Planar Homography Estimation from Traffic Streams via Energy Functional Minimization

by

Graham Beck

A dissertation submitted to The Johns Hopkins University in conformity with the requirements for the degree of Doctor of Philosophy.

Baltimore, Maryland

February, 2016

Abstract

The 3x3 homography matrix specifies the mapping between two images of the same plane as viewed by a pinhole camera. Knowledge of the matrix allows one to remove the perspective distortion and apply any similarity transform, effectively making possible the measurement of distances and angles on the image. A rectified road scene for instance, where vehicles can be segmented and tracked, gives rise to ready estimates of their velocities and spacing or categorization of their type.

Typical road scenes render the classical approach to homography estimation difficult. The Direct Linear Transform is highly susceptible to noise and usually requires refining via an further nonlinear penalty minimization. Additionally, the penalty is a function of the displacement between measured and calibrated coordinates, a quantity unavailable in a scene for which we have no knowledge of the road coordinates. We propose instead to achieve metric rectification via the minimization of an energy that measures the violation of two constraints: the divergence-free nature of the traffic flow and the orthogonality of the flow and transverse directions under the true transform.

Given that an homography is only determined up to scale, the minimization is

performed on the Lie group $SL(3)$, for which we develop a gradient descent algorithm. While easily expressed in the world frame, the energy must be computed from measurements made in the image and thus must be pulled back using standard differential geometric machinery to the image frame. We develop an enhancement to the algorithm by incorporating optical flow ideas and apply it to both a noiseless test case and a suite of real-world video streams to demonstrate its efficacy and convergence. Finally, we discuss the extension to a 3D-to-planar mapping for vehicle height inference and an homography that is allowed to vary over the image, invoking a minimization on Diff($SL(3)$).

Acknowledgments

This thesis is a testament to the floundering and circuitous route so often taken to arrive at a new idea that seems so clear and obvious in retrospect. I am tremendously thankful to those who were able to shorten the path a little. Firstly my advisor, Professor Daniel Naiman, who has an unparalleled ability to sniff out inconsistencies and whose enthusiasm for all aspects of statistics gave impetus to this project and livens up the department no end. Professor Gregory Chirikjian also fueled an avid and timely interest in Lie Groups with his prolific research which I spent many hours absorbed in. I would also like to thank my other committee members for serving with good humor even at great inconvenience: Professor Laurent Younes and Assistant Professor Nicolas Charon.

My heartfelt gratitude goes to my uncle and aunt, Peter Moll and Masami Kojima, for encouraging me on this path and providing such strong, unwavering support throughout. I owe a steely work ethic to my mother Jenny, and to my father Andrew a strong desire to accomplish a fraction of what he has managed ever since those first acknowledgments of his own family in that little green book. Finally, my

deepest appreciation to my lovely wife Vanessa who has never let my mathematical preoccupations dim the joy and laughter she brings.

Contents

List of Figures

Chapter 1

Introduction

1.1 Framework

We should soon be able to query the world around us for pertinent information that is superimposed on our field of vision. Smart phones may suggest the best choice from all product options on a supermarket shelf given our personal preferences, or show us how that art piece we've been eyeing up would look on the wall. Smart glasses may inform us whether or not the hotel we are passing has any vacancies ([1]). At the heart of such augmented reality is an understanding of how the 3D objects around us map to a 2D image and how the image changes with our orientation. While this thesis entertains somewhat less lofty aspirations, it is nevertheless an attempt to increase the amount of information we can extract from specific scenes by pursuing that same understanding.

We seek to use streaming traffic camera footage from a single camera to infer instantaneous quantities of interest such as the heavy-vehicle density, road congestion levels and average speed. One of the primary requirements is the classification of different vehicle types; in particular the ability to differentiate trucks from other traffic in order to keep a running measure of their flux. One of the most direct uses of this measure is as a 'Truckometer', a gauge of economic activity that has recently been suggested may lead GDP by six months[1]. We consider a solution to this problem valid only if it conforms to several criteria: it must allow for the typical low-resolution footage (several hundred pixels in each dimension) ubiquitous on the internet, be robust to ample noise in the image and converge rapidly.

Figure 1.1: Snapshot of a typical traffic scene

There are two broad approaches to the problem (see [2] for a review). The first employs an unsupervised learning of the characteristics associated with each moving object in the image ([3], [4], [5] for example). A model may be constructed that describes how a vehicle silhouette deforms as it approaches or recedes, and a

[1]Using data published by the New Zealand Transport Agency and compiled by ANZ Bank as described in *Spinning in the Mud?* April 2012, available at http://www.anz.co.nz/commercial-institutional/economic-markets-research/truckometer/

distribution of the silhouette area or other measurable learned at each point on the road so that distinct vehicle types are identified as belonging to distinct clusters. The second matches each object with a template derived from a 3D model mapped into the image via the homography ([6], [7], [8]). Our approach is most similar to the latter; the former suffers from the high number of training frames required as well as little generalization capability to the field of view. A panning camera for example would not require re-calibration under the second approach, but rather a simple adjustment to the estimated homography. A 3D template however can only be superimposed on an image if the intrinsic camera parameters are known. There are in general five of these, made up of a focal length in each coordinate direction, a principal point and a skew between axes. Our problem set-up only assumes uncalibrated cameras so the intrinsic parameters would need to be estimated; this is possible for scenes that include at least three non-coincident planes, a condition that unfortunately cannot realistically be met in typical traffic scenes for which only the road surface provides a clear plane. It is tempting then to cast the problem into one amenable to 'structure from motion' techniques: we imagine the camera to be moving past stationary vehicles, a frame of reference just as valid as that of the stationary camera and moving vehicles, as the background is usually filtered out. Tracking points of interest from frame to frame and exploiting the epipolar constraint - a constraint generated by the geometry of a scene viewed from more than one viewpoint - yields a reconstruction of those points in 3D space. While the change in coordinates of tracked vehicle vertices relative to

each other may overcome the previous lack of non-road-plane information available, the reconstruction is still only obtained up to a projective transformation without knowledge of the intrinsic parameters. This leaves us little better off in terms of mapping external 3D coordinates into the scene.

The upshot is that the best we can do, given tracked points that lie on or close to a single plane in the scene, is estimate the homography - the linear mapping of homogenous world coordinates on the road plane to homogenous image coordinates[2]. Being a 2D to 2D transform, some geometric information inherent in the scene is lost: objects that stand proud of the road plane appear distorted once the image has been rectified to recover metric quantities such as angles and distances. In most of the scenes we are interested in the distortion is not great enough to overwhelm the convenience of the planar assumption; where it is apparent, it does not necessarily make our task harder. Larger vehicles will take on larger rectified silhouettes especially if distorted due to their height, making them easier to categorize for instance.

The estimation of the homography from real-world scenes presents some challenges. Where road boundaries are clearly delineated and we can identify two boundary points exactly opposite one another, the geometrical reconstruction has been achieved by [9]. The boundaries and their tangents are not always found easily in an automated fashion however, especially when the image contrast is low or the background scene exhibits little color variation. A natural alternative to stationary

[2]See section 8.1 however for a proposed method to infer moving vehicle height when it extends significantly above the road plane with respect to the camera position.

points of identification is to use the trajectories of the vehicles themselves or other vehicle-based geometric cues. The authors in [10] achieve affine rectification (which represents an estimation of the homography up to similarity and skew transforms of the scene) by tracking two objects moving along nonparallel paths at constant speeds and exploiting an induced ratio constraint. Rather than using their trajectories, the dominant gradient orientations of the vehicles are used in [11] to enforce known information about the scene. This eliminates the need for nonparallel trajectories, a requirement that is seldom met in highway scenes, and our approach to geometrical constraints is similar: we also exploit constraint equations to estimate the affine transformation-related parameters, but the perspective parameters on the other hand we estimate by minimizing some measure of the violation of the constraints.

By setting up the homography estimation as a constrained minimization problem over the parameter space, we can borrow from the Beltrami flow regularization framework espoused by Sochen ([12], [13]). The main idea is to construct a functional on the image that accounts for variation in quantities to be regularized such as intensity, gradient or flow ([14], [15], [16]). An induced metric on the image domain summarizes the degree of variation at every point in the image, and a gradient descent iteration is then derived to perform the diffusion necessary to attain a state with lower variation. This framework is known to encompass image denoising by minimizing either of the L_1 and L_2 norms of the gradient as special cases. Tensor-valued fields can also be regularized in this manner as an extension to the orientation diffusion

applied to vector fields (see also [17] and [18] for a maximum likelihood methodology rather than gradient descent). In these cases an image is thought of as the section of a fibre bundle where each fibre is a Riemannian manifold. The 2D flow in an image for example is captured by the Lie group $SO(2)$ as fibre: each point in the image is associated with the direction of the flow. The most significant departure of our approach from this framework is that we do not modify elements of our image by the diffusive process; rather, by modifying the homography the diffusion occurs in the world frame that we are reconstructing. We expect then that as our homography estimates converge, we are recovering a world frame that minimizes variation in some sense while conforming to our constraints.

Given the importance of homography estimation to many robotics applications, it is in this field that we find much of the work most similar in spirit to our own. The homography g is only defined up to scale, a quality that allows us to specify $\det(g) = 1$, which in turn implies that g is an element of the Lie group $SL(3)$. Being a finite group, once we select a parameterization we can find the gradient of the functional to minimize in a straightforward manner but some care must be exercised to ensure that the minimization procedure yields a solution still on the manifold $SL(3)$. An efficient least-squares technique with second-order convergence properties for incremental tracking of the homography on $SL(3)$ is described in [19] and [20], while a set of coupled ODEs on $SL(3)$ is derived in [21] that is proved to drive tracking error to zero. The similarity to our objective is that we are also using the

difference between two views of the same plane to update an homography estimate, but in our case the second view is imaginary: we think of the world frame as an image taken with the camera pointing directly along the normal to the plane of the road (the so-called fronto-parallel view). The homography is then the mapping from this imagined camera position to the real one, so recovering the mapping can still be posed as a tracking problem, but since we do not have access to the coordinates in our world frame the same objective function cannot be used. We are left to create a new one from geometrical cues that we can get our hands on. In particular, we use the orthogonality of the vehicles' trajectories and their transverse vectors (the direction of least gradient variation over the vehicle) as well as the divergence-free nature of traffic flow.

1.2 Contribution of the Thesis

The cornerstone of our approach is the assertion that under the true homography, certain orthogonality and flow divergence conditions will be met in the world frame. That is, we should be able to reconstruct from the image a world frame with a divergence-free traffic flow for which the flow vectors are orthogonal to the transverse vehicle direction. Given that such metric rectification is classically achieved by Direct Linear Transform (DLT), a presentation is made of the standard techniques in computer vision for establishing correspondences between images, the invariants

that arise and their use in robust homography estimation. It is well known that the DLT itself suffers from inaccuracies due to an inherent amplification of image noise; we formalize this notion through a first-order analysis of the variance propagation from image coordinates through to the elements of the homography, and demonstrate the effect this can have on the rectification. The remedy is classically an additional high-dimensional, non-linear minimization step to refine the result; we instead propose a novel, lower-dimensional constrained energy minimization that achieves the homography estimation in a single procedure after being suitably initialized.

The minimization is performed on a Lie group, and a presentation is made of the pertinent results in differential geometry leading to the representation of the gradient descent algorithm both on the Lie algebra and in terms of the parameterization of the group $SL(3)$. Constraints are factored in through the use of Lagrange multipliers and are pulled back into the same space, effectively restricting the minimization to a submanifold of $SL(3)$. The energy we construct is a functional that measures the violation of our expected conditions in the world frame and we spend significant effort deriving the expression for its pullback as computed from the image before putting the algorithm to work on a noise-free mock scene to demonstrate its feasibility.

Thirdly, we tackle a suite of real-world traffic videos for which the vehicles must be segmented and both flow and transverse vector fields estimated before the minimization can be run. We profile the convergence of the algorithm and show that with a modification of the energy functional that incorporates well-known optical

flow ideas, the convergence can be improved by making incremental adjustments to the estimated vector fields. We also derive an improvement to the homography initialization when the imaged road is known to be straight. The objective of this work is thus achieved in that we arrive at a procedure that is more robust to noise in low-resolution videos than a standard application of the DLT while maintaining an efficient rate of convergence, though we also highlight instances where the algorithm will fail due to violation of the underlying assumptions.

1.3 Thesis Outline

Three primary research directions constitute the bulk of the thesis: the differential geometry that prescribes how objects should transform between spaces, 2D projective geometry and its application to computer vision, and the image processing techniques through which we capture the measurements needed in an application of the theory to real scenes. Chapters 2 and 3 set up the abstract differential geometric formalism before focusing on actions of a matrix Lie group on itself. Section 3.3 in particular describes how gradient descent is performed on the Lie group by stepping in a direction defined by a vector on the Lie algebra.

Before these results are used, Chapter 4 introduces the projective space in which the transformation of points and lines between different images is linear. The mapping itself is the homography, the estimation of which is dealt with in Chapter 5 in a

classical manner: using geometric information from the image to perform a metric rectification and resolving the remaining (similarity-oriented) degrees of freedom by gradient descent. An error analysis of this approach is given in section 5.5 that makes explicit the well-known need for an additional refinement procedure.

In Chapter 6 an energy functional is introduced, the minimization of which resolves the perspective- and skewness-oriented degrees of freedom while constraints are derived that fix the remaining five. The descent equations are calculated and applied in comparison to the DLT approach. The application to real-world traffic footage is made in Chapter 7 along with a discussion of the segmentation and measurement methods as well as an augmentation of the energy functional to incorporate optical flow. The rectifications achieved on the test suite are depicted and the algorithm's convergence rate profiled.

Finally, Chapter 8 summarizes the approach taken in this thesis and briefly discusses both its efficacy and directions that future work could take to improve and generalize it further.

Chapter 2

Differential Geometric Formalism

In establishing the mapping between the world and the image, we assume all geometric cues to describe the road plane itself; we will not initially be concerned with the height of the objects in the scene. Further, we expect the 2D-to-2D mapping to be not only invertible but also smooth and therefore a diffeomorphism which preserves structure from one space to the other. For instance, a smoothly varying traffic flow in the world should be mapped so that the flow in the image also varies smoothly. We should also have the freedom to specify the local coordinates in each frame without affecting the underlying transformation; the usual Euclidean framework with measurement by pixel in the image and metre in the world is just as valid as the choice of radial coordinates, or different measurement units for example. We need to be able to describe the relationships between objects in the world as a function of their appearance in the image: parallel lines delineating the road will almost certainly

not appear parallel in the image for instance, yet we need a way of recognizing their parallelism in the world frame under the correct mapping. The differential geometric tools to achieve this end are set out in the following section.

2.1 The Tangent Bundle

The road surface lies on a two dimensional manifold $\mathcal{W}$ embedded in the three dimensional ambient space $\mathbb{R}^3$. Both $\mathcal{W}$ and the scene image $\mathcal{I}$ are parameterized by choices of coordinates, the latter by the continuous *image coordinate chart* $\varphi : \mathcal{I} \to \mathbb{R}^2$ so that any location x on the image manifold is assigned the coordinate components $\varphi(x)$. The coordinate chart also by definition has a continuous inverse, rendering the *change of variables* formula given two such coordinate charts φ_1 and φ_2 as $\varphi_2 \circ \varphi_1^{-1}$. Moreover, given world coordinate chart φ_1 and image coordinate chart φ_2, a function $f : \mathcal{W} \to \mathcal{I}$ has the *representation in local coordinates*

$$\varphi_2 \circ f \circ \varphi_1^{-1} \tag{2.1}$$

which we usually simply identify as 'f'. The mappings that we are interested in preserve the local structure of their domains and in particular are *diffeomorphisms*; that is, they are differentiable and have differentiable inverses. More concretely this means that the composition $\varphi_2 \circ f \circ \varphi_1^{-1}$ is differentiable at all points of its domain (and for any choice of representation). Because it can be shown in the same sense that every coordinate chart of a submanifold of $\mathbb{R}^n$ is smooth, it is still solely the

nature of f itself that determines differentiability.

Traffic flow in the image is described by the differentiable paths $\gamma(t) \in \mathcal{I}$ the vehicles follow; these are the integral lines stemming from a vector field that associates each point in the image with a velocity, as will be explained below. The paths and their velocities are measured using local coordinates; a path through the point $x_0 \in \mathcal{I}$ at time $t = 0$ has local coordinates

$$q(t)|_{t=0} \doteq \varphi \circ \gamma(t)|_{t=0} = \varphi(x_0) \tag{2.2}$$

and velocity (in local coordinates)

$$\begin{aligned} \dot{q}(t)|_{t=0} &\doteq \left.\frac{d}{dt}\right|_{t=0} (\varphi \circ \gamma(t)) \\ &= \sum_i \frac{\partial \varphi}{\partial x_i}(x_0) \left.\frac{d}{dt}\right|_{t=0} \dot{\gamma}_i(t) \\ &= D\varphi(x_0)\dot{\gamma}(0) \\ &= (D\varphi \cdot v)(x_0) \end{aligned} \tag{2.3}$$

where $v(x_0) \doteq \left.\frac{d}{dt}\right|_{t=0} \gamma(t)$ is the definition of the velocity along the path at x_0. Of course, there are many different possible paths through x_0 with some specified velocity v_0, all of which will have identical measurements $(q, \dot{q})$. For this reason we say that the *tangent vector* v at x is the equivalence class of all paths $\gamma(t)$ through $x = \gamma(0)$ with velocity $v = \dot{\gamma}(0)$. The point x is the *base point* of the tangent vector, and the vector space $T_x\mathcal{I}$ of all tangent vectors on the manifold $\mathcal{I}$ at x is called the *tangent space* of $\mathcal{I}$ at x. The *tangent bundle* $T\mathcal{I}$ is the set of all tangent spaces over the

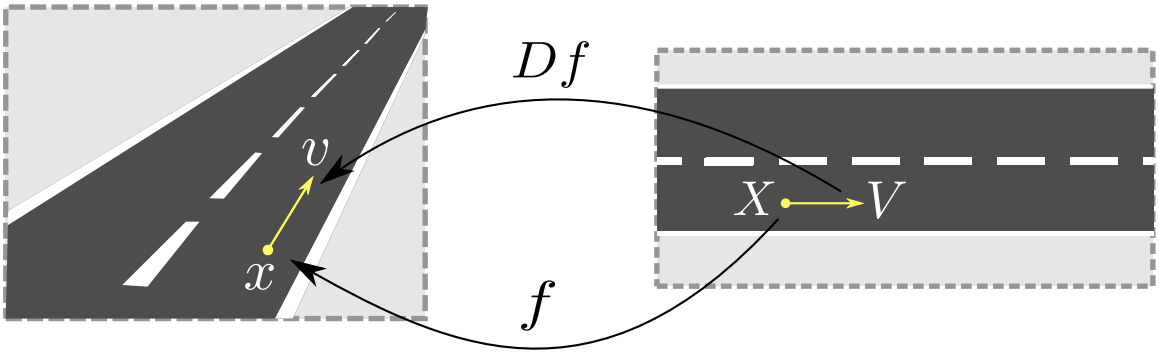


Figure 2.1: The actions of the mapping $f : \mathcal{W} \to \mathcal{I}$ and its derivative $Df : T\mathcal{W} \to T\mathcal{I}$.

manifold,

$$T\mathcal{I} = \cup_{x \in \mathcal{I}} T_x \mathcal{I} \, .$$

A map $f : \mathcal{W} \to \mathcal{I}$ also encodes the transformation of the tangent space from the world to the image manifold via its *tangent map* $Tf : T\mathcal{W} \to T\mathcal{I}$. At a base point $X \in \mathcal{W}$ the tangent map is the derivative

$$\begin{aligned} T_X f : T_X \mathcal{W} &\to T_{f(X)} \mathcal{I} \\ &\doteq (f(X), Df(X)) \end{aligned} \tag{2.4}$$

which takes a tangent vector $V \in T_X \mathcal{W}$ and returns a vector $v \in T_{f(x)} \mathcal{I}$,

$$v = Df(X) \cdot V \, ,$$

The components of a vector $V \in T_{X_0} \mathcal{W}$ are expressed in local coordinates with coordinate chart φ_1 using equations (2.2) and (2.3) as

$$\begin{aligned} (Q, \dot{Q}) &= (\varphi_1(X_0), D\varphi_1(X_0) \cdot V) \\ &= T_{X_0} \varphi_1(X_0, V) \end{aligned} \tag{2.5}$$

which is induced by the coordinate tangent map $T\varphi_1$. The components (2.5) are known as the *tangent lifted coordinates*. The mapping f can be expressed in tangent lifted coordinates by invoking (2.1) so that for local coordinates q on $\mathcal{I}$, the transformation

$$q = f(Q) \tag{2.6}$$

by which we more explicitly mean

$$q = (\varphi_2 \circ f \circ \varphi_1^{-1})(Q)$$

is extended to the transformation of the tangent space

$$\begin{aligned}
(q, \dot{q}) &= T_{x_0}\varphi_2 T_{X_0} f(\varphi_1^{-1}(Q), D\varphi_1^{-1}(Q) \cdot \dot{Q}) \\
&= T_{x_0}\varphi_2(f \circ \varphi_1^{-1}(Q), Df(\varphi_1^{-1}(Q)) \cdot D\varphi_1^{-1}(Q) \cdot \dot{Q}) \\
&= (\varphi_2 \circ f \circ \varphi_1^{-1}(Q), D\varphi_2(f \circ \varphi_1^{-1}(Q)) \cdot Df(\varphi_1^{-1}(Q)) \cdot D\varphi_1^{-1}(Q) \cdot \dot{Q}) \\
&= (f(Q), Df(Q) \cdot \dot{Q})
\end{aligned}$$

in the abbreviated notation of equation (2.6). Now if $X = \varphi_1^{-1}(Q)$, we have

$$\frac{\partial}{\partial Q_i}(X) = \frac{\partial}{\partial Q_i}\varphi_1^{-1}(Q) = D\varphi_1^{-1}(Q) \cdot E_i \tag{2.7}$$

where the E_i are the canonical basis vectors; $E_1 = (1, 0)^T$, $E_2 = (0, 1)^T$. The coordinate tangent map $D\varphi_1(X)$ is an isomorphism; this follows naturally from the fact that at all points $X \in \mathcal{W}$ the manifold "looks locally like $\mathbb{R}^2$" as implied by the mapping $\varphi(U) \to \mathbb{R}^2$ for some neighborhood $U \subset \mathcal{W}$ containing X. The map $D\varphi_1^{-1}(Q)$ is therefore also an isomorphism, which makes the set of vectors (2.7) a

valid basis set for $T_X\mathcal{W}$. Accordingly we can decompose any vector $V \in T_X\mathcal{W}$ into its components

$$V = \sum_i V_i \frac{\partial}{\partial Q_i}(X) \tag{2.8}$$

and in particular, the coordinates $\dot{Q} = (\dot{Q}_1, \dot{Q}_2)^T$ on $T_X\mathcal{W}$ are the components of the tangent-lifted vector $D\varphi_1^{-1}(Q) \cdot \dot{Q}$ with respect to this basis:

$$\begin{aligned} D\varphi_1^{-1}(Q) \cdot \dot{Q} &= \sum_i \dot{Q}_i D\varphi_1^{-1}(Q) \cdot E_i \\ &= \sum_i \dot{Q}_i \frac{\partial}{\partial Q_i}(X) \,. \end{aligned}$$

While the notion of a tangent map is a fundamental to how we will frame the mapping of vectors from the world to the image and vice versa, we need to extend our calculus to be able to take derivatives of a function defined over a manifold with respect to a vector field for instance, or even calculate the rate of change of one vector field with respect to another. This is tackled in section 2.2.

2.2 Flows & Lie Derivatives

A *section* of a tangent bundle (on the image manifold in this case) is a map $v : \mathcal{I} \to T\mathcal{I}$ that assigns every point $x_0 \in \mathcal{I}$ a tangent vector $v_{x_0} \in \mathbb{R}^2$. A *vector field* is then an instance of a section of the tangent bundle; the vector fields that interest us generate flows that vary smoothly over the flow lines themselves even if they may exhibit discontinuties in other directions. The rationale is that traffic

movement, even if accelerating or slowing down, changes smoothly in the direction of travel while vehicles in different lanes may travel at very different speeds or even opposite directions. The set of all vector fields on manifold $\mathcal{I}$ is denoted $\mathfrak{X}(\mathcal{I})$.

The tangent map $T_X f$ of equation (2.4) can be extended from vectors to deal with vector fields via the *pushforward* f_*. Provided f is differentiable and invertible, the pushforward of V takes for every point $x \in \mathcal{I}$ the tangent vector $V(X)$ at its preimage $X = f^{-1}(x)$ which it then tangent-lifts to $T_x\mathcal{I}$,

$$f_* V \doteq Tf \circ V \circ f^{-1} \,. \tag{2.9}$$

The *pullback* f^* of a diffeomorphism $f : \mathcal{W} \to \mathcal{I}$ is the opposite of a pushforward; rather than defining how the vector field $V \in \mathfrak{X}(\mathcal{W})$ should be transformed to $T\mathcal{I}$, the pullback defines how a vector field $v \in \mathfrak{X}(\mathcal{I})$ should be transformed to $T\mathcal{W}$,

$$f^* v \doteq Tf^{-1} \circ v \circ f \tag{2.10}$$

which by comparison with equation (2.9) is just the pushforward of f^{-1}:

$$f^* = (f^{-1})_* \,.$$

To calculate these transformations of vector fields we use local coordinates. Where $V(Q)$ is understood to be the vector $V(\varphi_1^{-1}(Q))$ covered by coordinate chart φ_1 and the abbreviated function signature (2.6) is used, the pushforward of V at Q is

$$(f_* V)(Q) = (f(Q), Df(Q) \cdot V(Q)) \,.$$

In particular we can identify the tangent lifted coordinates (2.5) as the components of the pushfoward by coordinate chart φ_1. The pullback of v is

$$\begin{aligned}(f^*v)(Q) &= (f^{-1}(f(Q)), D(f^{-1})(f(Q)) \cdot v(f(Q))) \\ &= (Q, (Df)^{-1}(Q) \cdot v(f(Q)))\end{aligned}$$

by the Inverse Function Theorem.

In the case that f is not a mapping to the image manifold but rather defines a scalar field $f : \mathcal{W} \to \mathbb{R}$, the *directional derivative* $\langle df(X_0), V(X_0)\rangle$ is the exactly the pushforward of $V(X_0)$ by f,

$$\begin{aligned}\langle df(X_0), V(X_0)\rangle &= Df(X_0) \cdot V(X_0) \\ &= \sum_i \frac{\partial f}{\partial X_i}(X_0) V_i(X_0) \end{aligned} \tag{2.11}$$

which has the usual interpretation as the derivative of f at X_0 in the direction of $V(X_0)$. The extension to the whole manifold is trivial:

$$\langle df(X_0), V(X_0)\rangle = \langle df, V\rangle (X_0)$$

which states that the directional derivative at X_0 is the evaluation there of another scalar field, namely the inner product of df with the vector field V. To put equation (2.11) in the context of equation (2.3), given a path $\gamma(t) \in \mathcal{W}$ for which $\gamma(0) = X_0$ and $\frac{d}{dt}\Big|_{t=0} \gamma(t) = V(X_0)$, the directional derivative is

$$\langle df, V\rangle (X_0) \doteq \frac{d}{dt}\bigg|_{t=0} f \circ \gamma(t)\,. \tag{2.12}$$

The scalar field $\langle df, V\rangle$ can then be understood as the derivative of f along the paths in the manifold that are exactly described by the vector field V; these are the *integral curves*. An integral curve of the differentiable vector field V is a differentiable path $\gamma : \mathbb{R} \to \mathcal{W}$ for which

$$\frac{d}{dt}\gamma(t) = V(\gamma(t)) \,, \tag{2.13}$$

that is, they "follow the flow" induced by V. In fact, the *flow* of V is the differentiable map $\Phi : \mathcal{W} \times \mathbb{R} \to \mathcal{W}$ where $\Phi_X(t)$ marks out the unique integral curve of V for which $\Phi_X(0) = X$ and $\frac{d}{dt}\Big|_{t=0} \Phi_X(t) = V(X)$. We will have more use for the equivalent notation $\Phi_t(X)$ which emphasises the flow field 'frozen' at time t. The equation (2.12) can now be extended to the whole manifold $\mathcal{W}$ with the identification

$$\langle df, V\rangle = \frac{d}{dt}\bigg|_{t=0} f \circ \Phi_t$$

which is simply the definition of the derivative of the pullback by Φ_t of function f,

$$\langle df, V\rangle = \frac{d}{dt}\bigg|_{t=0} \Phi_t^* f \,. \tag{2.14}$$

Being able to describe the derivative $\langle df, V\rangle$ of a function over a vector field leads to an important extension: the derivative of a vector field with respect to another vector field. The field $\langle df, V\rangle$ is known as the *Lie derivative of f along V* and denoted $\mathcal{L}_V f$; we would like to be able to pin down $\mathcal{L}_{V_1} V_2$, the Lie derivative of V_2 along V_1 which is itself a vector field. Taking inspiration from equation (2.14), what we mean by this derivative is the quantity

$$\mathcal{L}_{V_1} V_2 \doteq \frac{d}{dt}\bigg|_{t=0} \Phi_t^* V_2$$

where Φ_t maps out the flow induced by vector field V_1. Using equation (2.10) this becomes

$$\begin{aligned}
\mathcal{L}_{V_1} V_2 &= \left.\frac{d}{dt}\right|_{t=0} D\Phi_t^{-1} \circ V_2 \circ \Phi_t \\
&= \left[\frac{d}{dt}\left(D\Phi_t\right)^{-1} \circ V_2(\Phi_t) + \left(D\Phi_t\right)^{-1} \circ \frac{d}{dt} V_2(\Phi_t)\right]_{t=0} \\
&= \left[-\left(D\Phi_t\right)^{-1} \cdot \frac{d}{dt}\left(D\Phi_t\right) \cdot \left(D\Phi_t\right)^{-1} \circ V_2(\Phi_t) + \left(D\Phi_t\right)^{-1} \circ \frac{d}{dt} V_2(\Phi_t)\right]_{t=0} \\
&= -\left.\frac{d}{dt}\right|_{t=0} D\Phi_t \circ V_2 + \left.\frac{d}{dt}\right|_{t=0} V_2(\Phi_t) \\
&= -D\left(\left.\frac{d}{dt}\right|_{t=0} \Phi_t\right) \circ V_2 + DV_2 \circ \left(\left.\frac{d}{dt}\right|_{t=0} \Phi_t\right) \\
&= -DV_1 \circ V_2 + DV_2 \circ V_1
\end{aligned}$$

where at time $t = 0$, Φ_t is just the identity operator $\Phi_0(X) = X$ and its derivative $D\Phi_0(X)$ is the identity matrix. By DV_1, the Jacobian matrix with elements $(DV_1)_{ij} = \partial(V_1)_i/\partial Q_j$ in local coordinates is intended, though it is often more fruitful to treat $DV_1 \circ V_2$ in a coordinate free manner as the directional derivative

$$\left(DV_1 \circ V_2\right)(X) = \left.\frac{d}{dt}\right|_{t=0} V_1\left(X + tV_2(X)\right) \,. \tag{2.15}$$

The Lie derivative $\mathcal{L}_{V_1} V_2$ happens to equate to the *Jacobi-Lie bracket* $[V_1, V_2]$ on $\mathfrak{X}(\mathcal{W})$,

$$[V_1, V_2]_{\mathfrak{X}(\mathcal{W})} = DV_2 \circ V_1 - DV_1 \circ V_2 \,, \tag{2.16}$$

a useful fact in the discussion of minimization on Lie groups in section 6.3.3.

2.3 The Cotangent Space

Taking another look at the directional derivative in equation (2.12), we note that df exists in a natural dual space to V; the pairing between them results in a scalar field. The quantity $df(X)$ is known as a *cotangent vector* or the *differential of f at X*, and resides in the *cotangent space* $T^*_X\mathcal{W}$, itself a member of the *cotangent bundle* $T^*\mathcal{W}$. Just as any vector field $V \in T\mathcal{W}$ can be written in terms of basis vector fields $\frac{\partial}{\partial Q_i}(X)$ with coefficients $V_i(X)$, a differential P is written with the basis differentials $dQ_i(X)$,

$$P = \sum_i P_i(X) dQ_i(X) \, .$$

In the case of a Euclidean space, the bilinear pairing $\langle P, V \rangle$ should result in the scalar field $\sum_i P_i V_i$, so the subsitution

$$\begin{aligned} \langle P, V \rangle &= \left\langle \sum_i P_i dQ_i, \sum_j V_j \frac{\partial}{\partial Q_j} \right\rangle \\ &= \sum_{i,j} P_i V_j \left\langle dQ_i, \frac{\partial}{\partial Q_j} \right\rangle \end{aligned}$$

implies that we must have

$$\left\langle dQ_i, \frac{\partial}{\partial Q_j} \right\rangle = \delta_{ij} \, .$$

To explore what it means to perform a *cotangent lift* on a covector $P(X)$ given a mapping $f : \mathcal{W} \to \mathcal{I}$, we formalize the notion of duality in the tangent map transformation. Given the pairing $\langle P, V \rangle_X \doteq \langle P(X), V(X) \rangle$, the cotangent map T^*f is defined by

$$\langle T^*f \circ p, V \rangle_X \doteq \langle p, Tf \circ V \rangle_{f(X)} \tag{2.17}$$

where

$$T^*f \circ p = P\,. \tag{2.18}$$

The covector $p(x)$ lies in the vector space $T^*_x\mathcal{I}$, and therefore T^*f maps differentials in the opposite direction to the mapping of vector fields by Tf. For this reason it is normally more convenient to use the cotangent lift T^*f^{-1} when f is a diffeomorphism; this maps covector $P(X) \in T^*_X\mathcal{W}$ into $T^*_x\mathcal{I}$. It should be noted that (2.17) equates two pairings defined on different manifolds; in fact the pairing $\langle\cdot,\cdot\rangle_x : T^*_x\mathcal{I} \times T_x\mathcal{I} \to \mathbb{R}$ is *induced* here by f. A direct consequence of this is the orthogonality under this pairing of the induced basis vectors and differentials

$$\frac{\partial}{\partial q}(x) = T_X f \circ \frac{\partial}{\partial Q}(X)\ ,\ dq(x) = T^*_{f(X)}f^{-1} \circ dQ(X)$$

recalling that the components q are the local coordinates on $\mathcal{I}$. This follows by first noting that

$$\begin{aligned}
\langle P, V\rangle &= \left\langle P, Tf^{-1} \circ v\right\rangle \\
&= \left\langle T^*f^{-1} \circ P, v\right\rangle \\
&= \left\langle T^*f^{-1} \circ P, Tf \circ V\right\rangle \\
&= \left\langle T^*f \circ T^*f^{-1} \circ P, V\right\rangle
\end{aligned}$$

which implies that

$$(T^*f)^{-1} = T^*f^{-1}\,, \tag{2.19}$$

and using this result to show

$$\begin{aligned}\left\langle dq_i, \frac{\partial}{\partial q_j} \right\rangle_x &= \left\langle T^* f^{-1} \circ dQ_i, Tf \circ \frac{\partial}{\partial Q_j} \right\rangle_x \\ &= \left\langle dQ_i, (Tf)^{-1} \circ Tf \circ \frac{\partial}{\partial Q_j} \right\rangle_X \\ &= \left\langle dQ_i, \frac{\partial}{\partial Q_j} \right\rangle_X \\ &= \delta_{ij} \, .\end{aligned}$$

Now owing to the bilinearity of the pairings $\langle \cdot, \cdot \rangle$ on both $\mathcal{W}$ and $\mathcal{I}$, the orthonormality of the bases on both spaces and the fact that $T_X f$ is the linear operator $Df(X)$, the dual operator $T_X^* f$ is in fact the transposed matrix $(Df(X))^T$. Again making use of relation (2.19), this means that the local coordinate expression (2.18) can be rewritten in the convenient form

$$p(q) = (Df(Q))^{-T} P(Q) \, . \tag{2.20}$$

2.4 Metrics

Although we are now equipped with the tools to transform vector fields and differentials between the world manifold and the image, we need to be able to compare and measure transformed quantities. For instance, two vehicle trajectories on the road that are clearly nonparallel in the image should nevertheless be measureable as parallel under an appropriate measurement methodology. This is done with the right choice of *Riemannian metric*, a symmetric bilinear form which when defined

at $x \in \mathcal{I}$ has the signature $\langle \cdot, \cdot \rangle_{T_x\mathcal{I}} : T_x\mathcal{I} \times T_x\mathcal{I} \to \mathbb{R}$. As the similarity in notation suggests, there is a close tie (see equation (2.21)) between the metric and the pairing (2.11) that was introduced as the directional derivative. The metric will generally be differentiated in the following with a subscript indicating the vector space hosting its arguments.

Two vectors $v_1, v_2 \in T_x\mathcal{I}$ are orthogonal under the metric if and only if $\langle v_1, v_2 \rangle_{T_x\mathcal{I}} = 0$. The Euclidean metric is the canonical choice on world manifold $\mathcal{W}$, so for $V_1, V_2 \in T_X\mathcal{W}$,

$$\langle V_1, V_2 \rangle_{T_X\mathcal{W}} = V_1 \cdot V_2 \, .$$

We would like the metric on $\mathcal{I}$ to be specified such that for v_i the pushforward by $f : \mathcal{W} \to \mathcal{I}$ of V_i,

$$\begin{aligned} \langle v_1, v_2 \rangle_{T_x\mathcal{I}} &\doteq \left\langle T_x f^{-1} \circ v_1, T_x f^{-1} \circ v_2 \right\rangle_{T_X\mathcal{W}} \\ &= \langle V_1, V_2 \rangle_{T_X\mathcal{W}} \end{aligned}$$

where $x = f(X)$. In fact, this definition of the metric on $\mathcal{I}$ is simply the pullback by f^{-1} of the metric on $\mathcal{W}$, and is known as the *induced metric*. The Euclidean metric (used on the world manifold) is location-independent; two vectors at the same basepoint have a constant inner product no matter where that basepoint happens to be. On the other hand, the metric on the image inherits a location dependence through the tangent lifts of its arguments, which are certainly location dependent. This is how the distance effect is manifested: objects further away should appear

smaller in the image, or equally, a similarly sized object in the image positioned closer to the horizon should represent a larger object in the world.

Returning to the directional derivative, given a scalar field f defined over manifold $\mathcal{W}$, the *gradient of* f is the vector field ∇f defined by

$$\langle df, V \rangle_X = \langle \nabla f, V \rangle_{T_X \mathcal{W}} \tag{2.21}$$

for all $X \in \mathcal{W}$ and $V \in T_X \mathcal{W}$. Because the elements of $T_X \mathcal{W}$ here are the usual vectors and the metric is bilinear, it can be represented by the matrix $J(X)$ say,

$$\langle \nabla f, V \rangle_{T_X \mathcal{W}} = (\nabla f(X))^T \cdot J(X) \cdot V(X)$$

which, coupling equation (2.21) with equation (2.11) gives

$$\begin{aligned} \nabla f(X) &= J(X)^{-T} df(X) \\ &= J(X)^{-1} df(X) \end{aligned} \tag{2.22}$$

since $J(X)$ is symmetric. The gradient then is a linear transformation of the differential at each point in the manifold, but also varies over the manifold in a manner related to the variation of the metric. Equation (2.22) also provides us with a mapping $T^*\mathcal{W} \to T\mathcal{W}$ closely related to the momentum map of section 3.3.1 that will be useful in establishing gradient descent equations.

Chapter 3

Matrix Lie Groups & Lie Algebras

We shift focus now from operations on the world and image manifolds to the mapping between them. We have relied thus far on knowing the true mapping but in fact its accurate estimation is precisely what we want to achieve. We propose to do so by searching over the space of possible mappings for one that uniquely minimizes some error measure. The applicable space in this case is the Lie group $SL(3)$; each element of the group effects a different image for a fixed world scene. A *Lie group* is a smooth manifold where the group operation and its inverse are smooth in addition to the usual group properties. For real-valued matrix Lie groups in particular, the group operation is matrix multiplication; these are submanifolds of the set of all real-valued $n \times n$ matrices $\mathcal{M}(n, \mathbb{R})$ and subgroups of the general linear group of invertible real-valued matrices $GL(n, \mathbb{R})$.

3.1 The Lie Algebra

Lie groups describe succintly the continuously-varying nature of transformations such as rotations, translations or (as in our case) projectivities, and are traversed by means of left and right translations

$$L_h(g) = g \cdot h \,, \quad R_h(g) = g \cdot h$$

for g and h in G, the matrix Lie group of interest. The composition relations

$$L_h \circ L_g = L_{h \cdot g} \,, \quad R_h \circ R_g = R_{g \cdot h} \tag{3.1}$$

follow directly. Thinking of L_h simply as a mapping from G onto itself, we can construct the vector ξ as the velocity of a path $\gamma(t) \in G$ through g at $t = 0$,

$$\xi(g) \doteq \left. \frac{d}{dt} \right|_{t=0} \gamma(t) \tag{3.2}$$

just as in equation (2.3), and immediately apply the tangent lift (2.4),

$$\begin{aligned} T_g L_h \xi &\doteq \left. \frac{d}{dt} \right|_{t=0} h \circ \gamma(t) \\ &= (L_h)_* \, \xi \,. \end{aligned} \tag{3.3}$$

Equations (3.2) and (3.3) extend trivially to vector fields $\xi \in TG$ on G where the notion of a path is replaced with that of a flow. This gives rise to a particular vector field $V_\xi^L(g)$, the *left extension* of vector $\xi \in T_e G$

$$V_\xi^L(g) \doteq T_e L_g \xi$$

where e is the group identity. That is, V_ξ^L is the vector field generated by *left* tangent-lifting vector ξ from the identity. Now we say that a vector field V is *left-invariant* if

$$TL_g V = V$$

for any $g \in G$, and V_ξ^L is shown to be such a vector field;

$$\begin{aligned} T_h L_g V_\xi^L(h) &= T_h L_g \circ T_e L_h \xi \\ &= T_e(L_g \circ L_h)\xi \\ &= T_e L_{g\cdot h}\xi \\ &= V_\xi^L(g \cdot h)\,. \end{aligned}$$

Then given a left-invariant vector field V, by definition

$$V(g) = T_e L_g V(e) = V_{V(e)}^L(g) \tag{3.4}$$

and therefore any left-invariant vector field V is just the left extension of its own vector $V(e) \in T_eG$. We have now established an isomorphism λ between the vector space T_eG and the space $\mathfrak{X}^L(G)$ of all left-invariant vector fields on G, $\lambda(\xi) = V_\xi^L$. This is the first step toward defining the *Lie bracket* $[\cdot,\cdot]_{T_eG}$ to be bundled, along with the vector space T_eG, into the *Lie algebra* $\mathfrak{g}$ of Lie group G. The bracket is a bilinear operator that is both skew-symmetric and observes the Jacobi identity. The Jacobi-Lie bracket (2.16) is exactly such an operator on vector fields. What we need however, is a conforming operator on elements of T_eG. Luckily, the pullback of the

Jacobi-Lie bracket by λ will achieve exactly that, so we can define the Lie bracket as

$$\begin{aligned}
[\xi_1, \xi_2]_{T_eG} &\doteq \left(\lambda^{-1}\right)_* [\lambda(\xi_1), \lambda(\xi_2)]_{\mathfrak{X}^L(G)} \\
&= \left(\lambda^{-1}\right)_* \left[V^L_{\xi_1}, V^L_{\xi_2}\right]_{\mathfrak{X}^L(G)} \\
&= \left(\lambda^{-1}\right)_* \mathcal{L}_{V^L_{\xi_1}} V^L_{\xi_2}
\end{aligned} \tag{3.5}$$

which should be translated as "the unique element ξ of T_eG for which $\lambda(\xi) = \mathcal{L}_{V^L_{\xi_1}} V^L_{\xi_2}$". That the Jacobi-Lie bracket $[\lambda(\xi_1), \lambda(\xi_2)]_{\mathfrak{X}^L(G)}$ is still left-invariant is a straightforward follow-on the equality of mixed partials applied to the vector field definitions. Fortunately this simplifies a lot when dealing with matrix Lie groups. Given matrix $\xi \in T_gG$ where $\xi = \frac{d}{dt}\Big|_{t=0} \gamma(t)$ for a path $\gamma(t) \in G$, $\gamma(0) = g$, the tangent lift operation is mere matrix multiplication as we see from the specialization of equation (3.3),

$$T_g L_h \xi = \frac{d}{dt}\bigg|_{t=0} h \circ \gamma(t) = h \cdot \xi \,.$$

Further, applying (3.4) it is apparent that all left-invariant vector fields have the form

$$V(g) = T_e L_g V(e) = g \cdot V(e) \tag{3.6}$$

so that from equations (2.16) and (2.15) for two left-invariant vector fields V_1 and V_2,

$$\begin{aligned}
\left(\mathcal{L}_{V_1} V_2\right)(g) &= \left(DV_2 \cdot V_1\right)(g) - \left(DV_1 \cdot V_2\right)(g) \\
&= \frac{d}{dt}\bigg|_{t=0} \left[V_2\left(g + tV_1(g)\right) - V_1\left(g + tV_2(g)\right)\right] \\
&= \frac{d}{dt}\bigg|_{t=0} \left[\left(g + tV_1(g)\right) \cdot V_2(e) - \left(g + tV_2(g)\right) \cdot V_1(e)\right]
\end{aligned}$$

$$
\begin{aligned}
&= \left.\frac{d}{dt}\right|_{t=0} [(g + tg \cdot V_1(e)) \cdot V_2(e) - (g + tg \cdot V_2(e)) \cdot V_1(e)] \\
&= g \cdot (V_1 \cdot V_2)(e) - g \cdot (V_2 \cdot V_1)(e) \\
&= (V_1 \cdot V_2 - V_2 \cdot V_1)(g) .
\end{aligned}
$$

The pullback to the tangent space T_eG is evident; it is simply $V_1(e) \cdot V_2(e) - V_2(e) \cdot V_1(e)$. The Lie bracket (3.5) is then just

$$[\xi_1, \xi_2]_{T_eG} = \xi_1 \cdot \xi_2 - \xi_2 \cdot \xi_1$$

and is equivalent to the ad operator on matrix Lie algebras (introduced here for later reference only),

$$\mathrm{ad}_{\xi_1} \xi_2 \doteq [\xi_1, \xi_2]_{T_eG} .$$

Taking stock, we have established the Lie algebra $\mathfrak{g}$ as the tangent space T_eG coupled with the Lie bracket. We have already used the fact that every left invariant vector field on G is generated by some element $\xi \in T_eG$, and is in fact equivalent to the left extension V_ξ^L. To further connect the Lie group and Lie algebra we define the *one-parameter subgroup corresponding to* ξ as the integral curve $\gamma_\xi(t)$ (see equation (2.13)) for the vector field V_ξ^L, where $\gamma_\xi(0) = e$. The Lie *exponential map* $\exp : \mathfrak{g} \to G$ provides a formal mapping between the Lie algebra and the Lie group; it is defined

$$\exp(\xi) \doteq \gamma_\xi(1) .$$

Again this result is made more concrete in the case of matrix groups. Let us suppose the exponential map to be the usual matrix exponential $\exp(\xi) = 1 + \xi + \frac{\xi^2}{2!} + \ldots$ for

which it is easily seen, satisfies the ODE

$$\frac{d}{dt}\exp(t\xi) = \exp(t\xi)\xi \,.$$

However, as equation (3.6) states, $\exp(t\xi)\xi$ is just the evaluation at $\exp(t\xi)$ of the left extension of ξ, so

$$\frac{d}{dt}\exp(t\xi) = V_\xi^L(\exp(t\xi)) \,.$$

But this, along with the verification $\gamma_\xi(0) = \exp(0) = e$, is just the definition of the one-parameter subgroup $\gamma_\xi(t) = \exp(t\xi)$ which validates the matrix exponential as the form of the exponential map in the case of matrix groups. It can also be shown (see for example [22, pp. 103]) that not only is *every* one-parameter subgroup of G of the form $\exp(t\xi)$ for some $\xi \in T_eG$, but the exponential map is a diffeomorphism from $\mathfrak{g}$ to a neighborhood of $e \in G$. This means that provided we are not "too far" from the identity, the inverse of the exponential map can be used as a coordinate chart over that neighborhood. That is, we represent any suitable group element g by the components α_i corresponding to the basis vectors ξ_i of the Lie algebra element $\xi = \exp^{-1}(g)$,

$$g = \exp(\xi) = \exp\left(\textstyle\sum_i \alpha_i \xi_i\right) \,. \tag{3.7}$$

These are called the *canonical coordinates of the first kind*, and will appear in section 6.3.3 as an efficient parameterization of the Lie group.

3.2 Lie Group Actions

Lie groups interact with manifolds through *Lie group actions* to effect coordinate transformations such as rotations or displacement along a flow. In the case of a Lie group acting on itself, the result is a change in the state of the transformation the group itself describes. For any point $X \in \mathcal{W}$ for example, and any group elements $g, h \in G$ the *left* action $\phi_\cdot : G \times \mathcal{W} \to \mathcal{W}$ is a diffeomorphism defined by the properties

1. $\phi_e(X) = X$,

2. $\phi_h(\phi_g(X)) = \phi_{hg}(X)$

which is often abbreviated to $gX \doteq \phi_g(X)$. A simple example for concreteness is that of the rotation group $SO(3)$ operating on $\mathbb{R}^3$; the action ϕ_g for $g \in SO(3)$ on $X \in \mathbb{R}^3$ in this case is linear: $\phi_g(X) = g \cdot X$. The *right* action entertains the expected difference $\phi_h(\phi_g(X)) = \phi_{gh}(X)$ and is often shortened to $Xg \doteq \phi_g(X)$. The context will otherwise dictate whether we are dealing with a left or right action in the following.

The group action is a generalization of a flow, and induces the tangent lift described by equation (2.4) but this time from the tangent bundle $T\mathcal{W}$ onto itself

$$T\phi_g(X, V) = (\phi_g(X), T_X\phi_g(V)) \,, \tag{3.8}$$

while the cotangent lift consistent with equation (2.19) is

$$\begin{aligned} T^*\left(\phi_g\right)^{-1}(X,P) &= T^*\phi_{g^{-1}}(X,P) \\ &= (\phi_g(X), T^*_{\phi_g(X)}\phi_{g^{-1}}(P)) \end{aligned} \tag{3.9}$$

for $P \in T^*_X\mathcal{W}$. Recalling that given a path $\gamma(t) \in \mathcal{W}$ for which $\gamma(0) = X$ and $\frac{d}{dt}\Big|_{t=0}\gamma(t) = V$, the expression $T_X\phi_g(V)$ in the case of a matrix group left action becomes

$$\begin{aligned} T_X\phi_g(V) &\doteq \frac{d}{dt}\bigg|_{t=0} \phi_g(\gamma(t)) \\ &= \frac{d}{dt}\bigg|_{t=0} g\cdot\gamma(t) \\ &= g\cdot V\,, \end{aligned}$$

simplifying equation (3.8) to

$$T\phi_g(X,V) = (g\cdot X, g\cdot V)\,.$$

The further specialization to a matrix group acting on itself is the trivial replacement

$$T\phi_g(h,\xi) = (g\cdot h, g\cdot\xi) \tag{3.10}$$

for $\xi \in T_hG$. Similarly for the cotangent lift, we analyze

$$\begin{aligned} \left\langle T^*\left(\phi_g\right)^{-1}P, V\right\rangle_{\phi_g(X)} &= \left\langle P, T\phi_g^{-1}V\right\rangle_X \\ &= \left\langle P, g^{-1}\cdot V\right\rangle_X \\ &= \left\langle g^{-T}\cdot P, V\right\rangle_{\phi_g(X)} \end{aligned}$$

where $P \in T_X\mathcal{W}$, $V \in T_{\phi_g(X)}\mathcal{W}$ and the last line recalls the reasoning at the end of section 2.3. Then

$$T^* \left(\phi_g\right)^{-1} (X, P) = (g \cdot X, g^{-T} \cdot P) \tag{3.11}$$

and the analogous substitution to equation (3.10) can be made when the manifold is the Lie group itself.

We will be primarily interested in the action induced by a Lie algebra element rather than a group element. The *infinitesimal generator* $\xi_{\mathcal{W}}$ on manifold $\mathcal{W}$ is a vector field that describes how the element $\xi \in \mathfrak{g}$ should be expressed on the tangent bundle $T\mathcal{W}$ via the right action[1]

$$\xi_{\mathcal{W}}(X) \doteq \left.\frac{d}{dt}\right|_{t=0} \phi_{\exp(t\xi)}(X) = \left.\frac{d}{dt}\right|_{t=0} X \exp\left(t\xi\right) . \tag{3.12}$$

[1]Note that this definition is different from the one usually found in the literature (see for example [23, Definition 6.22]) where $\xi_{\mathcal{W}}(X)$ is given as the *left* action $\left.\frac{d}{dt}\right|_{t=0} \exp\left(t\xi\right) X$. The reason for this is that the latter is not in general left invariant, while we plan on exploiting such invariance later. In contrast, if we focus on the case of the Lie group acting on itself (rather than another manifold) and if we take a left invariant action to mean that

$$\left(L_g\right)_* \xi(h) = \xi(\phi_g^L(h))$$

for any vector field ξ and left action ϕ^L, our definition of the infinitesimal generator exhibits left invariance:

$$\begin{aligned}
\left(L_g\right)_* \xi_G(h) &= \left.\frac{d}{dt}\right|_{t=0} \phi_g^L(\phi_{\exp(t\xi)}(h)) \\
&= \left.\frac{d}{dt}\right|_{t=0} g\left(h \exp\left(t\xi\right)\right) \\
&= \left.\frac{d}{dt}\right|_{t=0} \phi_g^L(h) \exp\left(t\xi\right) \\
&= \xi_G(\phi_g^L(h)) .
\end{aligned}$$

In the case that the manifold being acted on by a matrix group is the group itself, equation (3.12) reduces to the expression

$$\xi_G(g) = \left.\frac{d}{dt}\right|_{t=0} g \cdot \exp(t\xi) = g \cdot \xi \tag{3.13}$$

which is also just the pushforward $(L_g)_* \xi$ viewed as a vector field (and evaluated at $g \in G$).

3.3 Gradient Descent on Matrix Lie Groups

It is of primary importance that given an energy functional (or penalization) $E(g)$, we are able to identify the group element $g^* \in G$ that minimizes the energy,

$$g^* = \arg\min_{g \in G} E(g) \quad . \tag{3.14}$$

An energy measuring the degree of violation of some constraint is then minimized by the set of transformations most consistent with the constraint. If we can ensure that the minimization yields a unique solution then we can identify the transformation most representative of the true one generating our data. We use gradient descent to steer a path to the solution on the group in both chapters 5 and 6, the basis of which we establish now.

3.3.1 Reduction to the Lie Algebra

The minimization over G requires some care when formulating the gradient descent equation. For instance, the iterative method

$$g_{k+1} = g_k - t\nabla E(g_k) \tag{3.15}$$

for some small t, valid on a Euclidean space, is inappropriate on G since the group operation (in our case) is multiplication and not addition, implying that the update g_{k+1} by (3.15) may not even lie in the group. Rather, taking the Taylor expansion of E at g_k in direction ξ (where $\xi \in \mathfrak{g}$) we get ([24])

$$E(g_k \exp(t\xi)) = \sum_{n=0}^{\infty} \frac{t^n}{n!} \left(\xi_G^n E\right)(g_k) \ , \qquad t \in [0,1] \tag{3.16}$$

where ξ_G is the infinitesimal generator (3.13), that is, $\xi_G(g) = T_e L_g \xi$. Now using equation (2.12) in reverse,

$$\left(\xi_G E\right)(g_k) \doteq \left.\frac{d}{dt}\right|_{t=0} E(g_k \exp(t\xi)) = \langle dE(g_k), \xi_G(g_k)\rangle$$

and we can expand (3.16) as

$$E(g_k \exp(t\xi)) = E(g_k) + t\,\langle dE(g_k), \xi_G(g_k)\rangle + o\left(\|\xi_G(g_k)\|^2\right) \ . \tag{3.17}$$

In analogy with the way in Euclidean space that the Taylor expansion

$$E(u_k + tw) = E(u_k) + t\,\langle dE(u_k), w\rangle + o\left(\|w\|^2\right)$$

yields the gradient descent update

$$u_{k+1} = u_k - t\nabla E(u_k) \tag{3.18}$$

by the identification of the tangent space with its dual, equation (3.17) suggests an update of the form

$$g_{k+1} = g_k \exp(-t\widetilde{\nabla E}(g_k)) \tag{3.19}$$

where $\widetilde{\nabla E}$ encompasses a transformation of the 1-form dE from T_g^*G to T_gG as well as an operation to account for the difference in tangent vectors $\xi_G(g_k)$ and $\xi = \xi_G(e)$. More succinctly,

$$(\xi_G E)(g_k) = \langle dE(g_k), \xi_G(g_k)\rangle = \left\langle \widetilde{\nabla E}, \xi \right\rangle_{\mathfrak{g}} \tag{3.20}$$

where $\langle \cdot, \cdot \rangle_{\mathfrak{g}} : \mathfrak{g} \times \mathfrak{g} \to \mathbb{R}$ is some chosen Riemannian metric on the group. We proceed to derive an expression for $\widetilde{\nabla E}$ but note first that depending on its parameterization, $\widetilde{\nabla E}$ may not lie in the Lie algebra $\mathfrak{g}$. This is easily seen if for example the coordinates used are those of the ambient space: the derivative of E with respect to the nine matrix entries g_{ij} making up the group element $g \in SL(3)$ lies in $\mathbb{R}^9$. On the other hand, the derivative with respect to any sufficient eight-dimensional coordinate system lies in the tangent bundle of $SL(3)$ and occupies only a subspace of $\mathbb{R}^9$; these derivatives may not coincide. So while for small t the iterates g_k in equation (3.19) should approximately follow the flow on the manifold generated by the ODE $g^{-1}\dot{g} = -\widetilde{\nabla E}(g)$, in practice they may drift away from the manifold. This can be fixed in two ways: either a projection of $\widetilde{\nabla E}$ back into $\mathfrak{g}$ can be made (§ 3.3.2), or a coordinate system chosen that ensures the update is made with a tangent vector guaranteed to lie in $\mathfrak{g}$ (§ 6.3.3).

3.3.2 The Coordinate-Free Approach

To find an expression for $\widetilde{\nabla E}$ we need to establish an isomorphism between T_g^*G and T_gG. Equipped with a link between these spaces, we can then exploit the vector space isomorphism $\xi_G \to \xi$ between left-invariant vector fields on G and $\mathfrak{g}$ to work out what $\widetilde{\nabla E}$ must be. The key to this formulation is the metric $\langle \cdot, \cdot \rangle_{\mathfrak{g}}$ which we extend to the whole of G by left tangent-lifting,

$$\langle u, v \rangle_{T_gG} \equiv \langle T_gL_{g^{-1}}u, T_gL_{g^{-1}}v \rangle_{\mathfrak{g}} = \left\langle g^{-1}u, g^{-1}v \right\rangle_{\mathfrak{g}} . \tag{3.21}$$

As suggested by equation (2.22) but now more generally, we introduce a positive definite, 'inertial' operator $J : TG \to T^*G$ such that

$$\langle J(u), v \rangle \equiv \langle u, v \rangle_{TG} , \tag{3.22}$$

(see [25, pp.323] where the inertial operator is restricted to the Lie algebra). We also define the metric $\langle \cdot, \cdot \rangle_{T^*G}$ by the desired relation $\langle J(u), J(v) \rangle_{T^*G} \equiv \langle u, v \rangle_{TG}$. Then the adjoint operator J^* is just the inverse J^{-1}: for $p \in T_g^*G$ and $u \in T_gG$ we have

$$\langle J(u), p \rangle_{T_g^*G} = \left\langle u, J^{-1}(p) \right\rangle_{T_gG} = \left\langle \left(J^{-1}\right)^* (u), p \right\rangle_{T_g^*G}$$

which by comparison of the first and last terms yields the result. By noting that the left action ϕ_g is just the left translation L_g, equation (3.11) states that the left cotangent-lift of covector p operates as

$$T^*L_{g^{-1}}p = g^{-T}p$$

for matrix Lie groups. This prompts us to write

$$J = g^{-T}\mathbb{I}g^{-1} \tag{3.23}$$

for some symmetric, positive definite operator $\mathbb{I} : \mathfrak{g} \to \mathfrak{g}^*$ that does not depend on g. Equation (3.23) simply states that J is the composition of a tangent-lift into the Lie algebra, a linear mapping into the dual Lie algebra followed by a cotangent-lift into the cotangent space T_g^*G.

If we choose the Euclidean matrix norm as metric

$$\langle \xi, \varphi \rangle_{\mathfrak{g}} \equiv \operatorname{trace}(\xi \, (\mathbb{I}\varphi)^T) \tag{3.24}$$

which extends over G in the following way,

$$\begin{aligned} \langle u, v \rangle_{T_gG} &= \left\langle g^{-1}u, g^{-1}v \right\rangle_{\mathfrak{g}} \\ &= \operatorname{trace}(g^{-1}u(\mathbb{I}g^{-1}v)^T) \\ &= \operatorname{trace}(u(J(v))^T) \end{aligned}$$

then we can also uncover how the dual metric $\langle \cdot, \cdot \rangle_{\mathfrak{g}^*}$ extends to the whole of G. Observing that J is a matrix when G is a matrix Lie group,

$$\begin{aligned} \langle p, Jv \rangle_{T_g^*G} &= \left\langle J^{-1}p, v \right\rangle_{T_gG} \\ &= \operatorname{trace}(J^{-1}p(Jv)^T) \\ &= \operatorname{trace}(pv^T) \end{aligned}$$

as J is symmetric. Due to (3.22) this is also the expression for the natural pairing $\langle p, \upsilon \rangle$. Now for $q \in T_g^*G$ we can make explicit the metric on the contangent bundle,

$$\begin{aligned} \langle p, q \rangle_{T_g^*G} &= \text{trace}(J^{-1}p(JJ^{-1}q)^T) \\ &= \text{trace}(p(J^{-1}q)^T) \quad , \end{aligned} \tag{3.25}$$

and substituting (3.23) into (3.25) gives

$$\langle p, q \rangle_{T_g^*G} = \text{trace}(p(g\mathbb{I}^{-1}g^T q)^T) \, .$$

The Lie algebra element $\widetilde{\nabla E}$ is dual to the *momentum map* $P : T^*G \to \mathfrak{g}^*$, $P(g,p) = J\left(\widetilde{\nabla E}\right)$, an important quantity because it explicitly links the concept of symmetry (or invariance to G) to conserved quantities. While this has later relevance to the description of 3D traffic flow that must yield the invariant quantity (8.4) under the action of G, for now we take P simply to be a mapping from the cotangent space to the dual Lie algebra. Following (3.20) in its identification of p with $dE(g)$ and making explicit the dependence of J on G, we have

$$\widetilde{\nabla E} = J^{-1}(e)P(g_k, p_k) = J^{-1}(e)P(g_k, dE(g_k)) \quad . \tag{3.26}$$

It remains to make explicit the momentum map P. Using the invariance of the inner

product on TG we have

$$\begin{aligned}\langle p, \xi_G(g)\rangle &= \left\langle J^{-1}(g)p, \xi_G(g)\right\rangle_{T_gG} \\ &= \left\langle g^{-1}J^{-1}(g)p, \xi\right\rangle_{\mathfrak{g}} \\ &= \left\langle J(e)g^{-1}J^{-1}(g)p, \xi\right\rangle \\ &= \langle P(g,p), \xi\rangle\end{aligned}$$

and thus

$$P(g,p) = J(e)g^{-1}J^{-1}(g)p \ . \tag{3.27}$$

This yields for the gradient descent vector (3.26) the simple expression

$$\widetilde{\nabla E} = g^{-1}J^{-1}(g)dE(g)$$

which when using the substitution (3.23) and observing that the coefficients of $dE(g)$ are denoted $\nabla_g E$, gives the vector coefficients

$$\widetilde{\nabla E} = \mathbb{I}^{-1}g^T\nabla_g E \quad . \tag{3.28}$$

Referring back to the remark made at the end of section 3.3.1, there is no guarantee that the calculation (3.28) actually lies in the Lie algebra. Aside from reasons of precision, this is due to the expression's coordinate-free nature. It is possible for instance to over-parameterize the group G by selecting the coordinates as those of the ambient space. This is certainly tempting in our case, parameterizing the eight-dimensional group elements $g \in SL(3)$ with the nine elements of the matrices g_{ij}.

Doing so, the derivatives $dE(g)$ then also lie in a nine-dimensional space rather than the tangent space to the manifold T^*G. To rectify this problem we use the projection

$$\mathbb{P}(\xi) \doteq \xi - \frac{\operatorname{trace}(\xi)}{3} I \tag{3.29}$$

which enforces the zero-trace characteristic of all elements in $\mathfrak{g}$.

The minimization problem can be summarized in the following steps (similar to [26, Definition 1]). Once an initial condition g_0 and inertial matrix $\mathbb{I}$ has been chosen,

$$\begin{aligned} \xi_{k+1} &= -\mathbb{P}\left(\mathbb{I}^{-1} g_k^T \nabla_{g_k} E\right) \\ t_{k+1} &= \arg\min_{0 \le t \le 1} E(g_k \exp(t\xi_{k+1})) \\ g_{k+1} &= g_k \exp(t_{k+1}\xi_{k+1}) \,. \end{aligned} \tag{3.30}$$

Chapter 4

2D Projective Geometry and the Homography

The road surface is flat and straight, at least locally, and assumed to be uncambered. In establishing the mapping between the world and the image, we assume all geometrical cues to describe the road plane itself; we will not initially be concerned with the height of the objects in the scene. The well-understood field of 2D planar geometry can then be exploited to reduce structural relationships such as parallel lines or orthogonality to algebraic conditions. We will use these to make a first pass at constructing an algorithm to recover the homography. Much of the foundation of this chapter is owed to [27].

4.1 Basic Elements

4.1.1 Homogeneous Points and Lines

A line in the plane is represented by the equation

$$aX + bY + c = 0 \tag{4.1}$$

for any point (X, Y) lying on the line. The triplet (a, b, c) is not unique however; any triplet $(\gamma a, \gamma b, \gamma c)$ for nonzero γ describes the same line. Therefore (a, b, c) is representative of the equivalence class of such vectors and the space of all such equivalence classes is the *projective space* $\mathbb{P}^2$. Now for any point lying on the line $\boldsymbol{L} = (a, b, c)$, equation (4.1) can also be written $(X, Y, 1) \cdot (a, b, c) = 0$ and in just the same way as the line, the point $(X, Y, 1)$ represents the equivalence class of all points $(\gamma X, \gamma Y, \gamma)$ for nonzero γ because any such point also satisfies (4.1). We say that the point $(X, Y) \in \mathbb{R}^2$ has *homogeneous coordinates* $\overline{\boldsymbol{X}} = (X, Y, 1)$ or equivalently, $(\gamma X, \gamma Y, \gamma)$ and the *inhomogeneous* coordinates are recovered from $\overline{\boldsymbol{X}} = (X_1, X_2, X_3)$ with the transformation $(X_1/X_3, X_2/X_3)$.

The intersection point of two lines $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$ is

$$\overline{\boldsymbol{X}} = \boldsymbol{L}_1 \times \boldsymbol{L}_2 \tag{4.2}$$

and similarly, the line joining two points $\overline{\boldsymbol{X}}_1$ and $\overline{\boldsymbol{X}}_2$ is

$$\boldsymbol{L} = \overline{\boldsymbol{X}}_1 \times \overline{\boldsymbol{X}}_2 \,. \tag{4.3}$$

These relationships are easily justified: taking the former, $\boldsymbol{L}_1 \cdot \overline{\boldsymbol{X}} = \boldsymbol{L}_1 \cdot (\boldsymbol{L}_1 \times \boldsymbol{L}_2) = \boldsymbol{L}_2 \cdot (\boldsymbol{L}_1 \times \boldsymbol{L}_1) = 0$ by the scalar triple product, so $\overline{\boldsymbol{X}}$ lies on $\boldsymbol{L}_1$. Similarly, we find that $\overline{\boldsymbol{X}}$ also lies on $\boldsymbol{L}_2$ so $\overline{\boldsymbol{X}}$ must be the point of intersection of the two lines. The latter relationship is justified in the same manner.

4.1.2 Points and Lines at Infinity

Taking two parallel lines $\boldsymbol{L}_1 = (a, b, c_1)$ and $\boldsymbol{L}_2 = (a, b, c_2)$, the point of their intersection is given by $\overline{\boldsymbol{X}} = (a, b, c_1) \times (a, b, c_2) = (c_2 - c_1)(b, -a, 0)$ which is equivalent to $(b, -a, 0)$. This indicates infinitely large inhomogeneous coordinates and highlights the notion that parallel lines meet only at infinity. Any homogeneous point $\overline{\boldsymbol{X}} = (X, Y, 0)$ is called a *point at infinity*. All such points lie on the *line at infinity* $\boldsymbol{L}_\infty = (0, 0, 1)$ as is evident with $(X, Y, 0) \cdot (0, 0, 1) = 0$.

The line at infinity $\boldsymbol{L}_\infty$ is not itself visible but when projected into images of traffic scenes where perspective effects are manifest, it has the physical interpretation as the horizon as depicted in Figure 7.6. In fact, given the line $\boldsymbol{L} = (a, b, c)$ traced out by the trajectories of the vehicles in one lane along a straight road for instance, the point at which $\boldsymbol{L}$ meets $\boldsymbol{L}_\infty$ is $(a, b, c) \times (0, 0, 1) = (b, -a, 0)$. As we have already seen, a parallel line traced out by the vehicles in another lane intersects at the same point on $\boldsymbol{L}_\infty$. A nonparallel line (a_2, b_2, c_2) however will intersect $\boldsymbol{L}_\infty$ at a different point at infinity: (a_2, b_2, c_2) is equivalent to $(a, b_2 a/a_2, c_2 a/a_2)$ which intersects $\boldsymbol{L}_\infty$ at

$(b_2a/a_2, -a, 0)$. The only difference is the first homogeneous coordinate, indicating that we can 'carve out' in homogeneous coordinates the line at infinity simply by varying the direction of the lines. The horizon, being the projection of $\boldsymbol{L}_\infty$ by a perspective transform into the image is no longer described by points at infinity and can be identified with a nonparallel set of either parallel line pairs or constant speed paths. This technique often forms the basis of affine rectification in images - the recovery of parallel lines - and the first step in estimating the full homography (see for example [10]).

4.1.3 Conics and their Duals

While the linear expressions so far are sufficient to resolve two of the eight degrees of freedom inherent in the world-to-image mapping (as will be be seen in §4.3.1), we can use a second order identity to resolve the two further degrees of freedom necessary for a metric rectification. This enables the measurement of angles between lines in the world from the image. In particular, while we can already establish vehicle trajectories in the image as parallel in the world, a metric rectification will enable us to measure orthogonality between a (box-like) vehicle's adjacent sides. Starting from the expression in inhomogeneous coordinates of a conic

$$aX^2 + bXY + cY^2 + dX + eY + f = 0 \tag{4.4}$$

and substituting for homogenous coordinates via $X = X_1/X_3$, $Y = X_2/X_3$, we can write equation (4.4) as

$$\overline{\boldsymbol{X}}^T \mathbf{C} \overline{\boldsymbol{X}} = 0 \quad \text{where} \quad \mathbf{C} = \begin{bmatrix} a & b/2 & d/2 \\ b/2 & c & e/2 \\ d/2 & e/2 & f \end{bmatrix}. \tag{4.5}$$

Any point $\overline{\boldsymbol{X}}$ satisfying equation (4.5) is said to lie on the conic $\mathbf{C}$. Given the symmetry we saw in the expressions above for the intersection of two lines and the line joining two points, it is unsurprising that a *dual conic* $\mathbf{C}^*$ exists as described by $\boldsymbol{L}^T \mathbf{C}^* \boldsymbol{L} = 0$ where $\mathbf{C}^*$ is the adjoint matrix to $\mathbf{C}$ in (4.5) and $\boldsymbol{L}$ is tangent to the conic at the point $\overline{\boldsymbol{X}}$ on the conic. It is primarily a particular singular conic $\mathbf{C}^*_\infty$ that is of interest to us (see equation (4.6)) but when the conic is not singular, $\mathbf{C}^* = \mathbf{C}^{-1}$ and the dual conic relation is established easily: first we realize that the line $\boldsymbol{L} = \mathbf{C}\overline{\boldsymbol{X}}$ is tangent to conic $\mathbf{C}$ because $\overline{\boldsymbol{X}}$ lies on $\mathbf{C}$ and $\overline{\boldsymbol{X}}$ must also lie on $\boldsymbol{L}$ as equation (4.5) states that $\overline{\boldsymbol{X}}^T \boldsymbol{L} = 0$. If it were possible for a second point $\overline{\boldsymbol{X}}_2$ on $\mathbf{C}$ to also lie on $\boldsymbol{L}$, that would imply $(\overline{\boldsymbol{X}} + \gamma\, \overline{\boldsymbol{X}}_2)^T \mathbf{C} (\overline{\boldsymbol{X}} + \gamma\, \overline{\boldsymbol{X}}_2) = 0$ for $0 \leq \gamma \leq 1$ and therefore that $\mathbf{C}$ is a degenerate conic, a contradiction. We can therefore write $\overline{\boldsymbol{X}}^T \mathbf{C} \overline{\boldsymbol{X}} = \boldsymbol{L}^T\, \mathbf{C}^{-T} \mathbf{C} \mathbf{C}^{-1} \boldsymbol{L} = \boldsymbol{L}^T \mathbf{C}^{-T} \boldsymbol{L} = \boldsymbol{L}^T \mathbf{C}^* \boldsymbol{L} = 0$.

Just as the transformation from the world to the image of the line at infinity $\boldsymbol{L}_\infty$ gives us a geometric cue to take advantage of in traffic scenes, so too does the rank 2

dual conic

$$\mathbf{C}_{\infty}^{*} = \mathbf{I}\mathbf{J}^{T} + \mathbf{J}\mathbf{I}^{T} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{4.6}$$

where $\mathbf{I} = (1, i, 0)$ and $\mathbf{J} = (1, -i, 0)$ are complex conjugate points at infinity known as the *circular points*. If the transformed locations of the circular points were somehow identified in the image, then metric rectification would be a matter of working out the transform necessary to move them back to $\mathbf{I}$ and $\mathbf{J}$. However, they are not directly visible but are found indirectly using the images of known angles, known length ratios or circles. By definition of the orthogonality of two lines $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$ and direct substitution,

$$\boldsymbol{L}_1^{T}\mathbf{C}_{\infty}^{*}\boldsymbol{L}_2 = 0 \tag{4.7}$$

but it turns out (see equation (4.13)) that this conjugacy relation holds true even under a projective transformation from the world to the image. We exploit this fact by identifying lines in the image we know are orthogonal in the world, enforcing equation (4.7) and solving for the parameters involved in the transformation that we make explicit now.

4.2 Transformation by the Homography

Having made several references to the transformation of geometric objects from the world to the image, we need to formalize this notion and introduce notation that

makes clear the distinction between world coordinates and those on the image. We reserve upper case coordinates (X, Y, Z) for the world frame (where $Z = 0$ indicates the road plane) and lower case (x, y) for the image. Further, as suggested above the coordinate vectors are homogenized with a bar: $\overline{\boldsymbol{x}}$ refers to $(x, y, 1) \sim (\gamma x, \gamma y, \gamma)$ and $\overline{\boldsymbol{X}}$ to $(X, Y, 1)$.

4.2.1 Projectivities

A *projectivity* is an invertible mapping $\phi : \mathbb{P}^2 \to \mathbb{P}^2$ such that $\phi(\boldsymbol{X}) = \boldsymbol{x}$ and colinearity is preserved. That is, if $\boldsymbol{X}_1$, $\boldsymbol{X}_2$ and $\boldsymbol{X}_3$ lie on the same line, so do $\phi(\boldsymbol{X}_1)$, $\phi(\boldsymbol{X}_2)$ and $\phi(\boldsymbol{X}_3)$. This in turn implies that ϕ can be represented by the 3×3 *homography matrix* g:

$$\overline{\boldsymbol{x}} = \overline{\phi(\boldsymbol{X})} = g\overline{\boldsymbol{X}} \tag{4.8}$$

or explicitly in elements of g,

$$\boldsymbol{x} = \phi(\boldsymbol{X}) = \begin{bmatrix} (g_{11}X + g_{12}Y + g_{13})/(g_{31}X + g_{32}Y + g_{33}) \\ (g_{21}X + g_{22}Y + g_{23})/(g_{31}X + g_{32}Y + g_{33}) \end{bmatrix} . \tag{4.9}$$

Clearly the mapping ϕ is defined only up to scale; its representation by g or γg makes no difference. There are then eight degrees of freedom associated with the projectivity that can be further categorized according to their effect on the image. The choice of scale is also up to us; specifying that $\det g = 1$ then anchors g to the group $SL(3)$. This also makes $\phi_g \equiv \phi$ a group action of $SL(3)$ on $\mathbb{R}^2$:

$$\phi_{g_2}(\phi_{g_1}(\boldsymbol{X})) = \phi_{g_2 g_1}(\boldsymbol{X}) .$$

4.2.2 Transformation Rules for Points, Lines and Conics

Now just as the homogeneous points are transformed according to $\overline{\boldsymbol{x}} = g\overline{\boldsymbol{X}}$, we have a dual transformation rule for lines

$$\boldsymbol{l} = g^{-T}\boldsymbol{L} \tag{4.10}$$

which preserves the incidence of points on lines (as it must by definition of ϕ): If the world point $\boldsymbol{X}$ lies on line $\boldsymbol{L}$ we have $0 = \boldsymbol{L} \cdot \overline{\boldsymbol{X}} = \left(g^T\boldsymbol{l}\right) \cdot (g^{-1}\overline{\boldsymbol{x}}) = \boldsymbol{l}^T g g^{-1}\overline{\boldsymbol{x}} = \boldsymbol{l}\cdot\overline{\boldsymbol{x}}$ so $\boldsymbol{x}$ also lies on $\boldsymbol{l}$. The line at infinity then is transformed to the image as

$$\begin{aligned}
\boldsymbol{l}_\infty^T &= \boldsymbol{L}_\infty^T g^{-1} \\
&= (0,0,1)g^{-1} \\
&= (\left(g^{-1}\right)_{31}, \left(g^{-1}\right)_{32}, \left(g^{-1}\right)_{33}) \\
&= (\det g)^{-1}\, \boldsymbol{M}_3^T
\end{aligned} \tag{4.11}$$

where $\boldsymbol{M}_j$ denotes the j'th column of the matrix of cofactors of g, using notation in line with that of [28]. We can establish the transformation rule for conics using the same principle. Incidence of points on conics must be preserved, so $0 = \overline{\boldsymbol{X}}^T\mathbf{C}\overline{\boldsymbol{X}} = \overline{\boldsymbol{x}}g^{-T}\mathbf{C}g^{-1}\overline{\boldsymbol{x}}$ from which we see that the conic transforms as $\mathbf{c} = g^{-T}\mathbf{C}g^{-1}$. Similarly, the dual conic transforms as

$$\mathbf{c}^* = g\mathbf{C}^*g^T. \tag{4.12}$$

We now have the tools to show the invariance of the conjugacy relation (4.7) to projective transformations:

$$\begin{aligned} \boldsymbol{l}_1^T \mathbf{c}_\infty^* \boldsymbol{l}_2 &= \boldsymbol{L}_1^T g^{-1} \left(g \mathbf{C}_\infty^* g^T \right) g^{-T} \boldsymbol{L}_2 \\ &= \boldsymbol{L}_1^T \mathbf{C}_\infty^* \boldsymbol{L}_2 \end{aligned} \tag{4.13}$$

so the transformed dual conic $\mathbf{c}^*$ is all that is needed to measure from the image the orthogonality of two lines in the world.

4.3 Homography Decomposition

4.3.1 Decomposition by Composed Transformations

For a 2D projective transform the homography matrix can be decomposed into a chain of three basic transformation types,

$$g = g_P g_A g_S = \begin{bmatrix} I & 0 \\ \omega^T & v \end{bmatrix} \begin{bmatrix} K & 0 \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} sR & t \\ 0^T & 1 \end{bmatrix} . \tag{4.14}$$

Matrix g_S provides the *similarity* transform component with rotational, translational and isotropic scaling elements R, t and s respectively. Altogether g_S contributes four degrees of freedom to the projectivity. Matrix g_A contributes a further two degrees of freedom to give an *affinity* in combination with g_S. Orthogonality is in general lost under an affinity, and scaling is not necessarily isotropic. The 2×2 matrix K is an element of the quotient group $SL(2)/SO(2)$; since any 2×2 matrix in $SL(2)$ can be decomposed into orthogonal and upper triangular factors Q and U, K must be an

upper triangular matrix (akin to U). We also specify that $\det K = 1$ so K has two degrees of freedom. The final two degrees of freedom are absorbed by the perspective parameters ω in g_P, the *projective* transform component. Also called an *elation*, it is this component alone that moves the line at infinity $\boldsymbol{L}_\infty$. The scale parameter υ is fixed by the requirement that $\det g = 1$,

$$\begin{aligned} \det g &= \det g_P \det g_A \det g_S = \upsilon s^2 \\ \implies \upsilon &= 1/s^2 \quad . \end{aligned} \tag{4.15}$$

4.3.2 Geometric Homography Decomposition

A second decomposition exists that is perhaps more intuitive from a geometrical viewpoint (see Figure 4.1). Both world and image frames assume 3D coordinate axes oriented with respect to the fixed road scene as if they both represent cameras observing the road. Of course, only the image frame is truly freely oriented with respect to the road; the world frame is actually aligned with and parallel to the road and lies a set distance away. Our objective here however is to reconcile a 3D decomposition often invoked in the literature with that of equation (4.14) by temporarily lifting those restrictions and denoting n as the normal to the road with respect to the world frame and d as its distance away. The transformation of a 3D point in the world frame to that of the image involves a rotation R, translation t and any scaling or shifting necessary to match the units the image is measured with.

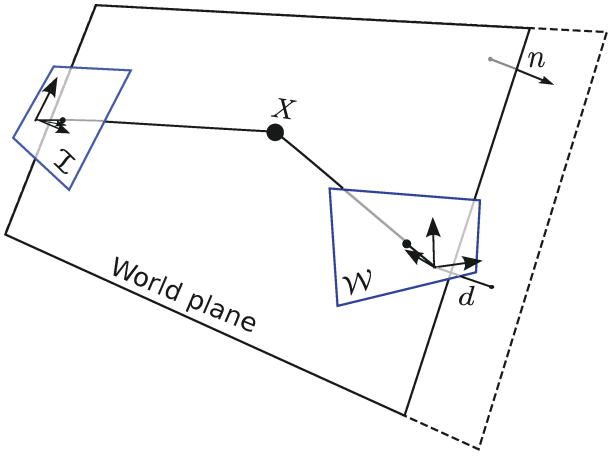


Figure 4.1: The homography maps a point X on the world plane as seen in the world frame $\mathcal{W}$ to the image frame $\mathcal{I}$.

The latter component is achieved by the 'intrinsic camera parameters' matrix usually denoted K (though different from the matrix K in equation (4.14)),

$$K = \begin{bmatrix} f_1 & \kappa & c_1 \\ 0 & f_2 & c_2 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.16}$$

where f_1, f_2 are the focal lengths in the x_1 and x_2 axis directions and act as scaling parameters, κ describes the skew between the axes and the c_i are the image origin offsets. Putting this together, a homogeneous image point is derived from the same point seen in the world frame according to

$$\bar{x} = K\left(RX + t\right) . \tag{4.17}$$

Now all the observed points X lie on the road plane, so $n^T X = d$ or $n^T X/d = 1$ for $d \neq 0$. Inserting this into (4.17),

$$\bar{x} = K\left(R + \frac{tn^T}{d}\right) X . \tag{4.18}$$

Traffic surveillance cameras have $\kappa = 0$ and because we are estimating the transformation parameters, the c_i can also be taken as zero by absorbing the difference into a compensating transformation. That is, denoting K' as the desired matrix K with $c_i = 0$,

$$K\left(R + \frac{tn^T}{d}\right) = K'R + K't\frac{n^T}{d} + cR_3 + t_3 c\frac{n^T}{d}$$

where c is the vector $(c_1, c_2, 0)^T$ and $R_3 = (R_{31}, R_{32}, R_{33})$ is the third row of R. But we can easily construct a vector b such that $K'b = c$; in fact our specification of K' yields $b = (c_1/f_1, c_2/f_2, 0)^T$. Substituting this in,

$$K\left(R + \frac{tn^T}{d}\right) = K'\left((R + bR_3) + (t + t_3 b)\frac{n^T}{d}\right) \tag{4.19}$$

so that R and t have been modified to $R + bR_3$ and $t + t_3 b$ respectively. We need to maintain orthogonality in the rotation component but as it stands $(R + bR_3)^T (R + bR_3) = (1 + b^T b)I$, so putting $R' = \left(1 + b^T b\right)^{-\frac{1}{2}} (R + bR_3)$ and $t' = \left(1 + b^T b\right)^{-\frac{1}{2}} (t + t_3 b)$ we rewrite (4.19) as

$$K\left(R + \frac{tn^T}{d}\right) = \left(1 + b^T b\right)^{\frac{1}{2}} K'\left(R' + \frac{t'n^T}{d}\right).$$

Recalling that scaling a homography matrix does not affect the inhomogeneous coordinates, the prefixed constant $\left(1 + b^T b\right)$ can effectively be ignored. The result is a decomposition of the original form but as if we had set $c_1 = c_2 = 0$, a simplification that we will take as granted in what follows.

If we do not know the (possibly modified) focal length values, we can take this a step further and similarly absorb them into a further-modified parameterization with

the resultant homography

$$K\left(R+\frac{tn^T}{d}\right)=R''+\frac{t''n^T}{d}\,. \tag{4.20}$$

but in doing so have destroyed the orthogonality of original matrix R. This unfortunately precludes the application of the relatively recent analytical decomposition technique of Malis ([28]) as well as the standard numerical decompositions ([29] and [30]) without knowledge of the focal lengths, though the homography form (4.20) reinforces the fact that we can still perform the calibration between world and image frames even if recovery of the true extrinsic parameters is not possible.

Our setup requires that the world frame and road plane be aligned, fixing $n=(0,0,1)^T$ and $X_3=d$. Then (4.18) becomes

$$\begin{aligned}K\left(R+\frac{tn^T}{d}\right)X &= K\left(RX+t\right)\\ &= K\begin{bmatrix}R_{11} & R_{12} & t_1+dR_{13}\\ R_{21} & R_{22} & t_2+dR_{23}\\ R_{31} & R_{32} & t_3+dR_{33}\end{bmatrix}\begin{bmatrix}X_1\\ X_2\\ 1\end{bmatrix}\end{aligned}$$

which, denoting the i'th column of R as R_i and recognizing the homogenization of point X, can be written $K\left[R_1\ R_2\ t+dR_3\right]\bar{X}$. For convenience we specify $d=0$, that the world frame sit on the road plane, and reuse the fact that a scaling of the transformation makes no difference to the result in inhomogeneous coordinates to get

$$\bar{x}=g\bar{X}\doteq\gamma K\left[R_1\ R_2\ t\right]\bar{X}\,, \tag{4.21}$$

normalizing by $\gamma = \det(K[R_1\ R_2\ t])^{-1}$. This is simply $\phi_g(X)$, the action on X of group element $g \in SL(3)$, which in spite of the different parameterization must match the transformation (4.14). The choice of which form to use depends on the application; it is often more convenient to use (4.14) for calibration purposes (see section 5.3 for example) and (4.21) for reconstruction (as exemplified in section 8.1).

4.4 Calculating the Transformations

The energy $E(g)$ to be minimized in (3.14) is some function of the degree to which the image and the world match. Under the true transformation ϕ_{g^*} all quantities such as lengths, angles and ratios can be measured on either the image or the world frame and yield the correct answer. When making such measurements however, we implicitly transform all quantities to the world frame; the image is where we can make the measurements but the world frame is where we know how they relate to one another. It is fundamental then that we understand how the group action (4.9) and its inverse transform coordinates and vectors between the spaces. We have already established the principles for these transforms in (2.9) and (2.10) but their calculation can be quite involved. Fortunately we can make substantial simplifications by taking (4.9) as the basis of a new set of coordinates on the image (and its inverse on the world). While this section applies to the action of any matrix Lie group, we specialize to the group $SL(3)$ for ease of notation and direct results that we will apply later.

The inverse to (4.9) maps image coordinates to the world,

$$\begin{aligned}
\phi_g^{-1}(\boldsymbol{x}) &= \phi_{g^{-1}}(\boldsymbol{x}) \\
&= \left(\left(g^{-1}\right)_{31} x + \left(g^{-1}\right)_{32} y + \left(g^{-1}\right)_{33}\right)^{-1} \begin{bmatrix} (g^{-1})_{11}\, x + (g^{-1})_{12}\, y + (g^{-1})_{13} \\ (g^{-1})_{21}\, x + (g^{-1})_{22}\, y + (g^{-1})_{23} \end{bmatrix} \\
&= \left(\boldsymbol{M}_3^T \bar{\boldsymbol{x}}\right)^{-1} \begin{bmatrix} \boldsymbol{M}_1^T \\ \boldsymbol{M}_2^T \end{bmatrix} \cdot \bar{\boldsymbol{x}}
\end{aligned}$$

which we will abbreviate to $\phi^{-1}(\boldsymbol{x}) = \left(\phi_1^{-1}(\boldsymbol{x}), \phi_2^{-1}(\boldsymbol{x})\right)^T$ and the label $\phi_3^{-1}(\boldsymbol{x}) \doteq \boldsymbol{M}_3^T \bar{\boldsymbol{x}}$. Strictly speaking, we should write $\phi^{-1}(\bar{\boldsymbol{x}})$ because the transformation (as an algebraic manipulation) applies equally well to homogeneous vectors $\bar{\boldsymbol{v}} = (v_1, v_2, 0)^T$. The inner products $\boldsymbol{M}_i^T \bar{\boldsymbol{x}}$ are calculated efficiently using Cramer's rule as the scaled determinants of the matrix g with $\bar{\boldsymbol{x}}$ inserted into the i'th column. For example, denoting

$$B_i = \begin{bmatrix} g_1 & \cdots & \underset{i'\text{th col}}{\bar{\boldsymbol{x}}} & \cdots & g_3 \end{bmatrix} \tag{4.22}$$

we have

$$\boldsymbol{M}_3^T \bar{\boldsymbol{x}} = |g|^{-1} |B_3| = |g|^{-1} \begin{vmatrix} g_{11} & g_{12} & x \\ g_{21} & g_{22} & y \\ g_{31} & g_{32} & 1 \end{vmatrix} .$$

Now an image vector is transformed according to equation (2.11) in each component

$f = \phi_i^{-1}$ to give the transformed elements

$$\begin{aligned}\left\langle d\phi_i^{-1}, \boldsymbol{v} \right\rangle &= D\phi_i^{-1}\boldsymbol{v} \\ &= \frac{1}{\boldsymbol{M}_3^T\bar{\boldsymbol{x}}}\left(D\left(\boldsymbol{M}_i^T\bar{\boldsymbol{x}}\right)\bar{\boldsymbol{v}} - \frac{\boldsymbol{M}_i^T\bar{\boldsymbol{x}}}{\boldsymbol{M}_3^T\bar{\boldsymbol{x}}}D\left(\boldsymbol{M}_3^T\bar{\boldsymbol{x}}\right)\bar{\boldsymbol{v}}\right) \\ &= \frac{\boldsymbol{M}_3^T\bar{\boldsymbol{v}}}{\boldsymbol{M}_3^T\bar{\boldsymbol{x}}}\left(\frac{\boldsymbol{M}_i^T\bar{\boldsymbol{v}}}{\boldsymbol{M}_3^T\bar{\boldsymbol{v}}} - \frac{\boldsymbol{M}_i^T\bar{\boldsymbol{x}}}{\boldsymbol{M}_3^T\bar{\boldsymbol{x}}}\right) \\ &= \frac{\phi_3^{-1}(\boldsymbol{v})}{\phi_3^{-1}(\boldsymbol{x})}\left(\phi_i^{-1}(\boldsymbol{v}) - \phi_i^{-1}(\boldsymbol{x})\right)\ , \end{aligned} \tag{4.23}$$

so that tangent vector $(\boldsymbol{x}, \boldsymbol{v})$ appears in the world frame as

$$\left(\phi^{-1}(\boldsymbol{x}), \frac{\phi_3^{-1}(\boldsymbol{v})}{\phi_3^{-1}(\boldsymbol{x})}\left(\phi^{-1}(\boldsymbol{v}) - \phi^{-1}(\boldsymbol{x})\right)\right)\ . \tag{4.24}$$

Though perhaps not obvious given the form of (4.23), it is of course a linear transformation of $\boldsymbol{v}$, the transformation matrix $D\phi^{-1}$ calculated as

$$D\phi^{-1} = \left(\boldsymbol{M}_3^T\bar{\boldsymbol{x}}\right)^{-1}\begin{bmatrix}\boldsymbol{M}_1^T \\ \\ \boldsymbol{M}_2^T\end{bmatrix}\left(\mathbf{I} - \left(\boldsymbol{M}_3^T\bar{\boldsymbol{x}}\right)^{-1}\boldsymbol{x}\boldsymbol{M}_3^T\right)\ .$$

In fact, equation (4.24) conforms to such a characteristic hallmark of transformations induced by a projectivity that we abbreviate the template

$$\frac{\phi_3^{-1}(\boldsymbol{v})}{\phi_3^{-1}(\boldsymbol{x})}\left(f(\boldsymbol{v}) - f(\boldsymbol{x})\right) \to f\left(\boldsymbol{v}, \boldsymbol{x}\right))$$

for any scalar, vector or matrix-valued function f that maps from the image frame to the world frame. Any tensor $h(\boldsymbol{V}, \boldsymbol{X})$ mapping in the opposite direction is of course expanded as

$$h\left(\boldsymbol{V}, \boldsymbol{X}\right) \to \frac{\phi_3(\boldsymbol{V})}{\phi_3(\boldsymbol{X})}\left(h(\boldsymbol{V}) - h(\boldsymbol{X})\right)\ .$$

It is precisely because vector fields transform linearly under inhomogeneous coordinates that we prefer the world coordinate system ϕ^{-1} to that of the homogeneous coordinates $\bar{\boldsymbol{X}}$, despite the fact that under the latter, points transform linearly. This is of primary importance to the approach taken in this thesis, allowing us to identify any vectors constructed from the image with tangent vectors transformed from the world frame. Without this property for instance, given an image vector that spans two points, the transformed tangent vector in the world frame may not span the same transformed points, rendering more difficult the calculation of quantities of interest. We will use transformation (4.24) extensively in the comparison with $(\boldsymbol{X}, \boldsymbol{V})$ - a predetermined world tangent vector we would expect to match under the true homography.

For simulation purposes we also need to transform world vectors into the image. This situation is symmetric to that above; the tangent vector $(\boldsymbol{X}, \boldsymbol{V})$ appears in the image as

$$(\phi(\boldsymbol{X}), \phi(\boldsymbol{V}, \boldsymbol{X})) = \left(\phi(\boldsymbol{X}), \frac{\phi_3(\boldsymbol{V})}{\phi_3(\boldsymbol{X})} \left(\phi(\boldsymbol{V}) - \phi(\boldsymbol{X})\right)\right) \tag{4.25}$$

where $\phi_3(\boldsymbol{X}) = g^3\bar{\boldsymbol{X}}$ and $\phi_i(\boldsymbol{X}) = g^i\bar{\boldsymbol{X}}/g^3\bar{\boldsymbol{X}}$, g^i denoting the i'th row of g.

To minimize a function of the transformed vector $D\phi^{-1}\boldsymbol{v}$ we also need its derivatives with respect to the parameterization of group element g. We can isolate the effect of a change in parameterization by observing

$$\frac{\partial}{\partial \alpha_j} D\phi_i^{-1}\boldsymbol{v} = \left\langle d\left(D\phi_i^{-1}\boldsymbol{v}\right), \frac{\partial g}{\partial \alpha_j} \right\rangle_g \tag{4.26}$$

where $\phi = \phi_g$ and the α_j are parameters of g. This means that we need only calculate

the derivative of $D\phi_i^{-1}\boldsymbol{v}$ with respect to the matrix entries g_{kl} once; the effect of each parameter is manifest through its derivative of g which is typically easy to compute. Recalling for any matrix B the identity $\partial |B| / \partial B_{ij} = \mathrm{Cof}(B)$, the matrix of cofactors of B, we have for the particular matrices B_k from (4.22)

$$\frac{\partial}{\partial g_{ij}} |B_k| = \begin{cases} (\mathrm{Cof}(B_k))_{ij} & j \neq k \\ \\ 0 & j = k \end{cases} . \tag{4.27}$$

This form is fundamental enough that we define the matrix-valued function

$$\Gamma_k(\boldsymbol{x}) \doteq \left(|g| \, \phi_k^{-1}(\boldsymbol{x}) \right)^{-1} \frac{\partial}{\partial g} |B_k| \tag{4.28}$$

which for $\Gamma_3(\boldsymbol{x})$ is explicitly

$$\begin{aligned} \Gamma_3(\boldsymbol{x}) &= \left(|g| \, \phi_3^{-1}(\boldsymbol{x}) \right)^{-1} \begin{bmatrix} g_2 \times \bar{\boldsymbol{x}} & \bar{\boldsymbol{x}} \times g_1 & \mathbf{0} \end{bmatrix} \\ &= \left(|g| \, \phi_3^{-1}(\boldsymbol{x}) \right)^{-1} \begin{bmatrix} -x_2 g_{32} + g_{22} & x_2 g_{31} - g_{21} & 0 \\ x_1 g_{32} - g_{12} & -x_1 g_{31} + g_{11} & 0 \\ -x_1 g_{22} + x_2 g_{12} & x_1 g_{21} - x_2 g_{11} & 0 \end{bmatrix} \end{aligned}$$

and is integral to the useful contractions

$$(\Gamma_3(\boldsymbol{x}))_i^T \, g_3 = -\phi_i^{-1}(\boldsymbol{x}) \tag{4.29}$$

and

$$g^3 \Gamma_3(\boldsymbol{x})^T g_3 = g_{33} - 1/\phi_3^{-1}(\boldsymbol{x}) \, , \tag{4.30}$$

that are easily verified through direct substitution. The matrices $\Gamma_i(\boldsymbol{v})$ follow similarly by replacing the homogeneous argument. Now starting with (4.24), using (4.27)

repeatedly and rearranging, we get matrix $\Theta_i(\boldsymbol{v}, \boldsymbol{x})$ for the components of $d\left(D\phi_i^{-1}\boldsymbol{v}\right)$ in the basis g_{jk}, that is,

$$\Theta_i(\boldsymbol{v}, \boldsymbol{x}) = \frac{\phi_3^{-1}(\boldsymbol{v})}{\phi_3^{-1}(\boldsymbol{x})} \left(\Theta_i(\boldsymbol{v}) - \Theta_i(\boldsymbol{x})\right) \tag{4.31}$$

$$\text{where } \Theta_i(\boldsymbol{a}) = \Gamma_i(\boldsymbol{a}) - \phi_i^{-1}(\boldsymbol{x})\Gamma_3(\boldsymbol{a}) - \phi_i^{-1}(\boldsymbol{a})\Gamma_3(\boldsymbol{x})\,, \tag{4.32}$$

reminiscent of the form of (4.23). Precomputing this matrix for each point of interest leads to an efficient procedure via equation (4.26) for energy minimization, as exemplified in section 5.3.

Chapter 5

Homography Estimation by Direct Linear Transform

The classical approach to estimating the homography is to coerce any information we have about objects in the image into revealing the coefficients of the transformed line at infinity $\boldsymbol{l}_\infty$ or dual conic $\mathbf{c}^*_\infty$. Any planar curve of order n yields a system of $n(n+3)/2$ equations ([31]), the linear and conic cases given by (4.10) and (4.12) respectively. We use their specializations to $\boldsymbol{l}_\infty$ and $\mathbf{c}^*_\infty$ in expressions (4.11) and (4.13) to respectively perform affine rectification followed by an inference task on the circular points to attain metric rectification (the *stratified* method), or direct metric rectification (the *unstratified* method). Both methods are described in [32], while the latter will be employed below. Either method establishes the homography only up to a similarity transform which is not sufficient for our purposes; we require the full homography if we are to match vehicle templates with the vehicle images for instance. To resolve the remaining degrees of freedom we provide additional constraints. This section sets up the estimation problem in a classical manner and as a test of its

efficacy, applies it to an artificial scene. We also assume access to the image vector field $\boldsymbol{v}$ describing the flow of the traffic and the vector field $\boldsymbol{w}$ specifying the direction of minimum gradient variation over the vehicles at each point. The latter field tends to point along the dominant vehicle edges as seen from the camera, perpendicular to the direction of movement so that in the world frame we have at each point $\boldsymbol{V} \cdot \boldsymbol{W} = 0$. We will see in section 7.1 how these quantities are captured from the image, but for now we take them to be noise-free.

5.1 Metric Rectification from Orthogonal Lines

To find the line $\boldsymbol{l}_v$ between points $\boldsymbol{x}$ and $\boldsymbol{x} + \boldsymbol{v}$ we observe the convention for vectors $\overline{\boldsymbol{v}} = (v_1, v_2, 0)^T$ and use equation (4.3) to get

$$\begin{aligned} \boldsymbol{l}_v &= \overline{\boldsymbol{x}} \times (\overline{\boldsymbol{x}} + \overline{\boldsymbol{v}}) \\ &= \overline{\boldsymbol{x}} \times \overline{\boldsymbol{v}} \\ &= (-v_2, v_1, x_1 v_2 - v_1 x_2)^T \end{aligned} \tag{5.1}$$

and similarly for line $\boldsymbol{l}_w$. Applying the knowledge that these are the transformed images of orthogonal lines in the world, we exploit equation (4.13) with six unknown conic parameters (analogous to equation (4.5)) to write

$$\boldsymbol{l}_v^T \mathbf{c}_\infty^* \boldsymbol{l}_w = \begin{bmatrix} -v_2 & v_1 & x_1 v_2 - v_1 x_2 \end{bmatrix} \begin{bmatrix} a & b/2 & d/2 \\ b/2 & c & e/2 \\ d/2 & e/2 & f \end{bmatrix} \begin{bmatrix} -w_2 \\ w_1 \\ x_1 w_2 - w_1 x_2 \end{bmatrix} = 0 \ . \tag{5.2}$$

This is rearranged into the inner product

$$\left(v_2w_2, -z, v_1w_1, x_2z - x_1v_2w_2, x_1z - x_2v_1w_1, x_1^2v_2w_2 - 2x_1x_2z + x_2^2v_1w_1\right) \cdot \boldsymbol{c} = 0 \tag{5.3}$$

where $\boldsymbol{c} = (a, b, c, d, e, f)^T$ and $z = (v_1w_2 + v_2w_1)/2$, which forms a set of linear equations in the unknowns once we stack equation (5.3) sufficiently many times over different points in the image into matrix $\mathbf{A}$. Though there are six unknowns, the dual conic $\mathbf{c}_\infty^*$ is only specified up to scale, implying the need for $\boldsymbol{x}$, $\boldsymbol{v}$ and $\boldsymbol{w}$ to be known at no fewer than five distinct points. The solution of overdetermined system $\mathbf{A}\boldsymbol{c} = \mathbf{0}$ with minimal Frobenius norm is given by the eigenvector of $\mathbf{A}^T\mathbf{A}$ corresponding to the smallest eigenvalue. We thus perform the SVD $\mathbf{A} = \mathbf{W}\mathbf{\Sigma}\mathbf{V}^T$ and take as the solution $\widehat{\boldsymbol{c}}$ the appropriate column of $\mathbf{V}$. Now we take advantage of the form of the dual conic transformation rule (4.12) in realizing that it reflects the spectral decomposition of a positive semidefinite matrix $\mathbf{B} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$. Matrix $\mathbf{c}_\infty^*$ as constructed from the elements of $\widehat{\boldsymbol{c}}$ has rank two and is symmetric and therefore certainly positive semidefinite, while $\mathbf{C}_\infty^*$ is diagonal. We assert that the homography, up to a similarity, is reconstructed by

$$g_i = \sqrt{\mathbf{\Lambda}_{ii}}\mathbf{U}_i \tag{5.4}$$

where g_i and $\mathbf{U}_i$ are the i'th columns of their respective matrices. To check this,

$$\begin{aligned}\mathbf{c}_\infty^* &= g\mathbf{C}_\infty^* g^T \\ &= \sum_{i=1}^{3} g_i \left(\mathbf{C}_\infty^*\right)_{ii} g_i^T \\ &= \sum_{i=1}^{3} \mathbf{\Lambda}_{ii}\mathbf{U}_i\mathbf{U}_i^T \\ &= \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T \quad , \end{aligned} \tag{5.5}$$

exactly the spectral decomposition we made to start with. The third line follows because $\mathbf{\Lambda}_{33}$ should be zero, $\mathbf{c}_\infty^*$ having rank two. In practice a non-zero value does not matter as we do not take $g_3 = \mathbf{0}$ as suggested by equation (5.4). Rather, the additional constraints we impose in section 5.3 to arrive at the full homography (not just up to similarity) dictate the entries g_3.

5.2 The Rectangle Ambiguity

The orthogonal lines we intend to exploit are those tracing out the trajectories of the vehicles and their transverse counterparts, derived from the vehicles themselves. When the road is straight however, these lines do not provide sufficient information to render matrix $\mathbf{A}$ of rank five. Known as the *rectangle ambiguity*, this stems from the fact that knowing three of the angles in a rectangle to be right angles implies the fourth must be too; supplying the fourth angle yields no additional information. This is easily understood by constructing two degenerate dual conics: With two parallel

lines in direction $\boldsymbol{v}$ labelled $\boldsymbol{l}_v^1$ and $\boldsymbol{l}_v^2$, their intersection on the line at infinity is $\overline{\boldsymbol{x}}_v = \boldsymbol{l}_v^1 \times \boldsymbol{l}_v^2$. Doing the same in direction $\boldsymbol{w}$ gives point $\overline{\boldsymbol{x}}_w$. The rank-one conic $\mathbf{c}_v^* = \overline{\boldsymbol{x}}_v \overline{\boldsymbol{x}}_v^T$ satisfies equation (5.2) for any combination ($i \in \{1,2\}, j \in \{1,2\}$) of $\boldsymbol{l}_v^{iT} \mathbf{c}_v^* \boldsymbol{l}_w^j$ because $\boldsymbol{l}_v^{1T} \overline{\boldsymbol{x}}_v = \boldsymbol{l}_v^{2T} \overline{\boldsymbol{x}}_v = 0$, that is, $\overline{\boldsymbol{x}}_v$ sits on both lines $\boldsymbol{l}_v^1$ and $\boldsymbol{l}_v^2$. This is also true however for the rank-one conic $\mathbf{c}_w^* = \overline{\boldsymbol{x}}_w \overline{\boldsymbol{x}}_w^T$, and therefore any linear combination of $\mathbf{c}_v^*$ and $\mathbf{c}_w^*$ satisfies (5.2). The fact that we have a one-parameter family of suitable conics means that matrix $\mathbf{A}$ is only of rank four, manifesting itself ultimately as an unknown aspect ratio built into the constructed homography (5.4) so that a full metric rectification has not actually been achieved.

We do not have the luxury of the usual resolution to this ambiguity: the incorporation of additional orthogonal line pairs in a different orientation. However, we can find the true aspect ratio by enforcing the condition that for two chosen vectors $\boldsymbol{v}_0$ and $\boldsymbol{w}_0$, their world-transformed, orthogonal counterparts have known lengths $\|\boldsymbol{V}_0\|$ and $\|\boldsymbol{W}_0\|$. We are required then to adopt an additional degree of freedom: the parameter r that specifies matrix K of equation (4.14) as[1]

$$K = \begin{bmatrix} r & 0 \\ 0 & 1/r \end{bmatrix},$$

an integral part of the calibration in the next section.

[1] In fact, K is of the more general form $\begin{bmatrix} r & \kappa \\ 0 & 1/r \end{bmatrix}$ for some skewness parameter κ. For most cameras the image coordinate axes are orthogonal and $\kappa = 0$, but if it were suspected that some skewness in the axes were present, we would need to take κ as a further degree of freedom to be calibrated alongside r.

5.3 Full Homography Estimation Procedure

Following the procedure in section 5.1 we recover the pseudo-metric properties of the world frame and can determine parallelism between objects in the world. The additional prescription of section 5.2 enables a full metric rectification from determination of the aspect ratio, but the rotation, translation and isotropic scaling of the scene are however undetermined. Fortunately the two translational degrees of freedom fall prey to a simple constraint. We choose a point $\boldsymbol{x}_0$ to serve as the image of the world origin $\boldsymbol{X}_0 = (0,0)^T$. Then

$$\begin{aligned} \overline{\boldsymbol{x}}_0 &= g\overline{\boldsymbol{X}}_0 \\ &= g_3 \end{aligned}$$

with the immediate consequence that

$$\begin{aligned} g_{31}/g_{33} &= x_0 \\ g_{32}/g_{33} &= y_0. \end{aligned} \tag{5.6}$$

So for any given value of g_{33}, elements g_{31} and g_{32} can be calculated. We incorporate this constraint directly into the decomposition of g as follows: From equations (4.14) and (4.15) we have

$$g = \begin{bmatrix} sKR & Kt \\ s\omega^T KR & \omega^T Kt + 1/s^2 \end{bmatrix} \tag{5.7}$$

but incorporating equations (5.6) we see

$$\left(\omega^T Kt + 1/s^2\right)\boldsymbol{x}_0 = Kt$$

which can be rearranged to give

$$\begin{aligned}\left(\boldsymbol{x}_0\omega^T - I\right)Kt &= -\left(1/s^2\right)\boldsymbol{x}_0 \\ Kt &= -\left(1/s^2\right)\left(\boldsymbol{x}_0\omega^T - I\right)^{-1}\boldsymbol{x}_0 \quad . \end{aligned} \tag{5.8}$$

Then

$$\omega^T Kt + 1/s^2 = \left(1/s^2\right)\left(1 - \omega^T\left(\boldsymbol{x}_0\omega^T - I\right)^{-1}\boldsymbol{x}_0\right) \tag{5.9}$$

and substituting this into equation (5.7) we arrrive at a matrix with t eliminated,

$$g = \begin{bmatrix} sKR & -\left(1/s^2\right)\left(\boldsymbol{x}_0\omega^T - I\right)^{-1}\boldsymbol{x}_0 \\ s\omega^T KR & \left(1/s^2\right)\left(1 - \omega^T\left(\boldsymbol{x}_0\omega^T - I\right)^{-1}\boldsymbol{x}_0\right) \end{bmatrix} = \begin{bmatrix} K' & t' \\ \omega'^T & v' \end{bmatrix}. \tag{5.10}$$

We proceed to resolve the final three degrees of freedom by minimizing an appropriate energy E. As mentioned in section 5.2 we choose vectors $\boldsymbol{v}_0$ and $\boldsymbol{w}_0$ at any point in the image that can be described in the world as $\boldsymbol{V}_0$ and $\boldsymbol{W}_0$; examples include an average vehicle velocity vector that points in a direction we take to be along the world X-axis, or the average vehicle transverse vector assumed to lie along the world Y-axis. Both such vectors have world-frame components that can be estimated a priori based on known average traffic speeds or average vehicle width. Moreover, for an objective such as vehicle classification where the vehicle images as transformed in the world frame are compared for size or shape, the moduli $\|\boldsymbol{V}_0\|$ and $\|\boldsymbol{W}_0\|$ can be

set arbitrarily. The energy is chosen as a robust penalization Ψ (see equation (5.21)) of the mismatching between $(\boldsymbol{v}_0, \boldsymbol{w}_0)$ transformed to the world and $(\boldsymbol{V}_0, \boldsymbol{W}_0)$,

$$E = \Psi\left(D\phi^{-1}\boldsymbol{v}_0 - \boldsymbol{V}_0\right) + \Psi\left(D\phi^{-1}\boldsymbol{w}_0 - \boldsymbol{W}_0\right) \tag{5.11}$$

which we calculate using equation (4.23). Matrix KR is an element of Lie group $SL(2)$ and can be decomposed into its constituents via the RQ decomposition, from which both the ratio r and the angle of rotation θ can be determined and updated. A coupled descent must also be made with respect to scale parameter s which is found from the current estimate of g using $s = \sqrt{\det K'}$. Crucially, the initial estimate of g must have been arrived at after (at least) affine rectification so that the perspective parameters ω can be uncovered, but the rectification of section 5.1 ensures this. The equations (5.4) guarantee that

$$\omega = K'^{-1}\omega' \tag{5.12}$$

gives the correct parameter values from the initialization. The other elements t' and v' are then completely determined, and all constraints will be met at each iterative update of matrix g. Each iteration requires the gradients $\partial E/\partial\alpha_j$ for $\alpha_j \in \{r, \theta, s\}$ in the update procedure (3.18); equation (4.26) implies that we need only calculate $d\left(D\phi_i^{-1}\boldsymbol{v}\right)$ once each iteration by applying (4.31) and taking the inner product with $\partial g/\partial\alpha_j$. These matrices are easy to compute and yield the expressions

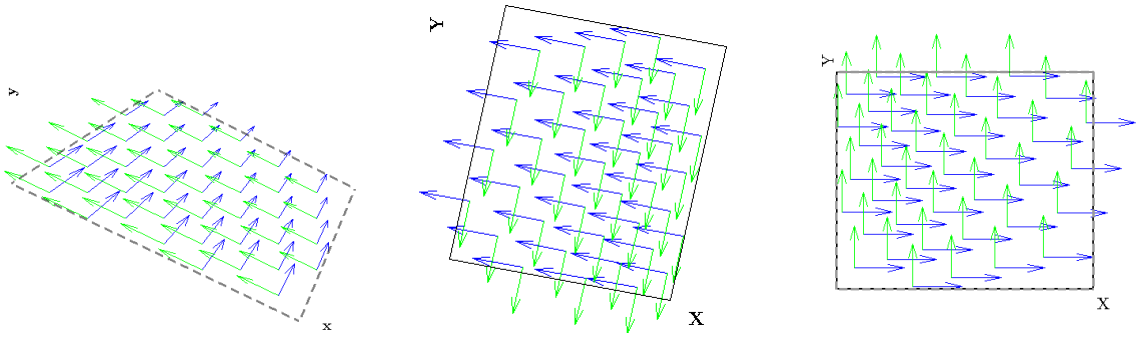
$$\frac{\partial g}{\partial s} = \frac{1}{s} g \Pi_s \quad \text{where} \quad \Pi_s = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -2 \end{bmatrix}, \tag{5.13}$$

$$\frac{\partial g}{\partial \theta} = g \Pi_\theta \qquad \text{where} \quad \Pi_\theta = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

$$\frac{\partial g}{\partial r} = \begin{bmatrix} \Pi_r & \mathbf{0} \\ \omega^T \Pi_r & 0 \end{bmatrix} \quad \text{where} \quad \Pi_r = s \begin{bmatrix} 1 & 0 \\ 0 & r^{-2} \end{bmatrix} R$$

5.4 A Toy Example

To test the utility of the classical estimation procedure in a noise-free setting, we construct an artificial scene by distorting a planar rectangle (the world frame) with a known homography. We also transform the vectors $\boldsymbol{V}$ that describe an imagined traffic flow and the corresponding transverse vectors $\boldsymbol{W}$ into the image using equation (4.25). From there we use equations (5.1) and (5.3) to set up the metric rectification problem using equally spaced grid points on the image. The vectors $\boldsymbol{v}$ and $\boldsymbol{w}$ at the grid points are interpolated from the transformed vectors. The initial value for g is arrived at by imposing (5.8) and (5.9) on (5.4) once an image origin has been chosen. Assuming we know the components of vector $\boldsymbol{V}_0$, the transformed image vector at the image origin, we update our estimates for parameters s, θ and



(a) Artificial scene produced with a known homography.

(b) The pseudo-rectified image after the calibration by the DLT.

(c) Fully rectified image at convergence of the gradient descent. The true homography would produce the dashed bounds; the alignment is very close.

Figure 5.1: Calibration by the DLT followed by a similarity transformation via energy minimization.

r via the derivatives of energy (5.11) in a standard gradient descent procedure. The calculation of these employs (4.26) and (5.13). Figure 5.1 illustrates the homography estimation process. From an initial scene (with the blue arrows reflecting an imagined traffic flow), the rectification by Direct Linear Transform delivers the center image. This is then rotated and anisotropically scaled by gradient descent which converges to machine precision after about 10^4 iterations as witnessed by the close alignment between true (dashed) and achieved (solid) bounding boxes. The energy minimization profile, largely an exponential function of the number of iterations, is shown in Figure 5.2. Some care must be taken to keep the relative step sizes larger for θ than s, at least initially, or risk an undesirable descent to the local minimum at $s = 0$. We observe a steady approach to the true minimum g^*, resulting in a mean square error in the elements $\hat{g} - g^*$ on the order of machine precision.

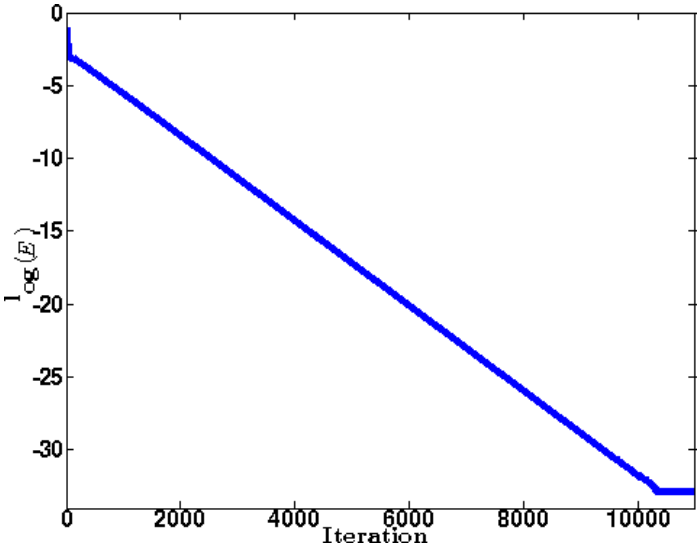


Figure 5.2: The energy on a logarithmic scale as a function of the gradient descent iterations.

It is evident that the convergence is somewhat slow; we would like to achieve an acceptable error tolerance within dozens or hundreds of iterations rather than thousands and preferably without the heightened risk of being trapped in local minima. These concerns are dwarfed however by the potential difficulty that the addition of noise into the scene poses. The measurements possible on actual traffic scenes exhibit a significant amount of noise which any estimation algorithm must be robust to. An analysis of this effect on the DLT is the subject of section 5.5.

5.5 Error Analysis of the DLT

It is imperative that we understand how our estimate of the homography is affected by noise in the image. It is well-known that such use of the SVD can amplify error

considerably; an estimate that exhibits fine sensitivity to noise levels compromises the validity of the vehicle template matching approach. We proceed by tracking the first-order error propagation from the variances associated with the z_i - the components of $\boldsymbol{x}$, $\boldsymbol{v}$ and $\boldsymbol{w}$ at each grid point in the image. Expressing the six-dimensional covariance matrix for these elements as Λ_x, the next step is to calculate Λ_c, the covariance of the solution vector $\hat{\boldsymbol{c}}$ in (5.3). Writing the Jacobian $\partial \hat{c}_i / \partial z_j$ as J_c, we have

$$\Lambda_c = J_c \Lambda_x J_c^T \,. \tag{5.14}$$

The difficulty is that $\hat{\boldsymbol{c}}$ is only an implicit function of the coordinates as outlined below equation (5.4); the use of an intermediate SVD prevents derivatives from being taken directly. However, Faugeras ([33, Theorem 5.28]) invokes the implicit function theorem to sidestep this issue in the following manner: Declaring the dependencies in $\mathbf{A}^T\mathbf{A}\hat{\boldsymbol{c}} = \mathbf{A}^T\mathbf{A}\boldsymbol{c}\big|_{c=\hat{c}} = 0$ we have

$$\begin{aligned}
\frac{\partial}{\partial z_j}\left(\mathbf{A}(\mathbf{z})^T\mathbf{A}(\mathbf{z})\boldsymbol{c}\right)_i &= \sum_k \frac{\partial}{\partial z_j}\left(\mathbf{A}(\mathbf{z})^T\mathbf{A}(\mathbf{z})\right)_{ik} \boldsymbol{c}_k + \sum_k \frac{\partial}{\partial c_k}\left(\mathbf{A}(\mathbf{z})^T\mathbf{A}(\mathbf{z})\boldsymbol{c}\right)_i \frac{dc_k}{dz_j} \\
&= \frac{\partial}{\partial z_j}\left(\mathbf{A}_i(\mathbf{z})^T\mathbf{A}(\mathbf{z})\right)\boldsymbol{c} + \left(\mathbf{A}_i(\mathbf{z})^T\right)\mathbf{A}(\mathbf{z})\frac{d\boldsymbol{c}}{dz_j} \\
&= 0 \text{ when } \boldsymbol{c} = \hat{\boldsymbol{c}}\,,
\end{aligned}$$

where $\mathbf{A}_i$ denotes the i'th column of matrix $\mathbf{A}$. This implies that

$$\frac{d\hat{\boldsymbol{c}}}{dz_j} = -\left(\mathbf{A}^T\mathbf{A}\right)^\dagger \frac{\partial}{\partial z_j}\left(\mathbf{A}^T\mathbf{A}\right)\hat{\boldsymbol{c}} \tag{5.15}$$

where the pseudo-inverse is taken due to $\mathbf{A}^T\mathbf{A}$ being singular. Now extending to all derivatives $D = d/d\mathbf{z}$,

$$\frac{\partial}{\partial \mathbf{z}}\left(\left(\mathbf{A}^T\mathbf{A}\right)\hat{\boldsymbol{c}}\right)_i = \left(\sum_k \hat{c}_k \mathbf{A}_k^T\right) D\mathbf{A}_i + \mathbf{A}_i^T \sum_k \hat{c}_k D\mathbf{A}_k$$
$$= \left(\mathbf{A}\hat{\boldsymbol{c}}\right)^T D\mathbf{A}_i + \mathbf{A}_i^T \sum_k \hat{c}_k D\mathbf{A}_k$$

so that to calculate (5.15) we need the matrices $D\mathbf{A}_i$. Matrix $D\mathbf{A}_1$ for instance has for the i'th row the form $\begin{bmatrix} 0 & 0 & 0 & w_2 & 0 & v_2 \end{bmatrix}$ where the vector components are those valid at the i'th point. Covariance (5.14) is now attainable but there was one more step in the procedure before reaching an affinely-calibrated homography g: the spectral decomposition (5.5). Here again we face an implicit function for which the derivatives are needed. Fortunately we can take advantage of a known analytical solution ([34]) for the eigenvalues of a 3×3 matrix and re-use the implicit approach for reconstructing the homography from the eigenvectors. The derivatives of the i'th column of g with respect to the components of $\hat{\boldsymbol{c}}$ are from (5.4),

$$J_g \doteq \frac{d}{d\hat{\boldsymbol{c}}}\left(\sqrt{\Lambda_{ii}}\mathbf{U}_i\right) = \frac{1}{2\sqrt{\Lambda_{ii}}}\mathbf{U}_i D\Lambda_{ii} + \sqrt{\Lambda_{ii}} D\mathbf{U}_i \text{ where } D = \frac{d}{d\hat{\boldsymbol{c}}}\,. \tag{5.16}$$

Putting $q \doteq \text{trace}(\mathbf{c}_\infty^*)/3$ and $p \doteq \sqrt{\text{trace}(\mathbf{c}_\infty^* - qI)^2/6}$, the eigenvalue matrix Λ is, by an affine transformation,

$$\Lambda = p\Lambda' + qI$$

where $\Lambda^{'}$ is the eigenvalue matrix for the decomposition of $\mathbf{B} \doteq p^{-1}\left(\mathbf{c}_{\infty}^{*} - qI\right)$. It turns out that the values $\Lambda_{ii}^{'}$ are given by

$$\Lambda_{ii}^{'} = 2\cos\left(\frac{1}{3}\arccos\left(\frac{\det \mathbf{B}}{2}\right) + \frac{2(i-1)\pi}{3}\right)$$

which in turn enables the computation of

$$\begin{aligned} D\Lambda_{ii} &= \frac{\partial \Lambda_{ii}}{\partial \Lambda_{ii}^{'}} \frac{d\Lambda_{ii}^{'}}{d\left(\det \mathbf{B}\right)} D\det \mathbf{B} + \frac{\partial \Lambda_{ii}}{\partial p} Dp + \frac{\partial \Lambda_{ii}}{\partial q} Dq \\ &= \frac{p}{3\sqrt{1 - \left(\frac{\det \mathbf{B}}{2}\right)^2}} \sin\left(\frac{1}{3}\arccos\left(\frac{\det \mathbf{B}}{2}\right) + \frac{2(i-1)\pi}{3}\right) D\det \mathbf{B} + \frac{1}{72p} Dp + Dq\,. \end{aligned}$$

We resort to Mathematica for the rather unwieldy derivatives $D\det \mathbf{B}$. If we now consider the eigenvectors $\mathbf{U}_i$ as the solutions to

$$\left(\mathbf{c}_{\infty}^{*} - \Lambda_{ii} I\right)\mathbf{U}_i = \mathbf{0} = \left(\mathbf{c}_{\infty}^{*} - \Lambda_{ii} I\right)^T \left(\mathbf{c}_{\infty}^{*} - \Lambda_{ii} I\right)\mathbf{U}_i\,,$$

then by the same reasoning leading to (5.15) we have

$$\frac{d\mathbf{U}_i}{d\hat{c}_j} = -\left(\left(\mathbf{c}_{\infty}^{*} - \Lambda_{ii} I\right)^T \left(\mathbf{c}_{\infty}^{*} - \Lambda_{ii} I\right)\right)^{\dagger} \left(\frac{d}{d\hat{c}_j}\left(\mathbf{c}_{\infty}^{*} - \Lambda_{ii} I\right)^T \left(\mathbf{c}_{\infty}^{*} - \Lambda_{ii} I\right)\right)\mathbf{U}_i\,. \quad (5.17)$$

Noting that all the matrices in (5.17) are symmetric and putting

$$\mathbf{B}^{'} \doteq \left(\frac{d}{d\hat{c}_j}\mathbf{c}_{\infty}^{*} - \Lambda_{ii} I + \mathbf{c}_{\infty}^{*} - \frac{d\Lambda_{ii}}{d\hat{c}_j} I\right)\left(\mathbf{c}_{\infty}^{*} - \Lambda_{ii} I\right)\,,$$

we process (5.17) as

$$\frac{d\mathbf{U}_i}{d\hat{c}_j} = -\left(\left(\mathbf{c}_{\infty}^{*} - \Lambda_{ii} I\right)^T \left(\mathbf{c}_{\infty}^{*} - \Lambda_{ii} I\right)\right)^{\dagger} \left(\mathbf{B}^{'} + \mathbf{B}^{'T}\right)\mathbf{U}_i\,.$$

All of the elements of (5.16) can now be computed to arrive at the homography covariance

$$\Lambda_g = J_g \Lambda_c J_g^T = J_g J_c \Lambda_x \left(J_g J_c\right)^T$$

which, recalling that the elements of g_3 are set subsequently given constraint (5.6), is valid for the elements $(\Lambda_g)_{ij},\ j \in (1,2)$.

Making the argument that our image grid points are known exactly but that each component of the vectors $\boldsymbol{v}$ and $\boldsymbol{w}$ have variance σ^2, the matrix of variances for $g_{ij},\ j \in (1,2)$ is for the toy example above rounded to

$$\sigma^2 \begin{bmatrix} 124 & 162 \\ 31 & 71 \\ 22 & 300 \end{bmatrix}, \tag{5.18}$$

a result that appears to be stable with the number of grid points used. For a variance in normalized coordinates of 0.0001 which typically understates the noise level in our images, we would expect to see an average variance over the homography elements of perhaps 0.012. To illustrate the significance of this error amplification, Figure 5.3 displays a typical instance of the toy scene image distorted in the same elements $g_{ij},\ j \in (1,2)$ with an average variance of 0.01 (while still constrained to be a valid homography). The difference is clearly large enough to render ineffective any template-matching algorithm that relies on an accurate representation of vehicle pose. This is the reason most applications of the DLT are on images with very clearly delineated features that can be identified with high precision. Evidently we are required to go beyond the DLT if we are to have any success with an automated homography estimation procedure on noisy images.

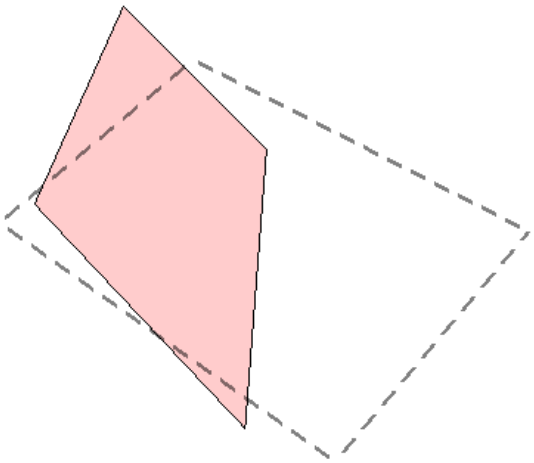


Figure 5.3: Example of planar scene disortion under an average variance per homography element of 0.01. The dashed plane is the original, the red plane its distortion.

5.6 Refining the Homography Estimation

The problem that noisy images pose to the classical homography estimation by DLT is well understood. In fact there are both pre- and post-processing steps that are generally accepted as compulsory. The DLT is not invariant to similarity transformations of the coordinates used, so data normalization is used to avoid spuriously large values of the Frobenius norm $\|\mathbf{A}\boldsymbol{c}\|$ of section 5.1 for example. The coordinates are scaled and centered so their centroid lies at the origin and the mean distance $n^{-1}\sum_i^n \|\boldsymbol{x}_i\|$ is $\sqrt{2}$. This step was performed on the data leading to the result (5.18), so there is nevertheless a strong amplification of noise inherent in the SVD procedure itself.

A nonlinear, iterative optimization is usually the tool of choice to further refine the homography estimate after initialization by the DLT. If we knew sufficiently many

world points $\boldsymbol{X}_i$ and their counterparts $\boldsymbol{x}_i$ on the image, we could minimize the total error $\sum_i^n d.\left(\overline{\boldsymbol{x}}_i, \overline{\boldsymbol{X}}_i\right)$ over the parameterization of g for one of the following error measures: the *algebraic distance*

$$d_{alg}\left(\overline{\boldsymbol{x}}_i, \overline{\boldsymbol{X}}_i\right) = \left(d_1^i\right)^2 + \left(d_2^i\right)^2 \text{ where } \boldsymbol{d}_i = \overline{\boldsymbol{x}}_i \times \hat{g}\overline{\boldsymbol{X}}_i \,,$$

or, denoting the inhomogeneous transform of $\hat{g}\overline{\boldsymbol{X}}_i$ as $\hat{\boldsymbol{x}}_i$, the *geometric distance*

$$d_{geo}\left(\overline{\boldsymbol{x}}_i, \overline{\boldsymbol{X}}_i\right) = \left\|\boldsymbol{x}_i - \hat{\boldsymbol{x}}_i\right\|^2 \,, \tag{5.19}$$

the *symmetric transfer error* that adds (5.19) together with its counterpart on the world $\left\|\boldsymbol{X}_i - \hat{\boldsymbol{X}}_i\right\|^2$, or the *reprojection error*

$$d_{rep}\left(\overline{\boldsymbol{x}}_i, \overline{\boldsymbol{X}}_i\right) = \min_{\boldsymbol{y}_i, \boldsymbol{Y}_i} \left\|\boldsymbol{x}_i - \boldsymbol{y}_i\right\|^2 + \left\|\boldsymbol{X}_i - \boldsymbol{Y}_i\right\|^2 \text{ where } \overline{\boldsymbol{y}}_i = \hat{g}\overline{\boldsymbol{Y}}_i \,.$$

Typically such a minimization would be executed with an algorithm such as Levenberg-Marquardt.

While we will also minimize a metric that is defined on one space in terms of quantities that have been transformed by the homography from the other, we have only a few quantities that we know anything about in both spaces, namely $\boldsymbol{x}_0$, $\boldsymbol{v}_0$ and $\boldsymbol{w}_0$ along with $\boldsymbol{X}_0$, $\boldsymbol{V}_0$ and $\boldsymbol{W}_0$. These do not provide sufficient information to enable a classical minimization, so we will pursue a metric of the form

$$d\left(\boldsymbol{V}, \boldsymbol{W}\right) = \Psi\left(\operatorname{div} \boldsymbol{V}\right) + \Psi\left(\boldsymbol{V} \cdot \boldsymbol{W}\right) \,, \tag{5.20}$$

that penalizes a non-zero divergence in the traffic flow as well as a lack of orthogonality between the trajectories and their transverse directions. Function Ψ is a 'robust

penalization' function, which when implemented in this thesis will take the form of an L1-norm,

$$\Psi(t) = \sqrt{t^2 + \epsilon}\Big|_{\epsilon \to 0} \tag{5.21}$$

so that

$$\Psi^{'}(t) = \text{sign}(t) \quad \text{and} \quad \Psi^{''}(t) = 0 \quad .$$

In so doing, we hope to obviate the need of the DLT as an initialization device by accounting for all eight degrees of freedom via both contraints and the minimization itself. We expect the minimization of such a metric to be somewhat more prone to local minima in the noiseless case than was seen in section 5.4 simply due to the greater number of degrees of freedom and the lack of a guarantee of convexity in the parameterization. On the other hand we expect it to exhibit far more robustness to noise, similar to the end result after a classical minimization but adapted to the situation where the coordinates in the world frame remain unknown. Of course, the expression (5.20) must be formulated with quantities we actually observe in the image; this is the grist of the next section.

Chapter 6

Homography Estimation by Energy Minimization

Once the affine rectification by Direct Linear Transform was achieved in chapter 5, we completed the calibration by minimizing an energy functional over the direct product group $SO(1) \times \mathbb{R} \times \mathbb{R}_+$. We could alternatively have realized the matrix K' in (5.10) by minimizing a similar energy over the Lie group $SL(2)$. This would have added the affine components to the transformation found through the minimization procedure. Having seen that the the DLT is highly sensitive to noise, we can take the minimization idea further and posit that estimating g itself over the group $SL(3)$,

$$g^* = \arg\min_g E(g) \,,$$

would afford an increased robustness. This again entails modifying the energy functional so that it encompasses addtional degrees of freedom. Equipped with such an energy $E : SL(3) \to \mathbb{R}$ that we first derive in section 6.1, we keep to gradient descent as a method to accomplish the minimization in section 6.3.

6.1 The Energy Functional

We posit that an appropriate energy functional to consider is

$$E = \int_{\mathcal{W}} \Psi\left(\operatorname{div} \boldsymbol{V}\right) \mathrm{d}\mathbf{X} \tag{6.1}$$

for penalty function Ψ. Vector field $\boldsymbol{V}$ is the pushforward by ϕ^{-1} of the vector field $\boldsymbol{v}$ induced by the traffic movement as observed in the image. The intuition behind (6.1) stems from the similarity between traffic flow and that of an incompressible fluid: While we expect to see a range of average velocities over any cross-section of the road, we also expect zero divergence over any surface area. This would not be the case if in the scene of interest the vehicles are accelerating for instance; fortunately on high capacity roads the times spent in a steady-state, constant velocity mode are expected to heavily outweigh the finite-divergence cases, and any fleeting departures from steady-state can in any case always be averaged out by selecting a long enough timeframe on which to aggregate the velocity information.

The only measurements we can make are on the image itself so we need to express E as a functional on $\mathcal{I}$. Assuming here only that ϕ is a continuously differentiable, bijective mapping we identify $\mathcal{W}$ as $\phi^{-1}(\mathcal{I})$ so that a change of variables gives

$$\begin{aligned}\int_{\phi^{-1}(\mathcal{I})} \Psi\left(\operatorname{div} \boldsymbol{V}\right) \mathrm{d}\mathbf{X} &= \int_{\mathcal{I}} (\phi^{-1})^{*} \Psi\left(\operatorname{div} \boldsymbol{V}\right) \left|\det D\phi^{-1}\right| \mathrm{d}\mathbf{x} \\ &= \int_{\mathcal{I}} \Psi\left((\phi^{-1})^{*} \operatorname{div} \boldsymbol{V}\right) \left|\det D\phi^{-1}\right| \mathrm{d}\mathbf{x}\end{aligned}$$

with $(\phi^{-1})^{*} \operatorname{div} \boldsymbol{V}$ simply the pullback of $\operatorname{div} \boldsymbol{V}$ on the image. Now $\operatorname{div} \boldsymbol{V} = \operatorname{tr}(D\boldsymbol{V})$, where by $D\boldsymbol{V}$ we are denoting the component matrix of Lie derivatives of $\boldsymbol{V}$ such that

$(D\boldsymbol{V})_{ij}$ is the i'th component of $\mathcal{L}_{\frac{\partial}{\partial X_j}}\boldsymbol{V}$. Because we need to express all quantities on $\mathcal{W}$ as induced by their counterparts on $\mathcal{I}$, we define

$$\mathcal{L}_{\frac{\partial}{\partial X_i}}\boldsymbol{V} \equiv (\phi^{-1})_* \mathcal{L}_{\phi_* \frac{\partial}{\partial X_i}} \boldsymbol{v} = (\phi^{-1})_* \left[\phi_* \frac{\partial}{\partial X_i}, \boldsymbol{v}\right]$$

which is just the pushforward of the Lie derivative of $\boldsymbol{v}$ with $\phi_* \frac{\partial}{\partial X_i}$. Denoting the i'th canonical basis vector (the zero vector with a 1 in the i'th position) as E_i, the components of $\phi_* \frac{\partial}{\partial X_i}$ at $\phi^{-1}(\boldsymbol{x})$ are written as $D\phi(E_i) \circ \phi^{-1}(\boldsymbol{x})$, the i'th column of matrix $D\phi \circ \phi^{-1}(\boldsymbol{x})$. But this is just

$$D\phi(E_i) \circ \phi^{-1}(\boldsymbol{x}) = \frac{\partial}{\partial X_i} \phi \circ \phi^{-1}(\boldsymbol{x})$$

so that in local image coordinates

$$\phi_* \frac{\partial}{\partial X_i} = \sum_j \frac{\partial \phi_j}{\partial X_i} \circ \phi^{-1}(\boldsymbol{x}) \frac{\partial}{\partial x_j} \ .$$

Then the expansion of the commutator becomes

$$\left[\phi_* \frac{\partial}{\partial X_i}, \boldsymbol{v}\right] = \sum_{j,k} \left(\frac{\partial \phi_j}{\partial X_i} \frac{\partial v_k}{\partial x_j} - v_j \sum_l \frac{\partial^2 \phi_k}{\partial X_l \partial X_i} \frac{\partial (\phi^{-1})_l}{\partial x_j} \right) \frac{\partial}{\partial x_k}$$

or in component form,

$$D\boldsymbol{v} \frac{\partial \phi}{\partial X_i} - \left(\frac{\partial}{\partial X_i} D\phi \right) D\phi^{-1} \boldsymbol{v} \ .$$

The pushforward by ϕ^{-1} of this vector gives the elements of DV,

$$(DV)_{ij} = \left(D\phi^{-1} D\boldsymbol{v} D\phi\right)_{ij} - \sum_{k,l} \left(D\phi^{-1}\right)_{ik} \left(D^2\phi\right)^k_{jl} \left(D\phi^{-1}\boldsymbol{v}\right)_l \tag{6.2}$$

where $(D^2\phi)^i_{jk} \doteq D_k \left(D\phi\right)_{ij}$ is shorthand for $\partial^2 \phi_i / \partial X_k \partial X_j$. However, we can simplify slightly the second term of (6.2) by exploiting the inverse function theorem to assert

that $D\phi^{-1} = (D\phi)^{-1}$. This is justified because ϕ is continuously differentiable and $D\phi$ has a non-zero determinant everywhere in $\mathcal{W}$ due to ϕ being a submersion. Then using the identity

$$\frac{\partial}{\partial X_i}(D\phi)^{-1} = -(D\phi)^{-1}\left(\frac{\partial}{\partial X_i}D\phi\right)(D\phi)^{-1}$$

which we equate to the application of the chain rule

$$\frac{\partial}{\partial X_i}\left(D\phi^{-1}\circ\phi(\boldsymbol{X})\right)_{jk} = \sum_l \frac{\partial}{\partial x_l}\left(D\phi^{-1}\right)_{jk}\frac{\partial}{\partial X_i}\phi_l = \sum_l \left(D^2\phi^{-1}\right)^j_{kl}(D\phi)_{li} \ ,$$

we can rewrite the second term of (6.2) as

$$\begin{aligned} -\sum_{k,l}\left(D\phi^{-1}\right)_{ik}\left(D^2\phi\right)^k_{jl}\left(D\phi^{-1}v\right)_l &= \sum_k \frac{\partial}{\partial X_k}\left(D\phi^{-1}\right)_{ij}v_k \\ &= \sum_{k,l}\left(D^2\phi^{-1}\right)^i_{kl}(D\phi)_{lj}\,v_k \end{aligned}$$

so that making all $\boldsymbol{x}$-dependence via the pullback $(\phi^{-1})^*$ explicit, (6.2) becomes

$$\boxed{(\phi^{-1})^*(DV)_{ij} = \left(D\phi^{-1}D\boldsymbol{v}D\phi\circ\phi^{-1}(\boldsymbol{x})\right)_{ij} + \sum_{k,l}\left(D^2\phi^{-1}\right)^i_{kl}\left(D\phi\circ\phi^{-1}(\boldsymbol{x})\right)_{lj}v_k} \ . \tag{6.3}$$

Taking the trace of (6.3) we get

$$\begin{aligned} (\phi^{-1})^*\operatorname{div}\boldsymbol{V} &= \sum_i\left((\phi^{-1})^*D\boldsymbol{V}\right)_{ii} \\ &= \operatorname{trace}\left(D\phi^{-1}D\boldsymbol{v}D\phi\right) + \sum_{i,k,l}\left(D^2\phi^{-1}\right)^i_{kl}(D\phi)_{li}\,v_k \\ &= \operatorname{trace}(D\boldsymbol{v}) + \sum_k\sum_{i,l}\left(D^2\phi^{-1}\right)^i_{lk}(D\phi)_{li}\,v_k \\ &= \operatorname{div}(\boldsymbol{v}) + \rho\cdot\boldsymbol{v} \end{aligned}$$

where

$$\rho_i = \text{trace}\left(\left(\frac{\partial}{\partial x_i} D\phi^{-1}\right) D\phi\right) = \text{trace}\left(D_i^2 \phi^{-1} D\phi\right) \quad .$$

We have now transformed the energy equation (6.1) into

$$E = \int_{\mathcal{I}} \Psi(\text{div}(\boldsymbol{v}) + \rho \cdot \boldsymbol{v}) \left|\det D\phi^{-1}\right| \mathrm{d}\mathbf{x} \,. \tag{6.4}$$

If we are able to sample from sufficiently many points in the image and are using the penalization function (5.21), we can benefit from a reduced computational burden by transforming (6.4) into a boundary line integral. In this case we split the image into domains $\mathcal{I}_+$ and $\mathcal{I}_-$ on which $c_\pm \doteq \text{sign}(\text{div}(\boldsymbol{v}) + \rho \cdot \boldsymbol{v})$ is positive and negative respectively. This is largely a didactic tool; while we expect the contour between domains to resemble a low-order polynomial[1], in practice we need only sample $c_\pm$ once. We typically want to assert the zero-divergence condition on a region of interest within the image, such as the area occupied by approaching vehicles - a subset of either $\mathcal{I}_+$ or $\mathcal{I}_-$ - which obviates the need for a full distinction between domains as shown in Figure 6.1. . On each we put $f = |\det D\phi^{-1}|$ and observe that

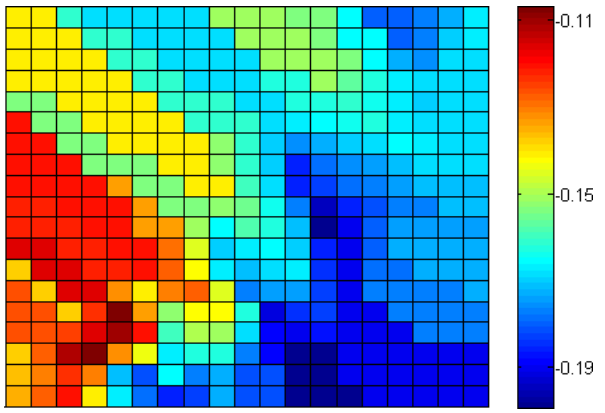
$$\int f \,\text{div}(v) \mathrm{d}\mathbf{x} = \int \text{div}(fv) - \nabla f \cdot v \mathrm{d}\mathbf{x} \quad , \tag{6.5}$$

so that the Divergence Theorem can be invoked to write

$$\begin{aligned} E &= \sum_{\mathcal{I}_\pm} c_\pm \int_{\mathcal{I}_\pm} \text{div}(fv) + (f\rho - \nabla f) \cdot v \mathrm{d}\mathbf{x} \\ &= \sum_{\mathcal{I}_\pm} c_\pm \left(\int_{\partial \mathcal{I}_\pm} fv \cdot n \mathrm{d}l + \int_{\mathcal{I}_\pm} (f\rho - \nabla f) \cdot v \mathrm{d}\mathbf{x} \right) \quad . \end{aligned}$$

[1]The contour between the analogous world domains $\mathcal{W}_\pm$ describing the sign of $\text{div}\,\boldsymbol{V}$ should coincide with the centerline between vehicles traveling in opposite directions. For a straight road this is a polynomial of order one. Recall that the smooth mapping ϕ preserves lines and conics.

(a) Region of interest encompassing approaching vehicles only.



(b) Heatmap of quantity $\mathrm{div}(\boldsymbol{v}) + \rho \cdot \boldsymbol{v}$ over the ROI, which clearly lies within $\mathcal{I}_-$. Vector field $\boldsymbol{v}$ has been nearest-neighbor extrapolated to cover the region.

Figure 6.1: Regions of interest are typically contained within either domain $\mathcal{I}_+$ or $\mathcal{I}_-$.

Now for any invertible parameterized matrix $A(x)$ we have the identity

$$\frac{d\,|\det A|}{dx} = |\det A|\,\mathrm{trace}\left(A^{-1}\frac{dA}{dx}\right)$$

which we apply to matrix $D\phi^{-1}$:

$$\begin{aligned}
\frac{\partial}{\partial x_i} f &= f\,\mathrm{trace}\left(\left(D\phi^{-1}\right)^{-1}\frac{\partial}{\partial x_i} D\phi^{-1}\right) \\
&= f\,\mathrm{trace}\left(D_i^2\phi^{-1} D\phi\right) \\
&= f\rho_i
\end{aligned}$$

where we have again used the inverse function theorem to assert $D\phi^{-1} = (D\phi)^{-1}$ as justified above. This simply states that $f\rho - \nabla f = 0$, finally yielding

$$E = \sum_{\mathcal{I}_\pm} c_\pm \int_{\partial\mathcal{I}_\pm} fv \cdot n\mathrm{d}l \tag{6.6}$$

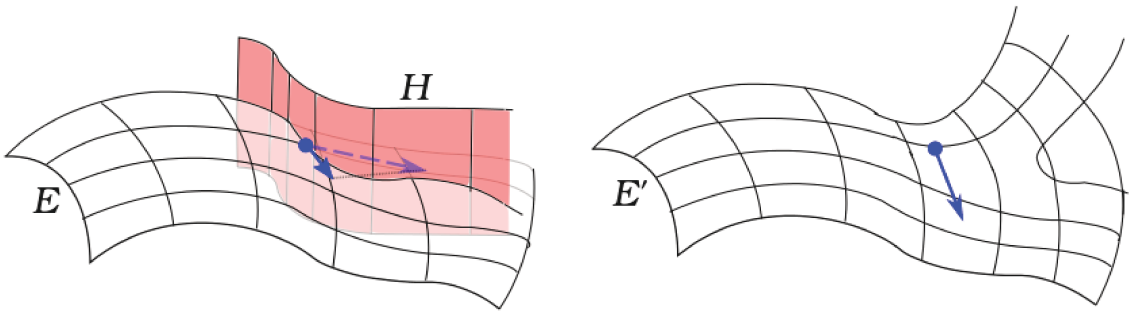
Comparing this with the divergence theorem applied directly to (6.1) under penalty

function (5.21),

$$E = \sum_{\mathcal{W}_\pm} c_\pm \int_{\mathcal{W}_\pm} \operatorname{div} \boldsymbol{V} \mathrm{d}\mathbf{X}$$
$$= \sum_{\mathcal{W}_\pm} c_\pm \int_{\partial\mathcal{W}_\pm} \boldsymbol{V} \cdot n \mathrm{d}l \tag{6.7}$$

which in comparison to (6.6) highlights the role of the Jacobian determinant f as the sole scaling factor needed to account for the differing divergence measurements between world frame and image. The boundary integrals (6.6) are less demanding than the surface integrals (6.4) to compute and in addition do not require the derivatives $D\boldsymbol{v}$ to be estimated, a task that augments the error involved in the minimization. On the other hand, for relatively few points in the image, equation (6.4) yields more accurate results and allows for a general penalty function.

A divergence-free criterion is not sufficient even for for affine rectification; in fact it is only able to resolve one of the two perspective-related degrees of freedom inherent in the homography. One option is to build a further geometric constraint into the minimization process that we need to constrain anyway in order to recover the similarity components of the transformation. The other option is to add a geometric term to the energy functional. The latter is preferable in this case: active contraints added to the minimization tend to impede the gradient descent by 'blocking off' the free descent direction and allowing only very small decrements in the energy (Figure 6.2). Additionally, imposing a direct constraint based on elements affected by noise detracts from the robustness of the method; a penalty term deals with noisy



(a) Active constraint H imposed on energy surface E. Only a small component of the descent step is allowed.

(b) The constraint added in as a penalty term to yield energy surface E', often resulting in a faster descent.

Figure 6.2: Constraint H added either directly or as a penalty term to the minimization of energy E.

measurements in a more natural manner (as opposed to the active constraints in Section 6.2 where measurements are assumed to be noiseless). The additional term we propose that will enable a pseudo-metric (affine plus skewness) rectification is

$$\int_{\mathcal{W}} \Psi\left(\boldsymbol{V}\cdot\boldsymbol{W}\right)\mathrm{d}\mathbf{X}\,,$$

which encourages orthogonality between vehicle trajectories and the transverse vectors. The modified energy balances this term against the divergence-free requirement with constant β,

$$E=\int_{\mathcal{W}} \Psi\left(\operatorname{div}\boldsymbol{V}\right)+\beta\Psi\left(\boldsymbol{V}\cdot\boldsymbol{W}\right)\mathrm{d}\mathbf{X}\,,$$

which is pulled back to the image by ϕ^{-1} in the following manner:

$$\begin{aligned}\left(\phi^{-1}\right)^{*}\Psi\left(\boldsymbol{V}\cdot\boldsymbol{W}\right) &= \Psi\left(\left(\phi_{*}^{-1}\boldsymbol{v}\right)\cdot\left(\phi_{*}^{-1}\boldsymbol{w}\right)\right)\\ &= \Psi\left(\left(D\phi^{-1}\boldsymbol{v}\right)\cdot\left(D\phi^{-1}\boldsymbol{w}\right)\right)\\ &= \Psi\left(\phi^{-1}(\boldsymbol{v},\boldsymbol{x})\cdot\phi^{-1}(\boldsymbol{w},\boldsymbol{x})\right)\,, \end{aligned} \tag{6.8}$$

using equation (4.23). For convenience we set the argument to Ψ in (6.8) equal to $\chi(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{w})$, to give

$$\boxed{E = \int_{\mathcal{I}} \Psi(\mathrm{div}(\boldsymbol{v}) + \rho \cdot \boldsymbol{v}) + \beta\Psi\left(\chi(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{w})\right) \left|\det D\phi^{-1}\right| \mathrm{d}\mathbf{x}}\,. \tag{6.9}$$

6.2 The Constraint Submanifold

The minimization of energy (6.9) leaves five degrees of freedom to be resolved: the four similarity degrees of freedom as well as the non-isotropic scaling associated with the rectangle ambiguity of section 5.2. We can ensure these are fixed by incorporating knowledge we have about the scene; in this case that the image vectors $\boldsymbol{v}_0, \boldsymbol{w}_0$ at the chosen origin correspond to known world vectors $\boldsymbol{V}_0, \boldsymbol{W}_0$. The following five constraints should then be observed:

$$\phi^{-1}(\boldsymbol{x}_0) = \mathbf{0} \tag{6.10}$$

$$D\phi^{-1}\boldsymbol{v}_0 = \boldsymbol{V}_0 \tag{6.11}$$

$$D\phi_i^{-1}\boldsymbol{w}_0 = (\boldsymbol{W}_0)_i \tag{6.12}$$

where index i in the last constraint can be set to fix either component of $\boldsymbol{W}_0$. Summarizing these constraints with the vector-valued condition $H(g) = \mathbf{0}$, the constraint submanifold on which the minimization is to take place is given by $G' = \{g \in SL(3) : H(g) = \mathbf{0}\}$. We follow [35] in establishing a modification to the gradient descent direction (3.28) on such a submanifold. For a $g^* \in G'$ that minimizes energy E, the

method of Lagrange multipliers states that there exists a vector $\lambda^* \in \mathbb{R}^5$ such that (g^*, λ^*) is a stationary point of the Lagrangian $\mathcal{L}(g, \lambda) = E(g) - \lambda^T H(g)$:

$$\begin{cases} \widetilde{\nabla \mathcal{L}} & = \widetilde{\nabla E} - \sum_j \lambda_j \widetilde{\nabla H}_j = 0 \\ \frac{\partial \mathcal{L}}{\partial \lambda} & = H(g) = \mathbf{0} \end{cases} . \tag{6.13}$$

Now the gradient descent takes the form

$$\xi = g^{-1}\dot{g} = -\widetilde{\nabla \mathcal{L}}(g, \lambda) \tag{6.14}$$

where λ satisfies $\partial \mathcal{L}/\partial \lambda = \mathbf{0}$. This is achieved by requiring a zero Gateaux derivative of the second expression in (6.13), thus specifying that the constraints must continue to be satisfied in the update direction,

$$\begin{aligned} 0 &= \frac{d}{dt} H_i(g \exp(t\xi)) \\ &= \left\langle \widetilde{\nabla H}_i, \xi \right\rangle_{\mathfrak{g}} \\ &= -\left\langle \widetilde{\nabla H}_i, \widetilde{\nabla \mathcal{L}} \right\rangle_{\mathfrak{g}} \\ &= -\left\langle \widetilde{\nabla H}_i, \widetilde{\nabla E} \right\rangle_{\mathfrak{g}} + \sum_j \lambda_j \left\langle \widetilde{\nabla H}_i, \widetilde{\nabla H}_j \right\rangle_{\mathfrak{g}} . \end{aligned}$$

Thanks to the definition of the metric $\langle \cdot, \cdot \rangle_{T_g^* G}$ in (3.25), vector λ is then the solution to the linear system of equations

$$\sum_j \lambda_j \langle dH_i, dH_j \rangle_{T_g^* G} = \langle dH_i, dE \rangle_{T_g^* G}$$

which is plugged into (6.14) to give by (3.28) the descent vector

$$\xi = -\mathbb{I}^{-1} g^T \left(\nabla_g E - \sum_j \lambda_j \nabla_g H_j \right) . \tag{6.15}$$

An infinitesimal step on the full manifold $SL(3)$ in this direction maintains the condition $H(g) = \mathbf{0}$ provided this held true at the initial estimate of g. For this reason we need to condition an initial guess (usually $g_k|_{k=0} = I$) with information derived from the constraints. This is simplest for the translational constraints (6.10) for which (5.6) must hold, yielding $g_{i3} = (\boldsymbol{x}_0)_i \, g_{33}$ for any initial scale g_{33}. The scaling and rotational constraints (6.11) and (6.12) are somewhat more involved but benefit from our knowledge of $\boldsymbol{V}_0$ and $\boldsymbol{W}_0$, enabling a replacement of the expression (4.24) with its inverse (4.25). The latter equation is simplified at $\boldsymbol{X}_0 = \mathbf{0}$,

$$(\boldsymbol{v}_0)_i = D\phi_i \boldsymbol{V}_0 = \frac{g^3 \bar{\boldsymbol{V}}_0}{g^3 \bar{\boldsymbol{X}}_0} \left(\frac{g^i \bar{\boldsymbol{V}}_0}{g^3 \bar{\boldsymbol{V}}_0} - \frac{g^i \bar{\boldsymbol{X}}_0}{g^3 \bar{\boldsymbol{X}}_0} \right) = \frac{1}{g_{33}} \left[g^i - \left(\frac{g_{i3}}{g_{33}} \right) g^3 \right] \bar{\boldsymbol{V}}_0 \,, \tag{6.16}$$

which for a given initial row g^3 can be used to solve for a valid initial leading 2×2 submatrix: Rearranging equations (6.16) as well as their counterparts for $\boldsymbol{w}_0$ and setting B as the leading 2×2 submatrix of $g_3 g^3$, we arrive at another linear system of equations to solve:

$$\begin{bmatrix} \boldsymbol{V}_0^T & \mathbf{0}^T \\ \mathbf{0}^T & \boldsymbol{V}_0^T \\ \boldsymbol{W}_0^T & \mathbf{0}^T \\ \mathbf{0}^T & \boldsymbol{W}_0^T \end{bmatrix} \begin{bmatrix} g_{11} \\ g_{12} \\ g_{21} \\ g_{22} \end{bmatrix} = g_{33} \begin{bmatrix} \boldsymbol{v}_0 \\ \boldsymbol{w}_0 \end{bmatrix} + \frac{1}{g_{33}} \begin{bmatrix} B & \mathbf{0} \\ \mathbf{0} & B \end{bmatrix} \begin{bmatrix} \boldsymbol{V}_0 \\ \boldsymbol{W}_0 \end{bmatrix} \,. \tag{6.17}$$

Populating the leading submatrix of the initial iterate with the solution to (6.17) ensures that the constraints (6.16) will at each iteration be met up to the accumulating error generated by the finite step size. To combat this we periodically reset the translational components of the homography and re-solve (6.16) with the new row values g^3 as a way of projecting back onto the manifold G'.

6.3 Computing the Descent Direction

6.3.1 The Energy Derivative

The vector (6.15) requires the matrices $\nabla_g E \doteq [\partial E/\partial g_{ij}]$ and $\nabla_g H \doteq [\partial H/\partial g_{ij}]$, from which the derivatives $\partial E/\partial \alpha_k$ and $\partial H/\partial \alpha_k$ can be calculated for the chosen parameterization α, just as in section 5.3. We start by establishing some useful identities. From (4.23) we have

$$\begin{aligned}\det D\phi^{-1} &= \frac{1}{\left(\boldsymbol{M}_3^T\bar{\boldsymbol{x}}\right)^2}\det\left(\begin{bmatrix}M_{11} & M_{21}\\ M_{12} & M_{22}\end{bmatrix} - \frac{1}{\boldsymbol{M}_3^T\bar{\boldsymbol{x}}}\begin{bmatrix}\boldsymbol{M}_1^T\bar{\boldsymbol{x}}\\ \boldsymbol{M}_2^T\bar{\boldsymbol{x}}\end{bmatrix}\begin{bmatrix}M_{13} & M_{23}\end{bmatrix}\right)\\ &= \frac{1}{\left(\boldsymbol{M}_3^T\bar{\boldsymbol{x}}\right)^2}\det\left(\begin{bmatrix}M_{11} & M_{21}\\ M_{12} & M_{22}\end{bmatrix}\right)\left(1 - \frac{1}{\boldsymbol{M}_3^T\bar{\boldsymbol{x}}}\begin{bmatrix}M_{13} & M_{23}\end{bmatrix}\begin{bmatrix}M_{11} & M_{21}\\ M_{12} & M_{22}\end{bmatrix}^{-1}\begin{bmatrix}\boldsymbol{M}_1^T\bar{\boldsymbol{x}}\\ \boldsymbol{M}_2^T\bar{\boldsymbol{x}}\end{bmatrix}\right).\end{aligned} \tag{6.18}$$

Now since matrix $\boldsymbol{M} = \det(g)g^{-T}$, $\det(\boldsymbol{M}) = (\det(g))^2$ and

$$\det(g)g = \det(\boldsymbol{M})\boldsymbol{M}^{-T}{=}\mathrm{Cof}(\boldsymbol{M})\,,$$

with the implication that

$$\det\left(\begin{bmatrix}M_{11} & M_{21}\\ M_{12} & M_{22}\end{bmatrix}\right) = \mathrm{Cof}(\boldsymbol{M})_{33} = \det(g)g_{33}.$$

Then

$$\begin{aligned}\begin{bmatrix}M_{13} & M_{23}\end{bmatrix}\begin{bmatrix}M_{11} & M_{21}\\ M_{12} & M_{22}\end{bmatrix}^{-1} &= \frac{1}{\det(g)g_{33}}\begin{bmatrix}M_{13} & M_{23}\end{bmatrix}\begin{bmatrix}M_{22} & -M_{21}\\ -M_{12} & M_{22}\end{bmatrix}\\ &= \frac{-1}{\det(g)g_{33}}\begin{bmatrix}\mathrm{Cof}(\boldsymbol{M})_{31} & \mathrm{Cof}(\boldsymbol{M})_{32}\end{bmatrix}\\ &= -g_{33}^{-1}\begin{bmatrix}g_{31} & g_{32}\end{bmatrix}\end{aligned}$$

and using the relation (4.29)

$$-\frac{1}{\boldsymbol{M}_3^T\bar{\boldsymbol{x}}}\begin{bmatrix}\boldsymbol{M}_1^T\bar{\boldsymbol{x}}\\ \boldsymbol{M}_2^T\bar{\boldsymbol{x}}\end{bmatrix} = -\phi^{-1}(\boldsymbol{x}) = \begin{bmatrix}(\Gamma_3(\boldsymbol{x}))_1^T\\ (\Gamma_3(\boldsymbol{x}))_2^T\end{bmatrix} g_3$$

the parenthesized factor in (6.18) becomes, with the help of (4.30),

$$\begin{aligned}1 - \frac{1}{g_{33}} g^3\Gamma_3(\boldsymbol{x})^T g_3 &= 1 - \frac{1}{g_{33}}\left(g_{33} - \frac{1}{\phi_3^{-1}(\boldsymbol{x})}\right)\\ &= \frac{1}{g_{33}\phi_3^{-1}(\boldsymbol{x})}\,.\end{aligned}$$

This brings us to the compact result

$$\det D\phi^{-1} = \frac{1}{\det(g)\left(\phi_3^{-1}(\boldsymbol{x})\right)^3}\,. \tag{6.19}$$

Following a similar though more laborious procedure, we also arrive at the expressions

$$\rho = -\frac{3}{\phi_3^{-1}(\boldsymbol{x})}\begin{bmatrix}M_{13} & M_{23}\end{bmatrix}\,,\ \rho\cdot\boldsymbol{v} = -3\frac{\phi_3^{-1}(\boldsymbol{v})}{\phi_3^{-1}(\boldsymbol{x})}\,,$$

the first of which can be used to verify the identity used earlier in equation (6.6), $f\rho - \nabla f = 0$: Again for clarity $f = |\det D\phi^{-1}|$, for which the derivative from (6.19) is

$$\nabla f = -\frac{3}{\det(g)\left(\phi_3^{-1}(\boldsymbol{x})\right)^4}\begin{bmatrix}M_{13} & M_{23}\end{bmatrix} = -f\rho \tag{6.20}$$

as required. Progressing with the derivative calculations and applying the fact that $\det(g) = 1$ (which up to now we have been prevented from doing because it is not

identically true),

$$\nabla_g f = 2f\boldsymbol{M} - 3f\Gamma_3(\boldsymbol{x})$$

$$\nabla_g(\rho \cdot \boldsymbol{v}) = \rho \cdot \boldsymbol{v}\left(\Gamma_3(\boldsymbol{v}) - \Gamma_3(\boldsymbol{x})\right)$$

$$\nabla_g \chi = \sum_i \phi_i^{-1}(\boldsymbol{w}, \boldsymbol{x})\Theta_i(\boldsymbol{v}, \boldsymbol{x}) + \phi_i^{-1}(\boldsymbol{v}, \boldsymbol{x})\Theta_i(\boldsymbol{w}, \boldsymbol{x})$$

using the matrices $\Theta_i(\cdot, \boldsymbol{x})$ from (4.32). We then have

$$\begin{aligned}\nabla_g E = \int_{\mathcal{I}} & \nabla_g f\left(\Psi(\operatorname{div}(\boldsymbol{v}) + \rho \cdot \boldsymbol{v}) + \beta\Psi\left(\chi\right)\right) \\ & + f\left(\Psi'(\operatorname{div}(\boldsymbol{v}) + \rho \cdot \boldsymbol{v})\nabla_g(\rho \cdot \boldsymbol{v}) + \beta\Psi'\left(\chi\right)\nabla_g\chi\right)\mathrm{d}\mathbf{x}\end{aligned}$$

which after substitution becomes

$$\boxed{\begin{aligned}\nabla_g E \;=\; & 2E\boldsymbol{M} + \int_{\mathcal{I}}\Big[\beta\Psi'\left(\chi\right)\textstyle\sum_{i=1}^{2}\phi_i^{-1}(\boldsymbol{w}, \boldsymbol{x})\Theta_i(\boldsymbol{v}, \boldsymbol{x}) + \phi_i^{-1}(\boldsymbol{v}, \boldsymbol{x})\Theta_i(\boldsymbol{w}, \boldsymbol{x}) \\ & + \Psi'(\operatorname{div}(\boldsymbol{v}) + \rho \cdot \boldsymbol{v})\rho \cdot \boldsymbol{v}\left(\Gamma_3(\boldsymbol{v}) - \Gamma_3(\boldsymbol{x})\right) \\ & - 3\left(\Psi(\operatorname{div}(\boldsymbol{v}) + \rho \cdot \boldsymbol{v}) + \beta\Psi(\chi)\right)\Gamma_3(\boldsymbol{x})\Big] f\mathrm{d}\mathbf{x}\,.\end{aligned}} \tag{6.21}$$

When the chosen inertial matrix $\mathbb{I}$ is a multiple of the identity, $\nabla_g E$ simplifies further under the observation that matrix $\boldsymbol{M}$ represents a direction orthogonal to the manifold $SL(3)$. Using equation (3.28), for any element $\xi \in \mathfrak{g}$,

$$\begin{aligned}\langle \boldsymbol{M}, \xi_G(g)\rangle &= \left\langle \mathbb{I}^{-1} g^T \boldsymbol{M}, \xi \right\rangle_{\mathfrak{g}} \\ &= \det(g)\left\langle \mathbb{I}^{-1} g^T g^{-T}, \xi \right\rangle_{\mathfrak{g}} \\ &\propto \operatorname{trace}(\xi) \\ &= 0\,,\end{aligned}$$

recalling that the Lie algebra $\mathfrak{g} = \mathfrak{sl}(3)$ is the space of real traceless matrices. Thus, the first term of (6.21) vanishes as a contribution to the descent direction on the manifold.

6.3.2 The Constraint Derivatives

The derivatives of the constraints are calculated in a straightforward manner. From constraints (6.10), (6.11) and (6.12) we find

$$\nabla_g H_j = \begin{cases} \Gamma_i(\boldsymbol{x}_0) - \phi_i^{-1}(\boldsymbol{x}_0)\Gamma_3(\boldsymbol{x}_0) & j \in \{1,2\} \\ \Theta(\boldsymbol{v}_0, \boldsymbol{x}_0) & j \in \{3,4\} \\ \Theta(\boldsymbol{w}_0, \boldsymbol{x}_0) & j = 5 \end{cases}$$

where $i \in \{1,2\}$ in each case.

6.3.3 Basis Coordinates of $SL(3)$

Section 3.3.2 developed the gradient descent equation in a coordinate-free framework, where the one-form dE was assumed to be an element of T^*G which would otherwise need to be enforced by using the projection like (3.29) of the associated Lie algebra element. As an alternative, we can parameterize the group $SL(3)$ with a minimal sufficient set of parameters that then guarantees $\mathbb{I}^{-1} g^T \nabla_g E$ as a member of the Lie algebra $\mathfrak{g}$, at least up to working precision. An obvious such set for example are the parameters α_i we relate physically to the homography: the three rotation,

three translation and two intrinsic parameters of equation (4.17), or the equivalent set in (4.14). A judicious choice of parameterization is likely to speed up the energy minimization algorithm by enabling larger step sizes. We focus now on the first order optimality condition $\dot{E}(g^*) = 0$ where the curve $t \to (\alpha_1(t), \alpha_2(t), ..., \alpha_n(t))$ denotes a path in the parameter space that in turn gives rise to the path $g = g(\alpha(t))$ through G. Because the effect of the constraints is just to modify $dE \to dE - \sum_j \lambda_j dH_j$, we can ignore their presence in the following development before reintroducing them again later. First we note that

$$\begin{aligned}
\dot{E}(g) &= \langle dE(g), \dot{g} \rangle \\
&= \left\langle dE(g), \sum_{i=1}^{n} \frac{\partial g}{\partial \alpha_i} \frac{d\alpha_i}{dt} \right\rangle \\
&= \sum_{i=1}^{n} \dot{\alpha}_i \langle dE(g), \partial_i g \rangle \\
&= \sum_{i=1}^{n} \dot{\alpha}_i \left\langle \mathbb{I}^{-1} g^T \nabla_g E, \xi_i \right\rangle_{\mathfrak{g}} \\
&= \sum_{i=1}^{n} \dot{\alpha}_i \, \mathrm{trace}(\mathbb{I}^{-1} g^T \nabla_g E \xi_i^T)
\end{aligned} \tag{6.22}$$

where we have introduced the vectors $\xi_i \in \mathfrak{g}$ that generate the infinitesimal action vector fields (recalling equation (3.13))

$$\begin{aligned}
(\xi_i)_G(g) &= \left. \frac{d}{dt} \right|_{t=0} g \exp(t\xi_i) \\
&= g\xi_i \\
&\doteq \partial g / \partial \alpha_i \, .
\end{aligned} \tag{6.23}$$

We emphasize the difference between the coordinate-free approach and this one; namely, instead of achieving gradient descent by directly finding the tangent vector $\xi \in \mathfrak{g}$, we solve for the parameter velocities $\dot{\alpha}_i$ given the vectors ξ_i that couple to the parameters α_i via the left-trivialized derivatives $\xi_i = g^{-1}\partial g/\partial \alpha_i$. We then employ an integration scheme such as the Euler method to arrive at an update to the parameter estimates, which updates the current configuration g in turn. This change of basis allows us to retain the matrix group approach while keeping the manifold structure (and in particular its dimensionality) explicit. Of course, a poorly chosen basis will make computation of the derivatives $\partial_i g$ difficult. The physical parameters mentioned above as a basis set have this problem. A perhaps more judicious basis set however is that of the Lie algebra $\mathfrak{sl}(3)$; the eight matrices A_i:

$$A_1 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad A_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad A_3 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad A_4 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$A_5 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad A_6 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad A_7 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad A_8 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} .$$

Each element $g \in G$ sufficiently close to the identity is then parameterized as

$$g = \exp\left(\sum_{i=1}^{8} \alpha_i A_i\right) \tag{6.24}$$

and the coordinates α_i are precisely the canonical coordinates of the first kind referenced in equation (3.7). Now the group $SL(3)$ is not Abelian, so the mapping $\exp : \mathfrak{g} \to G$ is not a homomorphism[2]. That is, for $\xi, \eta \in \mathfrak{g}$, $\exp(\xi)\exp(\eta) \neq \exp(\xi + \eta)$

[2]In fact, the exp mapping is not even surjective for group $SL(3)$ (see for example [36, pp. 468]). It can be shown that any matrix g in the group $SL(2)$ with $\text{trace}(g) < -2$ is not the exponential

but rather, the Baker-Campbell-Hausdorff formula provides the correction as a series of commutator nestings [37, pp. 17]:

$$\exp(\xi)\exp(\eta) = \exp(\xi + \eta + \frac{1}{2}[\xi,\eta] + \frac{1}{12}[\xi,[\xi,\eta]] - \frac{1}{12}[\eta,[\xi,\eta]] + ...) \quad .$$

A particular consequence of this pertains to the derivative of the exp mapping at $\xi \in \mathfrak{g}$ [38, pp. 107-108],

$$\begin{aligned}
\operatorname{dexp}_\xi \eta &= \left.\frac{d}{dt}\right|_{t=0} \exp(-\xi)\exp(\xi + t\eta) \\
&= \sum_{n=0}^{\infty} \frac{(-1)^n}{(n+1)!} \operatorname{ad}_\xi^n \eta \\
&= \eta - \frac{1}{2}[\xi,\eta] + \frac{1}{6}[\xi,[\xi,\eta]] - \frac{1}{24}[\xi,[\xi,[\xi,\eta]]] + ...
\end{aligned}$$

so that putting $\xi = \sum_i \alpha_i A_i$, the derivative $\partial_i g$ is calculated

$$\begin{aligned}
\partial_i g &= \partial_i \exp(\xi) \\
&= \left.\frac{d}{dt}\right|_{t=0} \exp(\xi + t\partial_i \xi) \\
&= \exp(\xi) \left.\frac{d}{dt}\right|_{t=0} \exp(-\xi)\exp(\xi + t\partial_i \xi) \\
&= \exp(\xi) \operatorname{dexp}_\xi \partial_i \xi \\
&= \exp\left(\sum\nolimits_j \alpha_j A_j\right) \operatorname{dexp}_{\sum_j \alpha_j A_j} A_i \ .
\end{aligned}$$

of any matrix in $\mathfrak{sl}(2)$. While the same analysis is more difficult in the $SL(3)$ case (and lack of a counterexample does not establish the principle), it appears as though surjectivity holds on the subgroup $g \in SL(3) : \text{trace}(g) > 0$ which holds true for any homography that we are interested in: Inspection of the form of the homography in (4.18) coupled with the fact that the camera must sit 'behind' the world plane yields this result. To our knowledge, the region on which surjectivity holds on $SL(3)$ is an open question.

This gives for the left-trivialized vectors ξ_i from equation (6.23),

$$\begin{aligned}\xi_i &= g^{-1}\partial_i g \\ &= \text{dexp}_{\sum_j \alpha_j A_j} A_i \\ &= A_i - \frac{1}{2}\sum\nolimits_j \alpha_j [A_j, A_i] + \frac{1}{6}\sum\nolimits_{j,k} \alpha_j \alpha_k [A_j, [A_k, A_i]] + ... \end{aligned} \tag{6.25}$$

While (6.25) may appear onerous to compute, the recursive series converges very quickly [15, for group $SO(3)$]. In fact if all $\alpha_i = 1$ we have $\xi_i = A_i$ with the exception of ξ_7, for which $\left\|\xi_7^{(100)} - \xi_7^{(5)}\right\|^2 \sim 10^{-6}$ under the Frobenius norm and where $\xi_i^{(n)}$ denotes the expression (6.25) calculated with the first n terms. Of course, for general vector α we expect $\xi_i \neq A_i$ and convergence will be slower, but in our experiments we found that about 16 terms were sufficient with typical vectors α to achieve an average tolerance $\left\|\xi_i^{(16)} - \xi_i^{(15)}\right\|^2 \sim 10^{-6}$ across all vectors ξ_i. To solve for $\dot{\alpha}_i$ we use the row-vectorize operator $\cdot^V$ and observe from (6.22) that $\dot{E} \leq 0$ is ensured when

$$\sum_{i=1}^{n} \dot{\alpha}_i \xi_i^V = -(\mathbb{I}^{-1} g^T \nabla_g E)^V \quad . \tag{6.26}$$

In matrix form where $B = [\xi_1^V \ \xi_2^V \ ... \ \xi_n^V]$, (6.26) becomes $B\dot{\alpha} = -(\mathbb{I}^{-1} g^T \nabla_g E)^V$ so the solution is

$$\dot{\alpha} = -B^{-1}(\mathbb{I}^{-1} g^T \nabla_g E)^V \quad . \tag{6.27}$$

To incorporate the constraints and ensure a parameter update that lies on the constraint manifold G', we simply replace $\nabla_g E$ with $\nabla_g E - \sum_j \lambda_j \nabla_g H_j$. Since the ODE (6.27) is on the Lie algebra which is a vector space, a simple integration scheme such

as Euler's method

$$\alpha_{k+1} = \alpha_k + t_{k+1}\dot{\alpha}$$

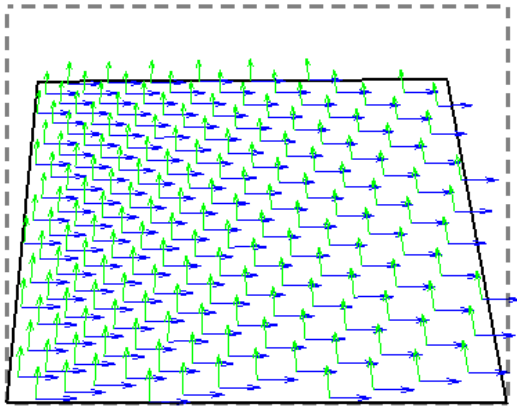
for small t_{k+1} suffices. In fact t_{k+1} may be selected by line search as a minimizer of the energy in the same manner as equation (3.30),

$$t_{k+1} = \arg\min_{0 \le t \le 1} E(g(\alpha_k + t\dot{\alpha}))$$

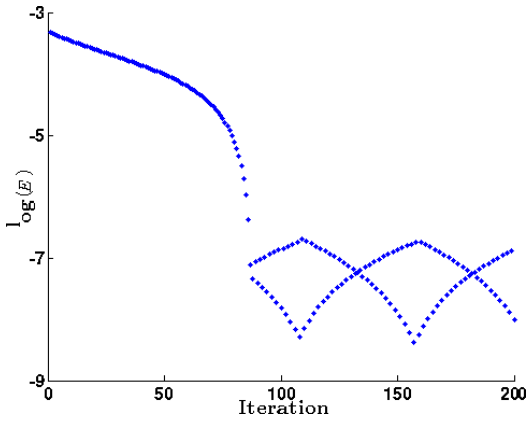
without fear of drifting off the manifold; the parameterization sees to it that even if the step size is too large for much confidence that the integration is following the flow on the Lie algebra generated by (6.27) very closely, the configuration updates g_k are at least confined to G.

6.4 Toy Example Revisited

To test the efficacy of the proposed algorithm we apply it to the model scene used in section 5.4. Although we have established that the gradient descent will work to minimize the divergence while maintaining the world-based constraints, a hurdle we still face in the noiseless case is the lack of a guarantee that the global minimium will be found. In practice as Figure 6.3 illustrates, the gradient descent is rapid but often falls prey to the wide basins of attraction typical of the local minima. We propose two methods to overcome this difficulty: a random perturbation to the parameters α_i from equation (6.24) once it is apparent that a local minimum has been located, or an initialization that puts the first iterate sufficiently close to the global minimum.



(a) World-frame reconstruction given the homography found at a local minimum.



(b) Descent into a local minimum. The branching pattern is due to the static step size with an $L1$-norm penalty function (5.21) which renders the energy surface non-smooth along local minima.

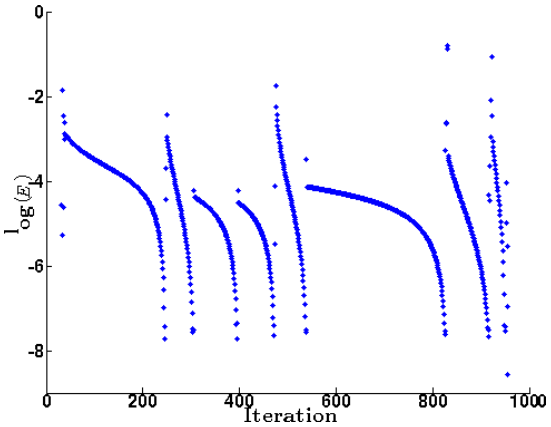
Figure 6.3: Example of a rapid gradient descent into a local minimum close to the global one. See Figures 5.1 and 5.2 for comparison.

The former method utilizes the fact that the true homography ensures

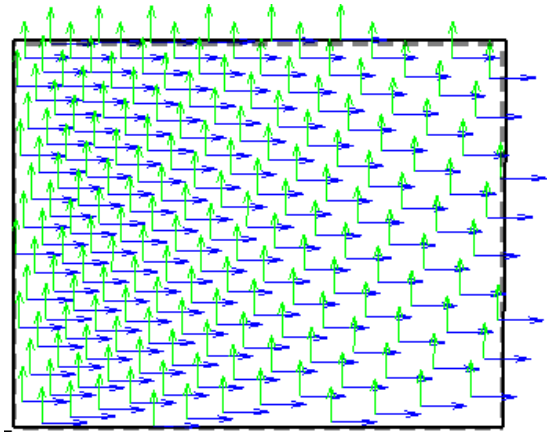
$$\int_{\mathcal{I}} \Psi(\chi) \left|\det D\phi^{-1}\right| \mathrm{d}\mathbf{x} = 0\,,$$

which cannot hold for any other homography that observes constraints (6.11) and (6.12). The condition $\int_{\mathcal{I}} \Psi(\chi) \left|\det D\phi^{-1}\right| \mathrm{d}\mathbf{x} > \epsilon$ for some suitable $\epsilon > 0$ is then symptomatic of a local minimum when further iterations fail to reduce the energy sufficiently. At this point the parameters α_i are randomly perturbed and the minimization procedure restarts. While the global minimum is achieved almost surely given sufficiently large perturbations, some of the performance gain with respect to the classical procedure in section 5.4 is given up due to the need for multiple runs. On the other hand, the latter method can be achieved by using the classical method as an initialization itself. This tends to result in extremely fast convergence to the

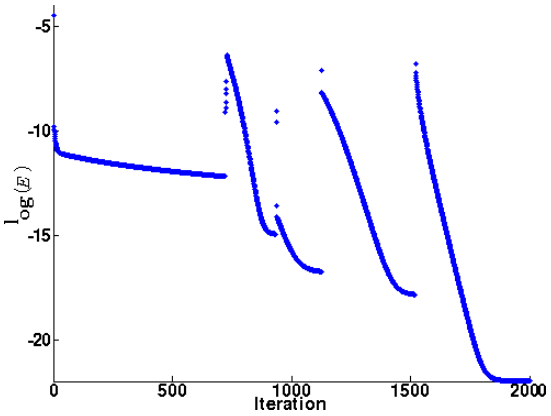
global minimum once the initialization has been performed (see Figure 6.4), but as depicted in Figure 5.2, that can be a somewhat delicate procedure requiring at least an order of magnitude more iterations for a comparable accuracy. A hybrid approach could be adopted where the minimization on $SL(3)$ is initialized with an homography only coarsely estimated via the classical method, though we have favored sole use of the random perturbation method for its greater robustness, particularly given our observation that noise in the real-world examples of chapter 7 often obviates the need for any multiple-starting at all. An average of around 10^3 iterations was sufficient to arrive at a typical error of 10^{-8} in each entry of g but for column g_2 for which the error was on the order of 10^{-3}.



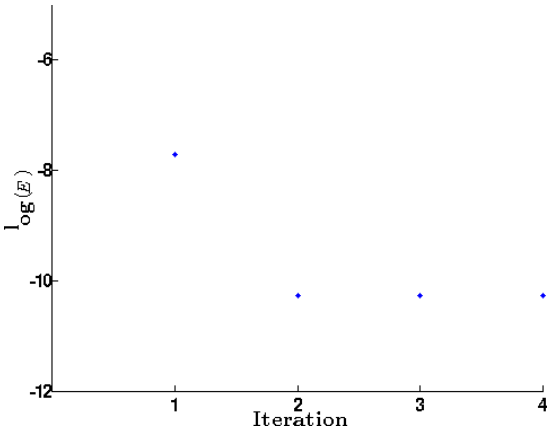
(a) Global minimum achieved via random perturbations of the parameters from local minima, using penalty function (5.21).



(b) The resulting world frame reconstruction showing a close match to the true (dashed) boundary.



(c) Example of the random perturbation method when the penalty function is $\Psi(t) = t^2$.



(d) Only a single iteration was needed for convergence to the global minimum in this example of an initialization with the homography produced by the classical DLT followed by the minimization procedure described in section 5.3.

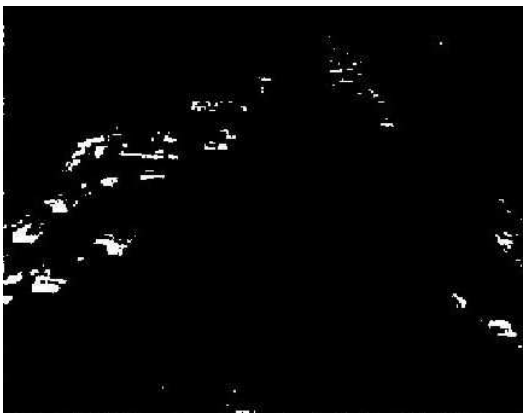
Figure 6.4: Global minimization of the energy via the random perturbation and classical-DLT-initialized methods.

Chapter 7

Traffic Streaming Application

The image vector fields $\boldsymbol{v}$ and $\boldsymbol{w}$ representing respectively the traffic flow and its transverse direction are critical inputs into the energy minimization approach to homography estimation. To establish the algorithm we have assumed the use of these fields, though in practice they need to be estimated. The methodology described in this chapter was arrived at after substantial experimentation with leading image processing techniques. As underpins the rationale for this homography estimation method as a whole, the techniques settled upon needed to exhibit robustness to varied image characteristics such as lighting and contrast.

(a) Typical highway scene video frame



(b) Unsatisfactory foreground decomposition.

Figure 7.1: Example segmentation via subtraction of a mean background image and variance thresholding. The segmentations assume local pixel independence, leading to high fragmentation, while noise produces spurious objects.

7.1 Vector Field Estimation

Both flow and transverse vector fields require the vehicles to be segmented in each frame of the video stream. This is often achieved by computing a background image, often a rolling mean image of the recent frames, and subtracting that from the current frame. The vehicles are identified as the high absolute intensity regions of the difference. While simple and fast, this method (and its extensions to a lesser degree) suffers from a relatively high level of noise (see Figure 7.3). This can be partly combatted with erosion/dilation operations and the dismissal of foreground objects that appear too small, though we found it difficult to attain robust segmentations in this manner. Instead, we implemented the Inexact Augmented Lagrangian Multiplier method ([18]) to separate from a stack of successive frames a low rank component and a sparse component. The former yields the stable, common element to each

(a) The frame to be decomposed into background and foreground elements.

(b) The background after applying the IALM method to 50 successive frames.

(c) The foreground objects, easily segmented by applying an intensity threshold.

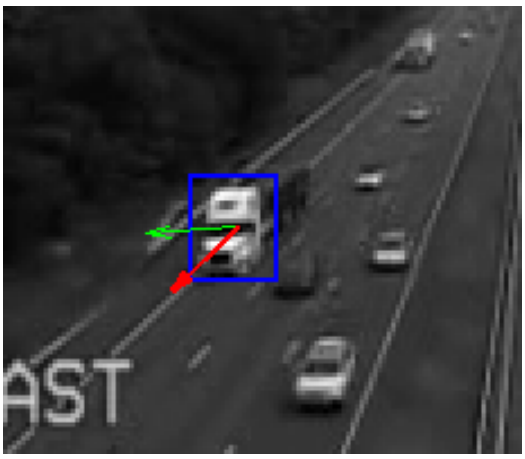
Figure 7.2: An example of vehicle segmentation with the Inexact Augmented Lagrangian Multiplier method.

frame which is immediately identified as the background while the latter picks up the transient elements which are almost always the vehicles as shown in Figure 7.2. This technique proved to be vastly superior to background subtraction, offering clean vehicle silhouettes while remaining impervious to changes in lighting conditions that were not too abrupt.

Each foreground object was used as a filter to convolve on the surrounding region of the previous frame. The position of the largest value in the convolution indicates where the object was located in that frame; the collection of all such displacement vectors over all objects and frames is precisely the traffic flow vector field $\boldsymbol{v}$. On the other hand, the transverse vector field $\boldsymbol{w}$ uses the spatial derivatives $\nabla I(\boldsymbol{x})$ over the object region $\mathcal{R}$ to build the structure tensor

$$\omega(\mathcal{R}) \begin{bmatrix} \int_{\mathcal{R}} \left(\partial_x I(\boldsymbol{r})\right)^2 d\boldsymbol{r} & \int_{\mathcal{R}} \partial_x I(\boldsymbol{r}) \partial_y I(\boldsymbol{r}) d\boldsymbol{r} \\ \int_{\mathcal{R}} \partial_x I(\boldsymbol{r}) \partial_y I(\boldsymbol{r}) d\boldsymbol{r} & \int_{\mathcal{R}} \left(\partial_y I(\boldsymbol{r})\right)^2 d\boldsymbol{r} \end{bmatrix} \tag{7.1}$$

where $\boldsymbol{r} \in \mathcal{R}$ and $\omega(\mathcal{R})$ is a normalization constant which we simply took as $1/\int_{\mathcal{R}} d\boldsymbol{r}$.



(a) A bounding box for the foreground object is used to isolate the region used to calculate the flow (red) and transverse (green) vectors.

(b) The flow (blue) and transverse (green) vectors after 50 frames.

Figure 7.3: The vector field construction process.

The largest eigenvalue of this matrix is associated with the orientation maximally aligned with the gradient over the object. Because the video streams are of traffic that is predominantly approaching or receding, the dominant lines on each vehicle image are those defining the windscreen, roof and chassis which in turn cause a large image gradient across these features. The eigenvector associated with the smallest eigenvalue then aligns itself *with* these features, effectively marking out a direction orthogonal to that of the traffic flow. The vector $\boldsymbol{w}(\boldsymbol{x})$ is exactly this eigenvector computed at the object centroid $\boldsymbol{x}$. The image derivatives are calculated by convolving the image of the object with the filter set described in [39]. After pruning the outlying vectors, those at an orientation significantly different to the local mean, the full fields were constructed as linear interpolations of the surviving vectors. This result is illustrated in Figure 7.3. It has been well established ([27]) that data normalization

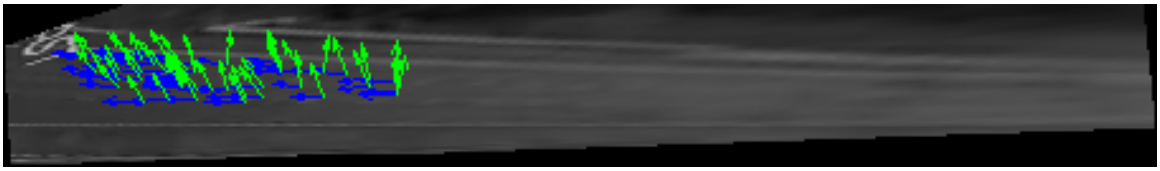
is an essential preprocessing step for the DLT due to the fact that the algorithm is not invariant to the units used. This remains true for the proposed approach; the image coordinates are normalized to the range [-1,1] in each dimension and the world origin $\boldsymbol{X}_0$ is taken to be the point at the image centroid. The vectors $\boldsymbol{V}_0$ and $\boldsymbol{W}_0$ are also coupled to the image vector fields at the centroid, with $\boldsymbol{W}_0$ scaled to make the ratio $\|\boldsymbol{V}_0\| \,/\, \|\boldsymbol{W}_0\|$ equal to $\|\boldsymbol{v}_0\| \,/\, \|\boldsymbol{w}_0\|$ provided the pixels are square. Further, the transverse vector field $\boldsymbol{w}$, for which $\|\boldsymbol{w}(\boldsymbol{x})\| = 1$ as a consequence of its construction, is scaled at each point so that $\|\boldsymbol{w}\| = \|\boldsymbol{v}\| \, \|\boldsymbol{w}_0\| \,/\, \|\boldsymbol{v}_0\|$. This serves to prevent an overweighting of any deviation from orthogonality of vectors $\boldsymbol{w}(\boldsymbol{x})$ and $\boldsymbol{v}(\boldsymbol{x})$ far from the camera, and although we have experimented with a modification to the energy functional that accounts for a normalization in vector lengths[1], the energy minimization is nevertheless better conditioned when the scaling is employed.

7.2 Initial Results

Initializing with the identity, we apply the system of equations (5.6) and (6.17) at the world origin $\boldsymbol{X}_0$ to establish the homography parameters associated with the similarity (and skew) component of the transformation between image and world frames. The resulting matrix $g_0^{'}$ may not lie in $SL(3)$ and is modified by inverting the relation (6.24) to find the initial coordinates α_i as the least-squares solution to

[1]Following the innovation as that described by [40] in the context of gradient vector diffusion for image segmentation. The same modification (but applied to the image gradient) appears in equation (7.9).

(a) The rectified image shown in Figure 7.3 after 50 gradient descent iterations.



(b) The flow and transverse vectors imposed on the rectified background image.

Figure 7.4: Transforming to the world frame via the solution homography.

the linear set of equations

$$\sum_{i=1}^{8} \alpha_i A_i = \log(g_0^{'}) \, ,$$

using the matrix logarithm operator. The initial iterate is then $g_0 = \exp(\sum_{i=1}^{8} \alpha_i A_i)$ and has already accounted for a large reduction in the energy (6.9); the remaining three degrees of freedom are resolved by the energy minimization. Setting the orthogonality penalty constant β to 10, the rectification achieved by the solution after 50 iterations is shown in Figure 7.4. The convergence of the algorithm, shown in Figure 7.5, is fast and robust to the considerable noise in the flow and transverse vectors, though it is evident that there is some systematic bias in the estimation of the latter, a result of the tendency for the transverse vectors to include a component along the flow the closer the vehicles are to the camera and the greater the visibility of their lateral lines. This affects the degree to which the perspective effect is

countered: some residual foreshortening is clearly visible. We can marginally improve on this result by incorporating an optical flow term into the energy functional (§7.4) and introducing an horizon constraint into the initialization (§7.3).

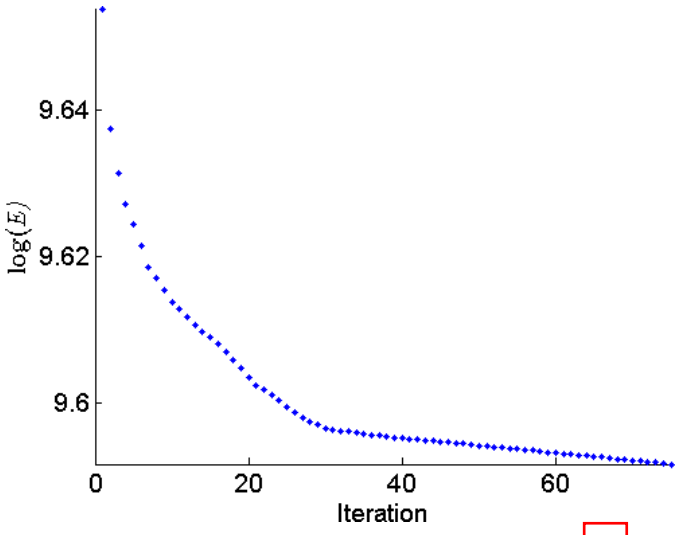


Figure 7.5: Convergence of the energy (6.9)

Nevertheless, the solution as it stands largely fulfills our objective: operating on the world frame rather than that of the image we are able to measure with some accuracy the various distances, areas and velocities of interest in the road scene. Moreover, we found little utility in randomly perturbing the parameters in order to overcome the convergence to a local mininum; the presence of noise appears to enlargen the basin of attraction to the global minimum.

7.3 Straight Road Initialization

The road segment in many scenes of interest may be well described, at least locally, as straight. One of the strengths of the divergence-based energy minimization technique for homography estimation is its indifference to the direction and curvature of the road as long as it lies on a plane. If the road is known to be straight however,

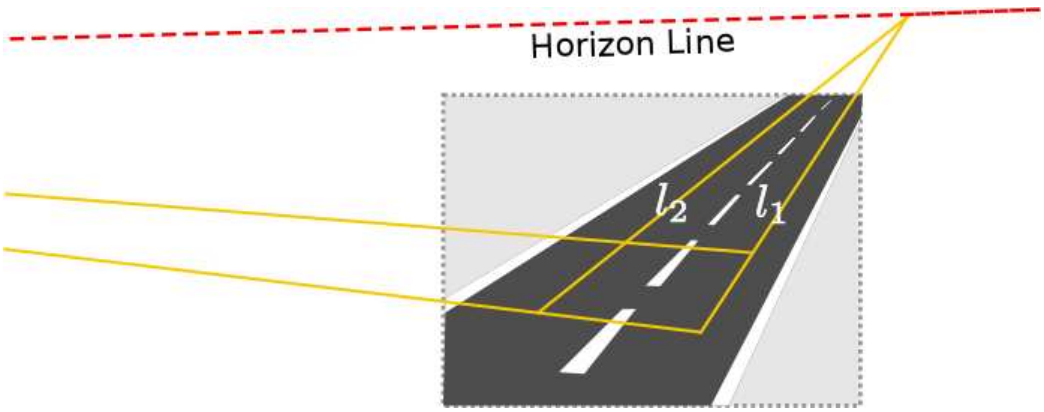


Figure 7.6: The line at infinity $\boldsymbol{l}_\infty$ as the bisection of points where the respective two sets of world-parallel lines intersect in the image.

the homography initialization can be improved by incorporating a horizon constraint that is worked in with the initial initialization (6.17) in an iterative fashion.

As mentioned in Section 4.1.2 and depicted in Figure 7.6, if lines $\boldsymbol{l}_1$ and $\boldsymbol{l}_2$ in the image trace out vehicle trajectories that are parallel in the world, they intersect at a point on the line at infinity $\boldsymbol{l}_\infty$ that is commonly identified as the horizon. Using equations (4.2) and (4.10) this is described

$$\left(g^T\boldsymbol{l}_1\right) \times \left(g^T\boldsymbol{l}_2\right) = \boldsymbol{L}_1 \times \boldsymbol{L}_2 = (b, -a, 0)^T \tag{7.2}$$

for the parameters a, b that orient the line at infinity. The constraint we exploit is the third equation in (7.2), $(\boldsymbol{L}_1 \times \boldsymbol{L}_2)_3 = 0$, the idea being to solve for the homography parameter g_{31} (or g_{32}) in terms of the other parameters.

While a solution can be derived by manipulating the constraint equation directly, it is easier to start with the following observation: For matrix $B_3(\boldsymbol{x})$ from (4.22), evaluated at any point $\boldsymbol{x}_\infty$ in the image that lies on $\boldsymbol{l}_\infty$, $|B_3(\boldsymbol{x}_\infty)| = 0$. This is

because $B_3(\boldsymbol{x}_\infty)$ is a rank-two matrix as follows from equation (4.8),

$$\begin{aligned} |B_3(\boldsymbol{x}_\infty)| &= \left| \begin{bmatrix} g_1 & g_2 & \overline{\boldsymbol{x}}_\infty \end{bmatrix} \right| \\ &= \left| \begin{bmatrix} g_1 & g_2 & g\overline{\boldsymbol{X}}_\infty \end{bmatrix} \right| \\ &= \left| \begin{bmatrix} g_1 & g_2 & g_1 b - g_1 a \end{bmatrix} \right| \\ &= 0 \end{aligned}$$

for the parameters a, b defining the point at infinity $\boldsymbol{X}_\infty$. Now using the Laplace expansion of the determinant,

$$|B_3(\boldsymbol{x}_\infty)| = g_{31}\mathrm{Cof}(B_3(\boldsymbol{x}_\infty))_{31} + g_{32}\mathrm{Cof}(B_3(\boldsymbol{x}_\infty))_{32} + (\boldsymbol{x}_\infty)_3\mathrm{Cof}(B_3(\boldsymbol{x}_\infty))_{33}$$

so that by rearranging and recognizing $\boldsymbol{l}_1 \times \boldsymbol{l}_2$ as a point on $\boldsymbol{l}_\infty$,

$$g_{31} = -\frac{g_{32}\mathrm{Cof}(B_3(\boldsymbol{l}_1 \times \boldsymbol{l}_2))_{32} + (\boldsymbol{l}_1 \times \boldsymbol{l}_2)_3\mathrm{Cof}(B_3(\boldsymbol{l}_1 \times \boldsymbol{l}_2))_{33}}{\mathrm{Cof}(B_3(\boldsymbol{l}_1 \times \boldsymbol{l}_2))_{31}} \,. \tag{7.3}$$

Incorporating the solution g_{31} into the initial homography iterate may violate the rotational and scaling constraints that were built into the initialization (6.17). For this reason, equations (7.3) and (6.17) are applied in an alternating manner for a few iterations. The energy minimization is then started once the initial matrix g_0 reaches an equilibrium, typically after 20 or so iterations.

7.4 Incorporating Optical Flow

The computation of the energy functional in section 6.1 assumes that the vector fields $\boldsymbol{v}$ and $\boldsymbol{w}$ on the image are static. While that has been the case thus far,

we can take advantage of the variational formulation under which optical flow is derived to incorporate it into our energy functional and iteratively improve the vector field estimates alongside the homography parameters. We benefit by this approach in that each local vector update is guided by the global information inherent in the homography, and on the other hand the resultant more accurate vector field should improve the convergence rate of the homography estimate. Optical flow methods estimate the motion in an image I_t at time t as the vector field v for which $I_{t+\Delta t}(\boldsymbol{x}+\boldsymbol{v}) = I_t(\boldsymbol{x})$ or equivalently, $I_t(\boldsymbol{x}-\boldsymbol{v}) = I_{t+\Delta t}(\boldsymbol{x})$. Linearizing this relation gives rise to the standard optical flow data term $\partial I/\partial t + \nabla I \cdot \boldsymbol{v}$ to be minimized over all fields $\boldsymbol{v}$ in a variational manner. Following the modern recommended practices listed in [41], the linearization has been 'delayed' so that the term to be minimized is

$$\Psi\left(I_t(\boldsymbol{x}-\boldsymbol{v}) - I_{t+1}(\boldsymbol{x})\right) \tag{7.4}$$

where t indexes the image frame number and rather than a standard quadratic penalty as in [42], a 'robust' penalization is achieved through the function Ψ, given by (5.21). We thus extend the energy expression (6.9) to

$$\boxed{E = \int_{\mathcal{I}} \Psi\left(I_t(\boldsymbol{x}-\boldsymbol{v}) - I_{t+1}(\boldsymbol{x})\right) \mathrm{d}\mathbf{x} + \alpha \int_{\mathcal{W}} \Psi\left(\operatorname{div} \boldsymbol{V}\right) \mathrm{d}\mathbf{X} + \beta \int_{\mathcal{W}} \Psi\left(\boldsymbol{V}\cdot\boldsymbol{W}\right) \mathrm{d}\mathbf{X}}. \tag{7.5}$$

Minization of the optical flow data term (7.4) is not by itself sufficient in the identification of a unique vector field. The well-known *aperture problem* highlights the ambiguity of the vector components perpendicular to the normal of an image boundary: given an edge that translates while keeping its orientation constant between

two successive frames, we cannot tell whether it has also moved in the direction of its alignment. To combat this a smoothness term is adopted, often something like $\Psi\left(\text{trace}\left(D\boldsymbol{v}D\boldsymbol{v}^T\right)\right)$, the addition of which both achieves uniqueness and encourages smoothness in the vector field. Because the divergence is not a local quantity, as shown below in equation (7.8) the second term in (7.5) cannot perform this regularization role. One option is to add an additional smoothness term. This can be done in a way that promotes non-isotropic diffusion, an important characteristic for a traffic scene where vehicles in neighboring lanes may travel at different speeds so that smoothing across lane boundaries is undesirable. However, it also increases substantially the complexity of the optical flow implementation and energy minimization. For this reason we fall back on the third term of (7.5) to provide a local regularization of the flow vectors as well as the metric rectification.

Pulling back the energy terms originating in the world frame of reference to that of the image, (7.5) becomes

$$\begin{aligned} E &= \int_{\mathcal{I}} \Psi\left(I_t\left(\boldsymbol{x}-\boldsymbol{v}\right)-I_{t+1}\left(\boldsymbol{x}\right)\right)\mathrm{d}\mathbf{x}+\alpha\int_{\mathcal{I}}(\phi^{-1})^{*}\Psi\left(\operatorname{div}\boldsymbol{V}\right)f\mathrm{d}\mathbf{x}+\beta\int_{\mathcal{I}}(\phi^{-1})^{*}\Psi\left(\boldsymbol{V}\cdot\boldsymbol{W}\right)f\mathrm{d}\mathbf{x} \\ &= \int_{\mathcal{I}} \Psi\left(I_t\left(\boldsymbol{x}-\boldsymbol{v}\right)-I_{t+1}\left(\boldsymbol{x}\right)\right)\mathrm{d}\mathbf{x}+\int_{\mathcal{I}}\left(\alpha\Psi\left((\phi^{-1})^{*}\operatorname{div}\boldsymbol{V}\right)+\beta\Psi\left(D\phi^{-1}\boldsymbol{v}\cdot D\phi^{-1}\boldsymbol{w}\right)\right)f\mathrm{d}\mathbf{x} \\ &= \int_{\mathcal{I}} \Psi\left(I_t\left(\boldsymbol{x}-\boldsymbol{v}\right)-I_{t+1}\left(\boldsymbol{x}\right)\right)+\alpha f\Psi\left(\operatorname{div}(\boldsymbol{v})+\rho\cdot\boldsymbol{v}\right)+\beta f\Psi\left(\boldsymbol{v}^T\left(D\phi^{-1}\right)^T D\phi^{-1}\boldsymbol{w}\right)\mathrm{d}\mathbf{x} \quad . \end{aligned} \tag{7.6}$$

We denote by φ a small perturbation in $\boldsymbol{v}$ that vanishes on the boundary of the image

$\partial\mathcal{I}$. Proceeding to take the variational derivative of the first term in (7.6),

$$\begin{aligned}
\delta E_1 &= \int_{\mathcal{I}} \left.\frac{\partial}{\partial\epsilon}\right|_{\epsilon\to 0} \Psi\left(I_t\left(\boldsymbol{x}-\boldsymbol{v}-\epsilon\varphi\right)-I_{t+1}\left(\boldsymbol{x}\right)\right)\mathrm{d}\mathbf{x} \\
&= -\int_{\mathcal{I}} \Psi'\left(I_t\left(\boldsymbol{x}-\boldsymbol{v}-\epsilon\varphi\right)-I_{t+1}\left(\boldsymbol{x}\right)\right)\nabla I_t(\boldsymbol{x}-\boldsymbol{v}-\epsilon\varphi)\cdot\varphi\mathrm{d}\mathbf{x}\Big|_{\epsilon\to 0} \\
&= -\int_{\mathcal{I}} \Psi'\left(I_t\left(\boldsymbol{x}-\boldsymbol{v}\right)-I_{t+1}\left(\boldsymbol{x}\right)\right)\nabla I_t\left(\boldsymbol{x}-\boldsymbol{v}\right)\cdot\varphi\mathrm{d}\mathbf{x}
\end{aligned}$$

while similarly for the second and third terms, the linearity of the divergence and inner product allow us to write

$$\begin{aligned}
\delta E_2 &= \alpha\int_{\mathcal{I}} f\Psi'\left(\mathrm{div}(\boldsymbol{v})+\rho\cdot\boldsymbol{v}\right)\left(\mathrm{div}(\varphi)+\rho\cdot\varphi\right)\mathrm{d}\mathbf{x} \qquad \text{and} \qquad (7.7)\\
\delta E_3 &= \beta\int_{\mathcal{I}} f\Psi'\left(\boldsymbol{v}^T\left(D\phi^{-1}\right)^T D\phi^{-1}\boldsymbol{w}\right)\left(\varphi\cdot\left(D\phi^{-1}\right)^T D\phi^{-1}\boldsymbol{w}\right)\mathrm{d}\mathbf{x} \quad .
\end{aligned}$$

To ease notation we introduce the vector field $\mu(\boldsymbol{w})$ and scalar field $\kappa(\boldsymbol{v})$:

$$\begin{aligned}
\mu(\boldsymbol{w}) &= \left(D\phi^{-1}\right)^T D\phi^{-1}\boldsymbol{w} \quad , \\
\kappa(\boldsymbol{v}) &= \mathrm{div}(\boldsymbol{v})+\rho\cdot\boldsymbol{v} \ ,
\end{aligned}$$

recognizing that $\boldsymbol{v}^T\mu(\boldsymbol{w}) = \boldsymbol{w}^T\mu(\boldsymbol{v}) = \chi$. In a similar application of the steps between expressions (6.4) and (6.6) , we have

$$\begin{aligned}
\int_{\mathcal{I}} f\Psi'\left(\kappa(\boldsymbol{v})\right)\mathrm{div}(\varphi)\mathrm{d}\mathbf{x} &= \int_{\mathcal{I}} \mathrm{div}\left(f\Psi'\left(\kappa(\boldsymbol{v})\right)\varphi\right)-\nabla\left(f\Psi'\left(\kappa(\boldsymbol{v})\right)\right)\cdot\varphi\mathrm{d}\mathbf{x} \\
&= \int_{\partial\mathcal{I}} f\Psi'\left(\kappa(\boldsymbol{v})\right)\varphi\cdot n\mathrm{d}l - \int_{\mathcal{I}} \nabla\left(f\Psi'\left(\kappa(\boldsymbol{v})\right)\right)\cdot\varphi\mathrm{d}\mathbf{x} \\
&= -\int_{\mathcal{I}} \left(f\nabla\Psi'\left(\kappa(\boldsymbol{v})\right)+\Psi'\left(\kappa(\boldsymbol{v})\right)\nabla f\right)\cdot\varphi\mathrm{d}\mathbf{x}
\end{aligned}$$

as φ vanishes on the boundary. Substituting this into (7.7) gives

$$\begin{aligned}\delta E_2 &= \alpha \int_{\mathcal{I}} \left(\Psi'\left(\kappa(\boldsymbol{v})\right)\left(f\rho - \nabla f\right) - f\nabla\Psi'\left(\kappa(\boldsymbol{v})\right)\right)\cdot\varphi \mathrm{d}\mathbf{x} \\ &= -\alpha \int_{\mathcal{I}} f\nabla\Psi'\left(\kappa(\boldsymbol{v})\right)\cdot\varphi \mathrm{d}\mathbf{x} \\ &= -\alpha \int_{\mathcal{I}} f\Psi''\left(\kappa(\boldsymbol{v})\right)\nabla\left(\kappa(\boldsymbol{v})\right)\cdot\varphi \mathrm{d}\mathbf{x} \\ &= 0 \end{aligned} \tag{7.8}$$

given our choice of penalty function (5.21). This lack of a contribution to the total energy from a vector field perturbation is a direct consequence of the non-local nature of the divergence. The forced boundary condition $\varphi = 0$ on $\partial\mathcal{I}$ means by virtue of the divergence theorem that the energy associated with the flow divergence remains independent of φ in the image interior. Piecing the terms back together,

$$\delta E = -\int_{\mathcal{I}} \left(\Psi'\left(I_t\left(\boldsymbol{x}-\boldsymbol{v}\right) - I_{t+1}\left(\boldsymbol{x}\right)\right)\nabla I_t\left(\boldsymbol{x}-\boldsymbol{v}\right) - \beta f\Psi'\left(\chi\right)\mu(\boldsymbol{w})\right)\cdot\varphi \mathrm{d}\mathbf{x}$$

from which we make the usual observation that a non-positive change in the energy is guaranteed by the ODE

$$\dot{\boldsymbol{v}} = \Psi'\left(I_t\left(\boldsymbol{x}-\boldsymbol{v}\right) - I_{t+1}\left(\boldsymbol{x}\right)\right)\nabla I_t\left(\boldsymbol{x}-\boldsymbol{v}\right) - \beta f\Psi'\left(\chi\right)\mu(\boldsymbol{w}) \quad .$$

Also in keeping with standard practice, we modify the gradient descent equation to

$$\dot{\boldsymbol{v}} = \Psi'\left(I_t\left(\boldsymbol{x}-\boldsymbol{v}\right) - I_{t+1}\left(\boldsymbol{x}\right)\right)\frac{\nabla I_t\left(\boldsymbol{x}-\boldsymbol{v}\right)}{\left\|\nabla I_t\left(\boldsymbol{x}-\boldsymbol{v}\right)\right\|} - \beta f\Psi'\left(\chi\right)\mu(\boldsymbol{w}) \tag{7.9}$$

in order that spuriously large gradients caused by noise in the image do not adversely affect the minimization. The vectors $\boldsymbol{w}$, as mentioned in section 7.1, are typically

normalized to reflect the same local length scale in the image as $\boldsymbol{v}$. We note that due to the subsitution in the energy of the usual optical flow smoothness term with that derived from the vectors $\boldsymbol{w}$, the gradient descent (7.9) is particularly easy to implement as all the vectors $\boldsymbol{v}$ are independent of one another. There is no need of a finite difference scheme over grid points with interpolated values for instance, but rather it is just the values at the pixels of interest that need to be accessed.

Performing gradient descent over the vector field $\boldsymbol{v}$ in this manner assumes that the transverse field $\boldsymbol{w}$ is fixed. As observed in Figure 7.3 however, $\boldsymbol{w}$ can exhibit a significant amount of noise. In fact it is quite typical as the vehicles draw nearer to the camera for systematic inaccuracies to appear: vehicle edges parallel to the traffic flow such as hood and roof lines become more apparent, leading to a bias in the flow direction of the eigenvector of the structure tensor (7.1) corresponding to the smallest eigenvalue. Rather than refining our image processing of the segmented vehicles we can simply apply the same variational procedure to $\boldsymbol{w}$ while keeping $\boldsymbol{v}$ fixed. The predicate here is that, at least in an average sense over the vehicle image, $I_t(\boldsymbol{x}+\boldsymbol{w}) = I_t(\boldsymbol{x})$. Working through the same process as that of the optical flow data term we arrive at the counterpart to gradient descent equation (7.9),

$$\dot{\boldsymbol{w}} = -\Psi'\left(I_t(\boldsymbol{x}+\boldsymbol{w}) - I_t(\boldsymbol{x})\right)\frac{\nabla I_t(\boldsymbol{x}+\boldsymbol{w})}{\|\nabla I_t(\boldsymbol{x}+\boldsymbol{w})\|} - \beta f \Psi'(\chi)\,\mu(\boldsymbol{v})\,. \tag{7.10}$$

(a) The rectified flow and transverse vectors when updated according to equations (7.9) and (7.10).

(b) A primitive vehicle tracking methodology applied to the rectified frame.

Figure 7.7: Rectification results once optical flow and straight road initialization is incorporated.

7.5 Improved Results

The spatial derivatives ∇I_t, just as in Section 7.1, were calculated using the optimized derivative filters described in [39]. Updating g via its coordinates α_i along with the vector fields $\boldsymbol{v}$ and $\boldsymbol{w}$ each iteration and using the straight road initialization when appropriate, the solution estimate $\hat{g}$ after 50 iterations is markedly improved (see Figure 7.7a) while maintaining a comparable speed of convergence given that the algorithm must now be able to access a lower energy state (Figure 7.8). Transforming the streaming video frames by $\hat{g}$ we use a simple rolling background subtraction method to identify the rectified vehicles that have been highlighted with bounding boxes in Figure 7.7b. This tracking methodology is primitive, making no use of the segmentations available in the last frame coupled with the learned vehicle flow field to enhance the current segmentations, nor any color information to more accurately distinguish a dark vehicle from the road surface. Such enhancements would certainly help eliminate the faults visible in the frame - a single vehicle identified

as two objects and the inability to distinguish a low-intensity vehicle - but we can nevertheless infer many quantities of interest, even with such a simple method. Comparing the sizes of the objects measured in the rectified frame yields a basic classification of vehicle type between all vehicles of class three or below (cars and light trucks), class four to seven (medium to heavy duty trucks) and class eight (tractor trailers). The same classification could be learned from the segmentation of the raw images but only within a narrow window around a single point in the image unless a model is learned that describes the deformation of the objects as they approach or recede. Much more difficult on the raw images is the measurement of distance between vehicles in order to classify the congestion level, the measurement of traffic velocity, identification of vehicle type when they are distant or the classification of common vehicle trajectories such as the merging from one lane to another, all of which are readily achieved once the image is rectified.

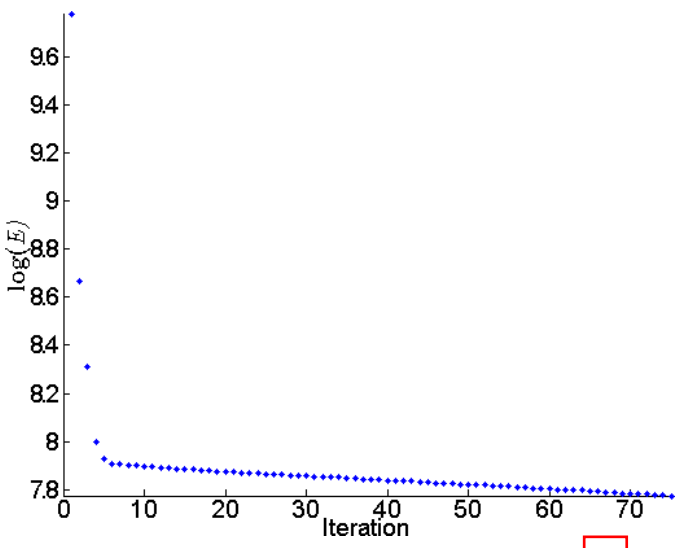


Figure 7.8: Convergence of the energy (7.5)

To get a sense of the robustness of the algorithm we compiled a suite of 10 different scenes for which the homography was to be estimated (Figure 7.10). The results

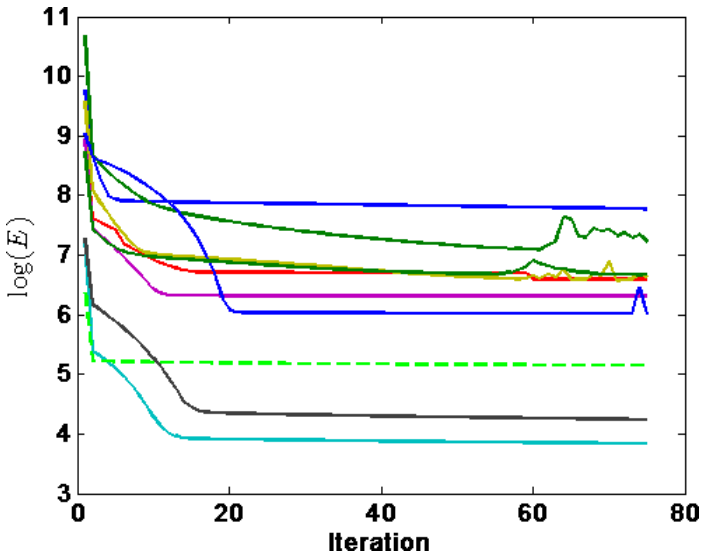
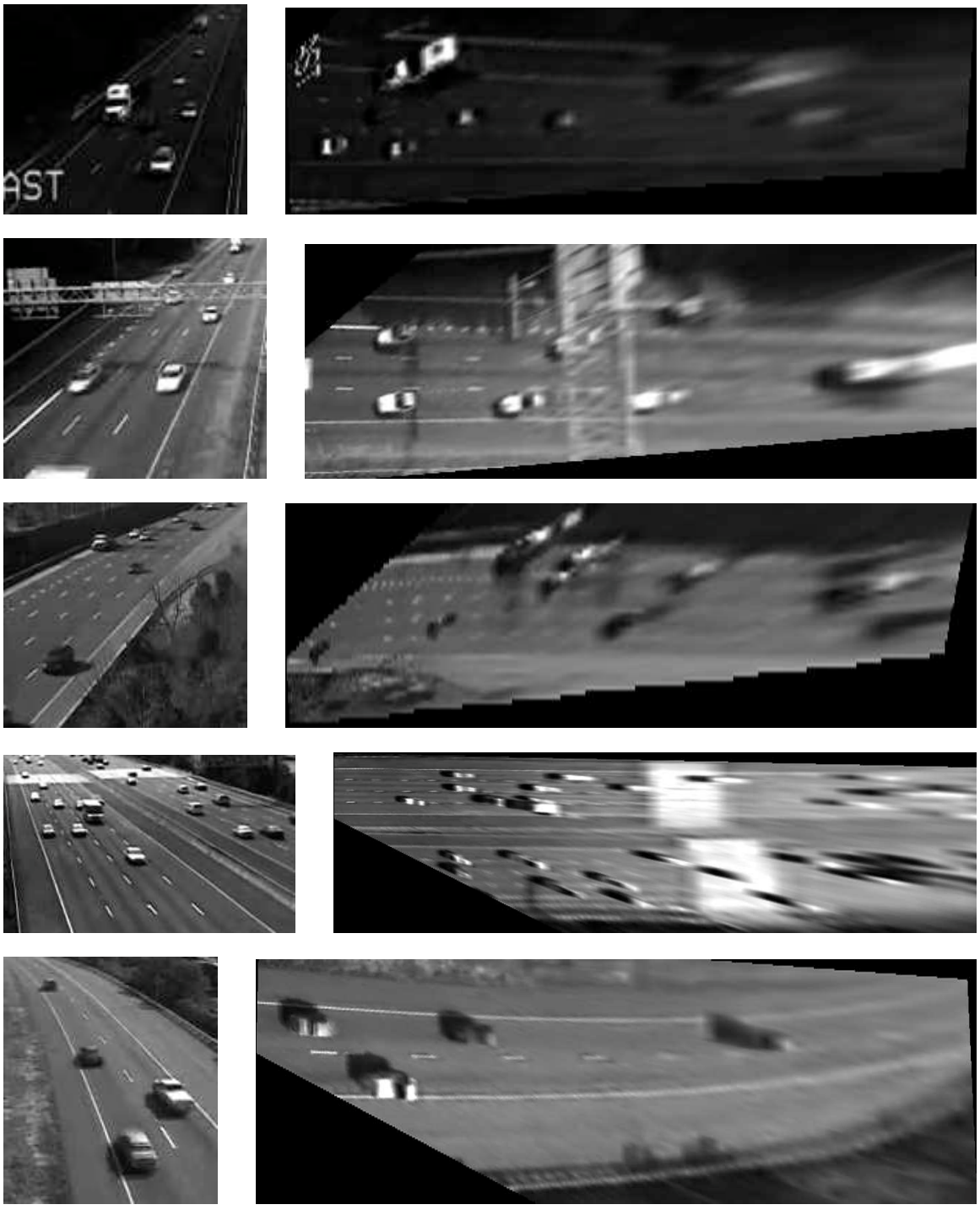


Figure 7.9: Convergence of the test suite.

are qualitatively pleasing: most scenes have been well rectified even when the road exhibits some curvature. Convergence is also fast, occuring within 20 iterations for 7 out of the 10 example video streams.

A notable exception is the last example (corresponding to the dashed line in Figure 7.9) that appears to have converged particularly quickly but in fact has not successfully rectified the perspective in the image. This is due to a violation in the scene of our fundamental assumption of a divergence-free traffic flow. The vehicles merging from the on-ramp were included in the estimation of the flow vector field which then must exhibit a non-zero divergence. This illustrates the attention that must be given to scenes where vehicles in different lanes need not travel parallel to one another; any image region displaying this characteristic should be excluded.


AST

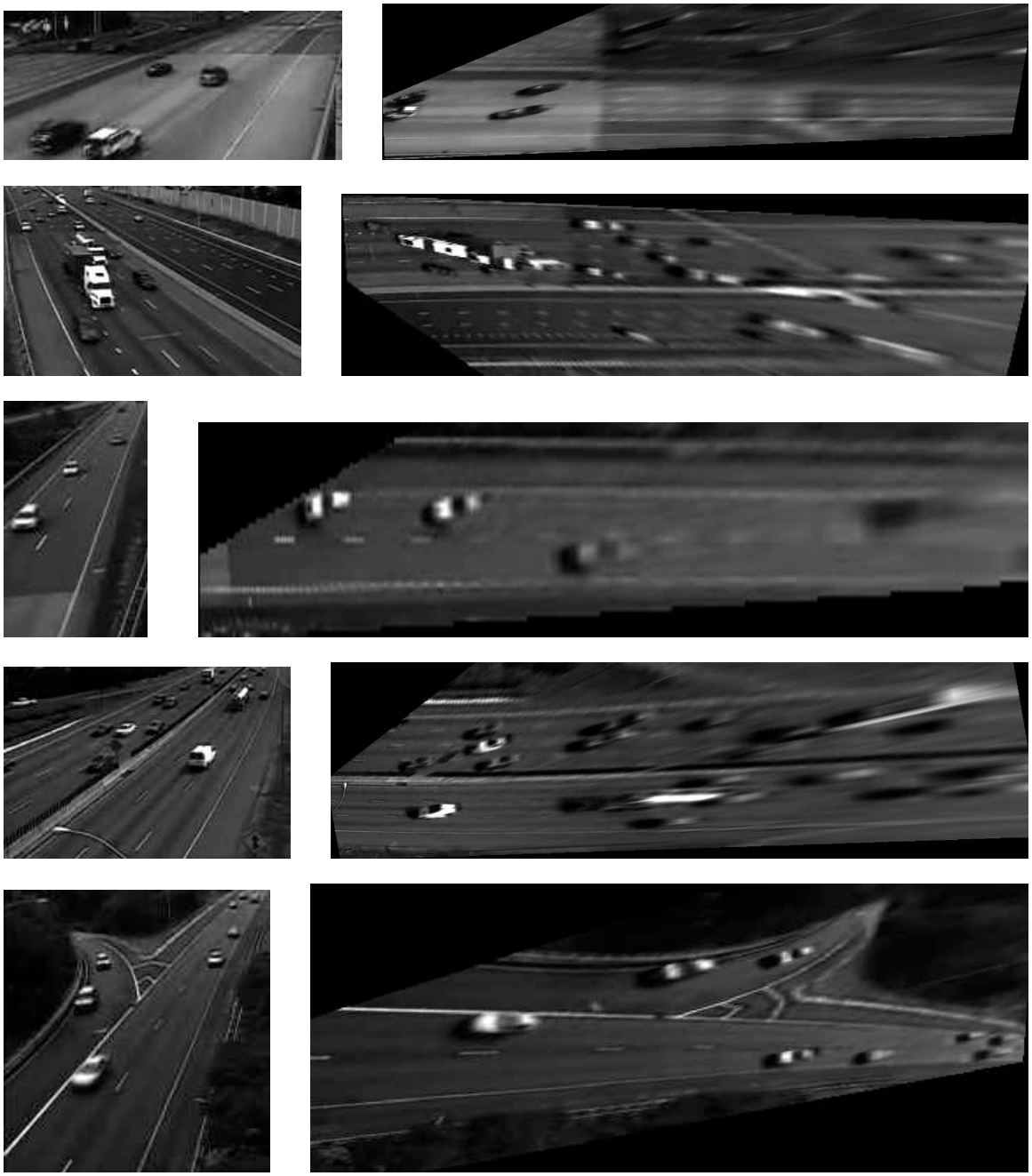


Figure 7.10: Rectification results once optical flow and straight road initialization is incorporated.

Chapter 8

Conclusion and Future Work

8.1 Object Height Inference

While the estimation of the planar homography allows measurement of many useful quantities, they are all restricted to what can be observed on the road's surface. The speed of a given vehicle is attainable for instance, but its height is not. This can detract from the accuracy with which vehicle classification or tracking (as exemplified in Figure 7.7) is performed: the projection of the vehicle image onto the road plane is certainly a unique attribute for that vehicle's class that can be categorized, but comparison with a template is more ambiguous. Moreover, the more aligned the camera's z-axis with the world plane and the further away the vehicle, the more pronounced the height-induced distortion, an undesirable feature for a representation that aims at conserving all measurable quantities at any point in the image. When

K, the camera intrinsic matrix (4.16) is known, the problem of finding the true world coordinates (X, Y, Z) for any trackable point on a moving object in the image boils down to a linear system of equations that is easily solved. More difficult is the case with unknown K, yet this would be a natural extension to the objectives achieved in this thesis. We outline a possible linear iterative approach and, in the spirit of the thesis' central tenet, suggest that it may more robustly be achieved by minimization of an extended energy functional.

The full 3×4 projection matrix $P = K \left[\begin{array}{c|c} R & t \end{array} \right]$ linearly transforms a 3D homogeneous world coordinate $\overline{\boldsymbol{X}} \sim (X, Y, Z, 1)^T$ to image coordinate $\overline{\boldsymbol{x}} \sim (x, y, 1)^T$ as $\overline{\boldsymbol{x}} = P\overline{\boldsymbol{X}}$, where P is known only up to scale. The homography g, for which all world points are assumed to lie on the plane $Z = 0$, is recovered as the 3×3 matrix $K \left[\begin{array}{cc|c} R_1 & R_2 & t \end{array} \right]$. Making the unknown scale constant s explicit and abusing notation slightly by fixing the equivalence class $\overline{\boldsymbol{x}}$ as $(x, y, 1)^T$ we can write

$$\boldsymbol{X} = (X, Y, Z)^T = (KR)^{-1} (s\overline{\boldsymbol{x}} - Kt) \ .$$

Now the world coordinate $\boldsymbol{X}'$ that we infer from image coordinate $\overline{\boldsymbol{x}}$ and which lies on the world plane $Z = 0$ has the homogenous representation

$$\overline{\boldsymbol{X}'} = (X', Y', 1)^T = \overline{\phi^{-1}(\boldsymbol{x})} = g^{-1}\overline{\boldsymbol{x}}$$

and because both $\boldsymbol{X}$ and $\overline{\boldsymbol{X}'}$ are members of the same equivalence class, that is $g\overline{\boldsymbol{X}} = g\overline{\boldsymbol{X}'} = \overline{\boldsymbol{x}}$, only the scale s differs between them. This implies that

$$\boldsymbol{X} = \boldsymbol{X}' + (s - s') (KR)^{-1} \overline{\boldsymbol{x}} \, , \tag{8.1}$$

and in fact it is not hard to prove (see for example [43]) that for any world coordindate $\boldsymbol{X}$ with counterpart $\boldsymbol{x}$ in the image,

$$\boldsymbol{X} - s\,(KR)^{-1}\,\overline{\boldsymbol{x}} = -R^{-1}t = -R^{T}t = \text{constant}\,, \tag{8.2}$$

where $-R^{T}t$ is the camera origin as seen in the world frame. Given a point on a moving object that is tracked between two frames and labelled $\boldsymbol{X}^1$ and $\boldsymbol{X}^2$ respectively, equation (8.2) yields

$$\boldsymbol{X}^2 - \boldsymbol{X}^1 = (KR)^{-1}\left(s^1\overline{\boldsymbol{x}}^1 - s^2\overline{\boldsymbol{x}}^2\right) \tag{8.3}$$

from which we first observe that the third component must be zero: the object has not changed in height. Further, the third row of $(KR)^{-1}$ is $\boldsymbol{M}_3^T$, a function solely of the elements of g so that

$$s^2\overline{\boldsymbol{x}}^2 \cdot \boldsymbol{M}_3 = s^1\overline{\boldsymbol{x}}^1 \cdot \boldsymbol{M}_3 \tag{8.4}$$

is a conserved quantity for any tracked point between frames.

8.1.1 The Known K Case

Given the estimated planar homography g, the columns of matrix R are recovered as

$$R_i = \begin{cases} (K^{-1}g)_i\,, & i \in \{1,2\} \\ R_1 \times R_2\,, & i = 3 \end{cases}$$

and $t = (K^{-1}g)_3$. Two distinct tracked points between two frames are sufficient to solve for the 'true' world coordinates $\boldsymbol{X}$ from those projected onto the road surface,

$\boldsymbol{X}$'. With superscripts indexing the frame and subscripts the tracked points, the first two components from (8.3) of the equality $\boldsymbol{X}_2^2 - \boldsymbol{X}_2^1 = \boldsymbol{X}_1^2 - \boldsymbol{X}_1^1$, which hold if the vehicle is assumed to have a constant orientation between frames, lend two equations in four unknowns s_i^j. The equalities (8.4) for each pair of points add two more equations. These can clearly be rearranged to form a linear system which once solved, leads directly to coordinates $\boldsymbol{X}$ via equation (8.2).

8.1.2 The Unknown K Case

We do not have the benefit in this case of prior knowledge of the first two rows of matrix $(KR)^{-1}$. However, it is easily verified for those rows that

$$\left((KR)^{-1}\right)^1 \cdot \overline{\boldsymbol{x}} = (\overline{\boldsymbol{x}} \times g_2) \cdot (KR_3) \quad \text{and} \quad \left((KR)^{-1}\right)^2 \cdot \overline{\boldsymbol{x}} = -(\overline{\boldsymbol{x}} \times g_1) \cdot (KR_3) \ ,$$

so that with knowledge of scales s'^j, the three unknowns KR_3 can be solved for in a linear system of equations derived from (8.3) applied to the vector difference $\boldsymbol{X}'^2 - \boldsymbol{X}'^1$ between the two tracked instances of a single point. The s'^j are however not known, and the resulting system of equations formed from stacking equalities (8.3) and (8.4) for both desired coordinates $\boldsymbol{X}_i^j$ and known projected coordinates $\boldsymbol{X}'^j_i$ is nonlinear. One possibility would be to initialize all s'^1_i and s_i^1 with guesses, infer the scales for subsequent frames from (8.4), solve the linear system (8.3) for KR_3 and continue iterating if it can be established that convergence is assured. Alternatively, for an

inner product

$$f\left(\boldsymbol{X}_1, \boldsymbol{X}_2\right) = f\left(\left(X_1, Y_1, Z_1\right), \left(X_2, Y_2, Z_2\right)\right) = \left\|\left(X_1 - X_2, Y_1 - Y_2\right)\right\|_2 + \left|Z_1\right| + \left|Z_2\right| \ ,$$

we could simply minimize a penalty function of the form

$$\sum_{i,j,k} \Psi\left(f\left(\boldsymbol{X}_i^{j+1} - \boldsymbol{X}_i^j, \boldsymbol{X}_k^{j+1} - \boldsymbol{X}_k^j\right)\right) \tag{8.5}$$

which expresses the expectation that while any vehicular tracked point remains at constant height, it should maintain a constant distance from other tracked points on the same vehicle.

8.1.3 3D Extension to the Energy Functional

Penalty (8.5) is in fact a 3D equivalent of the optical flow data term (7.4): Taking $\overline{\boldsymbol{V}}$ as a 3D vector field describing the traffic flow and $\overline{I}$ as a scalar field associating an intensity to any vehicular point in the (3D) world frame, we have $\partial\overline{I}/\partial t + \nabla\overline{I} \cdot \overline{\boldsymbol{V}} = 0$ along with the constraints

$$D\overline{\boldsymbol{V}} \cdot (\boldsymbol{W}, 0) = \boldsymbol{0} \quad \text{and} \quad D\overline{\boldsymbol{V}} \cdot (0, 0, 1) = \boldsymbol{0} \, . \tag{8.6}$$

The first constraint in (8.6) had no 2D counterpart in the development of the energy functional in section 6.1, in fact such a constraint was avoided in order to account for differing vehicle speeds in different lanes. By giving up that freedom here we are introducing the same information (but without the constant-orientation assumption) exploited in the linear system of section (8.1.1). But while that solution methodology

is restricted in applicability and naturally comes with the same caveats as the DLT in that it would be prone to noise amplification and likely benefit from some non-linear refinement, a minimization of (8.5) suggests that the height information and homography parameters could be inferred simultaneously by modifying the optical flow term in (7.5) and incorporating the additional constraints (8.6). The mechanics of pulling back the 3D optical flow equation to the image is left to future work.

8.2 Varying Homography

The planar homography has been assumed in this thesis to be constant over the image. This is reasonable for scenes of road segments that possess no vertical curvature nor cambering. On- and off-ramps, race tracks, bridges and undulations are however examples of scenes that may violate this assumption. By thinking of the road surface as the section of a fibre bundle where each point hosts the space of all planar homographies $SL(3)$, we can generalize to a scene where the homography varies over the road. Just as both data and regularization terms were needed to perform optical flow on the vehicle trajectories, we require their counterparts here to fit and regularize the vector-valued 'homography field' g. The latter term has been well studied in the context of diffusions on groups (see for example [15]) and is briefly outlined below; the latter requires new development.

8.2.1 Diffusive Flow on $SL(3)$

We imagine a smooth eleven-dimensional manifold parameterized by coordinates $\boldsymbol{Y}$ composed of the three 3D world-frame dimensions $\boldsymbol{X}$ in addition to the eight homography parameters. Given a metric $h_{\mathcal{E}}(\boldsymbol{Y})$ in this embedding space, we can calculate the induced metric $h_{\overline{\mathcal{W}}}(\boldsymbol{X})$ in the 3D world frame via the pullback

$$(h_{\overline{\mathcal{W}}})_{ij} = \sum_{k,l} (h_{\mathcal{E}})_{kl} \frac{\partial Y_k}{\partial X_i} \frac{\partial Y_l}{\partial X_j} \,. \tag{8.7}$$

Metric $h_{\overline{\mathcal{W}}}$ encodes the variation of the homography g over the world frame and can incorporate any information we have about expected homography parameters at any point. For instance, running the algorithm of chapter 6 on a small region of interest first allows us to extrapolate the homography translation parameters to any point in the scene, so when looking at the whole scene we can put greater weight on regions with greater deviation between the data and expectations. We collapse this information further to calculate the metric $h_{\mathcal{W}}$ induced on the 2D road manifold (itself embedded in the 3D ambient space) by adapting equation (8.7) again, and pull the Lagrangian $\int_{\mathcal{W}} \Psi\left(\sqrt{\det(h_{\mathcal{W}})}\right) \mathrm{d}\mathbf{X}$ back to the image using the techniques in section 6.1. Taking inspiration from the variational calculus on groups in [44, Chapter 13], the original eleven-dimensional surface area is minimized by application of the Euler-Lagrange equation. The result is an equation of motion for $\partial_t g(\boldsymbol{x})$ (or more preferably for $\partial_t \xi(\boldsymbol{x})$ on the Lie algebra) that induces a diffusive flow towards the minimum surface. In this case, where the Lagrangian is a function of g and not $\dot{g}$,

$\partial_t g$ simply describes gradient descent. A common example derived when Ψ is the identity function and $h_{\mathcal{E}}$ the identity matrix is mean curvature flow.

The data term must take the form $\int_{\mathcal{I}} \Psi\left(U\left(\nabla \boldsymbol{v}, \nabla \boldsymbol{w}\right)\right) \mathrm{d}\mathbf{x}$ for some potential U that penalizes any deviation in observed image vector fields $\boldsymbol{v}$ and $\boldsymbol{w}$ from the expected ones at $\boldsymbol{x}$, given homography $g(\boldsymbol{x})$ and its derivatives, and should also give rise to an equation of motion via Euler-Lagrange considerations. The specification of $h_{\mathcal{E}}$, U and the cranking through of the variational calculus machinery is left to future work.

8.2.2 Flow on Diff$(SL3)$

Rather than rely on the interaction between two opposing terms - the data term and regularization term - to extract an equation of motion on $SL(3)$ for every point in the image, it is possible to develop an equation of motion on Diff$(SL(3))$ itself, the group of diffeomorphisms of $SL(3)$, by constructing an appropriate Lagrangian. Here, the Lie algebra is Vect$(SL(3))$, the space of real vector fields on $SL(3)$, and invoking (3.21) we assume a left-invariant metric $h_G = \langle \cdot, \cdot \rangle_{T_g G}$ that induces the bilinear form

$$B\left(u, v\right) = \int_{SL(3)} \langle u, v \rangle_{T_g G} \, dV \ .$$

Constructing the Lagrangian $\mathscr{L}(g, v) = B\left(g^{-1}v, g^{-1}v\right) - U(g)$ for some potential U that emulates the role of the data term, we seek to extremize the energy functional

$$E(g) = \int_a^b \mathscr{L}(g(t), \dot{g}(t)) dt$$

for the resulting stationary-action path g arrived at after following the equation of motion on $\mathrm{Vect}(SL(3))$. For instance, when $U = 0$ the Euler equation describes a geodesic and has the form ([45, Chapter 13])

$$B\left(\partial_t v, u\right) = -B\left(v, \mathcal{L}_v u\right) \ ,$$

referring to the Lie derivative $\mathcal{L}$ of section 2.2, for any $u \in \mathrm{Vect}(SL(3))$. The advantage of this approach is the ability to regulate how quickly and smoothly the field $g(\boldsymbol{x})$ may evolve. Again, an appropriate metric and potential must be specified.

8.3 Conclusion

The estimation of the planar homography from road scenes has traditionally been challenging due to the lack of the usual direct geometric cues that metric rectification is performed with. We have presented a method that exploits the orthogonality between the vehicles' trajectories and their dominant edges when approaching or receding to enable rectification, but highlighted the inherent noise amplification of the classical Direct Linear Transform when applied to this end. Instead, we proposed a constrained nonlinear energy minimization that, while possessing no guarantee of convergence to the global minimum, has been shown to efficiently recover the homography in practice by updating the gradient descent steps on the Lie algebra $\mathfrak{sl}(3)$. The performance of the algorithm was also improved by the introduction of an optical flow term into the energy functional to promote the 'diffusing-out'

of estimation and systematic error in the image vector fields. As evidenced by a suite of examples, the resulting algorithm appears robust to noise and low video resolution, and stands as an effective first step toward automated vehicle classification and estimation of other traffic-related statistics.

Bibliography

[1] T. Langlotz, T. Nguyen, D. Schmalstieg, and R. Grasset, "Next-generation augmented reality browsers: Rich, seamless, and adaptive," *Proceedings of the IEEE*, vol. 102, no. 2, pp. 155–169, Feb 2014.

[2] N. Buch, S. Velastin, and J. Orwell, "A review of computer vision techniques for the analysis of urban traffic," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 12, no. 3, pp. 920 –939, sept. 2011.

[3] N. Ghosh and B. Bhanu, "Incremental unsupervised three-dimensional vehicle model learning from video," *Trans. Intell. Transport. Sys.*, vol. 11, no. 2, pp. 423–440, Jun. 2010. [Online]. Available: http://dx.doi.org/10.1109/TITS.2010.2047500

[4] S. Khan, H. C., D. Matthies, and H. Sawhney, "3d model based vehicle classification in aerial imagery," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, june 2010, pp. 1681 –1687.

[5] J. Choi, K. Sung, and Y. Yang, "Multiple vehicles detection and tracking

based on scale-invariant feature transform," in *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*, 30 2007-oct. 3 2007, pp. 528 –533.

[6] M. J. Leotta and J. L. Mundy, "Vehicle surveillance with a generic, adaptive, 3d vehicle model," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 7, pp. 1457–1469, Jul. 2011. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2010.217

[7] H. Kollnig and H.-H. Nagel, "3d pose estimation by fitting image gradients directly to polyhedral models," in *Proceedings of the Fifth International Conference on Computer Vision*, ser. ICCV '95. Washington, DC, USA: IEEE Computer Society, 1995, pp. 569–. [Online]. Available: http://dl.acm.org/citation.cfm?id=839277.840078

[8] P. Yarlagadda, O. Ozcanli, and J. Mundy, "Lie group distance based generic 3-d vehicle classification," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, dec. 2008, pp. 1 –4.

[9] A. Guiducci, "3d road reconstruction from a single view," *Comput. Vis. Image Underst.*, vol. 70, no. 2, pp. 212–226, May 1998. [Online]. Available: http://dx.doi.org/10.1006/cviu.1997.0633

[10] B. Bose and E. Grimson, "Ground plane rectification by tracking moving objects," in *IEEE International Workshop on Visual Surveillance and PETS*, 2004.

[11] Z. Zhang, M. Li, K. Huang, and T. Tan, "Robust automated ground plane rectification based on moving vehicles for traffic scene surveillance," in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, oct. 2008, pp. 1364 –1367.

[12] N. Sochen, R. Deriche, and L. Lopez-Perez, "Variational beltrami flows over manifolds," in *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, vol. 1, sept. 2003, pp. I – 861–4 vol.1.

[13] N. Sochen, R. Kimmel, and R. Malladi, "From high energy physics to low level vision," in *Presented in ONR workshop, UCLA*. Springer-Verlag, 1996, pp. 236–247.

[14] Y. Gur and N. Sochen, "Coordinate-free diffusion over compact lie-groups," in *Proceedings of the 1st international conference on Scale space and variational methods in computer vision*, ser. SSVM'07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 580–591. [Online]. Available: http://dl.acm.org/citation.cfm?id=1767926.1767983

[15] ——, "Regularizing flows over lie groups," *J. Math. Imaging Vis.*, vol. 33, no. 2, pp. 195–208, Feb. 2009. [Online]. Available: http://dx.doi.org/10.1007/s10851-008-0127-9

[16] R. Ben-Ari and N. Sochen, "A geometric framework and a new criterion in

optical flow modeling," *J. Math. Imaging Vis.*, vol. 33, no. 2, pp. 178–194, Feb. 2009. [Online]. Available: http://dx.doi.org/10.1007/s10851-008-0124-z

[17] D. Lin, E. Grimson, and J. Fisher, "Learning visual flows: A lie algebraic approach," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, june 2009, pp. 747 –754.

[18] ——, "Modeling and estimating persistent motion with geometric flows," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, june 2010, pp. 1 –8.

[19] C. Mei, S. Benhimane, E. Malis, and P. Rives, "Efficient homography-based tracking and 3-d reconstruction for single-viewpoint sensors," *Robotics, IEEE Transactions on*, vol. 24, no. 6, pp. 1352 –1364, dec. 2008.

[20] ——, "Homography-based tracking for central catadioptric cameras," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, oct. 2006, pp. 669 –674.

[21] E. Malis, T. Hamel, R. Mahony, and P. Morin, "Dynamic estimation of homography transformations on the special linear group for visual servo control," in *Proceedings of the 2009 IEEE international conference on Robotics and Automation*, ser. ICRA'09. Piscataway, NJ, USA: IEEE Press, 2009, pp. 1287–1292. [Online]. Available: http://dl.acm.org/citation.cfm?id=1703435.1703644

[22] F. Warner, *Foundations of Differentiable Manifolds and Lie Groups*, ser. Graduate texts in mathematics. Springer Verlage Publications, 1983. [Online]. Available: https://books.google.com/books?id=iiAquAAACAAJ

[23] D. D. Holm, T. Schmah, C. N. Stoica, and C. Ellis, David C. P, *Geometric mechanics and symmetry : from finite to infinite dimensions*, ser. Oxford texts in applied and engineering mathematics. New York, N.Y. Oxford University Press, 2009. [Online]. Available: http://opac.inria.fr/record=b1129333

[24] S. Helgason, *Differential Geometry, lie groups and symmetric space*. Academic Press, New York, 1978.

[25] V. Arnold, *Mathematical Methods of Classical Mechanics*. Springer, 1989, vol. 60.

[26] S. Ying and H. Qiao, "Lie group method: A new approach to image matching with arbitrary orientations," *International Journal of Imaging Systems and Technology*, vol. 20, no. 3, pp. 245–252, 2010.

[27] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. New York, NY, USA: Cambridge University Press, 2003.

[28] E. Malis and M. Vargas, "Deeper understanding of the homography decomposition for vision-based control," INRIA, Rapport de recherche RR-6303, 2007. [Online]. Available: http://hal.inria.fr/inria-00174036

[29] O. D. Faugeras and F. Lustman, "Motion and Structure From Motion in a Piecewise Planar Environment," *Intern. J. of Pattern Recogn. and Artific. Intelige.*, vol. 2, no. 3, pp. 485–508, 1988.

[30] Z. Zhang and A. R. Hanson, "3d reconstruction based on homography mapping," in *In ARPA Image Understanding Workshop*, 1996, pp. 0249–6399.

[31] A. Agarwal, C. V. Jawahar, and P. J. Narayanan, "A survey of planar homography estimation techniques," Tech. Rep., 2005.

[32] D. Liebowitz, "Camera calibration and reconstruction of geometry from images," Ph.D. dissertation, Merton College, University of Oxford, 2001.

[33] O. Faugeras, *Three-dimensional Computer Vision: A Geometric Viewpoint*, ser. Artificial intelligence. MIT Press, 1993. [Online]. Available: http://books.google.com/books?id=Aa6TTW9dWy0C

[34] O. K. Smith, "Eigenvalues of a symmetric 3 × 3 matrix," *Commun. ACM*, vol. 4, no. 4, pp. 168–, Apr. 1961. [Online]. Available: http://doi.acm.org/10.1145/355578.366316

[35] J. E. Solem and N. C. Overgaard, "A gradient descent procedure for variational dynamic surface problems with constraints," in *Variational, Geometric, and Level Set Methods in Computer Vision, Third International Workshop, VLSM*

2005, Beijing, China, October 16, 2005, Proceedings, 2005, pp. 332–343. [Online]. Available: http://dx.doi.org/10.1007/11567646_28

[36] J. Gallier, *Geometric Methods and Applications: For Computer Science and Engineering*. New York, NY: Springer New York, 2011.

[37] A. Iserles, H. Munthe-Kaas, S. Nørsett, and A. Zanna, "Lie group methods," *Acta Numerica 2005*, vol. 14, pp. 1–148, 2005.

[38] V. Varadarajan, *Lie groups, Lie algebras, and Their Representations*. Springer-Verlag, 1984.

[39] S. Roth and M. Black, "Steerable random fields," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, 2007, pp. 1–8.

[40] Z. Yu, C. Bajaj, and R. Bajaj, "Normalized gradient vector diffusion and image segmentation," in *In Proceedings of European Conference on Computer Vision*. Springer, 2002, pp. 517–530.

[41] H. Zimmer, A. Bruhn, J. Weickert, L. Valgaerts, A. Salgado, B. Rosenhahn, and H. Seidel, "Complementary optic flow," in *Proceedings of the 7th International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition*, ser. EMMCVPR '09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 207–220. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-03641-5_16

[42] B. K. Horn and B. G. Schunck, "Determining optical flow," Cambridge, MA, USA, Tech. Rep., 1980.

[43] V. Lepetit and P. Fua, "Monocular model-based 3d tracking of rigid objects," *Found. Trends. Comput. Graph. Vis.*, vol. 1, no. 1, Jan. 2005. [Online]. Available: http://dx.doi.org/10.1561/0600000001

[44] G. Chirikjian, *Stochastic Models, Information Theory, and Lie Groups, Volume 2: Analytic Methods and Modern Applications*, ser. Applied and Numerical Harmonic Analysis. Birkhäuser Boston, 2011. [Online]. Available: https://books.google.com/books?id=338COWrZDMUC

[45] J. E. Marsden and T. S. Ratiu, *Introduction to Mechanics and Symmetry: A Basic Exposition of Classical Mechanical Systems.* Springer Publishing Company, Incorporated, 2010.

Biographical Statement

Graham Beck was born December 6, 1980 in Umtata, South Africa before emigrating to New Zealand at the age of five. He received a BSc(Hons) in Physics and a PGDipSci in Math from the University of Otago in 2004 before receiving an MS in Applied Math from the Johns Hopkins University in 2012, the interlude taken up with work in the finance industry for several years. He has served as a Research Assistant in the Applied Math and Cognitive Science departments at Johns Hopkins and is currently employed as a Senior Quantitative Strategist at Allston Trading in Chicago.