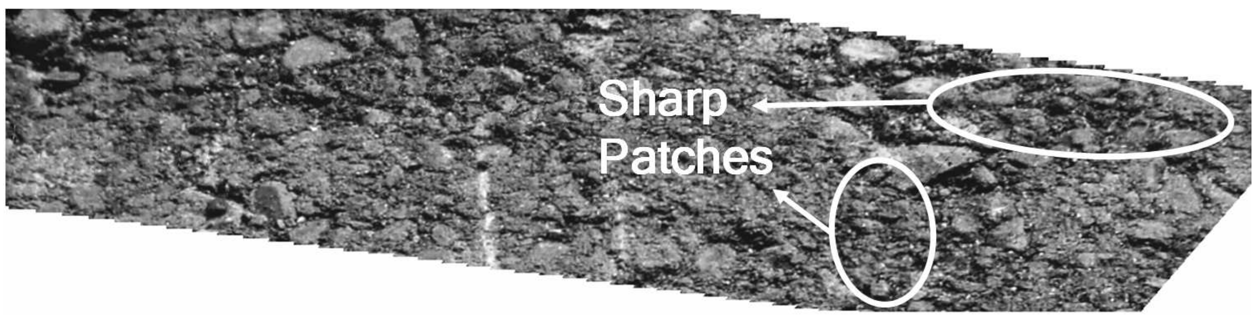

Figure 6.13: Mosaic built using our approach. It shows that our approach stays unaffected under false feature matching.

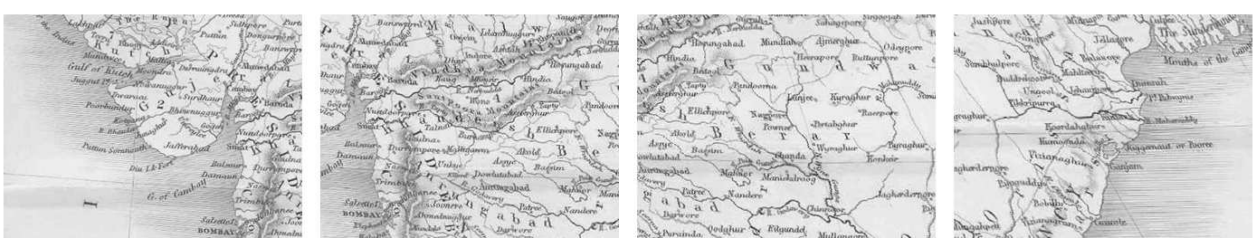

Figure 6.14: Sample frames of a map video

Figure 6.15: Mosaic built using pair-wise RANSAC followed by bundle-adjustment

Figure 6.16: Mosaic built using our approach. Our algorithm generates a better quality (more readable) mosaic than the one built by individually registering each pair of frames.

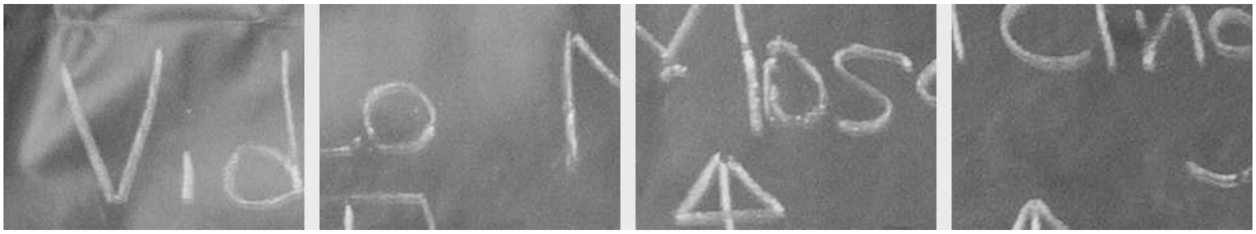

Figure 6.17: Sample frames of a blackboard video, captured by a hand-held web camera

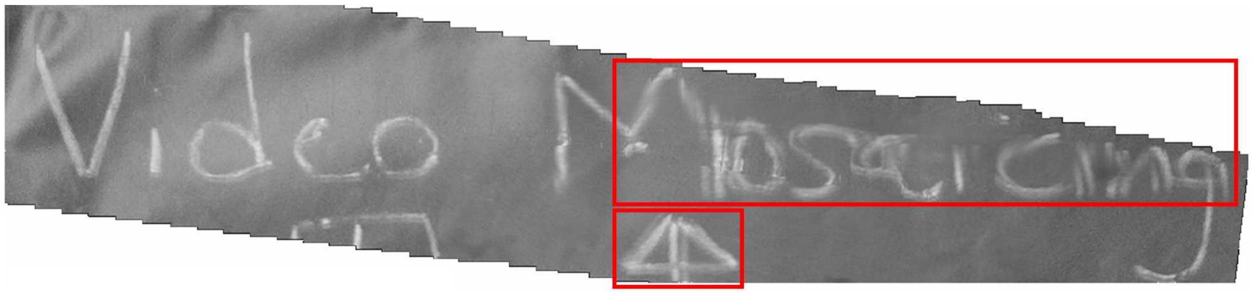

Figure 6.18: Mosaic built using the graph-based global alignment approach followed by bundle-adjustment

6.2.4 Blackboard

Fig 6.17 shows video frames, given by a hand-held web camera over a black board. The mosaic generated by the graph-based approach followed by bundle adjustment and the corresponding one built by our technique are shown in Fig 6.18 and Fig 6.19 respectively.

It can be noticed, in all the mosaicing examples, that our algorithms are robust to the error-accumulation phenomenon, described in Section 1. Table 6.1 quantifies the visual quality of the mosaics using the proposed measure and further highlights the superiority of our approach.

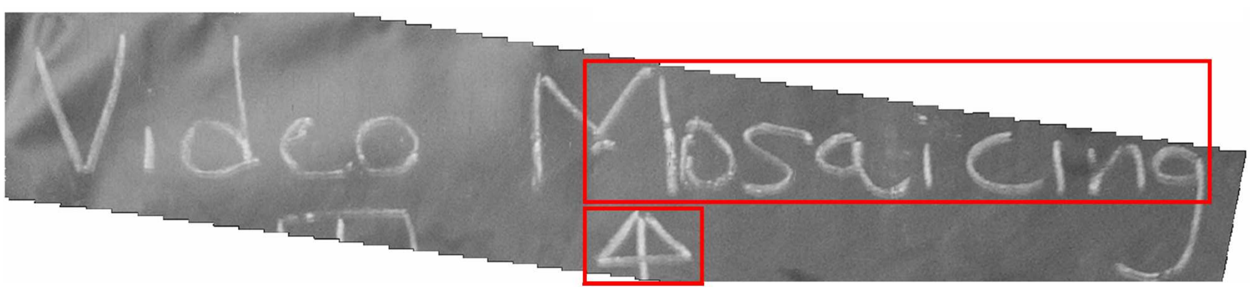

Figure 6.19: Mosaic built using our approach. The improvement in clarity can be easily seen, especially in the marked areas.

Technique	Desert Videos	Videos containing Repetitive Texture	Map Video	Blackboard Video
Graph-based	61.29	37.12	29.72	43.79
Bundle adjustment	54.94	49.97	35.90	53.10
Our	113.76	71.38	51.89	96.93

Table 6.1: The proposed quality metric for mosaics, for the four types videos considered in the thesis. Observe that the quality value, for each video, is maximum for our approach.

Chapter 7

Conclusions

We have presented video mosaicing algorithms with applicability to a variety of real-life scenarios. This includes the frequently occurring mosaicing issues like texture-less frames, repetitive texture and poor quality frames. The set of algorithms presented in this thesis account for calibrated well as uncalibrated set-ups i.e. if calibration information is available, we have a technique which can make use of it for robust registration and mosaicing. Experiments were carried out keeping in mind the practical mosaicing problems. Detailed analysis of all the algorithms was undertaken. Their sensitivity to many parameters like noise in the image measurements, calibration errors, poor quality frames, etc. was studied and in general, found to be well within acceptable limits.

Mosaics were generated using three popular existing techniques to facilitate comparison with mosaics generated through our algorithms. On almost all the videos that were used for testing, our approach, which exploits homography relationships stemming from the camera motion, is found to produce better quality mosaics on many videos. The improved quality can be readily perceived by the visually pleasing mosaics produced by our approach. However, we did not limit ourselves with that. To quantitatively evaluate the quality of mosaics, we devised a new measure. There exist a number of quality measures, mainly capturing the sharpness or the focus of the mosaic. Our measure complements them. It tries to capture the quality of alignment, the primary factor which can be attributed to the mosaic quality. The quantitative results were completely in tune with the (visually) perceived results. In other words, for each video, the quality metric returned maximum value for the mosaic generated using our technique capitalizing camera motion clues.

The type of mosaicing that we have considered in this work is planar mosaicing. In other words, the geometry and the camera motion are linked with planar homography. However, the ideas put forth can easily be adapted to rotational homography. Thus, robust panoramic mosaicing, using information stemming from the camera motion, can be achieved without deviating much from the core ideas presented in this thesis.

Another direction in which this work can be extended is super-resolution. With the mosaicing accuracy improving significantly, super-resolution mosaics can be generated as a part of the mosaicing process by scaling the resultant homographies appropriately. While the process is simplistic, the results can be quite impressive, owing to the accurate registration, made possible by the clues coming from the camera motion. In a similar way, these accurate mosaics can be geo-referenced using the telemetry information which often accompanies aerial video frames. Geo-registered mosaics are immensely useful in many places, especially in defense applications like object location and 3D visualization with walkthroughs.

Finally, in this thesis, the world has been assumed to completely planar. This assumption can be relaxed. The proposed ideas can be enhanced to cater to piece-wise planar objects. Instead of one homography, every pair of images will be related by a set of homographies. For a pair of images, each corresponding pair of matching points is ascribed to a unique homography, induced by a unique plane. If the object comprises n planes, correspondence for each frame pair can be viewed as n clusters of point pairs wherein the model charactering each cluster is the homography given by the corresponding plane. This enhancement will find usage in many interesting computer vision problems such as layer extraction, piece-wise planar 3D reconstruction and scene transfer.

Appendix

Appendix A

Homographies under Uniform Velocity Motion

In this chapter, we give detailed derivations of the homography relationships under uniform velocity motion. Before considering the general uniform velocity motion, we derive constraints on the incremental homographies $H_{i,i+1}$ and the inverse reference homographies $H_{0,i}$, for a purely translational motion. We show that there exist 11 parameter and 9 parameter relationships for the homographies in the case of linear translational motion.

A.1 Incremental Homographies

Theorem 1 *In presence of uniform translational motion, all the incremental homographies $H_{i,i+1}$ are related by a 11 parameter model.*

$$H_{i,i+1} = I + \frac{C}{c_1 + i.c_2} \tag{A.1}$$

where, C is a 3×3 matrix and c_1 and c_2 are scalars.

Proof: For a purely translational motion, the camera moving with velocity V (and acceleration A) is equivalent to its translational pose T moving with velocity $-V$ (and acceleration $-A$).
If T_i is the translational pose of the camera for frame I_i, assuming that the time elapsed between the capturing of successive frames is constant (t), we have

$$T_i = T_0 - i.t.V \tag{A.2}$$

and

$$T_{i+1} - T_i = -t.V$$

Substituting the above two equations in Equation 3.3,

$$
\begin{aligned}
H_{i,i+1} &= I - K\left[(-t.V)\frac{n^T}{d - n^T(T_0 - i.t.V)}\right]K^{-1} \\
&= I + \frac{t.KVn^TK^{-1}}{d - n^TT_0 + i(t.n^TV)}
\end{aligned}
$$

Substituting $C = t.KVn^TK^{-1}$, $c_1 = d - n^TT_0$ and $c_2 = t.n^TV$ in the above equation, directly results in Equation A.1.

Corollary 1 *If the camera motion is parallel to the world plane, all the incremental homographies are the same.*

Proof comes from the fact that, in presence of linear motion parallel to the world plane, $n^TV = 0$ leading to c_2 becoming zero in Equation A.1 and the incremental homographies becoming identical.

A.2 Inverse Reference Homographies

Theorem 2 *In presence of uniform translational motion, all the inverse reference homographies $H_{0,i}$ are related by a 9 parameter model.*

$$H_{0,i} = I + iK_c$$

where, K_c is a 3×3 matrix.

Proof: Substituting Equation A.2 in Equation 3.3,

$$H_{0,i} = I - K\left[-i.t.V\frac{n^T}{d^*}\right]K^{-1} \tag{A.3}$$

where, $d^* = d - n^TT_0$.

After simplification,

$$H_{0,i} = I + i\left[\underbrace{t.KV\frac{n^T}{d^*}K^{-1}}_{K_c}\right]$$

Corollary 2 *If the camera moves horizontally, imaging objects lying on a horizontal plane, the first order difference between the normalized inverse reference homographies is constant. By normalization, we mean that H(3,3) is set to 1.*

Proof: After normalization of the homographies (i.e.,after H(3,3) is set to unity),

$$H_{0,i} = \frac{I + iK_c}{1 + iK_c(3,3)} \tag{A.4}$$

As the camera moves horizontally, the z component of V is zero. Therefore, V is of the form $[x \;\; x \;\; 0]^T$. Moreover, the world plane is horizontal so vector n is of the form $[0 \;\; 0 \;\; x]^T$. We also know that the calibration matrix K is affine and so is its inverse.

Now, from Equation A.2, K_c is a multiplication of three matrices - K, Vn^T and K^{-1} and two scalars. From that, it directly follows that $K_c(3,3)$ is zero. Substituting that in A.4, after simplification,

$\Delta H_i = K_c = \text{constant}$

where, the first order difference $\Delta H_i = H_{0,i+1} - H_{0,i}$.

Note that when the camera moves parallel to an arbitrary plane and images objects on that plane, this corollary holds only when the camera's Z axis (which is usually the viewing axis) is perpendicular to the plane.

A.3 General Linear Motion

In the previous subsections, we derived constraints on incremental and inverse reference homographies for translational motion. In the case of general linear motion, we need to consider rotational velocity as well. We express the rotation in terms of the three Euler angles ϕ, θ and ψ represented in the vector form as L. The rotational velocity is denoted by V_r. L_0 denotes the initial configuration of Euler angels. Rewriting Equation 3.2 for the inverse reference homography of frame I_i,

$$H_{0,i} = K\left[R_iR_0^{-1} - (T_i - R_iR_0^{-1}T_0)\frac{n^TR_0^{-1}}{d - n^TR_0^{-1}T_0}\right]K^{-1}$$

where,

$$\begin{array}{lcll} K & = & f_K(\alpha, \beta, \vartheta, u_0, v_0) & - \quad 5 \text{ parameters} \\ R_i & = & f_R(L, V_r, i) & - \quad 6 \text{ parameters} \\ R_0 & = & f_r(L) & \\ T_i & = & f_T(T_0, V, i) & - \quad 6 \text{ parameters} \\ \pi & = & (n^T, d)^T & - \quad \underbrace{4 \text{ parameters}}_{21 \text{ parameters}} \end{array}$$

Thus, all inverse reference homographies are related by a 21 parameter model. In the same way, there exists a 21 parameter model for incremental homographies as well. The important point to note is that for general linear motion, the relationships or the constraints are non-linear whereas, for a purely translational case, the constraints derived are linear in the model parameters. Since the purely translational case is a special case of the general linear motion, there also exists a non-linear relationship between homographies for that case. However, linear relationships are always preferred as they can be exploited more robustly for video mosaicing.

It can be easily deduced from the results given in this section that under general linear motion, the homography between any pair of frames I_i and I_j can be computed from a 21 parameter model.

Appendix B

Homographies under Uniform Acceleration Motion

This chapter presents detailed derivations of the homography relationships under uniform acceleration motion. Again, before discussing the general uniform acceleration motion, we derive constraints on the incremental homographies and the inverse reference homographies, for a purely translational motion. In the last section, we discuss the general acceleration motion which contains (full) rotation in addition to translation.

B.1 Incremental Homographies

Theorem 3 *In presence of uniform acceleration motion, all the incremental homographies $H_{i,i+1}$ are related by a 21 parameter model.*

$$H_{i,i+1} = I - \left[\frac{E_1 + i.E_2}{b_1 + i.b_2 + i^2.b_3} \right] \tag{B.1}$$

where, E_1,E_2 are 3×3 matrices and b_1,b_2 and b_3 are scalars.

Proof: If V is the velocity of the camera and A is the acceleration of the camera,

$$T_i = T_0 - V.(i.t) - \frac{1}{2}.A(i.t)^2 \tag{B.2}$$

and

$$T_{i+1} - T_i = \underbrace{-(t.V + \frac{t^2}{2}.A)}_{D_1} + i.\underbrace{(-t^2.A)}_{D_2}$$

After simplification, we also have

$$d - n^T T_i = \underbrace{d - n^T T_0}_{b_1} + i.\underbrace{(t.n^T V)}_{b_2} + i^2 \underbrace{(\frac{t^2}{2}.n^T A)}_{b_3}$$

Substituting $T_{i+1} - T_i$ and $d - n^T T_i$ in Equation 3.3,

$$
\begin{aligned}
H_{i,i+1} &= I - \left[\frac{K(D_1 + i.D_2)n^T K^{-1}}{b_1 + i.b_2 + i^2.b_3}\right] \\
&= I - \left[\frac{(KD_1 n^T K^{-1}) + i.(KD_2 n^T K^{-1})}{b_1 + i.b_2 + i^2.b_3}\right]
\end{aligned}
$$

Setting $E_1 = KD_1 n^T K^{-1}$ and $E_2 = KD_2 n^T K^{-1}$, we arrive at Equation B.1.

Corollary 3 *If the camera moves horizontally, imaging objects lying on a horizontal plane, the first order difference between the normalized incremental homographies is constant.*

Proof: In presence of acceleration motion parallel to the world plane, $n^T V = 0$ and $n^T A = 0$. Using them, the normalized incremental homographies are given by:

$$H_{i,i+1} = \frac{I - E_1 - i.E_2}{1 - E_1(3,3) - i.E_2(3,3)} \tag{B.3}$$

As the camera moves horizontally and the world plane is horizontal, $E_1(3,3)$ is zero. Similarly, $E_2(3,3)$ can also be shown to be zero. Substituting them in B.3, after simplification,

$$\Delta H_i = E_2 = \text{constant}$$

where, the first order difference $\Delta H_i = H_{i+1,i+2} - H_{i,i+1}$.

B.2 Inverse Reference Homographies

Theorem 4 *In presence of uniform acceleration motion, all the inverse reference homographies $H_{i,i+1}$ are related by a 18 parameter model.*

$$H_{0,i} = I + i.M_1 + i^2.M_2$$

where, M_1,M_2 are 3×3 matrices.

Proof: Substituting Equation B.2 in Equation 3.3,

$$H_{0,i} = I - K\left[(i.(-t.V) + i^2.(-\frac{t^2}{2}.A)).\frac{n^T}{d}\right]K^{-1}$$

where,

$$d^* = d - n^T T_0$$

Simplifying,

$$H_{0,i} = I + i \underbrace{\left[\frac{t.KVn^TK^{-1}}{d^*}\right]}_{M_1} + i^2 \underbrace{\left[\frac{t^2.KAn^TK^{-1}}{2.d^*}\right]}_{M_2}$$

Corollary 4 *If the camera moves horizontally, imaging objects lying on a horizontal plane, the second order difference between the normalized inverse reference homographies is constant.*

Proof: After normalization,

$$H_{0,i} = \frac{I + i.M_1 + i^2.M_2}{1 + i.M_1(3,3) + i^2.M_2(3,3)}$$

As the camera motion is horizontal and the world plane is also horizontal, $M_1(3,3)$ and $M_2(3,3)$ are zero. Using that, the first order differences

$$\Delta H_i = (M_1 + M_2) + i(2M_2)$$

and the second order differences

$$\Delta H_{i+1} - \Delta H_i = 2M_2 = \text{constant}.$$

B.3 General Acceleration Motion

Under general uniform acceleration motion comprising rotation and translation, all inverse reference homographies are related by a 27 parameter model. Similarly, there is a 27 parameter model relating incremental homographies as well.

The proof is identical to the general linear motion case except that 3 additional parameters come in for rotational acceleration and 3 for translational acceleration. This constraint, like the one for the general linear motion, is non-linear. However, it can be made linear by introducing more parameters which are functions of one or more these 27 parameters. This is helpful when we want to use a linear technique to compute the model parameters in closed form.

Related Publications

- Pulkit Parikh, C. V. Jawahar, "Motion Constraints for Video Mosaicing", in *Proceedings of the Third IET International Conference on Visual Information Engineering*, 2006, pp. 494-499.
- Pulkit Parikh, C. V. Jawahar, "Enhanced Video Mosaicing using Camera Motion Properties", in *IEEE Workshop on Motion and Video Computing*, 2007.

Bibliography

[1] S. Nayar and A. Karmarkar, "360 x 360 Mosaics: Regular and Stereoscopic," in *Panoramic Vision*, pp. 291–307, April 2001.

[2] A. Levin, A. Zomet, S. Peleg, and Y. Weiss, "Seamless image stitching in the gradient domain," in *ECCV*, pp. 377–389, 2004.

[3] R. Marzotto, A. Fusiello, and V. Murino, "High resolution video mosaicing with global alignment," in *CVPR (1)*, pp. 692–698, 2004.

[4] R. Szeliski, "Video mosaics for virtual environments," *IEEE CG&A*, pp. 22–30, March 1996.

[5] H. S. Sawhney, S. C. Hsu, and R. Kumar, "Robust video mosaicing through topology inference and local to global alignment," in *ECCV (2)*, pp. 103–119, 1998.

[6] P. Wolf, *Elements of Photogrammetry.* McGraw-Hill, second ed., 1983.

[7] S. Kang, "A survey of image-based rendering techniques," *Technical Report CRL 97/4, Digital Equipment Corp. Cambridge Research Lab*, Aug 1997.

[8] J. Lengyel, "The convergence of graphics and vision," *Computer*, pp. 46–53, July 1998.

[9] S. E. Chen, "Quicktime vr: an image-based approach to virtual environment navigation," in *SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pp. 29–38, 1995.

[10] L. McMillan and G. Bishop, "Plenoptic modeling: an image-based rendering system," in *SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pp. 39–46, 1995.

[11] R. Szeliski and H.-Y. Shum, "Creating full view panoramic image mosaics and environment maps," in *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pp. 251–258, 1997.

[12] Y. Xiong and K. Turkowski, "Creating image-based VR using a self-calibrating fisheye lens," in *CVPR*, p. 237, 1997.

[13] S. Coorg, N. Master, and S. Teller, "Acquisition of a large pose-mosaic dataset," in *CVPR*, p. 872, 1998.

[14] S. E. Chen and L. Williams, "View interpolation for image synthesis," in *SIGGRAPH '93: Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pp. 279–288, 1993.

[15] H. Y. Shum, M. Han, and R. Szeliski, "Interactive construction of 3d models from panoramic mosaics," in *CVPR*, p. 427, 1998.

[16] S. B. Kang and R. Szeliski, "3-d scene data recovery using omnidirectional multibaseline stereo," in *CVPR*, p. 364, 1996.

[17] R. Szeliski, "Image mosaicing for tele-reality applications," in *IEEE Workshop on Applications of Computer Vision*, pp. 44–53, 1994.

[18] D. Nister, "Frame decimation for structure and motion," in *Workshop on Structure from Multiple Images of Large Environments*, pp. 17–34, 2000.

[19] C. Morimoto and R. Chellappa, "Fast 3d stabilization and mosaic construction," in *CVPR*, p. 660, 1997.

[20] M. Hansen, P. Anandan, K. Dana, G. Wal, and P. Burt, "Real-time scene stabilization and mosaic construction," in *IEEE Workshop on Applications of Computer Vision*, pp. 54–62, 1994.

[21] D. Capel and A. Zisserman, "Automatic mosaicing with super-resolution zoom.," in *CVPR*, pp. 885–891, 1998.

[22] S. Mann and R. W. Picard, "Virtual bellows: Constructing high quality stills from video," in *ICIP (1)*, pp. 363–367, 1994.

[23] M. Irani, S. Hsu, and P. Anandan, "Video compression using mosaic representations," *Signal Processing: Image Communication, 7*, pp. 529–552, 1995.

[24] M. Irani and P. Anandan, "Video indexing based on mosaic representations," in *IEEE Trans. on PAMI, 86(5)*, pp. 905–921, May, 1998.

[25] P. Saravanan, C. K. Narayanan, P. Prakash, and G. Prabhakara Rao, "Techniques for video mosaicing," *Transactions on Engineering, Computing and Technology V5*, April 2005.

[26] M. Irani, P. Anandan, and S. Hsu, "Mosaic based representations of video sequences and their applications," in *ICCV*, pp. 605–611, 1995.

[27] H. Sawhney and S. Ayer, "Compact representation of videos through dominant multiple motion estimation," in *IEEE Transactions on Pattern Analysis and Machine Intelligence, 18(8)*, pp. 814–830, 1996.

[28] M. Svedlow, C. McGillem, and P. Anuta, "Image registration: Simlarity measure and processing method comparisons," *AeroSys*, vol. 14, no. 1, pp. 141–149, 1978.

[29] L. Brown, "A survey of image registration techniques," *ACM Computing Surveys 24*, vol. 24, pp. 325–376, December 1992.

[30] B. Zitova and J. Flusser, "Image registration methods: a survey," *IVC*, vol. 21, pp. 977–1000, October 2003.

[31] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *IJCAI81*, pp. 674–679, 1981.

[32] D. Lowe, "Distinctive image features from scale-invariant keypoints," in *International Journal of Computer Vision*, vol. 20, pp. 91–110, 2003.

[33] E. D. Castro and C. Morandi, "Registration of translated and rotated images using finite fourier transforms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9, no. 5, pp. 700–703, 1987.

[34] P. A. Viola, "Alignment by maximization of mutual information," Tech. Rep. AITR-1548, 1995.

[35] S. Gumustekin and R. Hall, "Mosaic image generation on a flattened gaussian sphere," in *WACV96*, pp. 50–55, 1996.

[36] M. Uyttendaele, A. Eden, and R. Szeliski, "Eliminating ghosting and exposure artifacts in image mosaics," *CVPR*, vol. 02, pp. 509–516, 2001.

[37] E. H. Adelson, C. H. Anderson, J. R. Bergen, P. J. Burt, and J. M. Ogden, "Pyramid methods in image processing," *RCA Engineer*, vol. 29, no. 6, 1984.

[38] A. Agarwal, C. V. Jawahar, and P. J. Narayanan, "A survey of planar homography estimation techniques," Tech. Rep. IIIT/TR/2005/12, 2005.

[39] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge University Press, second ed., 2000.

[40] M. Fischler and R. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," in *RCV87*, pp. 726–740, 1987.

[41] K. Kanatani and N. Ohta, "Accuracy bounds and optimal computation of homography for image mosaicing applications," in *ICCV99*, pp. 73–78, 1999.

[42] E.-Y. Kang, I. Cohen, and G. G. Medioni, "A graph-based global registration for 2d mosaics.," in *ICPR*, pp. 1257–1260, 2000.

[43] H.-Y. Shum and R. Szeliski, "Systems and experiment paper: Construction of panoramic image mosaics with global and local alignment," *Int. J. Comput. Vision*, vol. 36, no. 2, pp. 101–130, 2000.

[44] U. Bhosle, S. D. Roy, and S. Chaudhuri, "Multispectral panoramic mosaicing," *Pattern Recogn. Lett.*, vol. 26, no. 4, pp. 471–482, 2005.

[45] S. Feng, H. Lu, and S. Ma, "Mosaic representations of video sequences based on slice image analysis," *Pattern Recogn. Lett.*, vol. 23, no. 5, pp. 513–521, 2002.

[46] A. Shashua and S. Avidan, "The rank 4 constraint in multiple view geometry," in *ECCV (II)*, pp. 196–206, 1996.

[47] Y. Sheikh and M. shah, "Aligning dissimilar images directly," in *Asian Conference on Computer Vision*, pp. 402–407, 2004.

[48] A. Rav Acha, Y. Pritch, D. Lischinski, and S. Peleg, "Dynamosaics: Video mosaics with non-chronological time," in *CVPR*, pp. I: 58–65, 2005.

[49] T. Mei, X. Hua, H. Zhou, S. Li, and H. Zhang, "Efficient video mosaicing based on motion analysis," in *ICIP*, pp. I: 861–864, 2005.

[50] M. Brown and D. G. Lowe, "Recognising panoramas," in *ICCV*, pp. 12–18, 2003.

[51] L. Zelnik-Manor and M. Irani, "Multiview constraints on homographies," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 2, pp. 214–223, 2002.

[52] O. Faugeras and F. Lustman, "Motion and structure from motion in a piecewise planar environment," *IJPRAI*, vol. 2, pp. 485–508, 1988.