HARVARD UNIVERSITY

Graduate School of Arts and Sciences

DISSERTATION ACCEPTANCE CERTIFICATE

The undersigned, appointed by the

Harvard John A. Paulson School of Engineering and Applied Sciences
have examined a dissertation entitled:

“A Lighting-Invariant Approach to Local Shape from Shading”

presented by: Kathryn Heal

Signature ____
Typed name: Professor T. Zickler

Signature ____
Typed name: Professor S. Gortler

Signature ____
Typed name: Professor Y. Lu

Signature ____
Typed name: Dr. C. Yu

December 4, 2020

A Lighting-Invariant Approach to Local Shape from Shading

A DISSERTATION PRESENTED
BY
KATHRYN HEAL
TO
THE DEPARTMENT OF ENGINEERING AND APPLIED SCIENCES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
IN THE SUBJECT OF
APPLIED MATHEMATICS

HARVARD UNIVERSITY
CAMBRIDGE, MASSACHUSETTS
DECEMBER 2020

Thesis advisor: Professor Todd Zickler Kathryn Heal

A Lighting-Invariant Approach to Local Shape from Shading

Abstract

Shape from shading is a classical problem in computer vision, in which the depth field of an object or a scene is reconstructed from a pattern of intensities in an image. This can be thought of in some sense as the inverse problem to geometry-based graphics rendering. In this context, shading is defined as a function of illumination, surface geometry, and surface characteristics like pattern or texture. Despite its enduring presence in the field, shape from shading is still largely unresolved. This dissertation shows that under the conventional diffuse shading model with unknown directional lighting, the set of quadratic surface shapes that are consistent with the spatial derivatives of intensity at a single image point is a two-dimensional algebraic variety embedded in the five-dimensional space of quadratic shapes. This work rigorously defines a family of such varieties, describes its geometry, and algebraically proves existence and uniqueness results in the areas of two-shot uncalibrated photometric stereo and coquadratic shape from shading. This work introduces a concise, feedforward model that computes an explicit, differentiable approximation of the variety from the intensity and its derivatives at any single image point. The result is a parallelizable processor that operates at each image point and produces a lighting-invariant descriptor of the continuous set of compatible surface shapes at the point. This processor is demonstrated on the two aforementioned application areas.

Contents

To the squirrels, rabbits – and yes, even the turkeys – of Harvard's campus.
May their fuzzy little bellies ever be warm and full of grain.

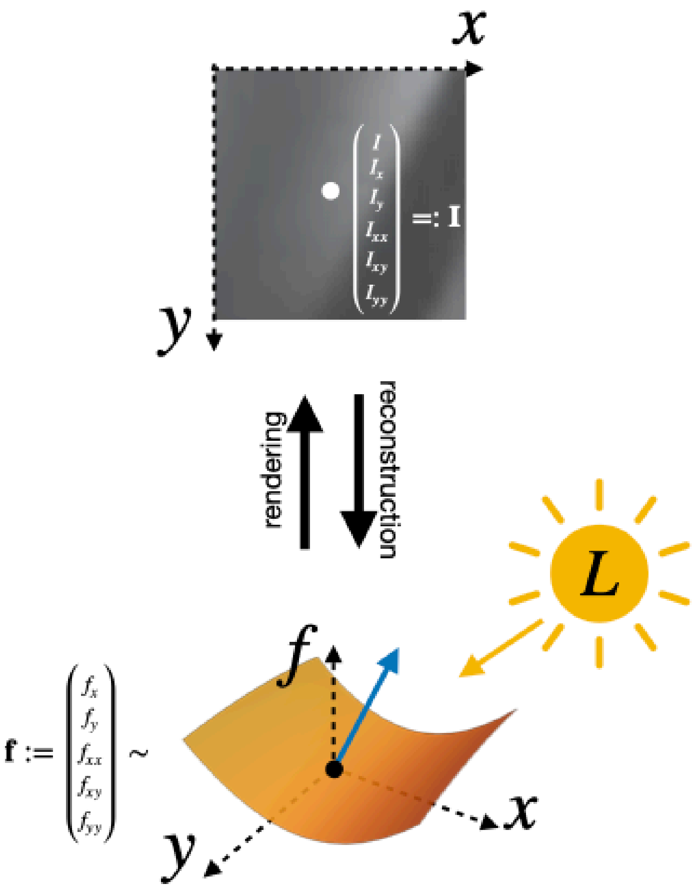
Acknowledgments

They say it takes a village to raise a thesis. Well, okay, they don't really say that, but in my case it has taken an ensemble of really remarkable people, for whom I will be eternally grateful. To Professors Todd Zickler, Shlomo Gortler, Yue Lu, Na Li, and Dr. Chris Yu, for their generous guidance, both academically and personally. To Harvard SEAS and Draper Laboratory, for providing support and stability through difficult circumstances. To my fiancé Kevin Ventullo, my mother Sheila Heal, and my father David Heal, for being the loves of my life. To Zick's Six: Emma Alexander, Qi Guo, Jialiang Wang, Dor Verbin, and Mia Polansky, for being a source of laughs, of comfort, and of knowledge. To Emma Rosenfeld, Julia Gonski, and Tamara Pico, for being only a hop or an Uber away. To the crew at 26 Ellery, including Ryan McKeown, Sam Dillavou, and Caity Seele, for keeping me fun. To Gracie Hailey, Dani Ward, and Christina Bailey, for their unconditional love and check-ins from across the country and around the world. To Dr. Scott Plotkin, for being along for the journey. Finally, to Wikipedia, StackExchange, and MathOverflow, to whom I should donate more.

0

Introduction

THE SHADING VARIATIONS in an image $I(x, y)$ of a diffuse, curved surface induce a perception of the surface shape $f(x, y)$; mimicking this perceptual capability in machines is referred to as recovering "shape from shading"[74,66,34]. Established techniques for recovering shape from shading exist in special cases where the strengths and locations of the light sources around the surface are known *a priori*, or are somehow accurately inferred before the depth reconstruction is attempted. These techniques can be understood as using a connected two-dimensional array of image "point processors", where each point processor reads the inten-


x
y
$\begin{pmatrix} I \\ I_x \\ I_y \\ I_{xx} \\ I_{xy} \\ I_{yy} \end{pmatrix} =: \mathbf{I}$
rendering
reconstruction
L
f
$\mathbf{f} := \begin{pmatrix} f_x \\ f_y \\ f_{xx} \\ f_{xy} \\ f_{yy} \end{pmatrix} \sim$
x
y

$$f(x,y) = f_x x + f_y y + \frac{1}{2} f_{xx} x^2 + f_{xy} xy + \frac{1}{2} f_{yy} y^2$$

Figure 1: The rendering model maps a surface to an image, which we locally represent as a mapping between surface and image coordinate space.

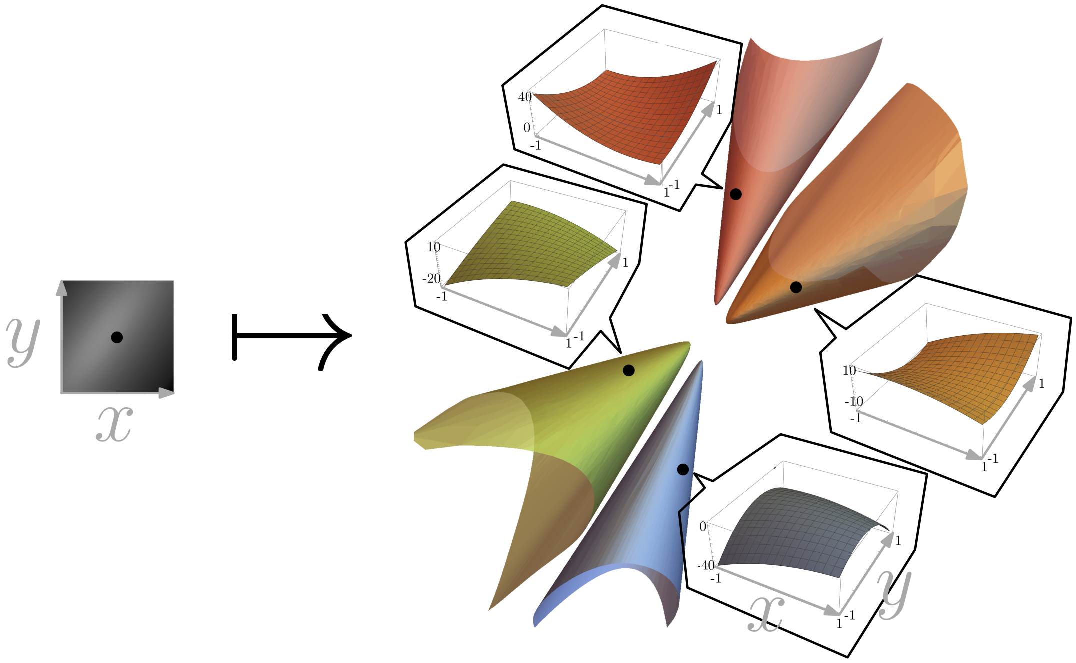


Figure 2: The set of local second order surface shapes $\{(f_x, f_y, f_{xx}, f_{xy}, f_{yy})\}$ that are consistent with the derivatives $\mathbf{I} = (I, I_x, I_y, I_{xx}, I_{xy}, I_{yy})$ at one image point (black circle, left) satisfy three polynomial equations. The zero locus (i.e., variety) is two-dimensional and is visualized here projected to three dimensions (f_{xx}, f_{xy}, f_{yy}). Each element of the variety is a local shape (four are called out) that produces the image derivatives under some light direction. We show that for any non-degenerate $\mathbf{I}$ the two-dimensional variety has four isomorphic components (colored in this example) and can be efficiently approximated by a coupled pair of shallow neural networks.

sity I at a single image point and, based on the known or estimated lighting conditions, calculates an intermediate numerical representation of the set of compatible local shapes at that point, comprising a set of (or probability density over) local surface orientations $\{(f_x, f_y)\}$ at the point. Each of the intermediate per-point orientation sets is ambiguous on its own, but when the array of point processors is connected together—by enforcing surface continuity and by including supplementary visual cues like occluding contours or top-down semantics—one can begin to recover shapes $f(x, y)$.

This has been the dominant paradigm for shape from shading for nearly fifty years[39], but it is far from satisfactory. Despite a half-century of research, the state of the art remains sensitive to non-idealities and is rarely deployed without substantial aid from a human annotator who first indicates occluding contours in an image or provides a segmentation of a relevant diffuse surface region (see for example Figure 1.1). One reason for this fragility is that lighting is typically non-uniform across surfaces, due to self-shadowing and other physical effects. This makes it difficult to infer the lighting conditions for each image point, which in turn distorts the per-point orientation sets $\{(f_x, f_y)\}$ upon which reconstruction is based. Moreover, even when lighting *is* uniform across a surface, the veridical location and strength of a scene's dominant light source can be impossible to infer from an image due to inherent mathematical ambiguities[9]. In comparison, monocular human vision seems to perform quite well at perceiving diffusely-shaded shape, at least modulo these ambiguities[44], despite being quite poor at inferring lighting[14].

This dissertation introduces a point processor for shading that might help address these deficiencies, by providing per-point constraints on shape *without* requiring knowledge of lighting[37]. The input to the processor is a measurement comprising a vector of spatial derivatives of intensity at one point, denoted by $\mathbf{I} := (I, I_x, I_y, I_{xx}, I_{xy}, I_{yy})$, Koenderink's 2-jet[43]. The internal structure of the processor is a coupled pair of shallow neural networks, and the processor's output is a compact representation of a continuous set of compatible local second-order shapes $F(\mathbf{I}) := \{(f_x, f_y, f_{xx}, f_{xy}, f_{yy})\}$ in the form of a parametrized two-dimensional manifold in $\mathbb{R}^5$. The processor provides useful per-point constraints because even though there are many compatible shapes $F(\mathbf{I})$, the overwhelming majority of shapes are ruled out.

Our main contribution is an algebraic analysis of Lambertian shading that provides the foundation for the point processor's internal structure and the format of its output. Specifically, we prove that the set of compatible local second-order shapes $F(\mathbf{I})$ are contained in the zero-set of three polynomial equations, i.e., are contained in a two-dimensional algebraic variety in $\mathbb{R}^5$. We show that special properties of this variety allow it to be represented in explicit form by a function from $\mathbb{R}^2$ to $\mathbb{R}^3$, which in turn can be approximated efficiently by a small system of simple neural networks.

The most important property of this point processor is that it is "invariant to illumination" in the sense that the output shape-set $F(\mathbf{I})$ always includes the veridical local second-order shape, regardless of how the surface is lit. This means that while a surface lit from different directions will generally induce different measurements $\mathbf{I}$ at a point, and while these different image measurements will in turn produce different shape-sets $F(\mathbf{I})$, all of the predicted shape-sets will include the true second-order shape at that point.

As examples of how the point processor can be used for image analysis, we describe two scenarios in which the intrinsic two-dimensional shape ambiguities $F(\mathbf{I})$ at each point can be reduced to a discrete four-way choice by exploiting additional constraints or information. One scenario is uncalibrated two-shot photometric stereo, where the input is two images of a surface under two unknown light directions. The other is quadratic shape from shading, where the input is a single image of a shape that is quadratic over an extended region. We demonstrate these using synthetic images, leaving the development of robust algorithms and deployment on captured photographs for (near) future work.

Throughout this dissertation, we assume a frame of reference such that our measurements are graphs of some polynomial function. We represent these local surface height and image values as vectors of their coefficients – applying the Monge-Taylor map – ignoring dependence of f_{xx} on f_x. This is to say that we are not attempting to solve any partial differential equations; instead, we are studying algebraic constraints in a local linear coefficient coordinate space.

This dissertation will be structured in the following way. We will introduce the problem of shape from shading, and historical attempts to solve it. We will then propose a new approach that is algebraic in nature. We will review existing ways to solve polynomial systems such as the one we propose, and describe why they are insufficient in this setting. We will propose our own local solver driven by a deep-learning inspired function approximation. Finally, we will explore applications and describe implications to various computer vision problems. We will introduce topic-specific related work at the beginning of each section as we incorporate new subfields.

1

Background and Related Work

MOST EXISTING APPROACHES TO SHAPE FROM SHADING rely on a per-point relationship between scalar intensity I and surface orientation (f_x, f_y). If the lighting comes from a single direction, for example, then the set of compatible surface orientations is a right-circular cone with axis equal to the light direction and apex angle proportional to intensity (see Figure 1.1). Similarly, if the lighting is a positive-valued function defined on the directional two-sphere then the set of compatible orientations is well approximated by a one-dimensional manifold defined by the light function's spherical harmonic coefficients up to third degree[62,5].

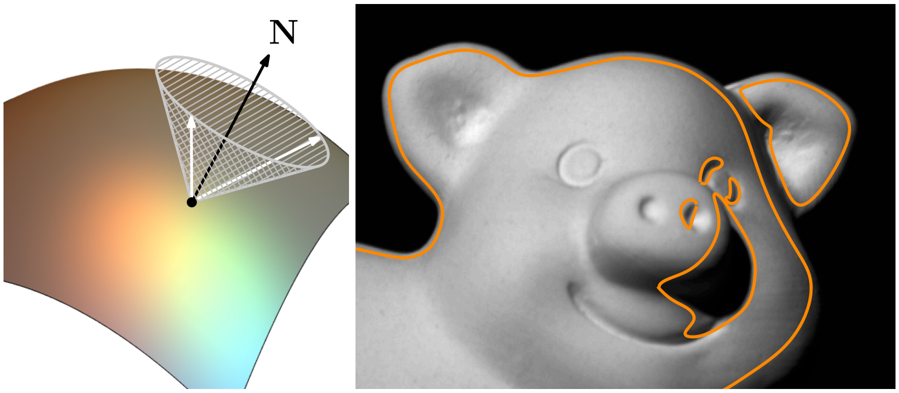


Figure 1.1: Left: Given an image intensity measurement at a point, and a known light source direction, the solution set is reduced to a one-parameter family of surface normals. Right: Segmentation into relevant diffuse surface regions.

Regardless, any such relation between intensity and surface orientation necessarily requires prior knowledge of, or accurate estimates of, the lighting at every surface point. Despite substantial recent progress[77,3,63,27], including the abilities to accommodate some amounts of non-uniform lighting and non-uniform surface material properties, obtaining useful results continues to require substantial help from a human, who must first label the region that contains a continuous surface and/or indicate the locations of occluding contours. We will refer to this requirement as the need for a prior "perceptual grouping" of the image into such regions.

In contrast, we follow Kunsberg and Zucker[50] by enhancing the per-point analysis to consider not just the intensity and surface orientation at a point, but also higher order derivatives of intensity (and shape). This allows eliminating the dependence on lighting entirely, and it suggests the possibility of a different approach where perceptual grouping and shape reconstruction can occur without explicit knowledge of lighting, and perhaps with lighting being (approximately) inferred later, as a by-product of shape perception. In this thesis I consider just the first step toward this possibility: the design of the essential point processor.

We are also motivated by the results of Xiong et al.[69], who consider a local area processor instead of a pure point processor, and show that the intensity values in an extended image patch determine the extended quadratic shape up to a discrete four-way choice. This four-way choice leads to the automorphism group that we describe in Section 2.

Many previous shape from shading studies assume prior knowledge of light source[57,60]. The lighting may be specified by a general reflectance map[3], by a directional source[69,50], or by its spherical harmonics[5]. On the other hand, some thought has been given to the setting of unknown light source, under the same types of lighting conditions. Our work addresses the case of an unknown directional lighting source, and expanding to more general lighting maps is an interesting area of future research.

The work in this dissertation is complementary to recent learning-based approaches[22,8] to monocular depth estimation that aim to exploit diffuse shading and many other bottom-up cues while also exploiting contextual cues in large image datasets. One of our goals is to explore alternative front-end architectures and interpretable intermediate representations that can improve the generality and efficiency of such systems in the future.

Shape from shading has traditionally been approached by solving a first-order PDE over a swath of an image; this differs from our more local, point-based approaches. However, these types of PDE methods implicitly assume that the image has been segmented beforehand into diffuse versus non-diffuse regions. These approaches also tend to use only first-order intensity information, rather than incorporating second-order as we do. A subset of the literature[58,21,49,51,50,38] phrases shape from shading not as solving a PDE, but as solving a system of polynomial or rational equations. Of course, the questions of existence and uniqueness also arise for algebraic systems; such a solution set is associated to a well-studied type of mathe-

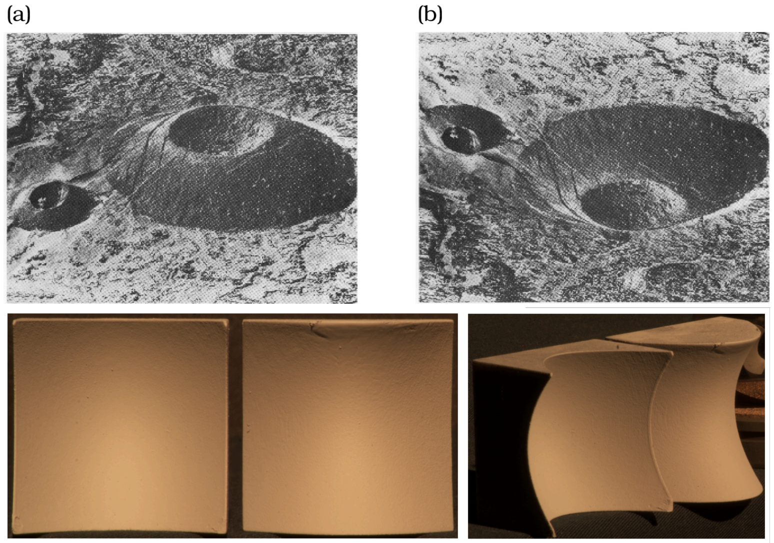


Figure 1.2: Examples of local visual ambiguities. One such phenomenon is known as the *fourfold ambiguity*, wherein a viewer not given the light source is unable to distinguish convex shapes from concave shapes, and saddle shapes from spherical shapes. The above images are due to Pentland[59], and the below images are due to Xiong et al.[70]

matical object called an *algebraic variety*.

The broadest objective of this dissertation is to describe the set of solutions to shape from shading when lighting is unknown. Shape from shading is a classical problem in computer vision, and this thesis reformulates it via a system of algebraic equations at each pixel. The solutions to this system represent the set of shapes consistent with a given image measurement, and this thesis proposes a feedforward way of describing this solution set that at each pixel that is compact and does not require online iterations.

But surely if artificial intelligence research is to claim victory over the vision problem, then it has to embrace the whole domain, understanding not only the problem solving aspects, but also the physical laws that underlie image formation and the corresponding symbolic constraints that enable the problem solving.

Berthold K. P. Horn, *Understanding Image Intensities*[40], 1977

2 Geometry of Local Shape Sets

THE ILLUMINATION-INVARIANT POINT PROCESSOR INTRODUCED IN THIS DISSERTATION is inspired by the work of Kunsberg and Zucker[50], who use differential geometry to derive three lighting-invariant rational equations that relate the image 2-jet $\mathbf{I}$ at a point to the surface height derivatives $\mathbf{f}$ at that point. We take an algebro-geometric approach instead, which provides an abbreviated derivation of equivalent equations and also reveals that the set of shapes that solves SfS is an algebraic variety with useful structure explored by this and the following chapter.

We show that this solution set is equipped with an automorphism group that naturally divides it into four isomorphic pieces, allowing the entire set to be represented by any one component. We then relate one piece of the shape set to a continuous function $\varphi_{\mathbf{I}}$ from $\mathbb{R}^2$ to $\mathbb{R}^3$, which implies that the point processor is equivalent to a map from vectors $\mathbf{I} \subset \mathbb{R}^6$ to continuous functions $\varphi_{\mathbf{I}} : \mathbb{R}^2 \mapsto \mathbb{R}^3$. This analysis provides the foundation for a fast, offline neural network approximation of the mapping from vectors $\mathbf{I}$ to functions $\varphi_{\mathbf{I}}$, which appears in Section 4.

2.0.1 Existence and Uniqueness in Shape from Shading

When confronted with an inverse problem, two questions must immediately be addressed, namely existence and uniqueness of a solution. Does a solution exist, or is the solution set empty? Is the solution unique, or are there finitely many – or even infinitely many – equally valid solutions? Well-posedness then means existence and uniqueness of a solution, while *ambiguity* refers to a solution that exists but is *not* unique.

PDE-based approaches have borne some compelling existence-uniqueness results. Brooks[13] studies ambiguity in images of planar surfaces, among others, that arises from the observation that any image having constant intensity must have been generated by some ruled surface. Introducing new information, such as boundary conditions, can reduce this ambiguity. Kozera et al.[45] is a more recent critique of that work. For a pinhole camera and a directional light source located at camera center, Prados et al.[61] proves that under appropriate boundary conditions, the proposed PDE has a unique solution.

The boundary conditions that allow the PDE approaches to in some cases establish existence and/or uniqueness of solutions to SfS are unfortunately not applicable when solving

SfS in a local way. Consequently, local SfS opens the door to many more possible ambiguities, as is noted in Ecker et al.[21]. A critical example of such ambiguities encountered in local SfS arise when the light source is unknown. Kunsberg and Zucker[50] use differential geometry to derive a set of equations relating surface to image that are invariant to light. Agreement with the equations derived in their main theorem is fundamental to our work. Shape from shading as a depth-perception tool is strengthened when combined with other visual cues and available information from photometric stereo, to specularity, to contours. Basri et al.[4] perform depth from photometric stereo, in the case where the lighting source is distant and isotropic, but otherwise unknown. Drbohlav et al.[20] argue that when lighting is unknown and the surface is specular, just two measurements will suffice to reconstruct the shape. Todorovic et al.[68] suggest that, given a two-dimensional shaded figure, the shape of its contours can can help a viewer resolve shape-from-shading ambiguities.

2.0.2 Shading and Surface Models

Our analysis applies to any point in a 2D image where the reflectance is roughly Lambertian. We assign the coordinates $(0, 0)$ to the point of interest and let $I(x, y)$ denote the intensity in a bounded local neighborhood $U \subset \mathbb{R}^2$ of that point. We refer to U as the *receptive field*, and its maximum size of will of course depend on the curvature of the surface $f(x, y)$; in traditional shape from shading one typically assumes f is reasonably smooth, so under this assumption U will be reasonably large. With this being said, to be safe we recommend choosing U that is no larger than is required to robustly compute a discrete approximation to the first and second spatial derivatives of $I(x, y)$ at the origin.

Ignoring its traditional definition, we use *pixel* to denote a single point in the continuum U.

This is done with the goal of distinguishing a physical point on the continuous image space with an abstract point in surface-solution space.

Within the neighborhood U, we assume that the image is the orthographic projection of a curved Lambertian surface, and that the surface can be represented by a height function $f(x,y)$. The surface albedo $\rho \in \mathbb{R}^+$ is assumed to be constant within U. We also assume that the lighting is uniform and directional within U, so that it can be represented by $\mathbf{L} \in \mathbb{R}^3$ with strength $\|\mathbf{L}\|$ and direction $\mathbf{L}/\|\mathbf{L}\|$. Under these assumptions the intensity is

$$I(x,y) = \rho \mathbf{L} \cdot \frac{\mathbf{N}(x,y)}{||\mathbf{N}(x,y)||}, \qquad (x,y) \in U, \tag{2.1}$$

where

$$\mathbf{N}(x,y) := \left(-\frac{\partial f}{\partial x}(x,y), -\frac{\partial f}{\partial y}(x,y), 1 \right)^T$$

is the normal field. Note that we allow for the projection, albedo, and lighting to vary outside of neighborhood U, so reasonable smoothness assumptions on these maps should suffice.

We assume that the surface f is locally smooth enough around the point (x,y) that we can ignore any third or higher order derivatives at that point.

$$f(x,y) = f_x x + f_y y + \frac{1}{2}\left(f_{xx} x^2 + 2 f_{xy} xy + f_{yy} y^2 \right). \tag{2.2}$$

We refer to $\mathbf{f} := (f_x, f_y, f_{xx}, f_{xy}, f_{yy}) \in \mathbb{R}^5$ as the *local shape* at the point (x,y). In particular we have $\mathbf{f}(0,0) = (a,b,c,d,e)$. We assume that all local shapes are not flat or cylindrical, or more precisely are nondegenerate in this sense:

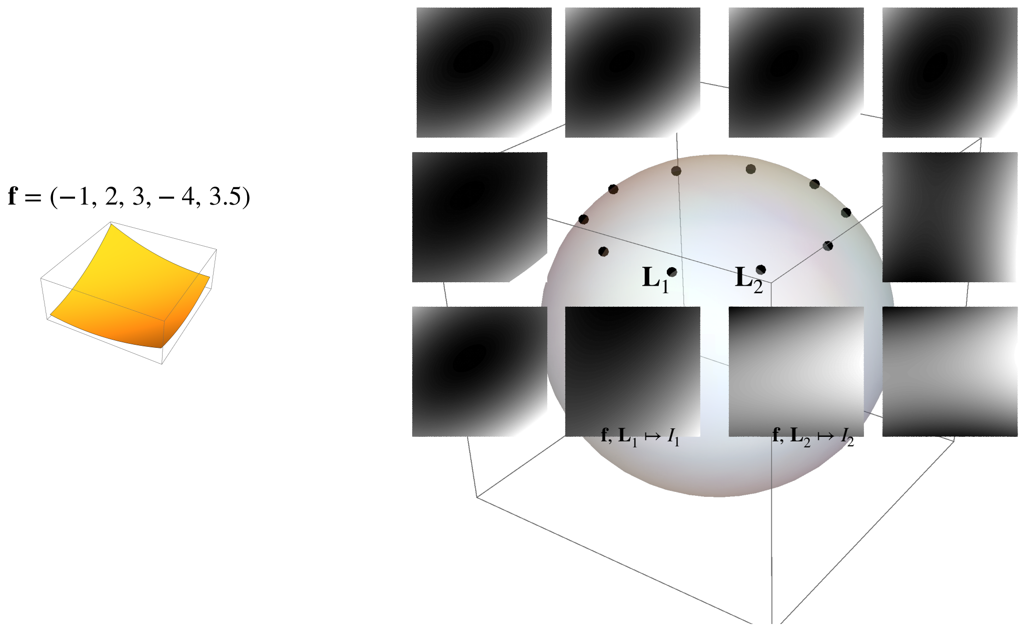


Figure 2.1: How the local image $\mathbf{I}_i$ changes with light source direction $\mathbf{L}_i$, for a fixed surface **f**.

Definition 1. *We say that a local shape* **f** *is nondegenerate if*

$$(f_{xx} + f_{yy})(f_{xx}f_{yy} - f_{xy}^2)(4f_{xy}^2 + (f_{xx} - f_{yy})^2) \neq 0.$$

Call the set of all nondegenerate local shapes Σ; *this set is full-dimensional in* $\mathbb{R}^5$.

Local shapes can produce many different image intensity patterns depending on the lighting direction; this is illustrated in Figure 2.1 for ten such light directions. We call the set of all possible image 2-jets generated by any combination of local shape and lighting *realizable*, and we say that a realizable image 2-jet produced by a particular shape is *consistent* with that shape.

Definition 2. *The set of realizable measurements* $\mathcal{I}$ *is the set of vectors* $\boldsymbol{v} \in \mathbb{R}^6$ *for which there exists a light direction* $\boldsymbol{L} \in \mathbb{R}^3$ *and nondegenerate local shape* **f** *such that* $\boldsymbol{v} = \mathbf{I}$ *when shape*

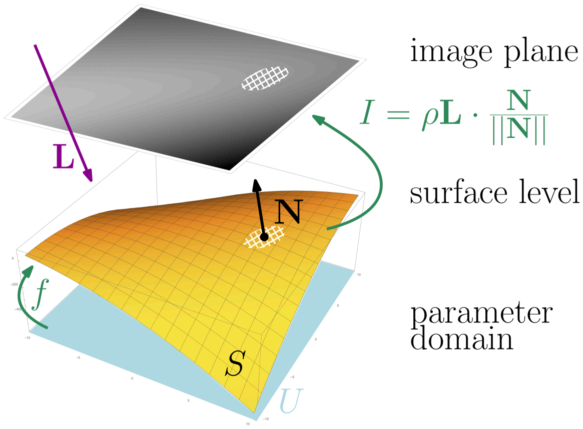


image plane
$I = \rho\mathbf{L} \cdot \frac{\mathbf{N}}{||\mathbf{N}||}$
L
surface level
N
f
parameter domain
S
U

Figure 2.2: We assume a Lambertian reflectance model, and measure local image measurements about a point.

model (2.2) *is combined with shading model* (2.1)*.*

Basically, an image measurement is realizable if it could have been generated by the models we assume.

Definition 3. *If for a pair* $(\mathbf{I}, \mathbf{f}) \in \mathcal{I} \times \mathbb{R}^5$ *there exists such an* ***L****, we say that* $\mathbf{I}$ *and* $\mathbf{f}$ *are consistent. This means that for some light direction,* $\mathbf{f}$ *is a valid explanation of image measurements* $\mathbf{I}$*.*

The objective of the point processor will be, given an observation **I**, to retrieve thew set of consistent local shapes **f**.

2.1 Local Shape Set

Our immediate goal is to characterize the set of shape that are consistent with observation **I** for *any* light direction. We'll refer to this set of shapes as $F(\mathbf{I})$. This set of admissible shapes turns out to be contained in the locus of real solutions to three polynomial equations. A crucial feature is that the albedo and lighting do not appear in these equations.

Theorem 1. *Assume the shading model of* (2.4) *and the surface model of* (2.2)*, and suppose we are given a measurement* $\mathbf{I} \in \mathcal{I}$ *generated by some unknown surface/lighting combination. Define polynomials*

$$\begin{aligned}
C_1(\mathbf{f};\mathbf{I}) \; &:= I\left(\left(f_x^2+1\right)f_{xy}^2 + f_{xx}\left(-2f_xf_{xy}f_y + f_{xx}f_y^2 + f_{xx}\right)\right) \\
&\quad + \left(f_x^2+f_y^2+1\right)\left(I_{xx}\left(f_x^2+f_y^2+1\right) + 2I_x(f_xf_{xx}+f_{xy}f_y)\right), \\
C_2(\mathbf{f};\mathbf{I}) \; &:= I\left(\left(f_x^2+1\right)f_{yy}^2 - 2f_xf_{xy}f_yf_{yy} + f_{xy}^2\left(f_y^2+1\right)\right) \qquad (2.3) \\
&\quad + \left(f_x^2+f_y^2+1\right)\left(I_{yy}\left(f_x^2+f_y^2+1\right) + 2I_y(f_xf_{xy}+f_yf_{yy})\right), \\
C_3(\mathbf{f};\mathbf{I}) \; &:= \; \left(f_x^2+f_y^2+1\right)\left(f_x^2I_{xy} + f_x(f_{xx}I_y + f_{xy}I_x) + f_y(f_{xy}I_y + f_yI_{xy} + f_{yy}I_x) + I_{xy}\right) \\
&\quad + I\left(f_{xy}\left(f_x^2f_{yy} - f_xf_{xy}f_y + f_{yy}\right) + f_{xx}\left(-f_xf_yf_{yy} + f_{xy}f_y^2 + f_{xy}\right)\right).
\end{aligned}$$

Then any nondegenerate local shape $\mathbf{f} \in \mathbb{R}^5$ *that is a valid explanation of measurements* $\mathbf{I}$ *will satisfy* $C_i = 0 \;\forall i$*. Equivalently, the affine variety* $F := \boldsymbol{V}(\sqrt{\langle C_1, C_2, C_3\rangle})$ *contains the set of all shapes* $\mathbf{f}$ *consistent with* $\mathbf{I}$*.*

Later in this document, we will allow $\mathbf{I}$ to vary freely, as a symbolic vector; this will turn Equations 2.3 into a parametrized system of polynomials, thus defining an incidence variety F in $\mathbb{R}_{\mathbf{f}} \times \mathbb{R}_{\mathbf{I}}$. See Amendola et al.[1] for some solutions to such incidence varieties. For now, assume we are given a fixed numeric image measurement $\mathbf{I}$ that defines our polynomial system. We will use $\mathcal{P}(\mathbf{I})$ to denote this system, or $\mathcal{P}$ in places where $\mathbf{I}$ is fixed (i.e. considered a numeric vector rather than a variable). Finally, we use the notation $\mathbf{V}(\cdot)$ to denote the variety corresponding to an ideal; this is essentially the zero locus of the generating polynomials.

Many times it will be useful to think about the set of local *curvatures* consistent with a lo-

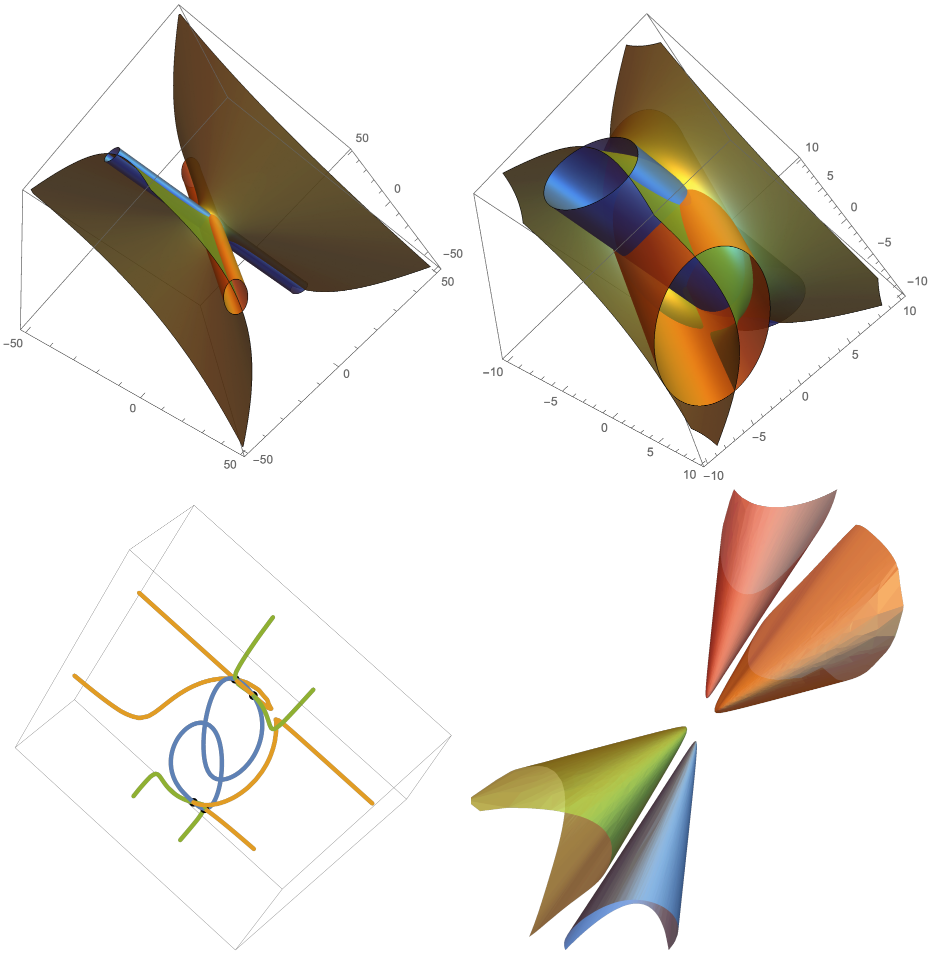


Figure 2.3: Clockwise from top left: For fixed image measurement **I** and fixed surface orientation f_x, f_y, the zero-loci of C_1, C_2, C_3, respectively blue, orange and yellow, (1) from afar and (2) zoomed in near the origin. (3) The space curves obtained from pairwise-intersecting these degenerate quadric surfaces. The black dots denote the pairwise intersections of the space curves. As f_x, f_y vary, so do these loci, and the corresponding black points trace out (4) the surface $\pi(X)$.

cal image measurement; that is, the projection $\pi(F)$, where we define $\pi : (f_x, f_y, f_{xx}, f_{xy}, f_{yy}) \mapsto (f_{xx}, f_{xy}, f_{yy})$. Such a projection is visualized in the lower right-hand corner of Figure 2.3.

2.2 Proof of Theorem 1

Proof. For brevity let $a, b, c, d, e = f_x, f_y, f_{xx}, f_{xy}, f_{yy}$. We begin with Lambert's law for some fixed **I** vector,

$$I(x,y) = \rho\mathbf{L} \cdot \frac{\mathbf{N}(x,y)}{||\mathbf{N}(x,y)||}, \qquad (x,y) \in U, \tag{2.4}$$

with $\mathbf{N}(x,y) := (-(\partial f/\partial x)(x,y), -(\partial f/\partial y)(x,y), 1)^T = (-a - cx - dy, -b - dx - ey, 1)^T$. Since we do not require $\mathbf{L}$ to be unit, we can effectively absorb ρ into it. This turns (2.4) into

$$I(x,y) = -\frac{(a + cx + dy)L_1 + (b + dx + ey)L_2 - L_3}{\sqrt{(a + cx + dy)^2 + (b + dx + ey)^2 + 1}} \tag{2.5}$$

$$= -w((a + cx + dy)L_1 + (b + dx + ey)L_2 - L_3), \tag{2.6}$$

with

$$w := \frac{1}{\sqrt{(a + cx + dy)^2 + (b + dx + ey)^2 + 1}}.$$

Rearranging this,

$$0 = w((a + cx + dy)L_1 + (b + dx + ey)L_2 - L_3) + I(x,y), \tag{2.7}$$

$$0 = w^2((a + cx + dy)^2 + (b + dx + ey)^2 + 1) - 1. \tag{2.8}$$

Taking first and second order partial derivatives of (2.7) with respect to x and y, then evaluating each resulting derivative at the point $(x, y) = (0, 0)$ gives us the local system

$$\begin{aligned}
s &:= w^2\left(a^2 + b^2 + 1\right) - 1 \\
r_0 &:= w(aL_1 + bL_2 - L_3) + I \\
r_x &:= w^3(-(ac + bd))(aL_1 + bL_2 - L_3) + w(cL_1 + dL_2) + I_x \\
r_y &:= w^3(-(ad + be))(aL_1 + bL_2 - L_3) + w(dL_1 + eL_2) + I_y \\
r_{xx} &:= w^3\left(3w^2(ac + bd)^2 - c^2 - d^2\right)(aL_1 + bL_2 - L_3) - 2w^3(ac + bd)(cL_1 + dL_2) + I_{xx} \\
r_{xy} &:= 3w^5(ac + bd)(ad + be)(aL_1 + bL_2 - L_3) - dw^3(c + e)(aL_1 + bL_2 - L_3) \\
&\quad - w^3(ad + be)(cL_1 + dL_2) - w^3(ac + bd)(dL_1 + eL_2) + I_{xy} \\
r_{yy} &:= w^3\left(3w^2(ad + be)^2 - d^2 - e^2\right)(aL_1 + bL_2 - L_3) - 2w^3(ad + be)(dL_1 + eL_2) + I_{yy}
\end{aligned}$$

and we refer to the vectors (r_0, r_x, r_y, r_{xx}), (r_0, r_x, r_y, r_{xy}), and (r_0, r_x, r_y, r_{yy}) as $\mathbf{r}_1$, $\mathbf{r}_2$, and $\mathbf{r}_3$, respectively. Each of these seven polynomials is linear in the L_i; thus, if $\mathbf{L} := (L_1, L_2, L_3, 1)$, we can write this system as $\mathbf{r}_i = A_i\mathbf{L}$, where $A_i = A_i(w, \mathbf{f}, \mathbf{I})$ is a square functional-entried matrix:

$$A_i = \begin{pmatrix} aw & bw & -w & I \\ cw\left(1-a^2w^2\right) - abdw^3 & dw\left(1-b^2w^2\right) - abcw^3 & w^3(ac+bd) & I_x \\ dw\left(1-a^2w^2\right) - abew^3 & ew\left(1-b^2w^2\right) - abdw^3 & w^3(ad+be) & I_y \\ \rho_{i1} & \rho_{i2} & \rho_{i3} & \rho_{i4} \end{pmatrix}$$

$$\begin{aligned}
\rho_{11} =& w^3\left(3a^3c^2w^2 + 6a^2bcdw^2 - a(d^2(1-3b^2w^2) + 3c^2) - 2bcd\right) \\
\rho_{12} =& w^3\left(bc^2(3a^2w^2 - 1) + 6ab^2cdw^2 - 2acd + 3b^3d^2w^2 - 3bd^2\right) \\
\rho_{13} =& w^3\left(c^2(1-3a^2w^2) - 6abcdw^2 + d^2(1-3b^2w^2)\right) \\
\rho_{21} =& w^3\left(3a^3d^2w^2 + 6a^2bdew^2 - a(e^2(1-3b^2w^2) + 3d^2) - 2bde\right) \\
\rho_{22} =& w^3\left(bd^2(3a^2w^2 - 1) + 6ab^2dew^2 - 2ade + 3b^3e^2w^2 - 3be^2\right) \\
\rho_{23} =& w^3\left(d^2(1-3a^2w^2) - 6abdew^2 + e^2(1-3b^2w^2)\right) \\
\rho_{31} =& w^3\left(3a^3cdw^2 + 3a^2bw^2(ce+d^2) - ad(-3b^2ew^2 + 3c + e) - b(ce+d^2)\right) \\
\rho_{32} =& w^3\left(-bd(-3a^2cw^2 + c + 3e) + 3ab^2w^2(ce+d^2) - a(ce+d^2) + 3b^3dew^2\right) \\
\rho_{33} =& w^3\left(c(-3a^2dw^2 - 3abew^2 + d) + d(-3abdw^2 - 3b^2ew^2 + e)\right) \\
\rho_{14} =& I_{xx} \qquad \rho_{24} = I_{yy} \qquad \rho_{34} = I_{xy}.
\end{aligned}$$

Notice that for each i, $\det A_i = w^3(d^2 - ce)C_i$, and that $s = 0 \implies w \neq 0$. Suppose $\neg(C_i = 0 \ \forall i)$. That is, $\exists i : C_i \neq 0$. Due to the non-degeneracy assumption and the constraint imposed by s that $w \neq 0$, this is equivalent to $\exists i : w^3(d^2 - ce)C_i = \det A_i \neq$

$0 \iff \exists i : \ker(A_i) = \{0\}$. This is equivalent to $\exists i : \forall \mathbf{L} \neq 0,\ A_i\mathbf{L} = \mathbf{r}_i \neq 0$, which is to say that there exists an *i* which, regardless of **L**, will always violate one of the Lambert partial derivative conditions. This implies $\neg((\mathbf{f}, \mathbf{I})$ are consistent). The contrapositive of this argument is that under the stated assumptions, $(\mathbf{f}, \mathbf{I})$ consistent implies that $C_i = 0\ \forall i$. □

Remark 1. *The real solutions to these equations are identical to those of Corollary 4.2 of Kunsberg et al.*[50]*; we offer our algebraic derivation as an alternative to the differential-geometric approach presented in that work.*

Theorem 1 states that the set of local shapes that are consistent with a given measurement $\mathbf{I} \in \mathcal{I}$ must satisfy a set of three algebraically independent polynomials and thus, by definition, is contained in a real two-dimensional algebraic variety embedded in the five-dimensional shape space. This variety is analogous to the one-dimensional manifold of surface orientations in classical shape from shading, and it provides substantial constraints on local shape, because although there are still infinitely many admissible local shapes, the vast majority of shapes are disqualified.

The variety for a particular measurement **I** is visualized in Figure 2.3, projected from the five-dimensional shape space to a three dimensional space that corresponds to the second-order shape dimensions (f_{xx}, f_{xy}, f_{yy}). Additional examples are in Figure 2.6, which shows how the varieties change for different measurements. As elements of $\mathbb{R}[\mathbf{f}]$, polynomials C_1, C_2, C_3 are determined by parameters $\mathbf{I} \in \mathcal{I}$, so the variety $F(\mathbf{I})$ is as well. We may therefore define a map $\Phi : \mathbf{I} \mapsto F(\mathbf{I})$. This leads us to the interpretation of the system as a smooth fiber bundle, with base space $\mathcal{I}$, fibers $F(\mathbf{I})$, and section Φ.

Remark 2. *Each polynomial C_i is irreducible in $\mathbf{f}, \mathbf{I}$, and therefore each $\boldsymbol{V}(C_i)$ is irreducible. However that does not imply that the intersection of those varieties F is irreducible. An example may convince the reader of this: the polynomials $g_1 = x^2 - y$ and $g_2 = x - 1$ are each irreducible, but the intersection $\boldsymbol{V}(g_1, g_2)$ reduces to components $\{(1,1)\}$ and $\{(-1,1)\}$.*

Remark 3. *Why not just solve this system of equations given a measurement $\mathbf{I}$, i.e. online? We prefer a robust feedforward engine that takes an image measurement and outputs a set of shapes compatible with that measurement, without needing to optimize anything online. This amounts to finding an explicit representation for the implicit form presented in Theorem 1, that applies to symbolic $\mathbf{I}$ rather than just numerical. However, as is often the case in algebraic geometry, such an explicit representation may not be readily available in an analytical form. In order to achieve this representation, we exploit symmetries inherent to the system and approximate it using a pair of coupled neural networks.*

2.3 Symmetries of the local shape set

At this stage we have an implicit description of a shape set $F(\mathbf{I})$ in terms of generating polynomials (2.3). For a useful point processor, we want instead an *explicit* representation, as well as an efficient way to calculate (and store) that explicit representation for any particular image 2-jet $\mathbf{I}$. An explicit *analytic* representation remains out of reach: when $\mathbf{I}$ and f_x, f_y are fixed, the solutions of (2.3). can be interpreted as the intersection of three quadric hypersurfaces (see Figure 2.3). Algebraic solvers for finding the intersection of three quadric *surfaces* have been proposed [16,46], but these will not suffice for the symbolic systems we seek to solve. Fortunately, the varieties exhibit three properties that make them easier to approximate.

In Section 2.3.1 we show that the variety is equipped with an automorphism group that naturally divides it into four isomorphic pieces, allowing the entire shape set to be represented by any one piece (Observation 1). Then Section 2.3.2 relates the one piece to a continuous function $\varphi_{\mathbf{I}}$ from $\mathbb{R}^2$ to $\mathbb{R}^3$, which implies that the point processor is equivalent to a map from vectors $\mathbf{I} \subset \mathbb{R}^6$ to continuous functions $\varphi_{\mathbf{I}} : \mathbb{R}^2 \mapsto \mathbb{R}^3$. As we will see later in Section 4, these two properties enable an efficient point processor in the form of a neural network approximation of the mapping from 2-jet $\mathbf{I}$ to functions $\varphi_{\mathbf{I}}$ (see Figure 4.2). Examples of how this representation can be used for shape from shading are described in Section 3.

2.3.1 A curvature-segmenting automorphism group

Each variety $F(\mathbf{I})$ exhibits two useful symmetries that allow it to be partitioned into four isomorphic components, and therefore represented more compactly by just a single component. This partition applies everywhere except on what is generically a single pair of points of $F(\mathbf{I})$. Thus, while we must technically define this partition over a "punctured" variety (what we will call F_0 below), in practice we can typically ignore this distinction, and may, in what follows, drop the subscript. The symmetries follow from those described for extended quadratic patches by Xiong et al.[70] and can be verified by substitution into Eqs (2.3).

Observation 1. *There exists a subset $F_{+}(\mathbf{I}) \subseteq F(\mathbf{I})$ whose orbit under the automorphism group* *generated by*

$$\rho_1 : (f_x, f_y, f_{xx}, f_{xy}, f_{yy}) \mapsto -(f_x, f_y, f_{xx}, f_{xy}, f_{yy})$$

$$\rho_2 : (f_x, f_y, f_{xx}, f_{xy}, f_{yy}) \mapsto \frac{1}{\sqrt{4f_{xy}^2 + (f_{xx} - f_{yy})^2}} \begin{pmatrix} f_x f_{xx} - f_x f_{yy} + 2f_y f_{xy} \\ 2f_x f_{xy} + f_y f_{yy} - f_y f_{xx} \\ f_{xx}^2 - f_{xx} f_{yy} + 2f_{xy}^2 \\ f_{xx} f_{xy} + f_{xy} f_{yy} \\ f_{yy}^2 - f_{xx} f_{yy} + 2f_{xy}^2 \end{pmatrix} \tag{2.9}$$

is precisely $F_0(\mathbf{I})$*, where*

$$F_0(\mathbf{I}) := F(\mathbf{I}) \backslash \boldsymbol{V}(4f_{xy}^2 + (f_{xx} - f_{yy})^2).$$

More details about the morphology of F_0 *are given in Section 2.4.*

For fixed $\mathbf{I}$ and f_x, f_y, there will be an even number, and up to eight, real solutions to Eqns (2.3), each of which corresponds to a local shape that is some combination of concave/convex and saddle/spherical. Figure 2.5 shows an example where the variety's four components are clearly visible, and where the four highlighted surfaces comprise one orbit.

We can choose any of the variety's four components to be the representative one. The component that corresponds to shapes with positive curvatures is convenient to characterize, so we choose that one and call it the *positive shape set*.

Definition 4. *We call the semi-algebraic set*

$$F_+ := \{\mathbf{f} \in F_0 : \ f_{xx} + f_{yy} > 0 \ \ \textit{and} \ \ f_{xx} f_{yy} - f_{xy}^2 > 0\} \tag{2.10}$$

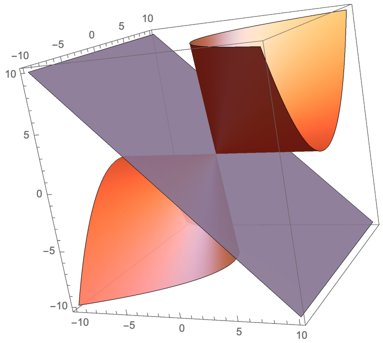


Figure 2.4: The fixed points $\{f_{xx}, f_{xy}, f_{yy}\}$ of ρ_1, ρ_2 when f_x, f_y are left free. These surfaces partition the ambient (curvature) space into regions where shapes have positive and negative Gaussian and mean curvature, respectively.

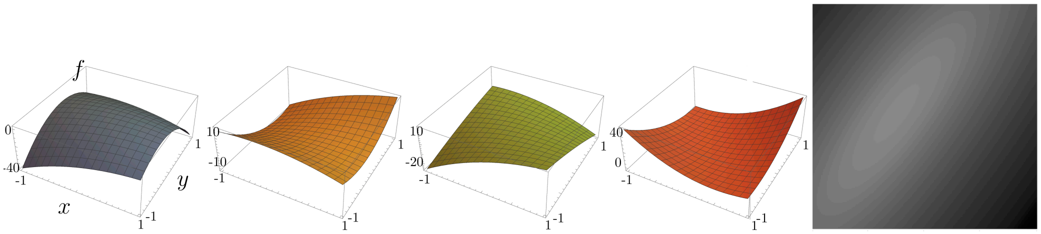


Figure 2.5: Four quadratic surfaces with a common orientation $(f_x, f_y) \approx (5.61, -4.03)$ at the origin. When each surface is lit from a particular direction it produces the image $I(x, y)$ shown right. The four surfaces make up a single orbit of the automorphism group in Observation 1. This figure is comparable to Figure 11 of Kunsberg et al.[50]

the positive shape set. This subset $F_+(\mathbf{I})$ is the set $F_0(\mathbf{I})$ modulo the group action of $\langle \rho_1, \rho_2 \rangle$.

This set $\pi(F_+)$ is depicted in Figure 2.6 among others as the red sheet. It is easily verified for non-planar images that $0 \notin F_+$, that there exist no real fixed points of ρ_2, and that by Definition 1 $4f_{xy}^2 + (f_{xx} - f_{yy})^2 \neq 0$ on $F_+(\mathbf{I})$. Therefore the maps ρ_1, ρ_2 are well-defined on $F_+(\mathbf{I})$.

2.3.2 $F_+(\mathbf{I})$ AS A GRAPH

Our aim is now to find a parsimonious representation of the positive shape subset $F_+(\mathbf{I})$ (and thus of the entire shape set $F(\mathbf{I})$) for general $\mathbf{I} \in \mathcal{I}$, as well as an efficient way to compute this representation for any particular measurement $\mathbf{I}$. Since $F(\mathbf{I})$ and its subset $F_+(\mathbf{I})$ are deter-

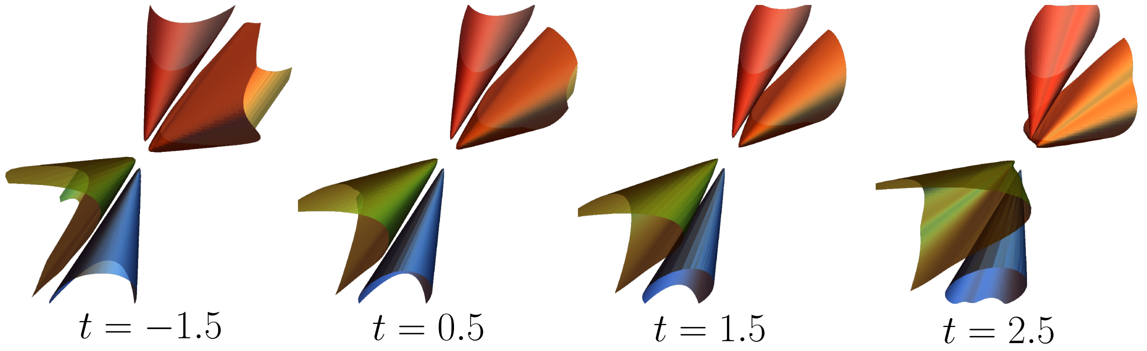

$t=-1.5$ $t=0.5$ $t=1.5$ $t=2.5$

Figure 2.6: Sets $(\pi \circ \Phi)(\mathbf{I})$ for $\mathbf{I} \approx (1-t, \text{-}4.10, \text{-}5.87, \text{-}12.41, \text{-}13.41, \text{-}20.30) + t$, varying t. If $\mathcal{I}$ is the base space of a variety-bundle, we can think of Φ as being a section of that bundle. Thus, this figure depicts fibers of a curve $\mathbf{I}(t) \subset \mathcal{I}$.

mined by $\mathbf{I}$, we may define a map

$$\Phi : \mathbf{I} \mapsto F_+(\mathbf{I}).$$

In order to represent this map Φ in a simpler way, we observe that $\pi(F_+)$ appears to be a (two-dimensional) surface in $\mathbb{R}^3$ that can be traced out by the surface normals. We see evidence of this by plugging in normals f_x, f_y to Equations (2.3) and then solving the resulting system for f_{xx}, f_{xy}, f_{yy}. Based on this we assume that each positive subset can be parametrized by surface orientation (f_x, f_y), so that the map $\Phi(\mathbf{I}) = \{(f_x, f_y, f_{xx}, f_{xy}, f_{yy})\}$ can be decomposed as a graph

$$\Phi(\mathbf{I}) = \{(f_x, f_y, \varphi_{\mathbf{I}}(f_x, f_y))\}, \tag{2.11}$$

with $\varphi_{\mathbf{I}} : \mathbb{R}^2 \mapsto \mathbb{R}^3$ a continuous function.

While we frame it here as an assumption, this decomposition may in fact be exact. The Implicit Function Theorem guarantees existence (and uniqueness) of a function $\varphi(f_x, f_y) = (f_{xx}, f_{xy}, f_{yy})$ in a local neighborhood of every $\mathbf{f}$ for which the Jacobian of system (C_1, C_2, C_3) is nonsingular. This Jacobian is taken with respect to f_{xx}, f_{xy}, f_{yy}, meaning that it is a 3×3 matrix.

While proving that the Jacobian is *always* non-singular—that is, non-singular for any $\mathbf{I} \in \mathcal{I}$ and any real (f_x, f_y)—remains an open problem due to the complicated expressions involved, we conjecture that it is true. Experimentally we have never witnessed a singular Jacobian, and we can prove non-singularity in simplified cases like the following.

Example 1. *Consider the case in which the measurements* $\mathbf{I}$ *satisfy* $I_x = I_y = 0$*, i.e. in which the images normal is parallel to the viewing direction. In this case the determinant of the Jacobian of system* (C_1, C_2, C_3) *with respect to surface curvatures is*

$$\det Jac = \gamma((1+f_y^2)f_{xx} - 2f_xf_yf_{xy} + (1+f_x^2)f_{yy})$$

where $\gamma = -4(f_{xx}f_{yy} - f_{xy}^2)/(1+f_x^2+f_y^2)^5$*. This has a real solution only if its discriminant taken with respect to* f_x*,*

$$discr_{f_x} \det Jac = \gamma((1+f_y^2)(f_{xx}f_{yy} - f_{xy}^2) + (f_{xy}^2 + f_{yy}^2)),$$

is strictly positive. On $F_+(\mathbf{I})$*, the term* $f_{xx}f_{yy} - f_{xy}^2 > 0$*, so* $discr_{f_x} \det Jac < 0$ *over* $\mathbb{R}$*. This implies that there are no points in* $F_+(\mathbf{I})$ *where the implicit function fails.*

Unfortunately, the function Φ is extremely complicated, and it is unlikely that an analytic form can be obtained; we approximate it instead using a data-driven approach in Section 4.

2.4 THE SHAPE OF F.

Depending on the image measurements, F may exhibit one of two morphologies[*], shown in Figures 2.7 and 2.8.

Observation 2. *For $c^*_{A,B}$ a function of $\mathbf{I}/I$ as defined by (2.12), if*

$$\max_{A,B\in\{-1,1\}} \bar{c}^*_{A,B}$$

is nonnegative, then $\rho_2(F_+)$ is a hyperbolic paraboloid. If it is positive, then $\rho_2(F_+)$ is a hyperboloid of one sheet. In either case, F_+ is a hyperboloid of one sheet.

Without loss of generality, assume the first coordinate of $\mathbf{I}$ as follows is 1; we can do this because uniformly scaling $\mathbf{I}$ has no effect on F. The change in morphology of $F_0(\mathbf{I})$ occurs when the sheet $F_+(\mathbf{I})$ intersects the line $\lambda(1,0,-1)$, $\lambda \in \mathbb{R}$. In this case, $f_{xx} = f_{yy}$ and $f_{xy} = 0$. This is the singularity of ρ_2.

Let $\tilde{C}_i(\mathbf{f}) := C_i(\tilde{\mathbf{f}})$ with $\tilde{\mathbf{f}} := \mathrm{diag}(1,1,1/(1+a^2+b^2),1/(1+a^2+b^2),1/(1+a^2+b^2))\mathbf{f}$. We define g as a Gröbner basis for the ideal $\tilde{J} := \sqrt{\langle \tilde{C}_1, \tilde{C}_2, \tilde{C}_3 \rangle} \cap \mathbb{R}[f_{xx}, f_{xy}, f_{yy}]$, evaluated at $x = y = 0$. Such a polynomial $g(f_{xx}, f_{xy}, f_{yy})$ is included in Appendix section A.9 for completeness (if not brevity).

Plugging $f_{xx} = f_{yy}$ and $f_{xy} = 0$ into the Gröebner basis for $\tilde{F}$ yields the polynomial

$k(c;\mathbf{I}) = c^{16} + 6c^{14}I_x^2 + 9c^{12}I_x^4 + 4c^{10}I_x^6 + 2c^{14}I_{xx} + 4c^{12}I_x^2I_{xx} + 2c^{10}I_x^4I_{xx} + c^{12}I_{xx}^2 + 2c^{10}I_x^2I_{xx}^2 + c^8I_x^4I_{xx}^2 - 2c^{12}I_{xy}^2 + 10c^{10}I_x^2I_{xy}^2 - 2c^{10}I_{xx}I_{xy}^2 + 2c^8I_x^2I_{xx}I_{xy}^2 + c^8I_{xy}^4 - 20c^{12}I_xI_{xy}I_y - 28c^{10}I_x^3I_{xy}I_y - 8c^8I_x^5I_{xy}I_y - 20c^{10}I_xI_{xx}I_{xy}I_y + 4c^8I_x^3I_{xx}I_{xy}I_y - 8c^8I_xI_{xx}^2I_{xy}I_y - 12c^8I_xI_{xy}^3I_y + 6c^{14}I_y^2 + 18c^{12}I_x^2I_y^2 +$

[*] This is relevant to anyone using a non-neural surface fitting approach, to help narrow down the family of approximating functions.

$$12c^{10}I_x^4I_y^2 + 14c^{12}I_{xx}I_y^2 + 18c^{10}I_x^2I_{xx}I_y^2 + 4c^8I_x^4I_{xx}I_y^2 + 12c^{10}I_{xx}^2I_y^2 + 4c^8I_{xx}^3I_y^2 + 10c^{10}I_{xy}^2I_y^2 + 36c^8I_x^2I_{xy}^2I_y^2 +$$
$$8c^8I_{xx}I_{xy}^2I_y^2 - 28c^{10}I_xI_{xy}I_y^3 - 16c^8I_x^3I_{xy}I_y^3 - 32c^8I_xI_{xx}I_{xy}I_y^3 + 9c^{12}I_y^4 + 12c^{10}I_x^2I_y^4 + 16c^{10}I_{xx}I_y^4 + 8c^8I_x^2I_{xx}I_y^4 +$$
$$8c^8I_{xx}^2I_y^4 - 8c^8I_xI_{xy}I_y^5 + 4c^{10}I_y^6 + 4c^8I_{xx}I_y^6 + 2c^{14}I_{yy} + 14c^{12}I_x^2I_{yy} + 16c^{10}I_x^4I_{yy} + 4c^8I_x^6I_{yy} + 4c^{12}I_{xx}I_{yy} +$$
$$4c^{10}I_x^2I_{xx}I_{yy} + 2c^{10}I_{xx}^2I_{yy} + 2c^8I_x^2I_{xx}^2I_{yy} - 2c^{10}I_{xy}^2I_{yy} + 8c^8I_x^2I_{xy}^2I_{yy} - 2c^8I_{xx}I_{xy}^2I_{yy} - 20c^{10}I_xI_{xy}I_yI_{yy} -$$
$$32c^8I_x^3I_{xy}I_yI_{yy} - 4c^8I_xI_{xx}I_{xy}I_yI_{yy} + 4c^{12}I_y^2I_{yy} + 18c^{10}I_x^2I_y^2I_{yy} + 8c^8I_x^4I_y^2I_{yy} + 4c^{10}I_{xx}I_y^2I_{yy} + 18c^8I_x^2I_{xx}I_y^2I_{yy} +$$
$$2c^8I_{xy}^2I_y^2I_{yy} + 4c^8I_xI_{xy}I_y^3I_{yy} + 2c^{10}I_y^4I_{yy} + 4c^8I_x^2I_y^4I_{yy} + c^{12}I_{yy}^2 + 12c^{10}I_x^2I_{yy}^2 + 8c^8I_x^4I_{yy}^2 + 2c^{10}I_{xx}I_{yy}^2 +$$
$$c^8I_{xx}^2I_{yy}^2 - 8c^8I_xI_{xy}I_yI_{yy}^2 + 2c^{10}I_y^2I_{yy}^2 + 2c^8I_{xx}I_y^2I_{yy}^2 + c^8I_y^4I_{yy}^2 + 4c^8I_x^2I_{yy}^3.$$

If for a given $\mathbf{I}$, the univariate polynomial $k_{\mathbf{I}}(c)$ has any real roots, then $\mathcal{V} \cap \{\lambda(1,0,-1)\} \neq \{0\}$. Replacing c^2 with $\bar{c}$, we can express $k(\bar{c}) = \bar{c}^4 \cdot O(\bar{c}^4)$. We only care in this case about the existence of positive real roots of this quartic $O(\bar{c}^4)$. The roots of this quartic take the form [†]

$$\bar{c}^*_{A,B} = -\frac{1}{2}\left(3X + W + A\delta + B2\sqrt{2}\sqrt{4I_xI_yI_{xy} + UV + X^2 + A\delta X}\right) \tag{2.12}$$

where $A, B \in \{-1, 1\}$, and

$$\delta = \sqrt{4(I_{xy} + I_xI_y)^2 + (U+V)^2}$$

$$U = I_{xx} - I_{yy} \qquad V = I_x^2 - I_y^2.$$

$$W = I_{xx} + I_{yy} \qquad X = I_x^2 + I_y^2.$$

If $\max_{A,B} \bar{c}^*_{A,B}$ is nonnegative, then $\mathcal{V} \cap \{\lambda(1,0,-1)\} \neq \{0\}$ and the solution set X is hyperbolic, as shown on the left of Figure 2.7. If else, it is the four cones, as shown on the right.

[†] Derivation in BifurcationCriterion.nb

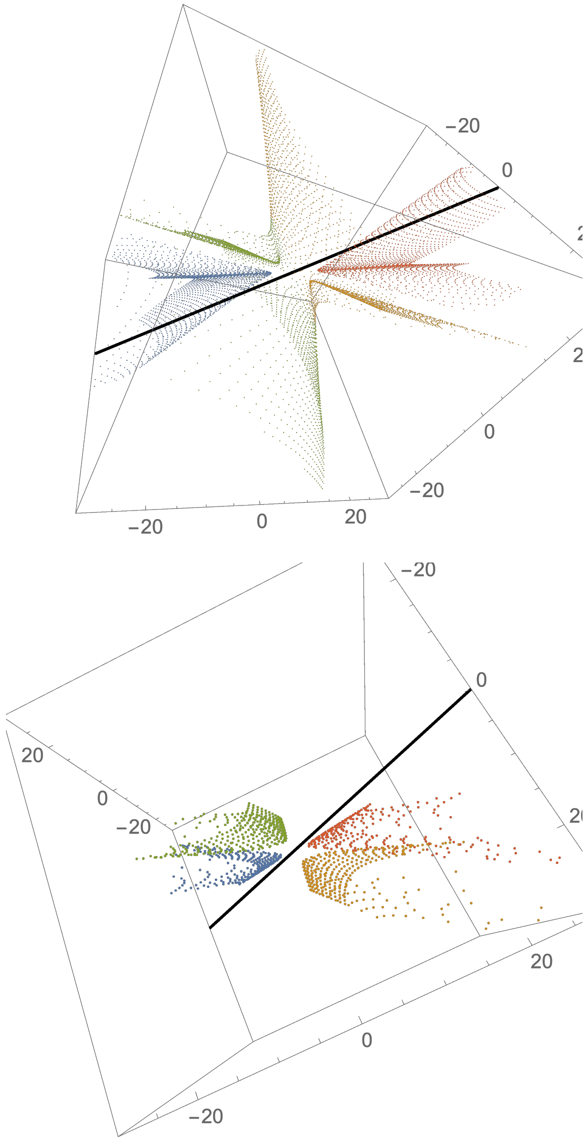


Figure 2.7: The two morphologies of F_0, projected onto $\mathbb{R}^3$ for visualization. The red sheet F_+ on the left figure intersects the black line $(\lambda, 0, -\lambda)$, whereas the right figure's does not.

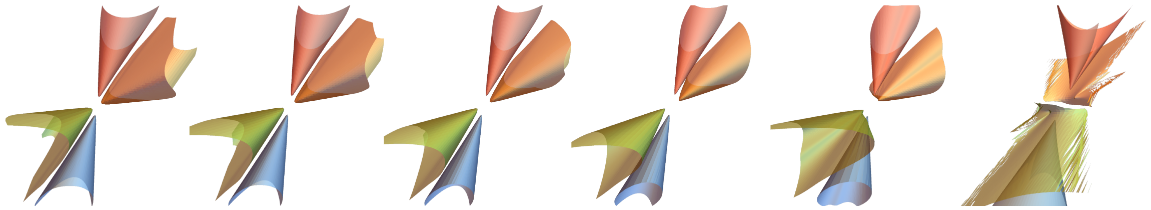


Figure 2.8: As **I** varies smoothly, it may cross the threshold for which $c^*_{A,B}$ in Observation 2 changes from negative to positive.

2.4.1 Related approaches to polynomial solving

Quadric intersections arise as solution sets in many science and engineering applications. This has motivated a large amount of research into solving quadric systems, analytically and approximately, depending on the application. Levin et al.[52,53] compute intersections of two (later, three) quadric surfaces using matrix algebra and parametric equations. Derive parametric expressions for intersection curve (called QSIC) of two surfaces. Idea: find a parametrization surface that contains the QSIC, which reduces to finding the roots of a cubic polynomial derived from the quadric surfaces' subdiscriminants. We believe this is worthy of future study in application to our problem. Kukelova et al.[47] solve the intersection of three quadric surfaces problem in the following way. They absorb one of the three variables into the coefficient ring. They then arrange the degree-two terms on the left side of the equation, and the degree-less-than-two terms on the right side in linear algebraic form. Here the coefficient matrices $M(x)$ are all univariate polynomial valued. Using a neat substitution trick, they reduce the problem to solving $M(x)z = 0$ for z and x. First, solve $\det(M(x)) = 0$ for x. Unfortunately, for the problem described in this thesis, $\det(M(x))$ is still a terribly messy polynomial since it's symbolic in not just x, z but also in other symbolic variables. Kukelova et al.[48] propose an algebraic elimination-based method tailored to solving vision-inspired systems where the image measurements only arise in linear polynomials. This method requires the prior (offline) computation of Gröbner bases. Homotopy continuation aims to solve a system of equations by constructing a smooth family of similar equations that includes a continuous path to an easily-solvable system within the family. An example of a software to do this is Bertini[65,64 7]. For the problem described in this thesis, we hope to approximation solutions to not one system, but to an entire family of systems. Therefore, we find that these existing methods will not suffice, and are forced to find an alternative.

The critical step in formulating the computational theory... is the discovery of additional constraints on the process that are imposed naturally and that limit the result sufficiently to allow a unique solution.

David Marr, *Vision* page 104

3

Consequences for Shape from Shading

We have seen that there is an unavoidable mathematical ambiguity in shape from shading for the per-point case when lighting is unknown. Not only is there not one unique solution to the ill-posed inverse problem, there is an entire two-parameter family of solutions. The proposed point processor $\hat{\Phi}$ transforms image values at a single point $\mathbf{I}$ into an intermediate representation of the consistent local shape-set in the form of a two-dimensional manifold parameterized by surface orientation, $\left(f_x, f_y, \hat{\varphi}_{\mathbf{I}}(f_x, f_y)\right)$. To demonstrate how this representation of per-point shapes can be used for image analysis, we consider two simple, idealized scenarios. In both cases, the per-point ambiguity is resolved (up to a discrete four-way choice) by exploiting additional information or assumptions.

In Chapter 2 we took the shading at a single point and turned that into a surface in $\mathbb{R}^5$ of possible solutions given that local shading measurement. What if we can see another point near the first? Or what if we can see the same point, but with a different light source direction? In each case, we will be presented with an additional set of surface solutions. The next task will be to find the set of solutions that are compatible with *both* measurements simultaneously, i.e. between both points, or between both light source directions. A crucial observation is as follows: the set compatible with both measurements is simply the intersection of their respective solution surfaces. More concretely, we'll be looking at the following scenarios.

1. Uncalibrated photometric stereo: given an unknown surface $\mathbf{f}^*$ and two unknown directional light sources $\mathbf{L}, \mathbf{L}'$, each of which generates an image measurement $\mathbf{I}, \mathbf{I}'$ at pixel $0 \in \mathcal{U}$, our point processor yields two solution sets F, F'. We will study the set

$$X := F \cap F' \subset \mathbb{R}^5.$$

2. Coquadratic[*] stereo: given an unknown surface $\mathbf{f}^*$, an unknown directional light source $\mathbf{L}$, and two pixel values $\mathbf{q}, \mathbf{q}' \in \mathcal{U}$, each of which generates an image measurement $\mathbf{I}, \mathbf{I}'$, our point processor yields two solution sets F, F'. We will study the set

$$X := TF \cap T'F' \subset \mathbb{R}^5,$$

where T, T' are simple but necessary linear transformations.

What do these intersections look like? Can we represent them concisely? What are their dimensions? One challenge is that with such messy generating polynomials as (2.3), many of our most immediate questions cannot be answered analytically and generally, even by computer algebra systems. Our goal in this chapter is to discover what we *can* prove about such

[*] We call two pixels *coquadratic* if they lie on a common quadratic surface patch.

intersections, and which properties then hold for *generic* surface/lighting pairings $(\mathbf{f}^*, \mathbf{L})$.

We purposely introduced our solution set F as an affine algebraic variety, and we will use that in this section. Any two measurements of the same scene should give rise to solution sets that intersect *at least* at the true solution, because the true solution should always solve the forward problem. Thus we expect $\mathbf{f}^* \in X$. But are there other solutions? When does the addition of a second measurement $\mathbf{I}'$ resolve the ambiguity of unknown light source up to finitely many points? This can be in general difficult to answer, but fortunately (extrapolating from our theorem) we will see that the solution sets are zero loci of two polynomial systems arising from the same family. Furthermore, we can study the intersection of these solution sets using algebraic geometry.

The kinds of geometric questions we want to ask are

- Does X have any real solutions? If not, then a physical modelling assumption we made earlier was probably violated: in the coquadratic case, this could mean that the two pixels are not in the same perceptual grouping.
- What is the dimension of the real part of X? If it is zero, then X consists of finitely many points.
- If X is of dimension zero, how many points does it consist of? Can we find analytical expressions for those points? Do they bear any relation to the orbit of $\mathbf{f}^*$ under the automorphism group $\langle \rho_1, \rho_2 \rangle$ from the previous chapter?

Many of these questions are easy to answer when variables like $\mathbf{I}, \mathbf{q}, \mathbf{f}^*$ are mostly fixed; see Examples 2 and 3. However, it quickly becomes much more difficult to answer questions in the fully general case; see Proposition 1. More generalization requires letting more variables be symbolic, which requires operating over larger polynomial rings, with wilder generating polynomials. In this setting many algorithms hang up or fail to converge to a solution. For an example of why generalizing makes things more difficult, see Section . In the following sections, we will push generalization as far as possible; future work might be able to leverage faster computer algebra algorithms as well as problem-specific symmetries to reach deeper generalization than we do here, while using the same general algebraic approach.

3.1 Polynomial complexity explodes when trying to generalize

The goal of this section is to rephrase a well-studied problem in an algebraic way, and to show that for fully general results the solution expressions quickly become unwieldy – see Appendix A.9 for more evidence of this. We will use a lower-dimensional toy example of *photometric stereo when light is known* for intuition; by this we mean that the surface set is temporarily restricted to planar surfaces (i.e. given only by the surface orientation parameters a, b). Given an image measurement *and its corresponding light source direction* at a single point (the origin), the set of consistent surface planes is contained in the zero-locus of

$$\begin{aligned} 0 &:= w^2\left(a^2+b^2+1\right)-1 \\ 0 &:= w(aL_1+bL_2-L_3)+I \end{aligned}$$

This is a system of two equations in three variables, so it defines a hypersurface. This is another example of ambiguity in shape from shading; it differs from our case because the lighting direction here is assumed to be known. Suppose, however that we add just another piece of information: the image intensity I' corresponding to another known light source direction $\mathbf{L}'$. This adds

$$0 := w(aL'_1+bL'_2-L'_3)+I'$$

to the set of constraints, thus making the system well-posed. Although we can now in principle solve this system, we are not guaranteed a concise or easily-analyzed expression for the solution. The two solutions (a, b) to the above example (illustrated in Figure 3.1 as the orange points) are given by:

$$
\begin{aligned}
a \mapsto{} & (I'^2 L_1 L_3 + L_2 L_3 L_1' L_2' - L_1 L_3 L_2'^2 + I^2 L_1' L_3' - L_2^2 L_1' L_3' + L_1 L_2 L_2' L_3' - II'(L_3 L_1' + L_1 L_3') \\
& + (-(I' L_2 - I L_2')^2 (I'^2 (L_1^2 + L_2^2 + L_3^2) - L_2^2 L_1'^2 - L_3^2 L_1'^2 + 2 L_1 L_2 L_1' L_2' - L_1^2 L_2'^2 - L_3^2 L_2'^2 \\
& + 2 L_3 (L_1 L_1' + L_2 L_2') L_3' - (L_1^2 + L_2^2) L_3'^2 - 2II'(L_1 L_1' + L_2 L_2' + L_3 L_3') \\
& + I^2 (L_1'^2 + L_2'^2 + L_3'^2)))^{1/2}) / (I'^2 (L_1^2 + L_2^2) - (L_2 L_1' - L_1 L_2')^2 - 2II'(L_1 L_1' + L_2 L_2') + I^2 (L_1'^2 + L_2'^2)), \\
b \mapsto{} & (I'^3 L_2^2 L_3 - I^3 L_2'^2 L_3' + I L_2' (L_2 L_1' - L_1 L_2')(L_3 L_1' - L_1 L_3') - II'^2 L_2 (2 L_3 L_2' + L_2 L_3') \\
& + I L_1' (-(I' L_2 - I L_2')^2 (I'^2 (L_1^2 + L_2^2 + L_3^2) - L_2^2 L_1'^2 - L_3^2 L_1'^2 + 2 L_1 L_2 L_1' L_2' - L_1^2 L_2'^2 - L_3^2 L_2'^2 \\
& + 2 L_3 (L_1 L_1' + L_2 L_2') L_3' - (L_1^2 + L_2^2) L_3'^2 - 2II'(L_1 L_1' + L_2 L_2' + L_3 L_3') + I^2 (L_1'^2 + L_2'^2 + L_3'^2)))^{1/2} \\
& + I'(-L_2 (L_2 L_1' - L_1 L_2')(L_3 L_1' - L_1 L_3') + I^2 L_2' (L_3 L_2' + 2 L_2 L_3') - L_1 ((I' L_2 - I L_2')^2 (-I'^2 (L_1^2 + L_2^2 + L_3^2) \\
& + L_2^2 L_1'^2 + L_3^2 L_1'^2 - 2 L_1 L_2 L_1' L_2' + L_1^2 L_2'^2 + L_3^2 L_2'^2 - 2 L_3 (L_1 L_1' + L_2 L_2') L_3' + (L_1^2 + L_2^2) L_3'^2 \\
& + 2II'(L_1 L_1' + L_2 L_2' + L_3 L_3') - I^2 (L_1'^2 + L_2'^2 + L_3'^2)))^{1/2})) / ((I' L_2 - I L_2')(I'^2 (L_1^2 + L_2^2) \\
& - (L_2 L_1' - L_1 L_2')^2 - 2II'(L_1 L_1' + L_2 L_2') + I^2 (L_1'^2 + L_2'^2))),
\end{aligned}
$$

and

$$
\begin{aligned}
a \mapsto\ & (I'^2 L_1 L_3 + L_2 L_3 L_1' L_2' - L_1 L_3 L_2'^2 + I^2 L_1' L_3' - L_2^2 L_1' L_3' + L_1 L_2 L_2' L_3' - II'(L_3 L_1' + L_1 L_3') \\
& - (-(I' L_2 - I L_2')^2 (I'^2 (L_1^2 + L_2^2 + L_3^2) - L_2^2 L_1'^2 - L_3^2 L_1'^2 + 2 L_1 L_2 L_1' L_2' - L_1^2 L_2'^2 \\
& - L_3^2 L_2'^2 + 2 L_3 (L_1 L_1' + L_2 L_2') L_3' - (L_1^2 + L_2^2) L_3'^2 - 2 II' (L_1 L_1' + L_2 L_2' + L_3 L_3') \\
& + I^2 (L_1'^2 + L_2'^2 + L_3'^2)))^{1/2}) / (I'^2 (L_1^2 + L_2^2) - (L_2 L_1' - L_1 L_2')^2 - 2 II' (L_1 L_1' + L_2 L_2') + I^2 (L_1'^2 + L_2'^2)), \\
b \mapsto\ & (I'^3 L_2^2 L_3 - I^3 L_2'^2 L_3' + I L_2' (L_2 L_1' - L_1 L_2')(L_3 L_1' - L_1 L_3') - II'^2 L_2 (2 L_3 L_2' + L_2 L_3') \\
& - I L_1' (-(I' L_2 - I L_2')^2 (I'^2 (L_1^2 + L_2^2 + L_3^2) - L_2^2 L_1'^2 - L_3^2 L_1'^2 + 2 L_1 L_2 L_1' L_2' - L_1^2 L_2'^2 - L_3^2 L_2'^2 \\
& + 2 L_3 (L_1 L_1' + L_2 L_2') L_3' - (L_1^2 + L_2^2) L_3'^2 - 2 II' (L_1 L_1' + L_2 L_2' + L_3 L_3') + I^2 (L_1'^2 + L_2'^2 + L_3'^2)))^{1/2} \\
& + I' (-L_2 (L_2 L_1' - L_1 L_2')(L_3 L_1' - L_1 L_3') + I^2 L_2' (L_3 L_2' + 2 L_2 L_3') + L_1 ((I' L_2 - I L_2')^2 (-I'^2 (L_1^2 + L_2^2 + L_3^2) \\
& + L_2^2 L_1'^2 + L_3^2 L_1'^2 - 2 L_1 L_2 L_1' L_2' + L_1^2 L_2'^2 + L_3^2 L_2'^2 - 2 L_3 (L_1 L_1' + L_2 L_2') L_3' + (L_1^2 + L_2^2) L_3'^2 \\
& + 2 II' (L_1 L_1' + L_2 L_2' + L_3 L_3') - I^2 (L_1'^2 + L_2'^2 + L_3'^2)))^{1/2})) / ((I' L_2 - I L_2')(I'^2 (L_1^2 + L_2^2) \\
& - (L_2 L_1' - L_1 L_2')^2 - 2 II' (L_1 L_1' + L_2 L_2') + I^2 (L_1'^2 + L_2'^2))).
\end{aligned}
$$

A cleaner factorization is possible but not obvious, especially in higher-dimensional settings, i.e. where the surfaces might not be planar.

3.2 An algebro-geometric strategy for analyzing intersections X

In each of the aforementioned scenarios, we study the intersection (a geometric object) using its set of generating polynomials (an algebraic object) as a proxy. In order to do this, we require the following components.

Theorem 2 (Ideal-Variety Correspondence)**.** *If k is an algebraically closed field, then there is a*

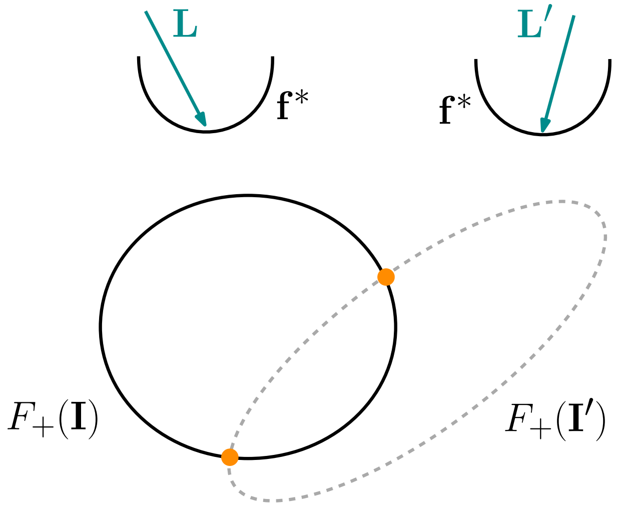


Figure 3.1: Classical calibrated photometric stereo: two simulated images of a quadratic surface $\mathbf{f}^*$ under different lights $\mathbf{L}$, $\mathbf{L'}$, with measurements $\mathbf{I}$, $\mathbf{I}'$ at the same pixel location, give rise to solution sets $F_+(\mathbf{I}), F_+(\mathbf{I}')$, the intersection of which represents the set of surfaces consistent with both light directions. Figure adapted from Marr's *Vision*[55].

bijection between radical ideals and affine varieties.

This means that we can study F and F' individually by studying $\sqrt{\langle C_1, C_2, C_3\rangle}$ and $\sqrt{\langle C_1', C_2', C_3'\rangle}$.

Lemma 1 (§3, Theorem 4[18]). *If I and J are ideals in $k[x_1, \ldots, x_n]$, then $\boldsymbol{V}(I+J) = \boldsymbol{V}(I) \cap \boldsymbol{V}(J)$.*

This means that we can study $F \cap F'$ by studying $\sqrt{\sqrt{\langle C_1, C_2, C_3\rangle} + \sqrt{\langle C_1', C_2', C_3'\rangle}}$.

Lemma 2 (§3, Exercise 7.b.[18]). *If $f_1, \ldots, f_r \in k[x_1, \ldots, x_n]$, where k is an arbitrary field. Then*

$$\sqrt{I+J} = \sqrt{\sqrt{I} + \sqrt{J}}.$$

This means that we can study $F \cap F'$ by studying $\sqrt{\langle C_1, C_2, C_3\rangle + \langle C_1', C_2', C_3'\rangle}$.

Lemma 3 (§3, Corollary 3 [18]). *If $f_1, \dots, f_r \in k[x_1, \dots, x_n]$, then*

$$\langle f_1 \rangle + \cdots + \langle f_r \rangle = \langle f_1, \dots, f_r \rangle.$$

This means that we can study $F \cap F'$ by studying $\sqrt{\langle C_1, C_2, C_3, C_1', C_2', C_3' \rangle}$.

Definition 5 (Radical decomposition of an ideal.). *The radical decomposition of an ideal J is defined as a sequence G of ideals, of length k, for which:*

1. *The ideals of G are prime.*
2. *The intersection of the ideals of G is $\sqrt{J}$.*
3. *G is minimal: no ideal of G contains the intersection of the rest of the ideals of G.*

In Magma, the decomposition G is sorted so that it is always unique.

We'll let $J := \langle C_1, C_2, C_3, C_1', C_2', C_3' \rangle$, and decompose its radical, for which we have established a bijection with X. Each element G_i of the radical decomposition will reveal a distinct irreducible component of the variety. One important note is that the following Magma computations require an algebraically closed field, so our base field will be $\mathbb{C}$. All this means is that we'll ignore those revealed components that are strictly complex, for the context of shape from shading.

Notation 1. *In the next section, we'll be discussing each equation C_i of system* (2.3) *as polynomials in surface* $\mathbf{f}$*, conditioned on true surface* $\mathbf{f}^*$*, true lighting* $\boldsymbol{L}$ *(both a priori unobservable by the viewer), and pixel position* $\boldsymbol{q}$*. That is,* $C_i(\mathbf{f}; \mathbf{f}^*, \boldsymbol{L}, \boldsymbol{q})$*. We will suppress* $\boldsymbol{q}$ *or* $\boldsymbol{L}$ *when their values are obvious from context.*

Each image measurement **I** is by assumption derived from the local image (2.5) given by-

$$I(x,y;\mathbf{f}^*,\mathbf{L}) = -\frac{(f_x^* + f_{xx}^* x + f_{xy}^* y)L_1 + (f_y^* + f_{xy}^* x + f_{yy}^* y)L_2 - L_3}{\sqrt{(f_x^* + f_{xx}^* x + f_{xy}^* y)^2 + (f_y^* + f_{xy}^* x + f_{yy}^* y)^2 + 1}}. \tag{3.1}$$

In the following section, our ultimate goal is that each statement should hold for every conceivable $\mathbf{I} \in \mathcal{I}$, that is, for all $\mathbf{f}^*$ and **L** pairs. Furthermore, in Section 3.4 we want that statement to hold at every pixel. Therefore, in this sense, for the remainder of this chapter **I** will be considered as a function of $\mathbf{f}^*$, **L**, and **q**, and therefore each equation in the system (2.3) will too. It should be noted that **I** will not naturally be a polynomial, or even a rational function. Fortunately, we can perform some convenient manipulation to the **I** vector to convert it to a polynomial (see the transformations given in Section 4.2).

3.3 Uncalibrated photometric stereo from two pixels

Our flavor of uncalibrated photometric stereo tackles the local shape from shading problem given two intensity measurements $\mathbf{I}, \mathbf{I}' \in \mathbb{R}^6$, generated by two unknown light source directions $\mathbf{L}, \mathbf{L}'$. We will examine the possible solutions to Equations (2.3), given various (unknown to the viewer) surface/lighting pairs. In an effort to generalize as much as is currently computationally possible, we let variables L_3, L_3' be free in the following illustrative example.

Example 2. *Consider two local image measurements given by* $\mathbf{f}^* = (-1,2,-4,3,3)$ *at the origin pixel: one with light* $\mathbf{L} = (1,-1,L_3)$ *and the other with light* $\mathbf{L}' = (5/6,3/4,L_3')$. *Then for generic* L_3, L_3', *precisely four real surfaces are consistent with both image measurements, and they are the orbit of* $\mathbf{f}^*$ *under the group* $\langle \rho_1, \rho_2 \rangle$.

Proof. The arguments in Section 3.2 suggest that we study the ideal

$$J := \langle C_1(\mathbf{f}; \mathbf{f}^*, \mathbf{L}), C_2(\mathbf{f}; \mathbf{f}^*, \mathbf{L}), C_3(\mathbf{f}; \mathbf{f}^*, \mathbf{L}), C_1(\mathbf{f}; \mathbf{f}^*, \mathbf{L}'), C_2(\mathbf{f}; \mathbf{f}^*, \mathbf{L}'), C_3(\mathbf{f}; \mathbf{f}^*, \mathbf{L}')\rangle,$$

where $\mathbf{L}$ and $\mathbf{L}'$ are substituted into Equations (2.3) where appropriate. The radical decomposition of J is

$$\begin{aligned}
G_1 &:= \langle e-3, d-3, c+4, b-2, a+1\rangle && \to \{\mathbf{f}^*\},\\
G_2 &:= \langle e+3, d+3, c-4, b+2, a-1\rangle && \to \{\rho_1\mathbf{f}^*\},\\
G_3 &:= \langle d+e/13, c-46e/39, b-8e/39, a-19e/39, e^2-1521/85\rangle && \to \{\rho_2\mathbf{f}^*\} \cup \{\rho_2 \circ \rho_1\mathbf{f}^*\},\\
G_4 &:= \langle L_3 - 2/3, L_3' + 3, ac - 10ad/3 + bd - 10be/3,\\
&\qquad a^2 + b^2 - 2ad - 2be + 1\rangle && \to \text{generically strictly complex},\\
G_5 &:= \langle L_3 - 2/3, L_3' + 3, a^2 + b^2 + 1\rangle && \to \text{strictly complex},\\
G_6 &:= \langle d^2 - ce, ad + be, ac + bd, a^2 + b^2 + 1\rangle && \to \text{strictly complex}.
\end{aligned}$$

The G_i represent all of the complex roots of system in J. The first two components are composed of single real points, the third is composed of two real points, and the following three components are generically strictly complex. By this we mean that G_4 is strictly complex whenever all of the following are satisfied: $L_3 = 2/3, L_3' = -3, 3(ac + bd) = 10(ad + be)$, and $a^2 + b^2 + 1 = 2(ad + be)$. This holds for generic L_3, L_3'. □

A more general result is desired, but the complexity of the expression for J prohibits that for the time being.

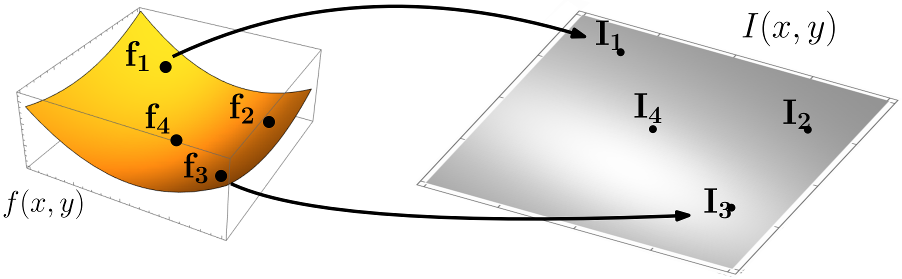


Figure 3.2: Two simulated images of a surface under different lights, with measurements $\mathbf{I}_1, \mathbf{I}_2$ at the same pixel location.

3.4 Coquadratic stereo from two pixels

In this section we assume that our true surface patch is a quadratic, so it is entirely determined by some single $\mathbf{f}^*$. Therefore the measurement $\mathbf{I}(\mathbf{q})$ is a function of $\mathbf{f}^*$ and $\mathbf{L}$ implicitly, and $\mathbf{q}$ explicitly. While the relationships (2.3) hold at every such $\mathbf{q} \in \mathcal{U}$, in the previous chapter we mostly discussed measurements near $\mathbf{q} = 0$. In order to compare measurements from multiple pixels, we need to broaden this framework to explicitly handle arbitrary $\mathbf{q}$. The following observation holds for quadratic local surfaces f. Quadratic surface vectors evaluated away from the origin are related linearly to those at the origin, by the transformation

$$\mathbf{f}(\mathbf{q}) = \begin{pmatrix} 1 & 0 & x & y & 0 \\ 0 & 1 & 0 & x & y \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \mathbf{f}_0 \qquad \text{for} \quad \mathbf{q} = (x,y) \in \mathcal{U}.$$

We will refer to this matrix as $T_\mathbf{q}$, with T_0 the 5×5 identity matrix. For brevity define $\mathbf{f}_0 = \mathbf{f}(0,0)$. Then by definition $\mathbf{f}(\mathbf{q}) = T_\mathbf{q}\mathbf{f}_0$. This change of basis ensures that we are comparing "apples to apples" with respect to our indeterminates $\mathbf{f}$ corresponding to the distinct pixels.

Example 3. *Consider two local image measurements given by* $\mathbf{f}^* = (-1,-1,3,1,-2)$ *and* $\boldsymbol{L} = (1,-1,15)$*: one at pixel center* $\boldsymbol{q} = (0,0)$*, and the other at* $\boldsymbol{q}' = (1,2)$*. Then precisely four real surfaces are consistent with both image measurements, and they are the orbit of* $\mathbf{f}^*$ *under the group* $\langle \rho_1, \rho_2 \rangle$*.*

Proof. Define the ideal

$$J := \Big\langle C_1(\mathbf{f};\mathbf{f}^*,0), C_2(\mathbf{f};\mathbf{f}^*,0), C_3(\mathbf{f};\mathbf{f}^*,0), C_1(T_{\mathbf{q}'}\mathbf{f};\mathbf{f}^*,\mathbf{q}'), C_2(T_{\mathbf{q}'}\mathbf{f};\mathbf{f}^*,\mathbf{q}'), C_3(T_{\mathbf{q}'}\mathbf{f};\mathbf{f}^*,\mathbf{q}')\Big\rangle.$$

The radical decomposition of J is

$$\begin{aligned}
G_1 &:= \langle a+1, b+1, c-3, d-1, e+2\rangle && \to \{\mathbf{f}^*\}\\
G_2 &:= \langle a-1, b-1, c+3, d+1, e-2\rangle && \to \{\rho_1\mathbf{f}^*\}\\
G_3 &:= \langle e^2 - 144/29, a + 7e/12, b - e/4, \\
&\qquad c - 17e/12, d - e/12\rangle && \to \{\rho_2\mathbf{f}^*\} \cup \{\rho_2 \circ \rho_1\mathbf{f}^*\}\\
G_4 &:= \langle a^2 + b^2 + 1, c, d, e\rangle && \to \text{strictly complex}\\
G_5 &:= \langle b^2 + 3be + 9e^2/4 + 1/5, a - 2b - 15e/4, \\
&\qquad c - e/4, d + e/2\rangle && \to \text{strictly complex}
\end{aligned}$$

where G_4 and G_5 generate components that lie strictly in $\mathbb{C}^5 \backslash \mathbb{R}^5$, so we can disregard them.

□

This decomposition tells us that in this case, not only do the points $\{\mathbf{f}^*, \rho_1\mathbf{f}^*, \rho_2\mathbf{f}^*, \rho_2 \circ \rho_1\mathbf{f}^*\}$ all satisfy the three principal equations (2.3) (a fact we could have deduced simply by plugging in those values to those equations), but they are the only real solutions to it, and therefore the real components of the solution set are all zero-dimensional. We could have used Bezout's bound to get an *upper bound* on the number of real solutions, but it would have been quite loose as it would not have differentiated real solutions from complex solutions; all we could ascertain from it would be that there are at most eight real solutions.

We should remember that these intersections we're trying to compute aren't just arbitrary 2-manifolds; they are members of a 2-parameter family of varieties, parametrized by (x, y). They are pairs of fibers of a variety bundle, like shown in Figure 3.3.

Example 4. *A simpler example of such a family is* $\mathfrak{G} := \{g(f_x, f_y; \mathbf{I}) = f_y I - f_x^2 : I \in \mathbb{R}\}$*, where we use the same variable names to show the ideological connection, but without any physical relevance. For any* $I \neq I'$*, the corresponding plane curves intersect only tangentially. The only pairwise intersection of their varieties in* $\mathbb{R}^2$ *satisfies* $f_y I - f_x^2 = 0, f_y I' - f_x^2 = 0$*, so is* $(f_x, f_y) = (0, 0)$*. At this point, the Jacobian matrix* $\begin{pmatrix} -2f_x & I \\ -2f_x & I' \end{pmatrix}$ *for the system is singular. This calculation was trivial because g happens to be a low-degree polynomial of few variables. However, even checking the singularity of the Jacobian of system* (2.3) *is very difficult, and even more so when non-origin pixels are considered.*

Through repeated experiments we have found that the same phenomena seen in Examples 2 and 3 occurs for many choices of $\mathbf{f}^*$, $\mathbf{L}$, $\mathbf{q}$, $\mathbf{q}'$, including when $\mathbf{q}$ is nonzero. We have been unable to compute the primary decomposition in the most general form, wherein all four of these variable vectors are symbolic. Therefore, we investigate other methods of obtaining a similar but slightly weaker statement for general $\mathbf{f}^*$, $\mathbf{L}$, $\mathbf{q}_i$.

When fully expanded (and using the trick to "polynomialize" $\mathbf{I}$ from Section 3.4), each polynomial $C_i(\mathbf{f}; \mathbf{L}) \in \mathbb{R}[\mathbf{f}]$ of system (2.3) can be decomposed as

$$C_i(\mathbf{f}; \mathbf{L}) = L_1 C_i^{(1)}(\mathbf{f}) + L_2 C_i^{(2)}(\mathbf{f}) + L_3 C_i^{(3)}(\mathbf{f})$$

for some $C_i^{(j)} \in \mathbb{R}[\mathbf{f}]$.

We conjecture that the linearity of C_i in $\mathbf{L}$ affords us generalization for free.

Conjecture 1. *Fix any $t = (t_1, t_2, t_3) \in \mathbb{R}^3$. Define the parametrized ideal*

$$J(t) := \sum_{i=1}^{3} \left\langle t_1 C_i^{(1)} + t_2 C_i^{(2)} + t_3 C_i^{(3)} \right\rangle .$$

in $\mathbb{R}[x]$. Let $K := \left\langle C_i^{(j)} \right\rangle_{i=1,2,3}^{j=1,2,3}$. Then $\sqrt{J(t)} = \sqrt{K}$, i.e. $\sqrt{J(t)} = \sqrt{J(t')}\ \forall t \neq t'$.

This remark would imply that the result given in Example 3 holds not only for $\mathbf{L} = (1, -1, 15)$ as stated, but also for arbitrary $\mathbf{L}$.

3.4.1 Generalizing to arbitrary oracle-surfaces and pixel positions

Doing so introduces the tradeoff of (1) greater generality, at the cost of (2) greatly increasing the degree of the $\mathfrak{C}_i$ and the number of variables. For this reason we are unable to compute a similar decomposition to Example 3 for arbitrary $\mathbf{f}^*, \mathbf{L}, \mathbf{q}$. Instead, we relax our statement and prove the following: *The set of solutions that are compatible with arbitrarily many co-quadratic pixels is precisely $\{\mathbf{f}^*, \rho_1\mathbf{f}^*, \rho_2\mathbf{f}^*, \rho_2 \circ \rho_1\mathbf{f}^*\}$.*

If the surface f^* is truly an extended quadratic, then the point $\mathbf{f}^*$ and its orbit should be common to every pair of pixels $\mathbf{q}_i, \mathbf{q}_j$. That is, we should have the orbit as a subset of $F_i \cap F_j$ for every i, j, or equivalently the orbit gives rise to four 2-dimensional $\mathbf{q}$-axis aligned subspaces

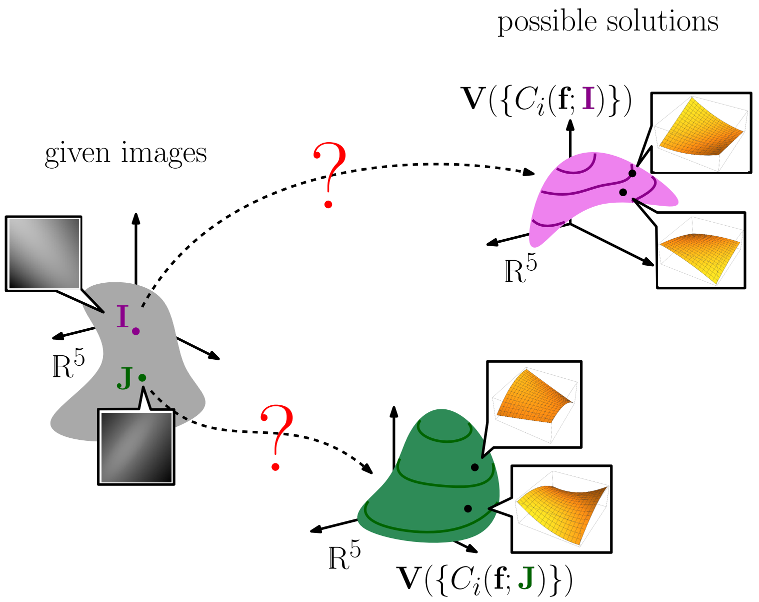


Figure 3.3: A view of $\mathfrak{F}$. Expanding $\mathbf{I}$ into functions of $\mathbf{f}^*, \mathbf{L}$, we can consider our generating system (2.3) as polynomials $\mathfrak{C}_i$ in the ring $\mathbb{F}[\mathbf{q}, \mathbf{f}, \mathbf{f}^*, \mathbf{L}]$ where $\mathbb{F}$ might be $\mathbb{C}$, $\mathbb{Q}$, or $\mathbb{R}$, and its corresponding variety an object $\mathfrak{F} \subset \mathbb{R}^{2+5+5+3}$.

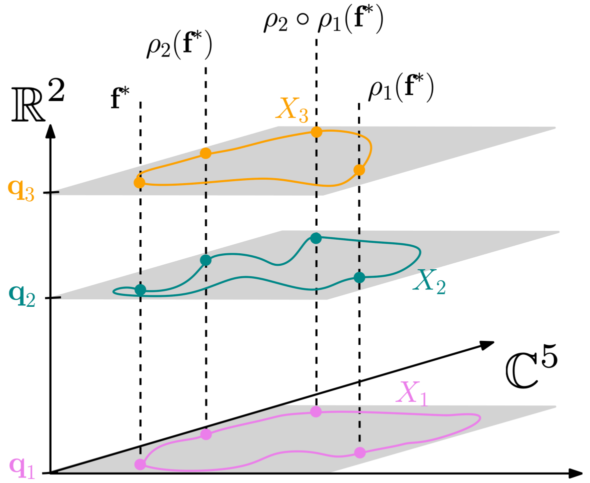


Figure 3.4: The idea behind finding 2-dimensional, $\mathbf{q}$-axis aligned, subspaces in the larger variety. If the surface f is truly an extended quadratic, then the point $\mathbf{f}^*$ and its orbit should be common to every pair of pixels $\mathbf{q}_i, \mathbf{q}_j$. That is, we should have the orbit as a subset of $F_i \cap F_j$ for every i, j, or equivalently the orbit gives rise to four 2-dimensional axis-aligned subspaces in the figure above. This inspires the question: are there any others?

in $\mathfrak{F}$.

We find the common intersection, for a particular **L**, by the following procedure. We rewrite Eqns (2.3) evaluated at this **L** as bivariate polynomials with variables (x, y). Evaluate each polynomial at $(0, 0)$ and concatenate with the polynomial at (x, y). The sets of coefficients of these six polynomials generate an ideal J_∞. These coefficients represent the set of conditions on $\mathbf{f}, \mathbf{f}^*$ such that Eqns (2.3) is zero-valued for *all* **q** in the extended patch.

This is in a similar spirit to identifying the base curve of a pencil of degenerate conics (2.3). If each variety in our family were related *linearly* by **q**, then in fact we could use the general algorithmic solution of Farouki et al.[24] . However, in the language of that work, the complicated relationship between the members of our family would make computation of the Segre characteristic very difficult.

Proposition 1. *Consider the image measurements given by* $\boldsymbol{L} = \{1, -1, 15\}$ *and arbitrary* $\mathbf{f}^* = (a^*, b^*, c^*, d^*, e^*) \in \mathbb{C}^5$ *at as many pixels as desired. Then precisely four real surfaces are consistent with all of those image measurements, and they are the orbit of* $\mathbf{f}^*$ *under the group* $\langle \rho_1, \rho_2 \rangle$*. That is,*

$$F_\infty \cap \mathbb{R}^5 = \{\mathbf{f}^*, \rho_1\mathbf{f}^*, \rho_1\mathbf{f}^*, \rho_2 \circ \rho_1\mathbf{f}^*, \rho_2\mathbf{f}^*\}$$

Proof. The primary decomposition[†] of the radical $\sqrt{J}$ is

$$
\begin{aligned}
G_1 &:= \langle e - e^*, d - d^*, c - c^*, b - b^*, a - a^* \rangle && \to \{\mathbf{f}^*\} \\
G_2 &:= \langle e + e^*, d + d^*, c + c^*, b + b^*, a + a^* \rangle && \to \{\rho_1 \mathbf{f}^*\} \\
G_3 &:= \langle ec^* - 2dd^* + ce^*, dc^* - cd^* + ed^* - de^*, && \\
&\quad cb^* + eb^* + bc^* - 2ad^* - be^*, ea^* - db^* - bd^* + ae^*, && \\
&\quad da^* + eb^* - ad^* - be^*, ca^* + db^* - ac^* - bd^*, && \\
&\quad ce + e^2 - 2d^{*2} + c^*e^* - e^{*2}, d^2 + e^2 - d^{*2} - e^{*2}, && \\
&\quad cd + de - c^*d^* - d^*e^*, bd - ae + b^*d^* - a^*e^*, && \\
&\quad ad + be - a^*d^* - b^*e^*, c^2 - e^2 - c^{*2} + e^{*2}, && \\
&\quad bc + be + b^*c^* - 2a^*d^* - b^*e^*, && \\
&\quad ac + ae - a^*c^* - 2b^*d^* + a^*e^*, a^2 + b^2 - a^{*2} - b^{*2} \rangle && \to \{\rho_2 \mathbf{f}^*\} \cup \{\rho_2 \circ \rho_1 \mathbf{f}^*\} \\
G_4 &:= \langle c^* + e^*, c + e, ea^* + db^* - bd^* - ae^*, && \\
&\quad da^* - eb^* - ad^* + be^*, d^2 + e^2 - d^{*2} - e^{*2}, && \\
&\quad bd - ae - b^*d^* + a^*e^*, ad + be - a^*d^* - b^*e^*, && \\
&\quad a^2 + b^2 - a^{*2} - b^{*2} \rangle && \to \text{strictly complex} \\
G_5 &:= \langle e^*, d^*, c^*, c + e, d^2 + e^2, bd - ae, ad + be, a^2 + b^2 \rangle && \to \text{strictly complex} \\
G_6 &:= \langle c^* + e^*, e, d, c, d^{*2} + e^{*2}, b^*d^* - a^*e^*, && \\
&\quad a^*d^* + b^*e^*, a^{*2} + b^{*2} \rangle && \to \text{strictly complex} \\
G_7 &:= \langle e^*, d^*, c^*, e, d, c \rangle && \to \text{strictly complex}
\end{aligned}
$$

where the last four ideals generate components that lie strictly in $\mathbb{C}^5 \backslash \mathbb{R}^5$, so we can disregard them. □

[†] FanoKH_update.txt

However, the variety $F_\infty := \mathbf{V}(J_\infty)$ isn't quite what we want; it's a subset of our target $F(\mathbf{q}_1) \cap F(\mathbf{q}_2)$. At this point it is still possible that in Figure 3.4 some pair X_i, X_j could still intersect nontrivially. For a two-shot result, we still need to prove that

$$F_{\text{junk}} := (F(\mathbf{q}_1) \cap F(\mathbf{q}_2)) \backslash F_\infty$$

is strictly complex. We saw anecdotal evidence for this in the previous section – in that case, the primary decomposition showed that F_{junk} was empty in $\mathbb{R}$. Exploring this in full generality would be an interesting avenue for future work.

Remark 4. *One may be tempted to simply minimize an objective like*

$$\varphi(\mathbf{I}, \boldsymbol{q}_1, \boldsymbol{q}_2) := \min_{\mathbf{f} \in \mathbb{R}^5} \sum_{j=1}^{2} \sum_{i=1}^{3} \alpha_{i,j} C_i(T_{\boldsymbol{q}_j}(\mathbf{f}), \mathbf{I}(\boldsymbol{q}_j))^2 \tag{3.2}$$

for fixed weights $\alpha_{i,j} > 0$. Unfortunately, this fails spectacularly if the surface patch is not quadratic or when error is present in the image measurements.

Minimizing a functional like (3.2) *is that the absolute magnitude of a generating polynomial is meaningless in the presence of noise. In an algebro-geometric sense, the only thing that matters is whether or not a given polynomial p evaluates to zero on a particular point. Any nonzero value (which is encountered for example when the measurements are noisy) can only indicate that the point does not belong to the variety: not how far away the point is from the variety with respect to whatever ambient space it may be embedded in.*

Remark 5. *Although not covered in this dissertation, it is conceivable that our two second-order image measurements will provide sufficient information to reconstruct the local intensity*

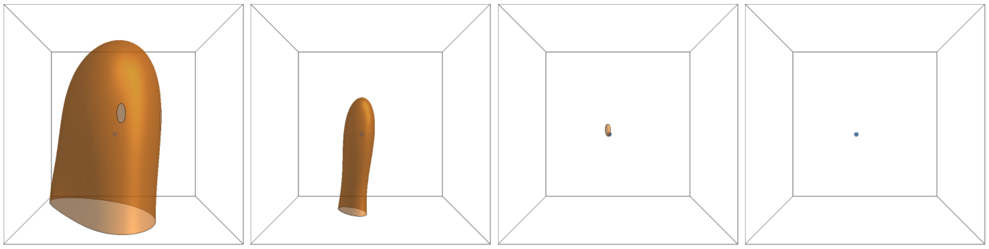


Figure 3.5: Level sets $\varphi((f^*_x, f^*_y, f_{xx}, f_{xy}, f_{yy}), \mathbf{I}, \mathbf{q}_1, \mathbf{q}_2) = \kappa$ for $\kappa = 10, 1, 0.1, 0.01$, left to right, weighted equally. Point plotted is the ground truth curvature corresponding to f^*_x, f^*_y. Figure axes are f_{xx}, f_{xy}, f_{yy}.

function, in which case one could perhaps derive uniqueness results from Xiong et al[71]*.*

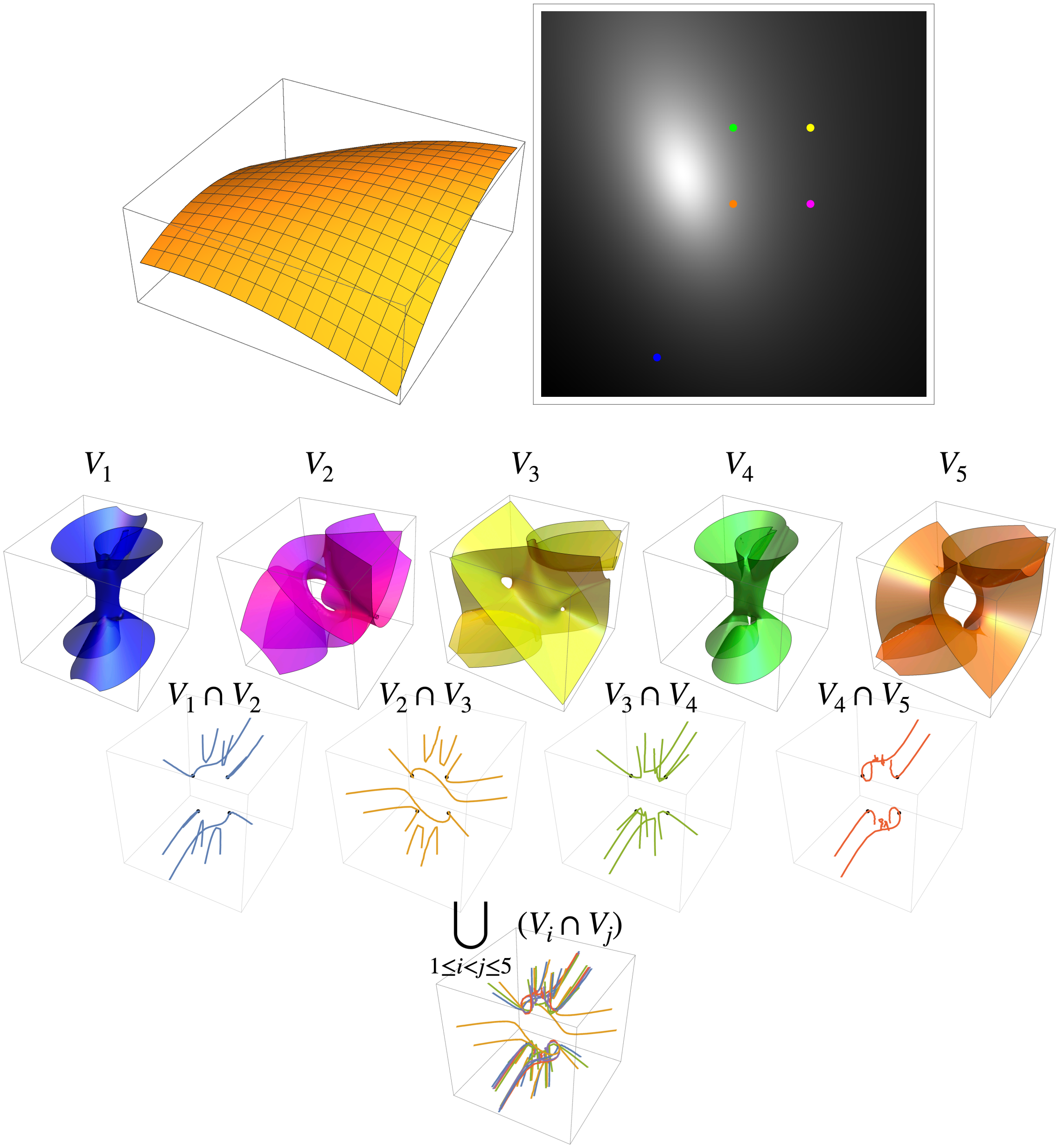


Figure 3.6: For fixed $\mathbf{f}^*$ ($f(x,y)$ shown in upper left), five pixels $\mathbf{q}_i$, and associated measurements $\mathbf{I}_i$ (upper right), the bottom row shows various levels intersections of Gröbner bases V_i (supersets of $\pi(F(\mathbf{I}_i))$. Pairwise intersections of the shape sets each include the ground truth $\mathbf{f}^*$ and its orbit under the automorphism group (denoted by the black points). Notice that the only intersection for *all five* pixels is the same four points. This set $\bigcap_{1 \leq i < j \leq 5}(V_i \cap V_j)$ gives some intuition toward Proposition 1.

4

A Neural Approximator for Φ

Chapter 2 established an implicit representation (2.3) of the local shape set. To use this in practice would involve solving an optimization problem for every image measurement as it is observed. In practice, a faster feedforward shape retrieval method is preferred, meaning one that does not need to perform this optimization with every new measurement. Unfortunately, as is many times the case in algebraic geometry, having access to an implicit representation of the variety does not imply that an explicit representation will be easy to find: there is no general-purpose algebraic "explicitization" procedure. This appears to be the case for the system investigated by this dissertation.

This chapter proposes a neural point processor that is “plug and play” in the sense that given an input $\mathbf{I} \in \mathcal{I}$, the processor returns a piecewise-linear approximation of $F(\mathbf{I})$, without requiring any optimizations to be performed online. We use a class of piecewise-linear functions derived from neural networks to approximate the function $\Phi(\mathbf{I}) = \{(f_x, f_y, \varphi_{\mathbf{I}}(f_x, f_y))\}$ suggested in Equation (2.11).

The function Φ has some properties that facilitate this approximation. Section 4.2 will explore some transforms under which the property of *consistency* is invariant. These are ways in which any consistent pair of measurement and shape $\mathbf{I}, \mathbf{f}$ can be simultaneously transformed without affecting consistency, which allows for a lossless compression of the input space $\mathcal{I}$ from six dimensions to only two. More concretely, we can derive a group of linear transforms T for which $\mathbf{f} \in F_+(\mathbf{I})$ if and only if $T^*\mathbf{f} \in F_+(T\mathbf{I})$. These transforms can then be used to “normalize” the vector $\mathbf{I}$, effectively reducing it to two dimensions.

4.1 Learning manifolds

Nonlinear manifold learning is a broad topic underpinned by the assumption that a given high-dimensional dataset lies on on or near to a low-dimensional manifold[†]. A common goal is dimension reduction: to represent such a dataset by learning a projection to an associated lower-dimensional domain. This work aims to do the inverse: to learn a graph φ. Manifold learning been approached from classical[41,11,12] and neural[15] directions.

Most relevant to our work is learning 2-surfaces embedded in $\mathbb{R}^3$. (1) AtlasNet[31] uses deep learning to approximate the atlas of charts that define a parametrized surface. This

[*] This is actually a principal submatrix of T.

[†] Although we are, strictly speaking, working with a variety and not a manifold, we strongly suspect that this variety is reasonably nonsingular and therefore behaves like a manifold.

work bears some similarity to our point processor in that each neural network is used to generate a continuous image of a plane, i.e. an explicit representation of a chart. In the language of that work, our $\mathbf{I}$ might be the shape feature. One difference is in our problem setup: the point processor introduced in our work learns a family of sets $\{F_+(\mathbf{I}) : \mathbf{I} \in \mathcal{I}\}$ of much simpler topology, where each atlas only has one chart. A technical difference is in the network architecture, where we decompose our neural system into two components. (2) Occupancy networks[56] approximate a surface by training a neural network on a classification task; the decision boundary of the classifier then approximates an implicit representation of the physical surface. Other work in deep implicit functions[29] decomposes three-dimensional physical space into a collection of learned implicit functions. These implicit methods are not directly relevant to the problem in this thesis, as the analytical form of the implicit function is already given by (2.3). A crucial spiritual distinction between these last two works and ours is that rather than representing the height map of the surface itself as a physical manifold, we imagine each candidate height map as being locally parametrizable by its first and second derivatives, and *those parameters* being a *point* on some abstract manifold. Thus, the mapping between Euclidean spaces our network learns does not correspond to the height map of a chair, say, but rather to a moduli space where every point represents a shape.

Since the structure of our target function is a graph, this problem becomes one of function approximation. To this end our choices are classical (wavelets, splines, free-knot splines, etc.) and deep. Neural networks have succeeded practically and theoretically in approximating functions of various kinds: ReLU networks' ability to express rational functions[67], univariate functions on a bounded interval[54], smooth functions in Sobolev spaces[72], and low-degree polynomials[2], to name a few. Probably the most famous result of this kind, the Universal Approximation Theorem[19] states that sigmoidal functions are dense in the class of continuous functions over the unit hypercube. This implies that any continuous function over the unit hypercube can be well-approximated by some single-layer neural network. The caveat is that the width of that layer may be arbitrarily large. These works inspire confidence that neural approximation is a promising route.

Each graph is represented by our point processor by a neural network of some height and width. We currently rely on intuition and experimentation to select network architecture, which lead us to the design choices in Section 4.2.4. Finding an "optimal" architecture[30,76] for these networks, or rigorously analyzing approximation error like the aforementioned works' bounds, is beyond the scope of this thesis.

4.2 Normalizing image measurement space

Ideally our perception machine will be invariant to rotations and translations of the image domain – like what is known as the class of nuisance transformations in computer vision. That is, for a certain family of functions $\mathcal{T}$ from $\mathcal{I}$ to itself, it should estimate the set $F_+(\mathbf{I})$ as accurately as it does $F_+(\mathcal{T}\mathbf{I})$. In this respect, perception needs to be stable, regardless of the viewer's position relative to the image.

Florack et al.[26] remarked that in order to introduce meaningful coordinates to the image domain $\mathcal{U}$, one must consider not only the coordinates, but also groups of coordinate transformations over which perception is to be stable. Only then can one identify those image descriptors that are invariant under this transformation group. In a similar spirit, in this section we will discuss symmetry transformations on maps $\mathcal{U} \mapsto \mathcal{I}$.

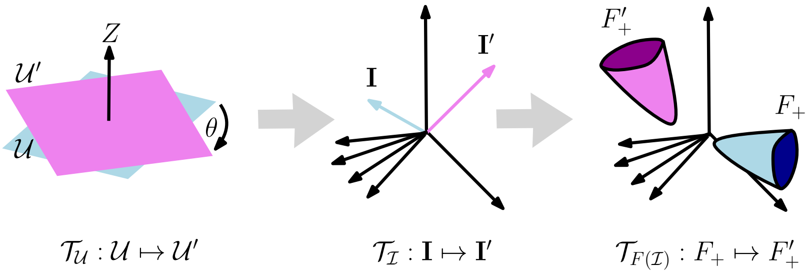

Z
$\mathcal{U}'$
$\mathcal{U}$
θ
$\mathbf{I}$
$\mathbf{I}'$
F'_+
F_+
$\mathcal{T}_{\mathcal{U}} : \mathcal{U} \mapsto \mathcal{U}'$
$\mathcal{T}_{\mathcal{I}} : \mathbf{I} \mapsto \mathbf{I}'$
$\mathcal{T}_{F(\mathcal{I})} : F_+ \mapsto F'_+$

Figure 4.1: A schematic of induced transformations and their notation.

We exploit some symmetries of the shape set, that are due to propagation of transformations on the image domain $\mathcal{U}$, through to transformations on the set of realizable 2-jets $\mathcal{I}$. In this section we describe several linear transformations $\mathcal{T}_{\mathcal{U}}$ of the image plane, and linear transformations $\mathcal{T}_{\mathcal{I}}, \mathcal{T}_{F(\mathcal{I})}$ that they induce. The action of one transformation can propagate to actions on different domains like image measurement space $\mathcal{I}$ and shape space $\mathbb{R}^5$. For example, rotating the camera, surface, and light source around the viewing direction without changing any other parameters induces a rotation of the image, which in turn induces an action on the solution space $\mathcal{F}$.

Three main types of transformations allow us to construct a bijection between the image measurement space $\mathcal{I} \subset \mathbb{R}^6$ and a "compressed" image measurement space $\tilde{\mathcal{I}} \subset \mathbb{R}^2$. We can think of this bijection as a sort of normalization. In Section 4.2.1 we describe such a normalization, and the bijection it induces on shape space. The following relationships are derived from the generating polynomials in the main theorem. Notice that the lighting vector $\mathbf{L}$ plays no role in the following discussion – this is a feature.

Case 1: $\mathcal{T}_{\mathcal{I}}$ is a constant scaling of the image that leaves the contrast unchanged: $\mathcal{T}_{\mathcal{I}} = c \cdot \mathrm{id}$ for $c \in \mathbb{R}_+$. Rescaling the image range does nothing to the shape set $F(\mathbf{I})$; in this case $\mathcal{T}_{F(\mathcal{I})} = \mathrm{id}$.

Case 2: $\mathcal{T}_{\mathcal{U}}$ is a rotation about the origin in the image domain. If $\mathcal{T}_{\mathcal{U}} \in SO(2)$,

$$\mathcal{T}_{\mathcal{U}} := \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix}, \qquad \tilde{\mathcal{T}}_{\mathcal{U}} := \begin{bmatrix} G_{11}^2 & 2G_{11}G_{21} & G_{21}^2 \\ G_{11}G_{12} & G_{11}G_{22} + G_{12}G_{21} & G_{21}G_{22} \\ G_{12}^2 & 2G_{12}G_{22} & G_{22}^2 \end{bmatrix},$$

then

$$\mathcal{T}_{\mathcal{I}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \mathcal{T}_{\mathcal{U}} & 0 \\ 0 & 0 & \tilde{\mathcal{T}}_{\mathcal{U}}^{-1} \end{bmatrix} \qquad \mathcal{T}_{F(\mathcal{I})} = \begin{bmatrix} \mathcal{T}_{\mathcal{U}} & 0 \\ 0 & \tilde{\mathcal{T}}_{\mathcal{U}}^{-1} \end{bmatrix}.$$

Deriving this is very straightforward; it relies on the linearity of the nth derivative.

Case 3: $\mathcal{T}_{\mathcal{U}}$ is isotropic rescaling of the image plane, i.e. $\mathcal{T}_{\mathcal{U}} = c \cdot \text{id}$ for $c \in \mathbb{R}$. Then

$$\mathcal{T}_{\mathcal{I}} = \text{diag}(1, c, c, c^2, c^2, c^2), \qquad \mathcal{T}_{F(\mathcal{I})} = \text{diag}(1, 1, c, c, c).$$

Substitution into (2.3) will confirm this.

Case 4: $\mathcal{T}_{\mathcal{U}}$ is a "transfer" in the image plane between coquadratic pixels, i.e. $\mathcal{T}_{\mathcal{U}} : (x, y) \mapsto (x + dx, y + dy)$ for $dx, dy \in \mathbb{R}$ within an extended quadratic surface. The study of translations is motivated by the desire to amalgamate shading information about multiple coquadratic points on the same surface patch. The idea is that information at those two pixels is not entirely redundant. If $\mathcal{T}_{\mathcal{U}} : (0, 0) \mapsto (p_x, p_y)$, then

$$\mathcal{T}_{F(\mathcal{I})} := \begin{bmatrix} 1 & 0 & p_x & p_y & 0 \\ 0 & 1 & 0 & p_x & p_y \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Case 4 is of a slightly different spirit than the others, and will be revisited in Section 4.3.2.

For Cases 1-3, one can verify that $\mathbf{f} \in F(\mathbf{I})$ if and only if $R_{\mathcal{I}}\mathbf{f} \in F(\mathcal{T}_{\mathcal{I}}\mathbf{I})$, with $R_{\mathcal{I}}$ the principal submatrix of $\mathcal{T}_{\mathcal{I}}$ obtained by removing its first row and column.

4.2.1 Normalizing to $\tilde{\mathcal{I}}$

In this section we choose strategic transformations that allow us to normalize $(I, I_x, I_y, I_{xx}, I_{xy}, I_{yy})$. First, send $I \mapsto 1$ through a constant rescaling. Second, send $I_y \mapsto 0$ through a rotation (see Lemma 4). Third, send $I_x \mapsto 1$ through an isotropic rescaling. Together this yields $\left(1, 0, 1, \tilde{I}_{xx}, \tilde{I}_{xy}, \tilde{I}_{yy}\right)$. We call the image of $\mathcal{I}$ under the composition of these three maps the space $\tilde{\mathcal{I}}$.

Lemma 4. *The rotation-induced transformation $\mathcal{T}_{\mathcal{I}}$ that maps $I_y \mapsto 0$, always maps I_{yy} to a nonpositive value.*

Proof. Under this transformation, $\mathcal{T}_{\mathcal{U}}$ is a rotation defined by

$$G_{11} = \frac{I_x}{\sqrt{I_x^2 + I_y^2}}, \quad G_{12} = \frac{I_y}{\sqrt{I_x^2 + I_y^2}}, \quad G_{21} = \frac{-I_y}{\sqrt{I_x^2 + I_y^2}}, \quad G_{22} = \frac{I_x}{\sqrt{I_x^2 + I_y^2}},$$

so

$$\mathcal{T}_{\mathcal{I}} : \mathbf{I} = \begin{pmatrix} I \\ I_x \\ I_y \\ I_{xx} \\ I_{xy} \\ I_{yy} \end{pmatrix} \mapsto \tilde{\mathbf{I}} = \begin{pmatrix} I \\ +\sqrt{I_x^2 + I_y^2} \\ 0 \\ \frac{I_x^2 I_{xx} + 2 I_x I_{xy} I_y + I_y^2 I_{yy}}{I_x^2 + I_y^2} \\ \frac{I_x^2 I_{xy} + I_x I_y (I_{yy} - I_{xx}) - I_{xy} I_y^2}{I_x^2 + I_y^2} \\ \frac{I_x^2 I_{yy} - 2 I_x I_{xy} I_y + I_{xx} I_y^2}{I_x^2 + I_y^2} \end{pmatrix} =: \begin{pmatrix} \tilde{I} \\ \tilde{I}_x \\ \tilde{I}_y \\ \tilde{I}_{xx} \\ \tilde{I}_{xy} \\ \tilde{I}_{yy} \end{pmatrix}$$

Over $\mathbb{R}$, $\tilde{I}_{yy}$ is the same sign as its numerator $W := I_x^2 I_{yy} - 2 I_x I_{xy} I_y + I_{xx} I_y^2$, so we'll study $\operatorname{sgn}(W)$ as a proxy. Recall that the point processor considers the point $(x, y) = (0, 0)$. By the quadratic patch assumption the true surface $f^*(x, y) = ax + by + \frac{1}{2}\left(c^2 x + 2dxy + ey^2\right)$, so at this point we have $\mathbf{f}^* = (a, b, c, d, e)^T$. Recall also that $I(0, 0) = \rho\mathbf{L} \cdot \mathbf{N}^*(x, y)/||\mathbf{N}^*(x, y)||$, where $\mathbf{N}^* = (-(\partial f/\partial x), -(\partial f/\partial y), 1)^T = (-a, -b, 1)^T$ and WLOG $\rho = 1$. Writing $\mathbf{I}(0, 0)$ as a function of $(\mathbf{f}^*, \mathbf{L}^*)$, we have

$$W = I_x^2 I_{yy} - 2I_x I_{xy} I_y + I_{xx} I_y^2 = \underbrace{\frac{(d^2 - ce)^2}{|(1 + a^2 + b^2)^{9/2}(L_1^2 + L_2^2 + L_3^2)^{3/2}|}}_{\text{strictly positive}} \underbrace{(aL_1 + bL_2 - L_3)}_{\text{strictly negative by assumption}} V,$$

$$\begin{aligned} V := {} & L_1^2 + b^2L_1^2 - 2abL_1L_2 + L_2^2 + a^2L_2^2 + 2aL_1L_3 + 2bL_2L_3 + a^2L_3^2 + b^2L_3^2 \\ = {} & (bL_1 - aL_2)^2 + (L_1 + aL_3)^2 + (L_2 + bL_3)^2. \end{aligned}$$

Since V can be written as the sum of squares, the entirety of W is nonpositive; therefore, $\tilde{I}_{yy}$ is always nonpositive. Furthermore, V can only be zero-valued when $(a, b) = (-L_1/L_3, -L_2/L_3)$, which generically will not occur. □

All in all, these normalization transformations take the form

$$\mathcal{T}_{\mathcal{I}}^{*} = \begin{pmatrix} \frac{1}{I} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{I_x}{I_x^2+I_y^2} & \frac{I_y}{I_x^2+I_y^2} & 0 & 0 & 0 \\ 0 & -\frac{I_y}{I_x^2+I_y^2} & \frac{I_x}{I_x^2+I_y^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{II_x^2}{\left(I_x^2+I_y^2\right)^2} & \frac{2II_xI_y}{\left(I_x^2+I_y^2\right)^2} & \frac{II_y^2}{\left(I_x^2+I_y^2\right)^2} \\ 0 & 0 & 0 & -\frac{II_xI_y}{\left(I_x^2+I_y^2\right)^2} & \frac{I\left(I_x^2-I_y^2\right)}{\left(I_x^2+I_y^2\right)^2} & \frac{II_xI_y}{\left(I_x^2+I_y^2\right)^2} \\ 0 & 0 & 0 & \frac{II_y^2}{\left(I_x^2+I_y^2\right)^2} & -\frac{2II_xI_y}{\left(I_x^2+I_y^2\right)^2} & \frac{II_x^2}{\left(I_x^2+I_y^2\right)^2} \end{pmatrix} \tag{4.1}$$

$$\mathcal{T}_{F(\mathcal{I})}^{*} = \begin{pmatrix} \frac{I_x}{\sqrt{I_x^2+I_y^2}} & \frac{I_y}{\sqrt{I_x^2+I_y^2}} & 0 & 0 & 0 \\ -\frac{I_y}{\sqrt{I_x^2+I_y^2}} & \frac{I_x}{\sqrt{I_x^2+I_y^2}} & 0 & 0 & 0 \\ 0 & 0 & \frac{II_x^2}{\left(I_x^2+I_y^2\right)^{3/2}} & \frac{2II_xI_y}{\left(I_x^2+I_y^2\right)^{3/2}} & \frac{II_y^2}{\left(I_x^2+I_y^2\right)^{3/2}} \\ 0 & 0 & -\frac{II_xI_y}{\left(I_x^2+I_y^2\right)^{3/2}} & \frac{I\left(I_x^2-I_y^2\right)}{\left(I_x^2+I_y^2\right)^{3/2}} & \frac{II_xI_y}{\left(I_x^2+I_y^2\right)^{3/2}} \\ 0 & 0 & \frac{II_y^2}{\left(I_x^2+I_y^2\right)^{3/2}} & -\frac{2II_xI_y}{\left(I_x^2+I_y^2\right)^{3/2}} & \frac{II_x^2}{\left(I_x^2+I_y^2\right)^{3/2}} \end{pmatrix}. \tag{4.2}$$

We see that

$$\mathcal{T}_{\mathcal{I}}^{*} : \mathbf{I} \mapsto \left(1, 1, 0, \frac{I(I_x^2 I_{xx} + 2I_xI_yI_{xy} + I_y^2 I_{yy})}{(I_x^2 + I_y^2)^2}, \frac{I((I_x^2 - I_y^2)I_{xy} + I_xI_y(-I_{xx} + I_{yy}))}{(I_x^2 + I_y^2)^2}, \frac{I(I_x^2 I_{yy} - 2I_xI_yI_{xy} + I_y^2 I_{xx})}{(I_x^2 + I_y^2)^2}\right) \tag{4.3}$$

By using these transformations to pre-process each of our point processor's inputs **I**, and to correspondingly post-process each output shape **f**, we reduce the effective image measurement space from $\mathcal{I} \subset \mathbb{R}^6$ to the normalized space $\tilde{\mathcal{I}} \subset \mathbb{R}^2 \times \mathbb{R}_+ \subset \mathbb{R}^3$. We may thus refer to $\tilde{\mathbf{I}} \in \tilde{\mathcal{I}}$ as a 3-vector.

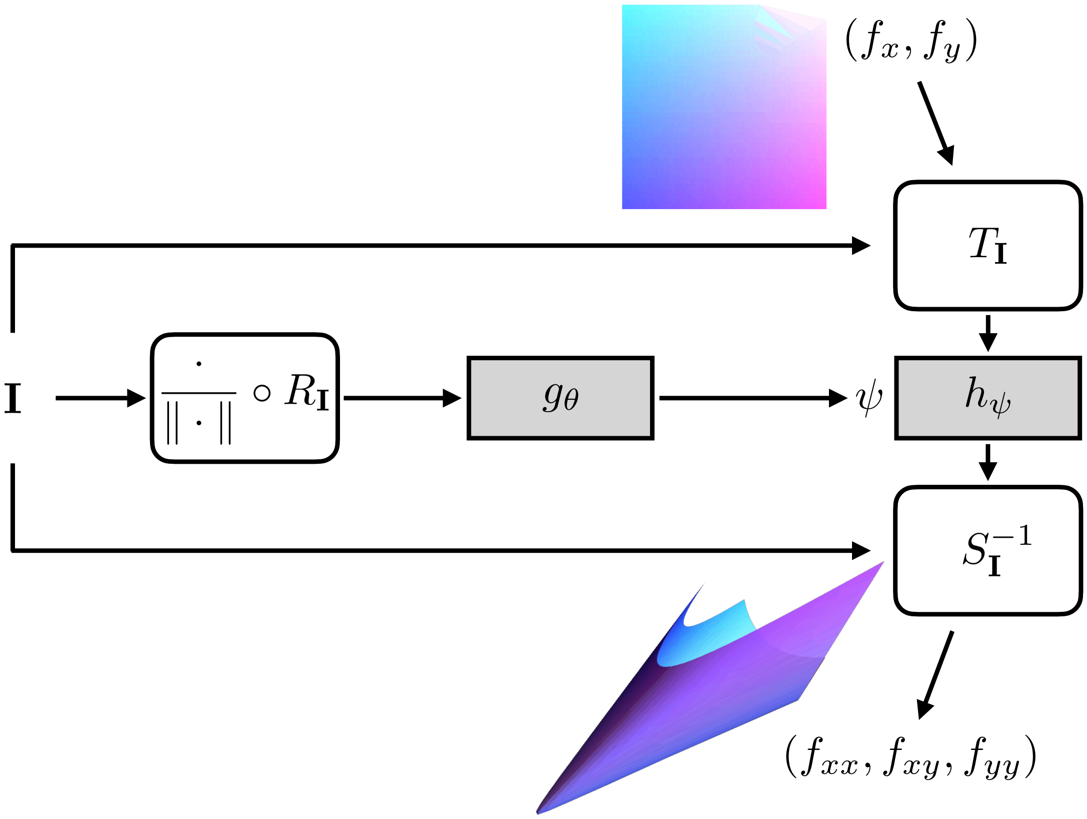


Figure 4.2: The structure of our two-stage network approximator $\hat{\varphi}_{\mathbf{I}}$ for the map from vectors **I** to functions $\varphi_{\mathbf{I}}$. The right shows orientation domain and output samples for the same **I** as in Figure 2.5.

4.2.2 Approximating F_+

In this section we consider the task of approximating the mapping from vectors $\mathbf{I} \in \tilde{\mathcal{I}}$ to functions $\varphi_{\mathbf{I}}$, where we assume the two-dimensional $\mathbf{I}$ discussed in the previous section. Let $\mathcal{S}$ be the class of multilayer perceptrons (MLP) with ReLU activations, of any size so long as the input layer is size 2 and the output layer is size 3. Let $\psi \in \mathbb{R}^M$ denote a concatenation of the weights and biases of the chosen architecture, with M the total number of parameters. Then every MLP $h_\psi \in \mathcal{S}$ is a piecewise-linear map and with a good choice of ψ will well-approximate $\varphi_{\mathbf{I}}$. Then our problem becomes: given $\mathbf{I} \in \tilde{\mathcal{I}}$, find the $\psi \in \mathbb{R}^M$ that best approximates the function $\varphi_{\mathbf{I}}$. Of course, this means that ψ depends on $\mathbf{I}$. It remains to learn a map between $\mathbf{I}$ and $\psi_{\mathbf{I}}$.

4.2.3 Our proposed neural network

One convenient way to learn the map $\mathbf{I} \mapsto \varphi_{\mathbf{I}}$ is to couple a pair of neural networks, with the output of one network providing the weights of the other (see Figure 4.2). That is, we can use

$$\hat{\varphi}_{\mathbf{I}}(f_x, f_y) := h_{g_\theta(\mathbf{I})}\left(f_x, f_y\right), \tag{4.4}$$

where $g_\theta : \mathbb{R}^3 \mapsto \mathbb{R}^M$ is a (fully-connected, few-layer) neural network with tunable weights $\theta \in \mathbb{R}^N$ and $h_\psi : \mathbb{R}^2 \mapsto \mathbb{R}^3$ is a (fully-connected, single-layer) neural network whose weights $\psi \in \mathbb{R}^M$ are provided by the output of g. This means that under the hood, ψ (and therefore $\hat{\varphi}_{\mathbf{I}}$) is a function of θ.

This is convenient because it provides a compact representation that can be efficiently fit to a large set of training samples. We can fit the weights θ by synthetically generating

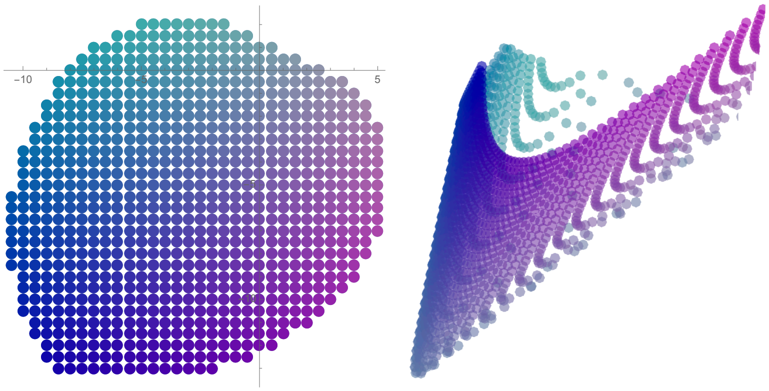


Figure 4.3: Samples of one of the $F_+(\mathbf{I})$ sets, $\mathbf{I} \in \tilde{\mathcal{I}}$, on which the networks train. The left side is uniformly sampled from U, and the right side is obtained by solving the now-square system (2.3) numerically.

many measurements $\mathbf{I}$ and for each one computing many samples $\mathbf{f}$ from the corresponding semi-algebraic set $F_+(\mathbf{I})$ using Theorem 1 and Observation 1. This produces a set of samples $\{(\mathbf{I}^{(j)}, \mathbf{f}^{(i,j)})\}_{i,j}$ that we can use to solve

$$\theta = \arg\min_{\theta} \sum_{j} \sum_{i} \left\| \left(f_{xx}^{(i,j)}, f_{xy}^{(i,j)}, f_{yy}^{(i,j)}\right) - h_{g_\theta(\mathbf{I}^{(j)})}\left(f_{x}^{(i,j)}, f_{y}^{(i,j)}\right) \right\|^2 \tag{4.5}$$

via stochastic gradient descent. We chose Euclidean distance as our loss, but it is not necessarily the most appropriate choice; for example, we could have used another suggested[42] shape-space metric.

4.2.4 Training Data and Network Architecture

Training requires samples $\mathbf{I}^{(j)} \in \tilde{\mathcal{I}}$ as well as samples of the positive set $F_+(\mathbf{I}^{(j)})$ for each 3-vector $\mathbf{I}^{(j)}$. We generate the former by sampling light source directions $\mathbf{L}$ and quadratic patches $\mathbf{f}$ and then applying Eqs. (2.4) and (2.2) (and their spatial derivatives) to render image 2-jets in $\mathcal{I}$, then compress them to get 3-vectors $\mathbf{I}^{(j)}$ in $\hat{\mathcal{I}}$. Specifically, we sample the light sources uniformly from the subset of S^2 contained in an angular radius of $\pi/4$ from the view direction $(0, 0, 1)$. We sample surface orientations f_x, f_y uniformly from the unit disk B^2. By Observation 1 it is sufficient to sample positive curvatures, so we sample f_{xx}, f_{xy}, f_{yy} uniformly from a bounded subset of $\mathbb{R}^3 \cap \{f_{xx}f_{yy} - (f_{xy})^2 > 0\} \cap \{f_{xx} + f_{yy} > 0\}$. We then use these to generate image measurement samples, and compress those into 3-vectors $\mathbf{I}^{(j)}$.

To create samples of the positive shape set $F_+(\mathbf{I}^{(j)})$, we first generate a dense set of sample orientations $\{(f_x^{(i)}, f_y^{(i)})\}$ from the unit disk (these will also serve as input to network h_φ later). Then, for each $\mathbf{I}^{(j)}$ and for each $f_x^{(i)}, f_y^{(i)}$ the corresponding "ground truth" second order shape values $(f_{xy}^{(i,j)}, f_{xy}^{(i,j)}, f_{yy}^{(i,j)})$ are computed by applying a numerical root-finder to (2.3). The result is a training set

$$\left\{(\mathbf{I}^{(j)}, \mathbf{f}^{(i,j)})\right\}_{i,j}.$$

Numerical root finding can be expensive, but the simplification of the domain of g_θ (see the previous section) in our case reduces the computational burden.

Our model effectively solves a continuous set of non-linear regression tasks, where each "task" is a regression from $\mathbb{R}^2$ to $\mathbb{R}^3$ for a particular image 3-vector $\mathbf{I}$. It is related to some approaches to meta-learning, which also try to learn models that handle many tasks simultaneously and adapt to new tasks quickly[25,75]. The differences are that our task-set is continuous, and that we are not concerned with efficient adaptability.

For network $g_\theta : \mathbb{R}^3 \mapsto \mathbb{R}^M$ we use $d_g = 1$ hidden layer with $w_g = 25$ ReLU nodes. For network $h_\psi : \mathbb{R}^2 \mapsto \mathbb{R}^3$ we use one hidden layer with $w_h = 50$ ReLU nodes. The total number of tunable parameters is $N = (3 + 1)w_g + (w_g + 1)M$ in a total of about 32 kilobytes, and once the model is trained, the output description of the shape-set $F(\mathbf{I})$ for any $\mathbf{I}$ consists of

$M = 3(2w_h + 1)$ rational numbers (the size of vector ψ). The entire shape set at each image point is therefore summarized by only $M = 303$ floating-point numbers, in just over 1 kilobyte. Figure 4.4 visualizes the quality of fit for a representative test measurement $\mathbf{I}$ that was not used during training. When this procedure is complete, we perform the inverse transformation mapping $\tilde{\mathcal{I}} \to \mathcal{I}$ to "decompress" the local shape sets.

4.3 Qualitative Analysis and Applications

This point processor provides evidence that bolsters our uniqueness claims for the photometric and the coquadratic applications from Section 3. The two experiments that follow do not use the isometric scaling compressions, but they do use the rotational compression.

4.3.1 Two-Shot Photometric Stereo

The per-point ambiguity can be resolved by capturing additional images of the same surface under distinct light directions. When the light directions are unknown this is called uncalibrated photometric stereo[73,23,9]. In the traditional formulation, which is based purely on surface orientation (f_x, f_y), it requires at least three images under three distinct lights[36]. Our point processor based on second-order shape provides a similar capability with only two input images instead of three.

Consider two measurements $\mathbf{I}_1, \mathbf{I}_2$ generated at the same point from two (unknown) light sources $\mathbf{L}_1, \mathbf{L}_2$. A simulated example is depicted in the top of Figure 4.6. The first measurement $\mathbf{I}_1$ limits the shape to being in the set $F_+(\mathbf{I}_1)$, but within this set all shapes are equally likely. Since the set is parameterized by surface orientation $\left(f_x, f_y, \hat{\varphi}_{\mathbf{I}_1}(f_x, f_y)\right)$, we can visu-

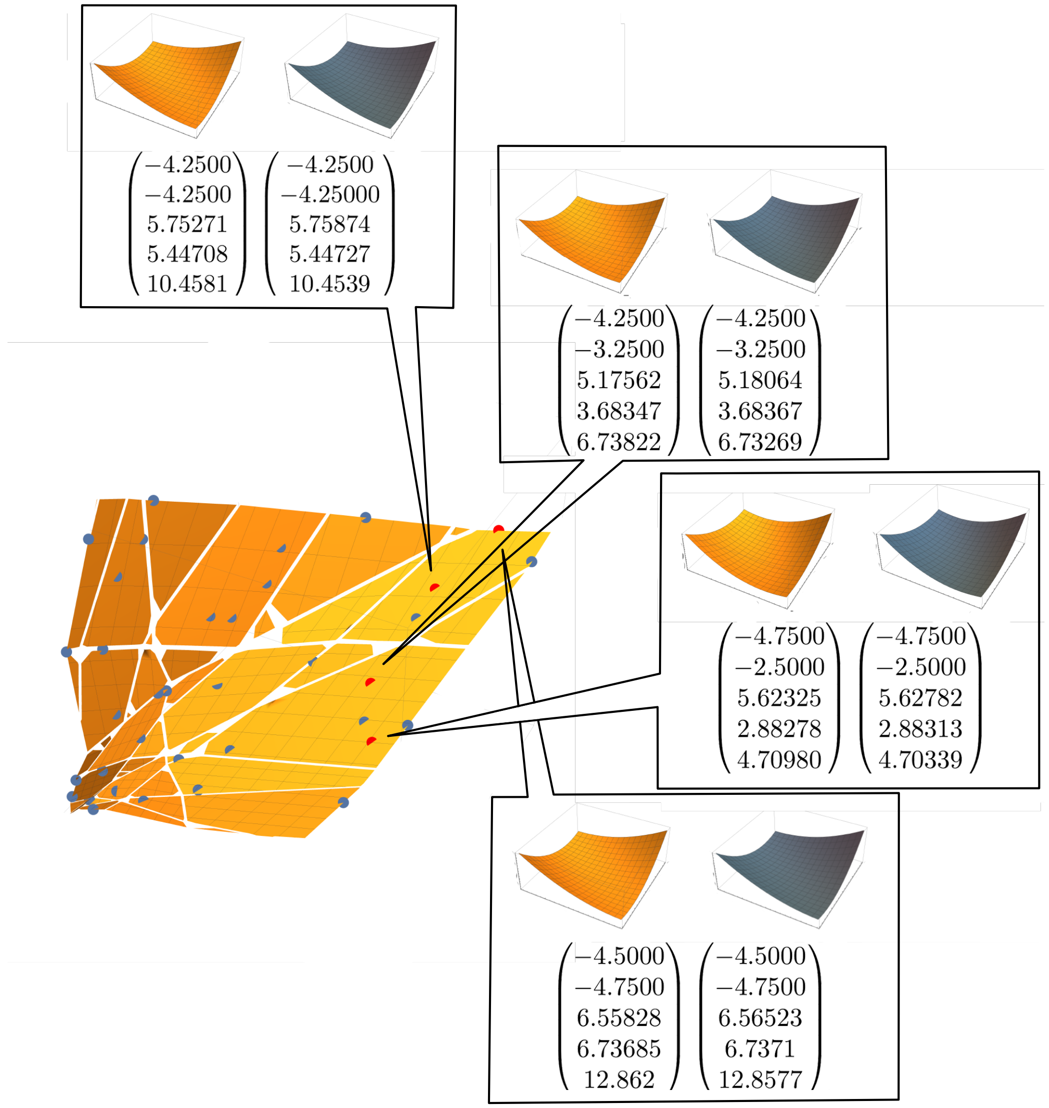

$\begin{pmatrix} -4.2500 \\ -4.2500 \\ 5.75271 \\ 5.44708 \\ 10.4581 \end{pmatrix} \begin{pmatrix} -4.2500 \\ -4.25000 \\ 5.75874 \\ 5.44727 \\ 10.4539 \end{pmatrix}$
$\begin{pmatrix} -4.2500 \\ -3.2500 \\ 5.17562 \\ 3.68347 \\ 6.73822 \end{pmatrix} \begin{pmatrix} -4.2500 \\ -3.2500 \\ 5.18064 \\ 3.68367 \\ 6.73269 \end{pmatrix}$
$\begin{pmatrix} -4.7500 \\ -2.5000 \\ 5.62325 \\ 2.88278 \\ 4.70980 \end{pmatrix} \begin{pmatrix} -4.7500 \\ -2.5000 \\ 5.62782 \\ 2.88313 \\ 4.70339 \end{pmatrix}$
$\begin{pmatrix} -4.5000 \\ -4.7500 \\ 6.55828 \\ 6.73685 \\ 12.862 \end{pmatrix} \begin{pmatrix} -4.5000 \\ -4.7500 \\ 6.56523 \\ 6.7371 \\ 12.8577 \end{pmatrix}$

Figure 4.4: Visualization of the approximator's interpolation error. This figure depicts $\hat{\varphi}_{\mathbf{I}}$ for an $\mathbf{I}$ that was randomly chosen from the convex hull of the training data set, but that was not used as a training sample. The inset shows the four randomly-chosen solutions for which our approximation performs worst, *i.e.* those $\mathbf{f}$ that maximize the error $||\mathbf{f} - (f_x, f_y, \hat{\varphi}_{\mathbf{I}}(f_x, f_y))||_2^2$.

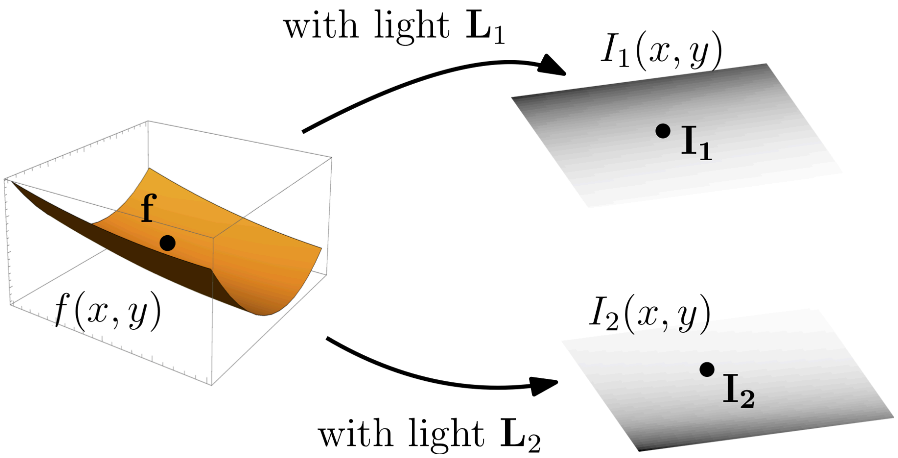


Figure 4.5: Image measurements taken at the same pixel, of the same surface, with two distinct light directions.

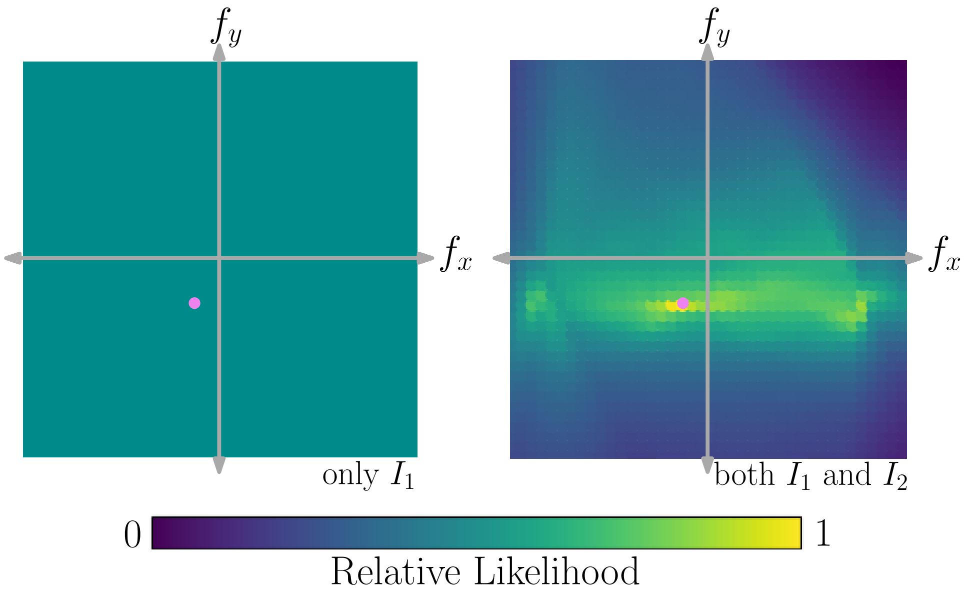


Figure 4.6: Uncalibrated two-shot photometric stereo. The "likelihood" of different shapes using only one measurement (left) or both measurements (right), visualized over the orientation domain. The magenta dot indicates true shape used for simulation.

alize the (uniform) "likelihood" over some reasonably-sized disk of the orientation domain (f_x, f_y). This is shown in the left of Figure 4.6, with the magenta dot indicating the orientation of the latent true shape $\mathbf{f}^*$ that was used for the simulation.

The second measurement $\mathbf{I}_2$ further restricts the shape to being in the intersection of sets $F_+(\mathbf{I}_1)$ and $F_+(\mathbf{I}_2)$. Thus, we can improve the "likelihood" based on how close each shape is to $F_+(\mathbf{I}_1) \cap F_+(\mathbf{I}_2)$. One way to quantify this is

$$L(f_x, f_y) := \left\| \hat{\varphi}_{\mathbf{I}_2}(f_x, f_y) - \hat{\varphi}_{\mathbf{I}_1}(f_x, f_y) \right\|_2^2 \tag{4.6}$$

for (f_x, f_y) in the disk. For our simulation, this updated two-measurement likelihood is shown on the right of Figure 4.6, where it provides a successful identification of the true shape.

Recovering the correct per-point shape (up to the four-way choice) by this simple strategy relies on the intersection $F_+(\mathbf{I}_1) \cap F_+(\mathbf{I}_2)$ being a single point, as seems to be the case for our simulation, as shown in Figure 4.6. Our experiments suggest this is typically the case, but analytically characterizing the conditions for uniqueness may be a worthwhile direction for future work. Also, resolving the four-way choice at each point would require making additional surface continuity assumptions, analogous to how "integrability" is used to reduce the inherent global linear ambiguity in traditional three-shot photometric stereo[73].

4.3.2 Spatial consistency in the coquadratic case

An alternative way to reduce the per-point ambiguity $F(\mathbf{I})$ is to design a 2D array of point processors that are connected together by enforcing surface continuity across an extended region of the input image. As a simple example of this, we consider the scenario in which the

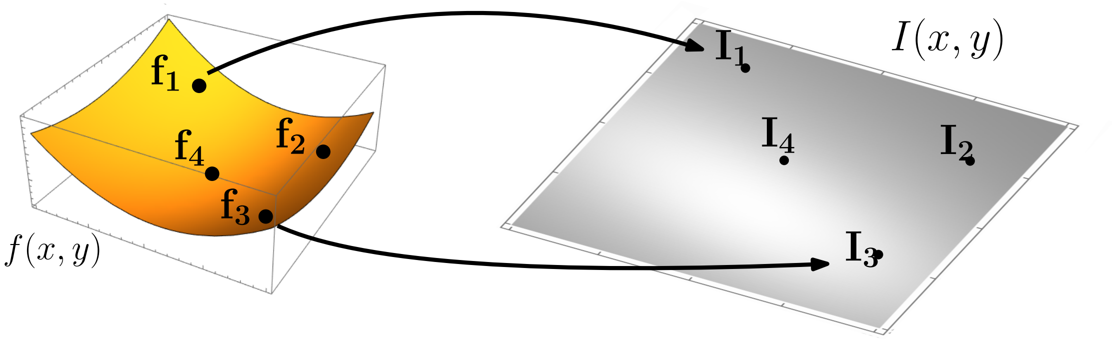


Figure 4.7: Four image measurements taken at four coquadratic points, i.e. four points from the same quadratic surface.

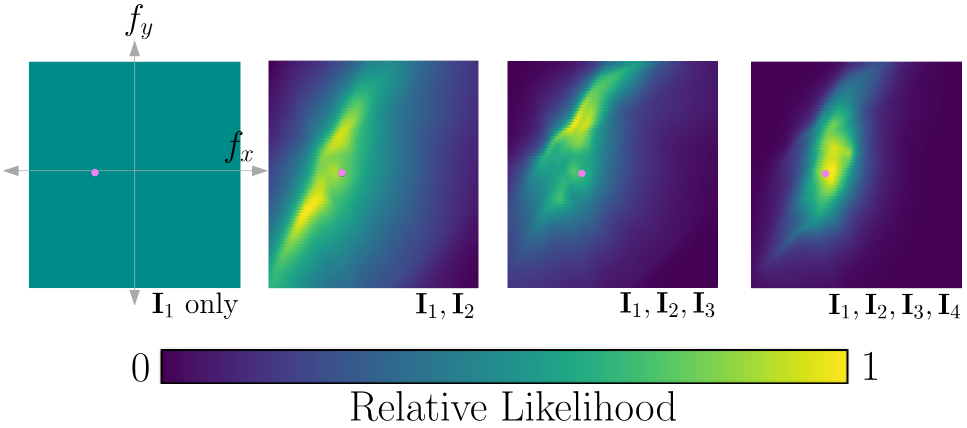


Figure 4.8: Combining shape information at multiple co-quadratic points. Given one measurement $\mathbf{I}_1$, all quadratic shapes in $F_+(\mathbf{I}_1)$ are equally likely. This is depicted on the left as a constant relative likelihood over the domain of function $\hat{\varphi}_\mathbf{I}$. Incorporating measurements $\mathbf{I}_i$ at two or more points modifies the likelihood to have a maximum that is close to the true shape (magenta dot) modulo ρ_1, ρ_2.

entire surface is an extended quadratic function, meaning one that satisfies (2.2) over the entire image $I(x,y)$ with some "true shape" values $\mathbf{f}^* = (f^*_x, f^*_y, f^*_{xx}, f^*_{xy}, f^*_{yy})$.

When the surface is known to be an extended quadratic, any single local shape $\mathbf{f} \in F_+(\mathbf{I}_1)$ at one point, say the image origin, immediately predicts a corresponding local shape $\mathbf{f}'$ at every other point (x,y) in the image, via $(f'_{xx}, f'_{xy}, f'_{yy}) = (f_{xx}, f_{xy}, f_{yy})$ and $(f'_x, f'_y) = (f_x, f_y) + A(x,y) \cdot (f_{xx}, f_{xy}, f_{yy})$ with matrix

$$A(x,y) = \begin{bmatrix} x & y & 0 \\ 0 & x & y \end{bmatrix}. \tag{4.7}$$

As before, we begin with a uniform relative likelihood over the shape set $F_+(\mathbf{I}_1)$ obtained by a single measurement at the origin in an input image of an extended quadratic surface (Figure 4.7). Then given a measurement $\mathbf{I}_2$ at one other point (x_2, y_2), we use that information to update the likelihood over the first set using (4.6), but with the term $\hat{\varphi}_{\mathbf{I}_2}(f_x, f_y)$ replaced by $\hat{\varphi}_{\mathbf{I}_2}\left((f_x, f_y) + A(x_2, y_2) \cdot \hat{\varphi}_{\mathbf{I}_1}(f_x, f_y)\right)$. The updated two-measurement likelihood is shown in Figure 4.8.

Despite our theoretical expectations, in our preliminary experiments the likelihood function L is sometimes not be a single pulse based on two coquadratic pixels alone. This *could* result from approximation errors, namely undesired folds in the piecewise-linear approximation to the varieties. These artifacts might arise from overfitting the network, which could potentially be resolved by increasing the number of training samples. One could resolve this issue by continuing the process: adding information from additional measurements, $\mathbf{I}_3$ at (x_3, y_3) and $\mathbf{I}_4$ at (x_4, y_4), each time updating the likelihood over the original set $L(f_x, f_y)$ by accumulating the intersection errors between $F_+(\mathbf{I}_i)$ and $F_+(\mathbf{I}_1)$. The evolution of this likelihood for three and four points is shown in Figure 4.8. We see that the composite likelihood function achieves its global maximum at a shape $\mathbf{f} \in F_+(\mathbf{I})$ that is very close to $\mathbf{f}^*$ modulo the irreconcilable four-way ambiguity. This is consistent with the area-based analysis of Xiong et al.[71] that proves the uniqueness of shape reconstruction for extended quadratric patches.

5

Conclusion

This dissertation takes preliminary steps toward a deployable point processor for shading that does not require knowledge of lighting at a point or rely on accurate estimates of that lighting. It suggests a new intermediate representation of the set of consistent second-order shapes at each image point, in the form of an explicit differentiable, parametrized two-dimensional manifold. It also provides two simple examples of how this new intermediate representation can be used for shape analysis. The distinguishing feature of this approach is that it has the potential to enable shape processing to succeed in real-world situations where the lighting varies across surfaces and is therefore difficult or impossible to accurately infer.

The contributions of this work are primarily theoretical, and turning this research into practice will require substantial progress in several directions. This may include combining multi-scale derivatives, creating spatial regularization schemes that are suitable for piece-wise smooth surfaces, extending the approach from local second-order shape to local third-order shape, and exploring the ability of the factored network architecture to represent more general (e.g. non-Lambertian) rendering models and to be trained from images instead of algebraic equations.

This dissertation is a jumping-off point for a number of interesting avenues of research. One interesting direction for future work could be the incorporation of contour cues along-side shading cues, or any number of other low-level vision cues. Another next avenue would be to continue this local-to-global approach, addressing the global "consistency-constraining" step. Finally, this work could benefit from a more thorough analysis conducted regarding the ability of the ReLU network proposed here to approximate such systems of rational functions, for example including error bounds. I look forward to the continued study of the inquiries originating in this dissertation.

A

Additional Detail

A.1 Algebra preliminaries

A mathematical prerequisite is Cox et al.[18,17], but other excellent algebro-geometric primers are Hartshorne[35] and Harris[33]. For real algebraic geometry, see Bochnak et al.[10] or Basu et al.[6] For differential topology, see Guillemin and Pollack[32]. For elementary algebra, see Gallian[28]. In this section, we will recall some particularly relevant definitions[35,18]. Operators will be written in bold to distinguish them from sets.

Definition 6. *Let G be a group acting on a set X. The orbit of an element x in X is the set of* *elements in X to which x can be moved by the elements of G. The orbit of x is denoted by* $G \cdot x = \{g \cdot x | g \in G\}$*.*

In this dissertation, we are particularly concerned with a particular such group $G = \langle \rho_1, \rho_2 \rangle$.

Definition 7. *The set of all orbits of X under the action of G is written X/G, and is called the quotient of the action. In geometric situations it may be called the orbit space.*

A.2 Algebro-geometric preliminaries

Definition 8 (affine n-space)**.** *Define affine n-space* $\mathbb{A}^n_k$ *(or simply* $\mathbb{A}^n$*) to be the set of all n-tuples of elements of a fixed field k. In our work, we will use* $k = \mathbb{R}$ *in the exact case, and* $k = \mathbb{Q}$ *in the approximate case. Note that neither of these fields are algebraically closed. We call an element* $q \in \mathbb{A}$ *a point, and if* $q = (a_1, \ldots, a_n)$ *with* $a_\nu \in k$*, then the* a_ν *will be called the coordinates of q.*

Definition 9 (zero-locus)**.** *Let* $A = k[x_1, \ldots, x_n]$ *be the polynomial ring in n variables over k. We can think of the elements* $g \in A$ *as functions* $g : \mathbb{A}^n \to k, q \mapsto g(q) := g(a_1, \ldots, a_n)$ *where* $g \in A$ *and* $q \in \mathbb{A}^n$*. Thus if* $T \subseteq A$ *is a set of polynomials, we can discuss their joint zero-locus*

$$\boldsymbol{V}(T) := \{q \in \mathbb{A}^n | g(q) = 0 \quad \forall g \in T\}.$$

Definition 10 (algebraic set, Zariski topology)**.** *A subset Y of* $\mathbb{A}^n$ *is called an algebraic set if there exists a subset* $T \subseteq A$ *such that* $Y = \boldsymbol{V}(T)$*. We define the Zariski topology on* $\mathbb{A}^n$ *by taking the open subsets to be the complements of the algebraic sets. We also call an algebraic set a variety.*

Definition 11 (variety defined by an ideal). *Given an ideal $I \subseteq k[x_1, \ldots, x_n]$, define the set*

$$\boldsymbol{V}(I) = \{a \in k^n | f(a) = 0 \textit{ for all } f \in I\}.$$

Definition 12 (ideal, elimination ideal). *For any subset $Y \subseteq \mathbb{A}^n$, define the ideal of Y in A by*

$$\boldsymbol{I}(Y) := \{g \in A | g(q) = 0 \ \ \forall q \in Y\}.$$

Given an ideal $I = \langle f_1, \ldots, f_s \rangle \subseteq k[x_1, \ldots, x_n]$, the jth elimination ideal is

$$I_j := I \cap k[x_{j+1}, \ldots, x_n] \subseteq k[x_{j+1}, \ldots, x_n].$$

We may also refer to I_j as the $x_1, \ldots, x_j$-elimination ideal.

Definition 13 (J. Levin). *The pencil of two quadric surfaces is the set of quadric surfaces linearly dependent on them. If the two intersect, then the intersection is contained in every surface of the pencil.*

Definition 14 (radical of an ideal). *Let $I \subseteq k[x_1, \ldots, x_n]$ be an ideal. The radical of I, denoted $\sqrt{I}$, is the set*

$$\{f : f^m \in I \textit{ for some integer } m \geq 1\}.$$

A.3 Other Derivations of the KZs

A variant of Equations 2.3 were first derived by Kunsberg et al.[50] using a differential-geometric approach. We provide an alternate tensor algebra-based derivation that is consistent with both our Theorem 1 and their Equations C.1-C.3.

A.3.1 Tensor Algebra Derivation

We include an alternative derivation for completeness.

Proposition 2.

$$\begin{aligned}\underbrace{(\nabla_d w^d)(\nabla_a I)v^a}_{D_v u \text{ term}} + \underbrace{(\nabla_d \nabla_a I)v^a w^d}_{I_{uu} \text{ term}} = \underbrace{I w^d dN^c_d II_{ca} v^a}_{\langle dNv, dNu\rangle_G \text{ term}} + \underbrace{n_3 l^c w^d (\partial_d H_{ca}) v_a}_{\text{third-order term}} \\ + \underbrace{w^d (\nabla f)^s \Big(II_{sd}(\nabla_a I) - (\nabla_d I) II_{sa} \Big) v^a}_{\text{the } \nabla I \text{ term}}\end{aligned}$$

Proof. Let w^b be an arbitrary vector, and let v^a be a vector field. We begin with the observation that $v^a \nabla_a I = g_{bc} l^c dN^b_a v^a$. Applying the covariant derivative operator to both sides,

$$LHS := \nabla_d(v^a \nabla_a I) = \nabla_d(g_{bc} l^c dN^b_a v^a) =: RHS,$$

and the product rule gives us

$$LHS = (\nabla_d v^a)(\nabla_a I) + (\nabla_d \nabla_a I)v^a.$$

Since $dN^b_a = -g^{be} II_{ea}$ and $g_{bc} g^{be} = \delta^e_c$, we have

$$\begin{aligned}RHS &= \nabla_d(g_{bc} l^c dN^b_a v^a) = \delta^e_c \nabla_D(l^c II_{ea} v^a) = \nabla_D(l^c II_{ca} v^a) \\ &= l^c \nabla_d(II_{ca} v^a) + l^c v^a (\nabla_d II_{ca}) = l^c II_{ca}(\nabla_d v^a) + l^c v^a \nabla_d(II_{ca}) + II_{ca} v^a \nabla_d l^c\end{aligned}$$

and by §2.3 of KZ, we have $\nabla_d l^c = IdN^c_d$. Suppose that v^a is a constant vector field, i.e. it's always "the horizontal direction in the image plane". Notice that $\nabla_d v^a$ can be decomposed into $\partial_d v^a + \Gamma^h_{ed} v^e$, but that $\partial_c v^a = 0$ for a constant vector field v^a. Therefore

$$
\begin{aligned}
RHS &= l^c II_{ca}(\nabla_d v^a) + l^c v^a \nabla_d (II_{ca}) + II_{ca} v^a IdN^c_d \\
&= l^c II_{ca} \Gamma^a_{ed} v^e + l^c v^a \nabla_d (II_{ca}) + II_{ca} v^a IdN^c_d
\end{aligned}
$$

We can also expand $\nabla_d II_{ca} = \partial_d II_{ca} - \Gamma^h_{cd} II_{ha} - \Gamma^h_{ad} II_{ch}$ from the second term above. We have

$$
\begin{aligned}
RHS &= l^c II_{ca} \Gamma^a_{ed} v^e + l^c v^a (\partial_d II_{ca} - \Gamma^h_{cd} II_{ha} - \Gamma^h_{ad} II_{ch}) + II_{ca} v^a IdN^c_d \\
&= l^c II_{ca} \Gamma^a_{ed} v^e + l^c v^a \partial_d II_{ca} - l^c v^a \Gamma^h_{cd} II_{ha} - l^c v^a \Gamma^h_{ad} II_{ch} + II_{ca} v^a IdN^c_d \\
&= l^c II_{ca} \Gamma^a_{ad} v^a + l^c v^a \partial_d II_{ca} - l^c v^a \Gamma^h_{cd} II_{ha} - l^c II_{ca} \Gamma^a_{ad} v^a + II_{ca} v^a IdN^c_d \\
&= l^c v^a \partial_d II_{ca} - l^c v^a \Gamma^h_{cd} II_{ha} + II_{ca} v^a IdN^c_d
\end{aligned}
$$

Since $l^c II_{cd}$ is proportional to $\nabla_d I$, the second term above is proportional to $l^c v^a \left(II_{cd} (\nabla f)^h\right) II_{ha} = (\nabla_d I)(\nabla f)^h II_{ha} v^a = (\nabla_d I)(\nabla f)^h II_{ha} v^a$. Furthermore, $\partial_d II_{ca} = \partial_d (n_3 H_{ca}) = (\partial_d n_3) H_{ca} + (\partial_d H_{ca}) n_3$ for $1/n_3 = \sqrt{1 + ||\nabla f||^2}$. Applying w^d to both sides, we have

$$
\begin{aligned}
(\nabla_d w^d v^a)(\nabla_a I) + (\nabla_d \nabla_a I) v^a w^d = l^c w^d (\partial_d n_3) H_{ca} v^a + n_3 l^c w^d (\partial_d H_{ca}) v_a \\
- w^d (\nabla_d I)(\nabla f)^h II_{ha} v^a + w^d II_{ca} v^a IdN^c_d)
\end{aligned}
$$

Further examining the first right-hand term, $l^c (\partial_d n_3) H_{ca} = l^c (\partial_d n_3)(II_{ca}/n_3) = (\partial_d n_3)(1/n_3)(\nabla_a I) = (\nabla f)^s II_{sd} n_3 (1/n_3)(\nabla_a I) = (\nabla f)^s II_{sd} (\nabla_a I)$. Finally, we will match these terms up to KZ's relations.

$$\underbrace{(\nabla_d w^d)(\nabla_a I)v^a}_{D_v u \text{ term}} + \underbrace{(\nabla_d \nabla_a I)v^a w^d}_{I_{uu} \text{ term}} = \underbrace{w^d (\nabla f)^s II_{sd}(\nabla_a I)v^a}_{\text{one of the } \nabla I \text{ terms}} + \underbrace{n_3 l^c w^d (\partial_d H_{ca})v_a}_{\text{third-order term}}$$
$$- \underbrace{w^d (\nabla_d I)(\nabla f)^h II_{ha} v^a}_{\text{one of the } \nabla I \text{ terms}} + \underbrace{I w^d dN^c_d II_{ca} v^a}_{\langle dNv, dNu \rangle_G \text{ term}}$$

and rearranging,

$$\underbrace{(\nabla_d w^d)(\nabla_a I)v^a}_{D_v u \text{ term}} + \underbrace{(\nabla_d \nabla_a I)v^a w^d}_{I_{uu} \text{ term}} = \underbrace{I w^d dN^c_d II_{ca} v^a}_{\langle dNv, dNu \rangle_G \text{ term}} + \underbrace{n_3 l^c w^d (\partial_d H_{ca})v_a}_{\text{third-order term}}$$
$$+ \underbrace{w^d (\nabla f)^s \Big(II_{sd}(\nabla_a I) - (\nabla_d I) II_{sa} \Big) v^a}_{\text{the } \nabla I \text{ term}} .$$

□

A.4 Linearity in the main polynomials

Restated from the main text, we begin with Equations (2.3),

$$
\begin{aligned}
C_1(\mathbf{f};\mathbf{I}) \;&:= I\left(\left(f_x^2+1\right)f_{xy}^2 + f_{xx}\left(-2f_xf_{xy}f_y + f_{xx}f_y^2 + f_{xx}\right)\right) \\
&\quad + \left(f_x^2+f_y^2+1\right)\left(I_{xx}\left(f_x^2+f_y^2+1\right) + 2I_x(f_xf_{xx}+f_{xy}f_y)\right), \\
C_2(\mathbf{f};\mathbf{I}) \;&:= I\left(\left(f_x^2+1\right)f_{yy}^2 - 2f_xf_{xy}f_yf_{yy} + f_{xy}^2\left(f_y^2+1\right)\right) \\
&\quad + \left(f_x^2+f_y^2+1\right)\left(I_{yy}\left(f_x^2+f_y^2+1\right) + 2I_y(f_xf_{xy}+f_yf_{yy})\right), \\
C_3(\mathbf{f};\mathbf{I}) \;&:= \left(f_x^2+f_y^2+1\right)\left(f_x^2I_{xy} + f_x(f_{xx}I_y + f_{xy}I_x) + f_y(f_{xy}I_y + f_yI_{xy} + f_{yy}I_x) + I_{xy}\right) \\
&\quad + I\left(f_{xy}\left(f_x^2f_{yy} - f_xf_{xy}f_y + f_{yy}\right) + f_{xx}\left(-f_xf_yf_{yy} + f_{xy}f_y^2 + f_{xy}\right)\right)
\end{aligned}
$$

Since these are linear in **I**, they can be rewritten as a matrix equality,

$$
0 = \begin{pmatrix}
\left(f_x^2+1\right)f_{xy}^2 - 2f_xf_yf_{xx}f_{xy} + f_{xx}^2\left(f_y^2+1\right) & 2\gamma(f_xf_{xx}+f_{xy}f_y) & 0 & \gamma^2 & 0 & 0 \\
\left(f_x^2+1\right)f_{yy}^2 - 2f_xf_yf_{xy}f_{yy} + f_{xy}^2\left(f_y^2+1\right) & 0 & 2\gamma(f_xf_{xy}+f_yf_{yy}) & 0 & 0 & \gamma^2 \\
f_{xy}\left(f_x^2f_{yy} - f_xf_yf_{xy} + f_{yy}\right) + f_{xx}\left(f_y^2f_{xy} - f_xf_yf_{yy} + f_{xy}\right) & \gamma(f_xf_{xy}+f_yf_{yy}) & \gamma(f_xf_{xx}+f_yf_{xy}) & 0 & \gamma^2 & 0
\end{pmatrix}\mathbf{I} \tag{A.1}
$$

with $\gamma = f_x^2 + f_y^2 + 1$.

In Section 3 we proved that certain properties hold for observed image measurements generated by ground truth light **L**. The goal of this section is to prove that that specification was not a terrible limitation. That is, does the property shown then also hold for arbitrary **L**? One way this might be solved is using the fact that Equations (2.3) are each linear in **I**, and **I** is also linear in **L**. Therefore the reconstruction problem can be reframed as a condition on a polynomial-valued 6×3 matrix M for which $0 = M(\mathbf{f};\mathbf{f}^*,\mathbf{q}_1,\mathbf{q}_2)\cdot\mathbf{L}$. This matrix can be derived by expanding **I** into its analytical form as a function of **L**.

Example 3 gives the solution set to

$$0 = M\left(\mathbf{f}; \begin{pmatrix} -1 \\ -1 \\ 3 \\ 1 \\ -2 \end{pmatrix}, (-1,1), (1,2)\right) \cdot \begin{pmatrix} 1 \\ -1 \\ 15 \end{pmatrix}.$$

By definition the surface **f** is consistent with $\mathbf{I}(\mathbf{f}^*, \mathbf{q}_1, \mathbf{q}_2)$ if and only if there exists nonzero $\mathbf{L} \in \ker M(\mathbf{f}; \mathbf{f}^*, \mathbf{q}_1, \mathbf{q}_2)$. What can we say about $\ker M(\mathbf{f}; \mathbf{f}^*, \mathbf{q}_1, \mathbf{q}_2)$ when $\mathbf{f}^*, \mathbf{q}_1, \mathbf{q}_2$ are not fixed? The kernel of M is nontrivial if and only if M is not full-rank, which in this case is a rank of 3. This turns the reconstruction problem into a single constraint

$$\mathbf{f} \sim \mathbf{f}^*, \mathbf{q}_1, \mathbf{q}_2 \iff \text{rank} M(\mathbf{f}; \mathbf{f}^*, \mathbf{q}_1, \mathbf{q}_2) < 3.$$

Following this path may provide a way of proving the statements made in this work for observed images from generic ground truth light **L**.

A.5 Sampling from Surface Space

In this work we opted for representing the surface patches as points in Euclidean space, with the usual metric. This choice will at least affect the way in which points are sampled for our data-driven network approach. However, it is possible that there exist more appropriate metrics. For example, one could use the shapelet-space metric proposed in Koenderink et al.[42] for the surface curvature representations.

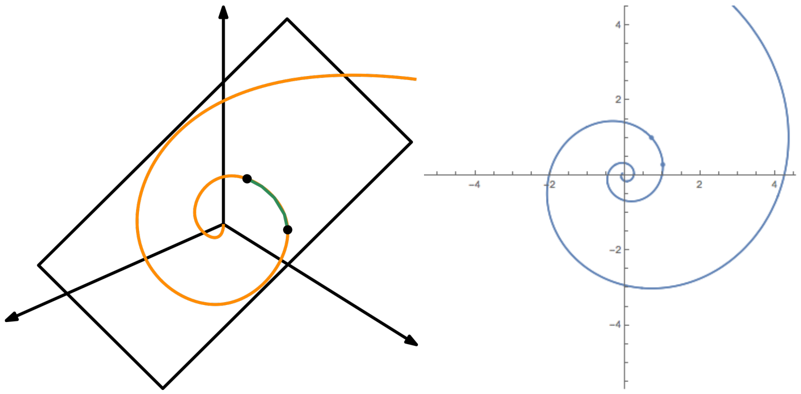
Their Cartesian coordinates are defined by

$$\begin{aligned}
r &= \frac{1}{2}(f_{xx} - f_{yy}) \\
s &= f_{xy} \\
t &= \frac{1}{2}(f_{xx} + f_{yy}),
\end{aligned} \tag{A.2}$$

and polar coordinates by

$$\begin{aligned}
C &= \sqrt{r^2 + s^2 + t^2} \\
\rho &= \arctan\left(\frac{t}{\sqrt{r^2 + t^2}}\right) \\
\varphi &= \frac{1}{2}\arctan\left(\frac{s}{r}\right).
\end{aligned} \tag{A.3}$$

The authors argue that geodesic paths in this shape space will be planar logarithmic spirals, on planes passing through the origin. We expand upon this observation.



$$\mathbb{R}^3 \xrightarrow{T} \mathbb{R}^3 \xrightarrow{\pi} \mathbb{R}^2$$

The image on the right depicts the result of the orthogonal linear transform T and projection π (call this $R := \pi \circ T$) associating the plane to $span\{e_1, e_2\}$. In particular, given any

$x, x' \in \mathbb{R}^3$, we define $n := x \times x'$, $u := \frac{x}{||x||}$, $v := \frac{n}{||n||}$, $w := \frac{x \times n}{||x \times n||}$, and

$$T := \begin{bmatrix} | & | & | \\ u & v & w \\ | & | & | \end{bmatrix}^T = \begin{bmatrix} | & | & | \\ \frac{x}{||x||} & \frac{x \times x'}{||x \times x'||} & \frac{x \times (x \times x')}{||x \times (x \times x')||} \\ | & | & | \end{bmatrix}^T .$$

These spirals can be defined in polar coordinates by $\rho = ae^{b\theta}$. Given two surflets (corresponding to two points $y = Rx, y' = Rx' \in \mathbb{R}^2$ in post-R shape space) there is a unique logarithmic spiral passing through those points and the origin. Express y, y' in polar coordinates, giving (r, φ) and (r', φ'). Then the unique logarithmic spiral is given by equation

$$\rho = r \left(\frac{r}{r'}\right)^{\frac{\theta - \varphi}{\varphi - \varphi'}} \iff a = r \left(\frac{r}{r'}\right)^{\frac{-\varphi}{\varphi - \varphi'}} \quad \text{and} \quad b = \left(\frac{1}{\varphi - \varphi'}\right) \ln\left(\frac{r}{r'}\right) = \frac{1}{\varphi} \ln\left(\frac{r}{a}\right).$$

Since R is orthogonal, the geodesic distance between x, x' will be the same as that of y, y'. That is,

Lemma 5. *The geodesic distance between $x, x' \neq 0$ is*

$$d(x, x') = |r - r'| \sqrt{\left(\frac{\varphi - \varphi'}{\ln\left(\frac{r}{r'}\right)}\right)^2 + 1}$$

where $Rx =: (r, \varphi)$ and $Rx' =: (r', \varphi')$.

Proof. First notice that $\frac{r}{a} = \left(\frac{r}{r'}\right)^{\frac{\varphi}{\varphi - \varphi'}} \implies \ln\left(\frac{r}{a}\right) = \ln\left(\left(\frac{r}{r'}\right)^{\frac{\varphi}{\varphi - \varphi'}}\right) = \frac{\varphi}{\varphi - \varphi'} \ln\left(\frac{r}{r'}\right)$ and

$\left(\frac{r}{a}\right)^{\frac{\varphi'}{\varphi}} = \left(\frac{r}{r'}\right)^{\frac{\varphi'}{\varphi-\varphi'}}$, so

$$\begin{aligned}
d(x,x') = |\mathrm{arclength}_0(y) - \mathrm{arclength}_0(y')| &= \left|\frac{a\sqrt{1+b^2}}{b}\left(e^{b\varphi}-e^{b\varphi'}\right)\right| \\
&= \left|\frac{a\sqrt{1+\left(\frac{1}{\varphi}\ln\left(\frac{r}{a}\right)\right)^2}}{\frac{1}{\varphi}\ln\left(\frac{r}{a}\right)}\left(e^{\frac{1}{\varphi}\ln\left(\frac{r}{a}\right)\varphi}-e^{\frac{1}{\varphi}\ln\left(\frac{r}{a}\right)\varphi'}\right)\right| \\
&= \left|\frac{a\sqrt{\varphi^2+\left(\ln\left(\frac{r}{a}\right)\right)^2}}{\ln\left(\frac{r}{a}\right)}\left(\frac{r}{a}-\left(\frac{r}{a}\right)^{\frac{\varphi'}{\varphi}}\right)\right| = \left|\frac{a\sqrt{\varphi^2+\left(\ln\left(\frac{r}{a}\right)\right)^2}}{\ln\left(\frac{r}{a}\right)}\left(\frac{r}{a}-\left(\frac{r}{a}\right)^{\frac{\varphi'}{\varphi}}\right)\right| \\
&= \left|a\sqrt{\left(\frac{\varphi}{\ln\left(\frac{r}{a}\right)}\right)^2+1}\left(\frac{r}{a}-\left(\frac{r}{a}\right)^{\frac{\varphi'}{\varphi}}\right)\right| = \left|a\sqrt{\left(\frac{\varphi}{\frac{\varphi}{\varphi-\varphi'}\ln\left(\frac{r}{r'}\right)}\right)^2+1}\left(\frac{r}{a}-\left(\frac{r}{a}\right)^{\frac{\varphi'}{\varphi}}\right)\right| \\
&= \left|r\left(\frac{r}{r'}\right)^{\frac{-\varphi}{\varphi-\varphi'}}\left(\left(\frac{r}{r'}\right)^{\frac{\varphi}{\varphi-\varphi'}}-\left(\frac{r}{r'}\right)^{\frac{\varphi'}{\varphi-\varphi'}}\right)\sqrt{\left(\frac{\varphi-\varphi'}{\ln\left(\frac{r}{r'}\right)}\right)^2+1}\right| \\
&= \left|r\left(\left(\frac{r}{r'}\right)^{\frac{\varphi-\varphi}{\varphi-\varphi'}}-\left(\frac{r}{r'}\right)^{\frac{\varphi'-\varphi}{\varphi-\varphi'}}\right)\sqrt{\left(\frac{\varphi-\varphi'}{\ln\left(\frac{r}{r'}\right)}\right)^2+1}\right| \\
&= \left|r\left(1-\left(\frac{r'}{r}\right)\right)\sqrt{\left(\frac{\varphi-\varphi'}{\ln\left(\frac{r}{r'}\right)}\right)^2+1}\right| = \left|r-r'\right|\sqrt{\left(\frac{\varphi-\varphi'}{\ln\left(\frac{r}{r'}\right)}\right)^2+1}
\end{aligned}$$

□

A.6 Code: Cone Fitting in Mathematica

```
model = {#1^2, #2^2, #1, #2, #3, #1 #2, #1 #3, #2 #3, #3^2} & @@@ pp;

lmf = LinearModelFit[ model, {1, x2, y2, x, y, z, xy, xz, yz},

         {x2, y2, x, y, z, xy, xz, yz}];

fun[u_, v_, w_] := lmf["BestFitParameters"].{1, u^2, v^2, u, v, w,

         u v, u w, v w};
```

```
Print[fun[u, v, w] - w^2 == 0];
```

where pp is the point cloud in $\mathbb{R}^3$.

A.7 Code: Radical Decomposition code in Magma

```
R := RationalField();


PQlg<a,b,c,d,e,w,x,y> := PolynomialRing(R, 8, "grevlex");

PQsm<ra,rb,rc,rd,re>  := PolynomialRing(R, 5, "grevlex");


ta := -1;

tb :=  2;

tc := -4;

td :=  3;

te :=  3;


l1 :=  1;

l2 := -1;

l3 := 15;


x1 :=  0;

y1 :=  0;

x2 := -1;

y2 :=  2;


F1 := hom<PQlg -> PQsm |  [ra, rb, rc, rd, re, 1, x1, y1]>;
```

```
F2 := hom<PQlg -> PQsm |  [ra, rb, rc, rd, re, 1, x2, y2]>;

// From GetEqs.nb and saved to KH PC. loads variable "KZh".
load "Equations/KZs_expand_hom.txt";

// Evaluate sys at xi,yi.
sysF1 := F1(KZh);
sysF2 := F2(KZh);

fullSysF := sysF1 join sysF2;

idF := ideal<PQsm | fullSysF>;

dF,_ := Dimension(idF);
primesF := RadicalDecomposition(idF);

allprimes := primesF[1] meet primesF[2] meet primesF[3] meet
        primesF[4] meet primesF[5];
mprimes:=MinimalBasis(allprimes);

// MB of Rad(I) is the same as MB of rad decomp of I
ridF:=Radical(idF);
mridF:=MinimalBasis(ridF);
```

A.8 Code: Finding 2D Subspaces in Magma

```
tCoefs := function(f, t)

   fout := f;

   for j in t do

      fout := &cat[Coefficients(fout[i], j) : i in [1..#fout]];

   end for;

   return Exclude(Seqset(fout),0);

end function;


R := RationalField();


PQog<a, b, c, d, e, w, x, y, ta, tb, tc, td, te> :=

        PolynomialRing(R, 13, "grevlex");

PQlg<t1, t2, s, pa, pb, pc, pd, pe, tta, ttb, ttc, ttd, tte> :=

        PolynomialRing(R, 13, "grevlex");

PQsm<ppa, ppb, ppc, ppd, ppe, pta, ptb, ptc, ptd, pte> :=

        PolynomialRing(R, 10, "grevlex");


l1 :=  1;

l2 := -1;

l3 := 15;

w  := 1;


num  := [-1, 2, -4, 3, 3];

sym  := [tta, ttb,ttc,ttd,tte];
```

```
sv   := RSpace(PQlg, 5) ! sym; // Cast as a 5-vector in PQlg space.


load "Equations/KZs_expand_hom.txt";


ContractToPQsm := hom<PQlg -> PQsm | 0, 0, 0, ppa, ppb, ppc,
        ppd, ppe, pta, ptb, ptc, ptd, pte>;


FanoMat := Transpose(RMatrixSpace(PQlg, 13, 3) !
                               [0, 0,  pa,
                                0, 0,  pb,
                                0, 0,  pc,
                                0, 0,  pd,
                                0, 0,  pe,
                                0, 0,  1,
                                1, 0,  0,
                                0, 1,  0,
                                0, 0,  sv[1],
                                0, 0,  sv[2],
                                0, 0,  sv[3],
                                0, 0,  sv[4],
                                0, 0,  sv[5]]);
F  := hom<PQog -> PQlg | Eltseq(RSpace(PQlg, 3) ! [t1, t2, s] * FanoMat)>;
Fh := [elem : elem in F(KZh)];
Fc := tCoefs(Fh, {t1, t2, s});


fullSys := ContractToPQsm(Fc);
```

```
fano := ideal<PQsm | fullSys>;

fanoMB := MinimalBasis(fano);

GroebnerBasis(fano);

print(fano);


print RadicalDecomposition(fano);
```

A.9 A Gröbner Basis for the second elimination ideal of $\langle C_1, C_2, C_3\rangle$

This polynomial is relevant to Section 2.4. Inclusion of this polynomial in its entirety is intended to convey the difficulty in working with or analyzing it in the fully-symbolic case, i.e. where **I** is allowed to vary freely. This polynomial can be derived from taking the second elimination ideal generated by the three equations C_1, C_2, C_3 in the main theorem. Buchberger's algorithm was used to generate this polynomial; because finding S-polynomials involves a polynomial multiplication at each iteration, it can return an expression that is quite unwieldy. This equation is significant because its zero locus is a fairly tight superset of the set $\pi(X; \mathbf{I})$.

$$g(c,d,e;\mathbf{I}) = d^{16} - 8cd^{14}e + 28c^2d^{12}e^2 - 56c^3d^{10}e^3 + 70c^4d^8e^4 - 56c^5d^6e^5 + 28c^6d^4e^6 - 8c^7d^2e^7 + c^8e^8 + 6d^{14}I_x^2 - 36cd^{12}eI_x^2 + 90c^2d^{10}e^2I_x^2 + 6d^{12}e^2I_x^2 -$$
$$120c^3d^8e^3I_x^2 - 36cd^{10}e^3I_x^2 + 90c^4d^6e^4I_x^2 + 90c^2d^8e^4I_x^2 - 36c^5d^4e^5I_x^2 - 120c^3d^6e^5I_x^2 + 6c^6d^2e^6I_x^2 + 90c^4d^4e^6I_x^2 - 36c^5d^2e^7I_x^2 + 6c^6e^8I_x^2 + 9d^{12}I_x^4 - 36cd^{10}eI_x^4 + 54c^2d^8e^2I_x^4 +$$
$$18d^{10}e^2I_x^4 - 36c^3d^6e^3I_x^4 - 72cd^8e^3I_x^4 + 9c^4d^4e^4I_x^4 + 108c^2d^6e^4I_x^4 + 9d^8e^4I_x^4 - 72c^3d^4e^5I_x^4 - 36cd^6e^5I_x^4 + 18c^4d^2e^6I_x^4 + 54c^2d^4e^6I_x^4 - 36c^3d^2e^7I_x^4 + 9c^4e^8I_x^4 + 4d^{10}I_x^6 -$$
$$8cd^8eI_x^6 + 4c^2d^6e^2I_x^6 + 12d^8e^2I_x^6 - 24cd^6e^3I_x^6 + 12c^2d^4e^4I_x^6 + 12d^6e^4I_x^6 - 24cd^4e^5I_x^6 + 12c^2d^2e^6I_x^6 + 4d^4e^6I_x^6 - 8cd^2e^7I_x^6 + 4c^2e^8I_x^6 + 2d^{14}I_{xx} - 12cd^{12}eI_{xx} + 30c^2d^{10}e^2I_{xx} +$$
$$2d^{12}e^2I_{xx} - 40c^3d^8e^3I_{xx} - 12cd^{10}e^3I_{xx} + 30c^4d^6e^4I_{xx} + 30c^2d^8e^4I_{xx} - 12c^5d^4e^5I_{xx} - 40c^3d^6e^5I_{xx} + 2c^6d^2e^6I_{xx} + 30c^4d^4e^6I_{xx} - 12c^5d^2e^7I_{xx} + 2c^6e^8I_{xx} + 4d^{12}I_x^2I_{xx} -$$
$$16cd^{10}eI_x^2I_{xx} + 24c^2d^8e^2I_x^2I_{xx} + 8d^{10}e^2I_x^2I_{xx} - 16c^3d^6e^3I_x^2I_{xx} - 32cd^8e^3I_x^2I_{xx} + 4c^4d^4e^4I_x^2I_{xx} + 48c^2d^6e^4I_x^2I_{xx} + 4d^8e^4I_x^2I_{xx} - 32c^3d^4e^5I_x^2I_{xx} - 16cd^6e^5I_x^2I_{xx} + 8c^4d^2e^6I_x^2I_{xx} +$$
$$24c^2d^4e^6I_x^2I_{xx} - 16c^3d^2e^7I_x^2I_{xx} + 4c^4e^8I_x^2I_{xx} + 2d^{10}I_x^4I_{xx} - 4cd^8eI_x^4I_{xx} + 2c^2d^6e^2I_x^4I_{xx} + 6d^8e^2I_x^4I_{xx} - 12cd^6e^3I_x^4I_{xx} + 6c^2d^4e^4I_x^4I_{xx} + 6d^6e^4I_x^4I_{xx} - 12cd^4e^5I_x^4I_{xx} +$$
$$6c^2d^2e^6I_x^4I_{xx} + 2d^4e^6I_x^4I_{xx} - 4cd^2e^7I_x^4I_{xx} + 2c^2e^8I_x^4I_{xx} + d^{12}I_{xx}^2 - 4cd^{10}eI_{xx}^2 + 6c^2d^8e^2I_{xx}^2 + 2d^{10}e^2I_{xx}^2 - 4c^3d^6e^3I_{xx}^2 - 8cd^8e^3I_{xx}^2 + c^4d^4e^4I_{xx}^2 + 12c^2d^6e^4I_{xx}^2 + d^8e^4I_{xx}^2 -$$
$$8c^3d^4e^5I_{xx}^2 - 4cd^6e^5I_{xx}^2 + 2c^4d^2e^6I_{xx}^2 + 6c^2d^4e^6I_{xx}^2 - 4c^3d^2e^7I_{xx}^2 + c^4e^8I_{xx}^2 + 2d^{10}I_x^2I_{xx}^2 - 4cd^8eI_x^2I_{xx}^2 + 2c^2d^6e^2I_x^2I_{xx}^2 + 6d^8e^2I_x^2I_{xx}^2 - 12cd^6e^3I_x^2I_{xx}^2 + 6c^2d^4e^4I_x^2I_{xx}^2 + 6d^6e^4I_x^2I_{xx}^2 -$$
$$12cd^4e^5I_x^2I_{xx}^2 + 6c^2d^2e^6I_x^2I_{xx}^2 + 2d^4e^6I_x^2I_{xx}^2 - 4cd^2e^7I_x^2I_{xx}^2 + 2c^2e^8I_x^2I_{xx}^2 + d^8I_x^4I_{xx}^2 + 4d^6e^2I_x^4I_{xx}^2 + 6d^4e^4I_x^4I_{xx}^2 + 4d^2e^6I_x^4I_{xx}^2 + e^8I_x^4I_{xx}^2 - 4cd^{13}I_{xy} + 24c^2d^{11}eI_{xy} - 4d^{13}eI_{xy} -$$
$$60c^3d^9e^2I_{xy} + 24cd^{11}e^2I_{xy} + 80c^4d^7e^3I_{xy} - 60c^2d^9e^3I_{xy} - 60c^5d^5e^4I_{xy} + 80c^3d^7e^4I_{xy} + 24c^6d^3e^5I_{xy} - 60c^4d^5e^5I_{xy} - 4c^7de^6I_{xy} + 24c^5d^3e^6I_{xy} - 4c^6de^7I_{xy} - 8cd^{11}I_x^2I_{xy} +$$
$$32c^2d^9eI_x^2I_{xy} - 8d^{11}eI_x^2I_{xy} - 48c^3d^7e^2I_x^2I_{xy} + 24cd^9e^2I_x^2I_{xy} + 32c^4d^5e^3I_x^2I_{xy} - 16c^2d^7e^3I_x^2I_{xy} - 8d^9e^3I_x^2I_{xy} - 8c^5d^3e^4I_x^2I_{xy} - 16c^3d^5e^4I_x^2I_{xy} + 32cd^7e^4I_x^2I_{xy} + 24c^4d^3e^5I_x^2I_{xy} -$$
$$48c^2d^5e^5I_x^2I_{xy} - 8c^5de^6I_x^2I_{xy} + 32c^3d^3e^6I_x^2I_{xy} - 8c^4de^7I_x^2I_{xy} - 4cd^9I_x^4I_{xy} + 8c^2d^7eI_x^4I_{xy} - 4d^9eI_x^4I_{xy} - 4c^3d^5e^2I_x^4I_{xy} + 12c^2d^5e^3I_x^4I_{xy} - 8d^7e^3I_x^4I_{xy} - 8c^3d^3e^4I_x^4I_{xy} +$$
$$12cd^5e^4I_x^4I_{xy} - 4d^5e^5I_x^4I_{xy} - 4c^3de^6I_x^4I_{xy} + 8cd^3e^6I_x^4I_{xy} - 4c^2de^7I_x^4I_{xy} - 4cd^{11}I_{xx}I_{xy} + 16c^2d^9eI_{xx}I_{xy} - 4d^{11}eI_{xx}I_{xy} - 24c^3d^7e^2I_{xx}I_{xy} + 12cd^9e^2I_{xx}I_{xy} + 16c^4d^5e^3I_{xx}I_{xy} -$$
$$8c^2d^7e^3I_{xx}I_{xy} - 4d^9e^3I_{xx}I_{xy} - 4c^5d^3e^4I_{xx}I_{xy} - 8c^3d^5e^4I_{xx}I_{xy} + 16cd^7e^4I_{xx}I_{xy} + 12c^4d^3e^5I_{xx}I_{xy} - 24c^2d^5e^5I_{xx}I_{xy} - 4c^5de^6I_{xx}I_{xy} + 16c^3d^3e^6I_{xx}I_{xy} - 4c^4de^7I_{xx}I_{xy} -$$
$$8cd^9I_x^2I_{xx}I_{xy} + 16c^2d^7eI_x^2I_{xx}I_{xy} - 8d^9eI_x^2I_{xx}I_{xy} - 8c^3d^5e^2I_x^2I_{xx}I_{xy} + 24c^2d^5e^3I_x^2I_{xx}I_{xy} - 16d^7e^3I_x^2I_{xx}I_{xy} - 16c^3d^3e^4I_x^2I_{xx}I_{xy} + 24cd^5e^4I_x^2I_{xx}I_{xy} - 8d^5e^5I_x^2I_{xx}I_{xy} - 8c^3de^6I_x^2I_{xx}I_{xy} +$$
$$16cd^3e^6I_x^2I_{xx}I_{xy} - 8c^2de^7I_x^2I_{xx}I_{xy} - 4cd^7I_x^4I_{xx}I_{xy} - 4d^7eI_x^4I_{xx}I_{xy} - 12cd^5e^2I_x^4I_{xx}I_{xy} - 12d^5e^3I_x^4I_{xx}I_{xy} - 12cd^3e^4I_x^4I_{xx}I_{xy} - 12d^3e^5I_x^4I_{xx}I_{xy} - 4cde^6I_x^4I_{xx}I_{xy} - 4de^7I_x^4I_{xx}I_{xy} +$$
$$4c^2d^{10}I_{xy}^2 - 2d^{12}I_{xy}^2 - 16c^3d^8eI_{xy}^2 + 20cd^{10}eI_{xy}^2 + 24c^4d^6e^2I_{xy}^2 - 62c^2d^8e^2I_{xy}^2 + 4d^{10}e^2I_{xy}^2 - 16c^5d^4e^3I_{xy}^2 + 88c^3d^6e^3I_{xy}^2 - 16cd^8e^3I_{xy}^2 + 4c^6d^2e^4I_{xy}^2 - 62c^4d^4e^4I_{xy}^2 +$$

$24c^2d^6e^4I_{xy}^2+20c^5d^2e^5I_{xy}^2-16c^3d^4e^5I_{xy}^2-2c^6e^6I_{xy}^2+4c^4d^2e^6I_{xy}^2+8c^2d^8I_x^2I_{xy}^2+10d^{10}I_x^2I_{xy}^2-16c^3d^6eI_x^2I_{xy}^2-24cd^8eI_x^2I_{xy}^2+8c^4d^4e^2I_x^2I_{xy}^2+36c^2d^6e^2I_x^2I_{xy}^2+18d^8e^2I_x^2I_{xy}^2-$
$40c^3d^4e^3I_x^2I_{xy}^2-40cd^6e^3I_x^2I_{xy}^2+18c^4d^2e^4I_x^2I_{xy}^2+36c^2d^4e^4I_x^2I_{xy}^2+8d^6e^4I_x^2I_{xy}^2-24c^3d^2e^5I_x^2I_{xy}^2-16cd^4e^5I_x^2I_{xy}^2+10c^4e^6I_x^2I_{xy}^2+8c^2d^2e^6I_x^2I_{xy}^2+4c^2d^6I_x^4I_{xy}^2+8cd^6eI_x^4I_{xy}^2+$
$8c^2d^4e^2I_x^4I_{xy}^2+4d^6e^2I_x^4I_{xy}^2+16cd^4e^3I_x^4I_{xy}^2+4c^2d^2e^4I_x^4I_{xy}^2+8d^4e^4I_x^4I_{xy}^2+8cd^2e^5I_x^4I_{xy}^2+4d^2e^6I_x^4I_{xy}^2-2d^{10}I_{xx}I_{xy}^2+8cd^8eI_{xx}I_{xy}^2-12c^2d^6e^2I_{xx}I_{xy}^2-2d^8e^2I_{xx}I_{xy}^2+8c^3d^4e^3I_{xx}I_{xy}^2+$
$8cd^6e^3I_{xx}I_{xy}^2-2c^4d^2e^4I_{xx}I_{xy}^2-12c^2d^4e^4I_{xx}I_{xy}^2+8c^3d^2e^5I_{xx}I_{xy}^2-2c^4e^6I_{xx}I_{xy}^2+2d^8I_x^2I_{xx}I_{xy}^2-4cd^6eI_x^2I_{xx}I_{xy}^2+2c^2d^4e^2I_x^2I_{xx}I_{xy}^2+4d^6e^2I_x^2I_{xx}I_{xy}^2-8cd^4e^3I_x^2I_{xx}I_{xy}^2+$
$4c^2d^2e^4I_x^2I_{xx}I_{xy}^2+2d^4e^4I_x^2I_{xx}I_{xy}^2-4cd^2e^5I_x^2I_{xx}I_{xy}^2+2c^2e^6I_x^2I_{xx}I_{xy}^2+4cd^9I_{xy}^3-16c^2d^7eI_{xy}^3+4d^9eI_{xy}^3+24c^3d^5e^2I_{xy}^3-16cd^7e^2I_{xy}^3-16c^4d^3e^3I_{xy}^3+24c^2d^5e^3I_{xy}^3+4c^5de^4I_{xy}^3-$
$16c^3d^3e^4I_{xy}^3+4c^4de^5I_{xy}^3-4cd^7I_x^2I_{xy}^3+8c^2d^5eI_x^2I_{xy}^3-4d^7eI_x^2I_{xy}^3-4c^3d^3e^2I_x^2I_{xy}^3+4cd^5e^2I_x^2I_{xy}^3+4c^2d^3e^3I_x^2I_{xy}^3-4d^5e^3I_x^2I_{xy}^3-4c^3de^4I_x^2I_{xy}^3+8cd^3e^4I_x^2I_{xy}^3-4c^2de^5I_x^2I_{xy}^3+d^8I_{xy}^4-$
$4cd^6eI_{xy}^4+6c^2d^4e^2I_{xy}^4-4c^3d^2e^3I_{xy}^4+c^4e^4I_{xy}^4-12cd^{13}I_xI_y+72c^2d^{11}eI_xI_y-12d^{13}eI_xI_y-180c^3d^9e^2I_xI_y+72cd^{11}e^2I_xI_y+240c^4d^7e^3I_xI_y-180c^2d^9e^3I_xI_y-180c^5d^5e^4I_xI_y+$
$240c^3d^7e^4I_xI_y+72c^6d^3e^5I_xI_y-180c^4d^5e^5I_xI_y-12c^7de^6I_xI_y+72c^5d^3e^6I_xI_y-12c^6de^7I_xI_y-36cd^{11}I_x^3I_y+144c^2d^9eI_x^3I_y-36d^{11}eI_x^3I_y-216c^3d^7e^2I_x^3I_y+108cd^9e^2I_x^3I_y+$
$144c^4d^5e^3I_x^3I_y-72c^2d^7e^3I_x^3I_y-36d^9e^3I_x^3I_y-36c^5d^3e^4I_x^3I_y-72c^3d^5e^4I_x^3I_y+144cd^7e^4I_x^3I_y+108c^4d^3e^5I_x^3I_y-216c^2d^5e^5I_x^3I_y-36c^5de^6I_x^3I_y+144c^3d^3e^6I_x^3I_y-36c^4de^7I_x^3I_y-$
$24cd^9I_x^5I_y+48c^2d^7eI_x^5I_y-24d^9eI_x^5I_y-24c^3d^5e^2I_x^5I_y+72c^2d^5e^3I_x^5I_y-48d^7e^3I_x^5I_y-48c^3d^3e^4I_x^5I_y+72cd^5e^4I_x^5I_y-24d^5e^5I_x^5I_y-24c^3de^6I_x^5I_y+48cd^3e^6I_x^5I_y-24c^2de^7I_x^5I_y-$
$8cd^{11}I_xI_{xx}I_y+32c^2d^9eI_xI_{xx}I_y-8d^{11}eI_xI_{xx}I_y-48c^3d^7e^2I_xI_{xx}I_y+24cd^9e^2I_xI_{xx}I_y+32c^4d^5e^3I_xI_{xx}I_y-16c^2d^7e^3I_xI_{xx}I_y-8d^9e^3I_xI_{xx}I_y-8c^5d^3e^4I_xI_{xx}I_y-16c^3d^5e^4I_xI_{xx}I_y+$
$32cd^7e^4I_xI_{xx}I_y+24c^4d^3e^5I_xI_{xx}I_y-48c^2d^5e^5I_xI_{xx}I_y-8c^5de^6I_xI_{xx}I_y+32c^3d^3e^6I_xI_{xx}I_y-8c^4de^7I_xI_{xx}I_y-8cd^9I_x^3I_{xx}I_y+16c^2d^7eI_x^3I_{xx}I_y-8d^9eI_x^3I_{xx}I_y-8c^3d^5e^2I_x^3I_{xx}I_y+$
$24c^2d^5e^3I_x^3I_{xx}I_y-16d^7e^3I_x^3I_{xx}I_y-16c^3d^3e^4I_x^3I_{xx}I_y+24cd^5e^4I_x^3I_{xx}I_y-8d^5e^5I_x^3I_{xx}I_y-8c^3de^6I_x^3I_{xx}I_y+16cd^3e^6I_x^3I_{xx}I_y-8c^2de^7I_x^3I_{xx}I_y-4cd^9I_xI_{xx}^2I_y+8c^2d^7eI_xI_{xx}^2I_y-$
$4d^9eI_xI_{xx}^2I_y-4c^3d^5e^2I_xI_{xx}^2I_y+12c^2d^5e^3I_xI_{xx}^2I_y-8d^7e^3I_xI_{xx}^2I_y-8c^3d^3e^4I_xI_{xx}^2I_y+12cd^5e^4I_xI_{xx}^2I_y-4d^5e^5I_xI_{xx}^2I_y-4c^3de^6I_xI_{xx}^2I_y+8cd^3e^6I_xI_{xx}^2I_y-4c^2de^7I_xI_{xx}^2I_y-$
$4cd^7I_x^3I_{xx}^2I_y-4d^7eI_x^3I_{xx}^2I_y-12cd^5e^2I_x^3I_{xx}^2I_y-12d^5e^3I_x^3I_{xx}^2I_y-12cd^3e^4I_x^3I_{xx}^2I_y-12d^3e^5I_x^3I_{xx}^2I_y-4cde^6I_x^3I_{xx}^2I_y-4de^7I_x^3I_{xx}^2I_y+16c^2d^{10}I_xI_{xy}I_y-20d^{12}I_xI_{xy}I_y-64c^3d^8eI_xI_{xy}I_y+$
$152cd^{10}eI_xI_{xy}I_y+96c^4d^6e^2I_xI_{xy}I_y-428c^2d^8e^2I_xI_{xy}I_y+16d^{10}e^2I_xI_{xy}I_y-64c^5d^4e^3I_xI_{xy}I_y+592c^3d^6e^3I_xI_{xy}I_y-64cd^8e^3I_xI_{xy}I_y+16c^6d^2e^4I_xI_{xy}I_y-428c^4d^4e^4I_xI_{xy}I_y+$
$96c^2d^6e^4I_xI_{xy}I_y+152c^5d^2e^5I_xI_{xy}I_y-64c^3d^4e^5I_xI_{xy}I_y-20c^6e^6I_xI_{xy}I_y+16c^4d^2e^6I_xI_{xy}I_y+16c^2d^8I_x^3I_{xy}I_y-28d^{10}I_x^3I_{xy}I_y-32c^3d^6eI_x^3I_{xy}I_y+144cd^8eI_x^3I_{xy}I_y+$
$16c^4d^4e^2I_x^3I_{xy}I_y-216c^2d^6e^2I_x^3I_{xy}I_y-12d^8e^2I_x^3I_{xy}I_y+112c^3d^4e^3I_x^3I_{xy}I_y+112cd^6e^3I_x^3I_{xy}I_y-12c^4d^2e^4I_x^3I_{xy}I_y-216c^2d^4e^4I_x^3I_{xy}I_y+16d^6e^4I_x^3I_{xy}I_y+144c^3d^2e^5I_x^3I_{xy}I_y-$
$32cd^4e^5I_x^3I_{xy}I_y-28c^4e^6I_x^3I_{xy}I_y+16c^2d^2e^6I_x^3I_{xy}I_y-8d^8I_x^5I_{xy}I_y+16cd^6eI_x^5I_{xy}I_y-8c^2d^4e^2I_x^5I_{xy}I_y-16d^6e^2I_x^5I_{xy}I_y+32cd^4e^3I_x^5I_{xy}I_y-16c^2d^2e^4I_x^5I_{xy}I_y-8d^4e^4I_x^5I_{xy}I_y+$
$16cd^2e^5I_x^5I_{xy}I_y-8c^2e^6I_x^5I_{xy}I_y+16c^2d^8I_xI_{xx}I_{xy}I_y-20d^{10}I_xI_{xx}I_{xy}I_y-32c^3d^6eI_xI_{xx}I_{xy}I_y+112cd^8eI_xI_{xx}I_{xy}I_y+16c^4d^4e^2I_xI_{xx}I_{xy}I_y-168c^2d^6e^2I_xI_{xx}I_{xy}I_y-4d^8e^2I_xI_{xx}I_{xy}I_y+$
$80c^3d^4e^3I_xI_{xx}I_{xy}I_y+80cd^6e^3I_xI_{xx}I_{xy}I_y-4c^4d^2e^4I_xI_{xx}I_{xy}I_y-168c^2d^4e^4I_xI_{xx}I_{xy}I_y+16d^6e^4I_xI_{xx}I_{xy}I_y+112c^3d^2e^5I_xI_{xx}I_{xy}I_y-32cd^4e^5I_xI_{xx}I_{xy}I_y-20c^4e^6I_xI_{xx}I_{xy}I_y+$
$16c^2d^2e^6I_xI_{xx}I_{xy}I_y+16c^2d^6I_x^3I_{xx}I_{xy}I_y+4d^8I_x^3I_{xx}I_{xy}I_y+24cd^6eI_x^3I_{xx}I_{xy}I_y+36c^2d^4e^2I_x^3I_{xx}I_{xy}I_y+24d^6e^2I_x^3I_{xx}I_{xy}I_y+48cd^4e^3I_x^3I_{xx}I_{xy}I_y+24c^2d^2e^4I_x^3I_{xx}I_{xy}I_y+36d^4e^4I_x^3I_{xx}I_{xy}I_y+$
$24cd^2e^5I_x^3I_{xx}I_{xy}I_y+4c^2e^6I_x^3I_{xx}I_{xy}I_y+16d^2e^6I_x^3I_{xx}I_{xy}I_y-8d^8I_xI_{xx}^2I_{xy}I_y+16cd^6eI_xI_{xx}^2I_{xy}I_y-8c^2d^4e^2I_xI_{xx}^2I_{xy}I_y-16d^6e^2I_xI_{xx}^2I_{xy}I_y+32cd^4e^3I_xI_{xx}^2I_{xy}I_y-16c^2d^2e^4I_xI_{xx}^2I_{xy}I_y-$
$8d^4e^4I_xI_{xx}^2I_{xy}I_y+16cd^2e^5I_xI_{xx}^2I_{xy}I_y-8c^2e^6I_xI_{xx}^2I_{xy}I_y-16c^3d^7I_xI_{xy}^2I_y+20cd^9I_xI_{xy}^2I_y+32c^4d^5eI_xI_{xy}^2I_y-128c^2d^7eI_xI_{xy}^2I_y+20d^9eI_xI_{xy}^2I_y-16c^5d^3e^2I_xI_{xy}^2I_y+216c^3d^5e^2I_xI_{xy}^2I_y-$
$128cd^7e^2I_xI_{xy}^2I_y-128c^4d^3e^3I_xI_{xy}^2I_y+216c^2d^5e^3I_xI_{xy}^2I_y-16d^7e^3I_xI_{xy}^2I_y+20c^5de^4I_xI_{xy}^2I_y-128c^3d^3e^4I_xI_{xy}^2I_y+32cd^5e^4I_xI_{xy}^2I_y+20c^4de^5I_xI_{xy}^2I_y-16c^2d^3e^5I_xI_{xy}^2I_y-$
$16c^3d^5I_x^3I_{xy}^2I_y-8cd^7I_x^3I_{xy}^2I_y-32c^2d^5eI_x^3I_{xy}^2I_y-8d^7eI_x^3I_{xy}^2I_y-24c^3d^3e^2I_x^3I_{xy}^2I_y-40cd^5e^2I_x^3I_{xy}^2I_y-40c^2d^3e^3I_x^3I_{xy}^2I_y-24d^5e^3I_x^3I_{xy}^2I_y-8c^3de^4I_x^3I_{xy}^2I_y-32cd^3e^4I_x^3I_{xy}^2I_y-$
$8c^2de^5I_x^3I_{xy}^2I_y-16d^3e^5I_x^3I_{xy}^2I_y+28cd^7I_xI_{xx}I_{xy}^2I_y-56c^2d^5eI_xI_{xx}I_{xy}^2I_y+28d^7eI_xI_{xx}I_{xy}^2I_y+28c^3d^3e^2I_xI_{xx}I_{xy}^2I_y-28cd^5e^2I_xI_{xx}I_{xy}^2I_y-28c^2d^3e^3I_xI_{xx}I_{xy}^2I_y+28d^5e^3I_xI_{xx}I_{xy}^2I_y+$
$28c^3de^4I_xI_{xx}I_{xy}^2I_y-56cd^3e^4I_xI_{xx}I_{xy}^2I_y+28c^2de^5I_xI_{xx}I_{xy}^2I_y-24c^2d^6I_xI_{xy}^3I_y-12d^8I_xI_{xy}^3I_y+48c^3d^4eI_xI_{xy}^3I_y-24c^4d^2e^2I_xI_{xy}^3I_y+24c^2d^4e^2I_xI_{xy}^3I_y-24d^6e^2I_xI_{xy}^3I_y+$
$48cd^4e^3I_xI_{xy}^3I_y-12c^4e^4I_xI_{xy}^3I_y-24c^2d^2e^4I_xI_{xy}^3I_y+6c^2d^{12}I_y^2+6d^{14}I_y^2-36c^3d^{10}eI_y^2-36cd^{12}eI_y^2+90c^4d^8e^2I_y^2+90c^2d^{10}e^2I_y^2-120c^5d^6e^3I_y^2-120c^3d^8e^3I_y^2+90c^6d^4e^4I_y^2+$
$90c^4d^6e^4I_y^2-36c^7d^2e^5I_y^2-36c^5d^4e^5I_y^2+6c^8e^6I_y^2+6c^6d^2e^6I_y^2+54c^2d^{10}I_x^2I_y^2+18d^{12}I_x^2I_y^2-216c^3d^8eI_x^2I_y^2+324c^4d^6e^2I_x^2I_y^2-162c^2d^8e^2I_x^2I_y^2+54d^{10}e^2I_x^2I_y^2-216c^5d^4e^3I_x^2I_y^2+$
$288c^3d^6e^3I_x^2I_y^2-216cd^8e^3I_x^2I_y^2+54c^6d^2e^4I_x^2I_y^2-162c^4d^4e^4I_x^2I_y^2+324c^2d^6e^4I_x^2I_y^2-216c^3d^4e^5I_x^2I_y^2+18c^6e^6I_x^2I_y^2+54c^4d^2e^6I_x^2I_y^2+60c^2d^8I_x^4I_y^2+12d^{10}I_x^4I_y^2-120c^3d^6eI_x^4I_y^2+$
$72cd^8eI_x^4I_y^2+60c^4d^4e^2I_x^4I_y^2-108c^2d^6e^2I_x^4I_y^2+72d^8e^2I_x^4I_y^2-48c^3d^4e^3I_x^4I_y^2-48cd^6e^3I_x^4I_y^2+72c^4d^2e^4I_x^4I_y^2-108c^2d^4e^4I_x^4I_y^2+60d^6e^4I_x^4I_y^2+72c^3d^2e^5I_x^4I_y^2-120cd^4e^5I_x^4I_y^2+$
$12c^4e^6I_x^4I_y^2+60c^2d^2e^6I_x^4I_y^2+4c^2d^{10}I_{xx}I_y^2+14d^{12}I_{xx}I_y^2-16c^3d^8eI_{xx}I_y^2-76cd^{10}eI_{xx}I_y^2+24c^4d^6e^2I_{xx}I_y^2+178c^2d^8e^2I_{xx}I_y^2+4d^{10}e^2I_{xx}I_y^2-16c^5d^4e^3I_{xx}I_y^2-232c^3d^6e^3I_{xx}I_y^2-$
$16cd^8e^3I_{xx}I_y^2+4c^6d^2e^4I_{xx}I_y^2+178c^4d^4e^4I_{xx}I_y^2+24c^2d^6e^4I_{xx}I_y^2-76c^5d^2e^5I_{xx}I_y^2-16c^3d^4e^5I_{xx}I_y^2+14c^6e^6I_{xx}I_y^2+4c^4d^2e^6I_{xx}I_y^2+12c^2d^8I_x^2I_{xx}I_y^2+18d^{10}I_x^2I_{xx}I_y^2-$
$24c^3d^6eI_x^2I_{xx}I_y^2-48cd^8eI_x^2I_{xx}I_y^2+12c^4d^4e^2I_x^2I_{xx}I_y^2+72c^2d^6e^2I_x^2I_{xx}I_y^2+30d^8e^2I_x^2I_{xx}I_y^2-72c^3d^4e^3I_x^2I_{xx}I_y^2-72cd^6e^3I_x^2I_{xx}I_y^2+30c^4d^2e^4I_x^2I_{xx}I_y^2+72c^2d^4e^4I_x^2I_{xx}I_y^2+$
$12d^6e^4I_x^2I_{xx}I_y^2-48c^3d^2e^5I_x^2I_{xx}I_y^2-24cd^4e^5I_x^2I_{xx}I_y^2+18c^4e^6I_x^2I_{xx}I_y^2+12c^2d^2e^6I_x^2I_{xx}I_y^2+4d^8I_x^4I_{xx}I_y^2-8cd^6eI_x^4I_{xx}I_y^2+4c^2d^4e^2I_x^4I_{xx}I_y^2+8d^6e^2I_x^4I_{xx}I_y^2-16cd^4e^3I_x^4I_{xx}I_y^2+$
$8c^2d^2e^4I_x^4I_{xx}I_y^2+4d^4e^4I_x^4I_{xx}I_y^2-8cd^2e^5I_x^4I_{xx}I_y^2+4c^2e^6I_x^4I_{xx}I_y^2+2c^2d^8I_{xx}^2I_y^2+12d^{10}I_{xx}^2I_y^2-4c^3d^6eI_{xx}^2I_y^2-44cd^8eI_{xx}^2I_y^2+2c^4d^4e^2I_{xx}^2I_y^2+66c^2d^6e^2I_{xx}^2I_y^2+14d^8e^2I_{xx}^2I_y^2-$
$48c^3d^4e^3I_{xx}^2I_y^2-48cd^6e^3I_{xx}^2I_y^2+14c^4d^2e^4I_{xx}^2I_y^2+66c^2d^4e^4I_{xx}^2I_y^2+2d^6e^4I_{xx}^2I_y^2-44c^3d^2e^5I_{xx}^2I_y^2-4cd^4e^5I_{xx}^2I_y^2+12c^4e^6I_{xx}^2I_y^2+2c^2d^2e^6I_{xx}^2I_y^2+6c^2d^6I_x^2I_{xx}^2I_y^2+12cd^6eI_x^2I_{xx}^2I_y^2+$
$12c^2d^4e^2I_x^2I_{xx}^2I_y^2+6d^6e^2I_x^2I_{xx}^2I_y^2+24cd^4e^3I_x^2I_{xx}^2I_y^2+6c^2d^2e^4I_x^2I_{xx}^2I_y^2+12d^4e^4I_x^2I_{xx}^2I_y^2+12cd^2e^5I_x^2I_{xx}^2I_y^2+6d^2e^6I_x^2I_{xx}^2I_y^2+4d^8I_{xx}^3I_y^2-8cd^6eI_{xx}^3I_y^2+4c^2d^4e^2I_{xx}^3I_y^2+8d^6e^2I_{xx}^3I_y^2-$
$16cd^4e^3I_{xx}^3I_y^2+8c^2d^2e^4I_{xx}^3I_y^2+4d^4e^4I_{xx}^3I_y^2-8cd^2e^5I_{xx}^3I_y^2+4c^2e^6I_{xx}^3I_y^2-8c^3d^9I_{xy}I_y^2-8cd^{11}I_{xy}I_y^2+32c^4d^7eI_{xy}I_y^2+24c^2d^9eI_{xy}I_y^2-8d^{11}eI_{xy}I_y^2-48c^5d^5e^2I_{xy}I_y^2-16c^3d^7e^2I_{xy}I_y^2+$
$32cd^9e^2I_{xy}I_y^2+32c^6d^3e^3I_{xy}I_y^2-16c^4d^5e^3I_{xy}I_y^2-48c^2d^7e^3I_{xy}I_y^2-8c^7de^4I_{xy}I_y^2+24c^5d^3e^4I_{xy}I_y^2+32c^3d^5e^4I_{xy}I_y^2-8c^6de^5I_{xy}I_y^2-8c^4d^3e^5I_{xy}I_y^2-24c^3d^7I_x^2I_{xy}I_y^2+48cd^9I_x^2I_{xy}I_y^2+$
$48c^4d^5eI_x^2I_{xy}I_y^2-264c^2d^7eI_x^2I_{xy}I_y^2+48d^9eI_x^2I_{xy}I_y^2-24c^5d^3e^2I_x^2I_{xy}I_y^2+432c^3d^5e^2I_x^2I_{xy}I_y^2-264cd^7e^2I_x^2I_{xy}I_y^2-264c^4d^3e^3I_x^2I_{xy}I_y^2+432c^2d^5e^3I_x^2I_{xy}I_y^2-24d^7e^3I_x^2I_{xy}I_y^2+$
$48c^5de^4I_x^2I_{xy}I_y^2-264c^3d^3e^4I_x^2I_{xy}I_y^2+48cd^5e^4I_x^2I_{xy}I_y^2+48c^4de^5I_x^2I_{xy}I_y^2-24c^2d^3e^5I_x^2I_{xy}I_y^2+32cd^7I_x^4I_{xy}I_y^2-64c^2d^5eI_x^4I_{xy}I_y^2+32d^7eI_x^4I_{xy}I_y^2+32c^3d^3e^2I_x^4I_{xy}I_y^2-32cd^5e^2I_x^4I_{xy}I_y^2-$
$32c^2d^3e^3I_x^4I_{xy}I_y^2+32d^5e^3I_x^4I_{xy}I_y^2+32c^3de^4I_x^4I_{xy}I_y^2-64cd^3e^4I_x^4I_{xy}I_y^2+32c^2de^5I_x^4I_{xy}I_y^2-8c^3d^7I_{xx}I_{xy}I_y^2-28cd^9I_{xx}I_{xy}I_y^2+16c^4d^5eI_{xx}I_{xy}I_y^2+88c^2d^7eI_{xx}I_{xy}I_y^2-28d^9eI_{xx}I_{xy}I_y^2-$

$$8c^5d^3e^2I_{xx}I_{xy}I_y^2-120c^3d^5e^2I_{xx}I_{xy}I_y^2+88cd^7e^2I_{xx}I_{xy}I_y^2+88c^4d^3e^3I_{xx}I_{xy}I_y^2-120c^2d^5e^3I_{xx}I_{xy}I_y^2-8d^7e^3I_{xx}I_{xy}I_y^2-28c^5de^4I_{xx}I_{xy}I_y^2+88c^3d^3e^4I_{xx}I_{xy}I_y^2+16cd^5e^4I_{xx}I_{xy}I_y^2-$$

$$28c^4de^5I_{xx}I_{xy}I_y^2-8c^2d^3e^5I_{xx}I_{xy}I_y^2-24c^3d^5I_x^2I_{xx}I_{xy}I_y^2-12cd^7I_x^2I_{xx}I_{xy}I_y^2-48c^2d^5eI_x^2I_{xx}I_{xy}I_y^2-12d^7eI_x^2I_{xx}I_{xy}I_y^2-36c^3d^3e^2I_x^2I_{xx}I_{xy}I_y^2-60cd^5e^2I_x^2I_{xx}I_{xy}I_y^2-60c^2d^3e^3I_x^2I_{xx}I_{xy}I_y^2-$$

$$36d^5e^3I_x^2I_{xx}I_{xy}I_y^2-12c^3de^4I_x^2I_{xx}I_{xy}I_y^2-48cd^3e^4I_x^2I_{xx}I_{xy}I_y^2-12c^2de^5I_x^2I_{xx}I_{xy}I_y^2-24d^3e^5I_x^2I_{xx}I_{xy}I_y^2-16cd^7I_{xx}^2I_{xy}I_y^2+32c^2d^5eI_{xx}^2I_{xy}I_y^2-16d^7eI_{xx}^2I_{xy}I_y^2-16c^3d^3e^2I_{xx}^2I_{xy}I_y^2+$$

$$16cd^5e^2I_{xx}^2I_{xy}I_y^2+16c^2d^3e^3I_{xx}^2I_{xy}I_y^2-16d^5e^3I_{xx}^2I_{xy}I_y^2-16c^3de^4I_{xx}^2I_{xy}I_y^2+32cd^3e^4I_{xx}^2I_{xy}I_y^2-16c^2de^5I_{xx}^2I_{xy}I_y^2+8c^4d^6I_{xy}^2I_y^2+18c^2d^8I_{xy}^2I_y^2+10d^{10}I_{xy}^2I_y^2-16c^5d^4eI_{xy}^2I_y^2-$$

$$40c^3d^6eI_{xy}^2I_y^2-24cd^8eI_{xy}^2I_y^2+8c^6d^2e^2I_{xy}^2I_y^2+36c^4d^4e^2I_{xy}^2I_y^2+36c^2d^6e^2I_{xy}^2I_y^2+8d^8e^2I_{xy}^2I_y^2-24c^5d^2e^3I_{xy}^2I_y^2-40c^3d^4e^3I_{xy}^2I_y^2-16cd^6e^3I_{xy}^2I_y^2+10c^6e^4I_{xy}^2I_y^2+18c^4d^2e^4I_{xy}^2I_y^2+$$

$$8c^2d^4e^4I_{xy}^2I_y^2+24c^4d^4I_x^2I_{xy}^2I_y^2+24c^2d^6I_x^2I_{xy}^2I_y^2+36d^8I_x^2I_{xy}^2I_y^2+48c^3d^4eI_x^2I_{xy}^2I_y^2-96cd^6eI_x^2I_{xy}^2I_y^2+24c^4d^2e^2I_x^2I_{xy}^2I_y^2+264c^2d^4e^2I_x^2I_{xy}^2I_y^2+24d^6e^2I_x^2I_{xy}^2I_y^2-96c^3d^2e^3I_x^2I_{xy}^2I_y^2+$$

$$48cd^4e^3I_x^2I_{xy}^2I_y^2+36c^4e^4I_x^2I_{xy}^2I_y^2+24c^2d^2e^4I_x^2I_{xy}^2I_y^2+24d^4e^4I_x^2I_{xy}^2I_y^2+18c^2d^6I_{xx}I_{xy}^2I_y^2+8d^8I_{xx}I_{xy}^2I_y^2-36c^3d^4eI_{xx}I_{xy}^2I_y^2+4cd^6eI_{xx}I_{xy}^2I_y^2+18c^4d^2e^2I_{xx}I_{xy}^2I_y^2-24c^2d^4e^2I_{xx}I_{xy}^2I_y^2+$$

$$18d^6e^2I_{xx}I_{xy}^2I_y^2+4c^3d^2e^3I_{xx}I_{xy}^2I_y^2-36cd^4e^3I_{xx}I_{xy}^2I_y^2+8c^4e^4I_{xx}I_{xy}^2I_y^2+18c^2d^2e^4I_{xx}I_{xy}^2I_y^2-4c^3d^5I_{xy}^3I_y^2-4cd^7I_{xy}^3I_y^2+8c^4d^3eI_{xy}^3I_y^2+4c^2d^5eI_{xy}^3I_y^2-4d^7eI_{xy}^3I_y^2-4c^5de^2I_{xy}^3I_y^2+$$

$$4c^3d^3e^2I_{xy}^3I_y^2+8cd^5e^2I_{xy}^3I_y^2-4c^4de^3I_{xy}^3I_y^2-4c^2d^3e^3I_{xy}^3I_y^2-36c^3d^9I_xI_y^3-36cd^{11}I_xI_y^3+144c^4d^7eI_xI_y^3+108c^2d^9eI_xI_y^3-36d^{11}eI_xI_y^3-216c^5d^5e^2I_xI_y^3-72c^3d^7e^2I_xI_y^3+$$

$$144cd^9e^2I_xI_y^3+144c^6d^3e^3I_xI_y^3-72c^4d^5e^3I_xI_y^3-216c^2d^7e^3I_xI_y^3-36c^7de^4I_xI_y^3+108c^5d^3e^4I_xI_y^3+144c^3d^5e^4I_xI_y^3-36c^6de^5I_xI_y^3-36c^4d^3e^5I_xI_y^3-80c^3d^7I_x^3I_y^3-48cd^9I_x^3I_y^3+$$

$$160c^4d^5eI_x^3I_y^3-48c^2d^7eI_x^3I_y^3-48d^9eI_x^3I_y^3-80c^5d^3e^2I_x^3I_y^3+192c^3d^5e^2I_x^3I_y^3-48cd^7e^2I_x^3I_y^3-48c^4d^3e^3I_x^3I_y^3+192c^2d^5e^3I_x^3I_y^3-80d^7e^3I_x^3I_y^3-48c^5de^4I_x^3I_y^3-48c^3d^3e^4I_x^3I_y^3+$$

$$160cd^5e^4I_x^3I_y^3-48c^4de^5I_x^3I_y^3-80c^2d^3e^5I_x^3I_y^3-8c^3d^7I_xI_{xx}I_y^3-36cd^9I_xI_{xx}I_y^3+16c^4d^5eI_xI_{xx}I_y^3+120c^2d^7eI_xI_{xx}I_y^3-36d^9eI_xI_{xx}I_y^3-8c^5d^3e^2I_xI_{xx}I_y^3-168c^3d^5e^2I_xI_{xx}I_y^3+$$

$$120cd^7e^2I_xI_{xx}I_y^3+120c^4d^3e^3I_xI_{xx}I_y^3-168c^2d^5e^3I_xI_{xx}I_y^3-8d^7e^3I_xI_{xx}I_y^3-36c^5de^4I_xI_{xx}I_y^3+120c^3d^3e^4I_xI_{xx}I_y^3+16cd^5e^4I_xI_{xx}I_y^3-36c^4de^5I_xI_{xx}I_y^3-8c^2d^3e^5I_xI_{xx}I_y^3-16cd^7I_x^3I_{xx}I_y^3+$$

$$32c^2d^5eI_x^3I_{xx}I_y^3-16d^7eI_x^3I_{xx}I_y^3-16c^3d^3e^2I_x^3I_{xx}I_y^3+16cd^5e^2I_x^3I_{xx}I_y^3+16c^2d^3e^3I_x^3I_{xx}I_y^3-16d^5e^3I_x^3I_{xx}I_y^3-16c^3de^4I_x^3I_{xx}I_y^3+32cd^3e^4I_x^3I_{xx}I_y^3-16c^2de^5I_x^3I_{xx}I_y^3-4c^3d^5I_xI_{xx}^2I_y^3-$$

$$12c^2d^5eI_xI_{xx}^2I_y^3-4c^3d^3e^2I_xI_{xx}^2I_y^3-12cd^5e^2I_xI_{xx}^2I_y^3-12c^2d^3e^3I_xI_{xx}^2I_y^3-4d^5e^3I_xI_{xx}^2I_y^3-12cd^3e^4I_xI_{xx}^2I_y^3-4d^3e^5I_xI_{xx}^2I_y^3+16c^4d^6I_xI_{xy}I_y^3-12c^2d^8I_xI_{xy}I_y^3-28d^{10}I_xI_{xy}I_y^3-$$

$$32c^5d^4eI_xI_{xy}I_y^3+112c^3d^6eI_xI_{xy}I_y^3+144cd^8eI_xI_{xy}I_y^3+16c^6d^2e^2I_xI_{xy}I_y^3-216c^4d^4e^2I_xI_{xy}I_y^3-216c^2d^6e^2I_xI_{xy}I_y^3+16d^8e^2I_xI_{xy}I_y^3+144c^5d^2e^3I_xI_{xy}I_y^3+112c^3d^4e^3I_xI_{xy}I_y^3-$$

$$32cd^6e^3I_xI_{xy}I_y^3-28c^6e^4I_xI_{xy}I_y^3-12c^4d^2e^4I_xI_{xy}I_y^3+16c^2d^4e^4I_xI_{xy}I_y^3-48c^2d^6I_x^3I_{xy}I_y^3-16d^8I_x^3I_{xy}I_y^3+96c^3d^4eI_x^3I_{xy}I_y^3-32cd^6eI_x^3I_{xy}I_y^3-48c^4d^2e^2I_x^3I_{xy}I_y^3+96c^2d^4e^2I_x^3I_{xy}I_y^3-$$

$$48d^6e^2I_x^3I_{xy}I_y^3-32c^3d^2e^3I_x^3I_{xy}I_y^3+96cd^4e^3I_x^3I_{xy}I_y^3-16c^4e^4I_x^3I_{xy}I_y^3-48c^2d^2e^4I_x^3I_{xy}I_y^3+16c^4d^4I_xI_{xx}I_{xy}I_y^3+12c^2d^6I_xI_{xx}I_{xy}I_y^3-32d^8I_xI_{xx}I_{xy}I_y^3+40c^3d^4eI_xI_{xx}I_{xy}I_y^3+$$

$$152cd^6eI_xI_{xx}I_{xy}I_y^3+12c^4d^2e^2I_xI_{xx}I_{xy}I_y^3-144c^2d^4e^2I_xI_{xx}I_{xy}I_y^3+12d^6e^2I_xI_{xx}I_{xy}I_y^3+152c^3d^2e^3I_xI_{xx}I_{xy}I_y^3+40cd^4e^3I_xI_{xx}I_{xy}I_y^3-32c^4e^4I_xI_{xx}I_{xy}I_y^3+12c^2d^2e^4I_xI_{xx}I_{xy}I_y^3+$$

$$16d^4e^4I_xI_{xx}I_{xy}I_y^3-16c^5d^3I_xI_{xy}^2I_y^3-24c^3d^5I_xI_{xy}^2I_y^3-8cd^7I_xI_{xy}^2I_y^3-32c^4d^3eI_xI_{xy}^2I_y^3-40c^2d^5eI_xI_{xy}^2I_y^3-8d^7eI_xI_{xy}^2I_y^3-8c^5de^2I_xI_{xy}^2I_y^3-40c^3d^3e^2I_xI_{xy}^2I_y^3-32cd^5e^2I_xI_{xy}^2I_y^3-$$

$$8c^4de^3I_xI_{xy}^2I_y^3-24c^2d^3e^3I_xI_{xy}^2I_y^3-16d^5e^3I_xI_{xy}^2I_y^3+9c^4d^8I_y^4+18c^2d^{10}I_y^4+9d^{12}I_y^4-36c^5d^6eI_y^4-72c^3d^8eI_y^4-36cd^{10}eI_y^4+54c^6d^4e^2I_y^4+108c^4d^6e^2I_y^4+54c^2d^8e^2I_y^4-$$

$$36c^7d^2e^3I_y^4-72c^5d^4e^3I_y^4-36c^3d^6e^3I_y^4+9c^8e^4I_y^4+18c^6d^2e^4I_y^4+9c^4d^4e^4I_y^4+60c^4d^6I_x^2I_y^4+72c^2d^8I_x^2I_y^4+12d^{10}I_x^2I_y^4-120c^5d^4eI_x^2I_y^4-48c^3d^6eI_x^2I_y^4+72cd^8eI_x^2I_y^4+$$

$$60c^6d^2e^2I_x^2I_y^4-108c^4d^4e^2I_x^2I_y^4-108c^2d^6e^2I_x^2I_y^4+60d^8e^2I_x^2I_y^4+72c^5d^2e^3I_x^2I_y^4-48c^3d^4e^3I_x^2I_y^4-120cd^6e^3I_x^2I_y^4+12c^6e^4I_x^2I_y^4+72c^4d^2e^4I_x^2I_y^4+60c^2d^4e^4I_x^2I_y^4+2c^4d^6I_{xx}I_y^4+$$

$$18c^2d^8I_{xx}I_y^4+16d^{10}I_{xx}I_y^4-4c^5d^4eI_{xx}I_y^4-64c^3d^6eI_{xx}I_y^4-60cd^8eI_{xx}I_y^4+2c^6d^2e^2I_{xx}I_y^4+90c^4d^4e^2I_{xx}I_y^4+90c^2d^6e^2I_{xx}I_y^4+2d^8e^2I_{xx}I_y^4-60c^5d^2e^3I_{xx}I_y^4-64c^3d^4e^3I_{xx}I_y^4-$$

$$4cd^6e^3I_{xx}I_y^4+16c^6e^4I_{xx}I_y^4+18c^4d^2e^4I_{xx}I_y^4+2c^2d^4e^4I_{xx}I_y^4+24c^2d^6I_x^2I_{xx}I_y^4+8d^8I_x^2I_{xx}I_y^4-48c^3d^4eI_x^2I_{xx}I_y^4+16cd^6eI_x^2I_{xx}I_y^4+24c^4d^2e^2I_x^2I_{xx}I_y^4-48c^2d^4e^2I_x^2I_{xx}I_y^4+$$

$$24d^6e^2I_x^2I_{xx}I_y^4+16c^3d^2e^3I_x^2I_{xx}I_y^4-48cd^4e^3I_x^2I_{xx}I_y^4+8c^4e^4I_x^2I_{xx}I_y^4+24c^2d^2e^4I_x^2I_{xx}I_y^4+c^4d^4I_{xx}^2I_y^4+8d^8I_{xx}^2I_y^4+4c^3d^4eI_{xx}^2I_y^4-32cd^6eI_{xx}^2I_y^4+54c^2d^4e^2I_{xx}^2I_y^4-32c^3d^2e^3I_{xx}^2I_y^4+$$

$$4cd^4e^3I_{xx}^2I_y^4+8c^4e^4I_{xx}^2I_y^4+d^4e^4I_{xx}^2I_y^4-4c^5d^5I_{xy}I_y^4-8c^3d^7I_{xy}I_y^4-4cd^9I_{xy}I_y^4+8c^6d^3eI_{xy}I_y^4+12c^4d^5eI_{xy}I_y^4-4d^9eI_{xy}I_y^4-4c^7de^2I_{xy}I_y^4+12c^3d^5e^2I_{xy}I_y^4+8cd^7e^2I_{xy}I_y^4-$$

$$4c^6de^3I_{xy}I_y^4-8c^4d^3e^3I_{xy}I_y^4-4c^2d^5e^3I_{xy}I_y^4+32c^3d^5I_x^2I_{xy}I_y^4+32cd^7I_x^2I_{xy}I_y^4-64c^4d^3eI_x^2I_{xy}I_y^4-32c^2d^5eI_x^2I_{xy}I_y^4+32d^7eI_x^2I_{xy}I_y^4+32c^5de^2I_x^2I_{xy}I_y^4-32c^3d^3e^2I_x^2I_{xy}I_y^4-$$

$$64cd^5e^2I_x^2I_{xy}I_y^4+32c^4de^3I_x^2I_{xy}I_y^4+32c^2d^3e^3I_x^2I_{xy}I_y^4-4c^5d^3I_{xx}I_{xy}I_y^4-4c^3d^5I_{xx}I_{xy}I_y^4-12c^4d^3eI_{xx}I_{xy}I_y^4-12c^2d^5eI_{xx}I_{xy}I_y^4-12c^3d^3e^2I_{xx}I_{xy}I_y^4-12cd^5e^2I_{xx}I_{xy}I_y^4-$$

$$4c^2d^3e^3I_{xx}I_{xy}I_y^4-4d^5e^3I_{xx}I_{xy}I_y^4+4c^6d^2I_{xy}^2I_y^4+8c^4d^4I_{xy}^2I_y^4+4c^2d^6I_{xy}^2I_y^4+8c^5d^2eI_{xy}^2I_y^4+16c^3d^4eI_{xy}^2I_y^4+8cd^6eI_{xy}^2I_y^4+4c^4d^2e^2I_{xy}^2I_y^4+8c^2d^4e^2I_{xy}^2I_y^4+4d^6e^2I_{xy}^2I_y^4-$$

$$24c^5d^5I_xI_y^5-48c^3d^7I_xI_y^5-24cd^9I_xI_y^5+48c^6d^3eI_xI_y^5+72c^4d^5eI_xI_y^5-24d^9eI_xI_y^5-24c^7de^2I_xI_y^5+72c^3d^5e^2I_xI_y^5+48cd^7e^2I_xI_y^5-24c^6de^3I_xI_y^5-48c^4d^3e^3I_xI_y^5-24c^2d^5e^3I_xI_y^5-$$

$$16c^3d^5I_xI_{xx}I_y^5-16cd^7I_xI_{xx}I_y^5+32c^4d^3eI_xI_{xx}I_y^5+16c^2d^5eI_xI_{xx}I_y^5-16d^7eI_xI_{xx}I_y^5-16c^5de^2I_xI_{xx}I_y^5+16c^3d^3e^2I_xI_{xx}I_y^5+32cd^5e^2I_xI_{xx}I_y^5-16c^4de^3I_xI_{xx}I_y^5-16c^2d^3e^3I_xI_{xx}I_y^5-$$

$$8c^4d^4I_xI_{xy}I_y^5-16c^2d^6I_xI_{xy}I_y^5-8d^8I_xI_{xy}I_y^5+16c^5d^2eI_xI_{xy}I_y^5+32c^3d^4eI_xI_{xy}I_y^5+16cd^6eI_xI_{xy}I_y^5-8c^6e^2I_xI_{xy}I_y^5-16c^4d^2e^2I_xI_{xy}I_y^5-8c^2d^4e^2I_xI_{xy}I_y^5+4c^6d^4I_y^6+12c^4d^6I_y^6+$$

$$12c^2d^8I_y^6+4d^{10}I_y^6-8c^7d^2eI_y^6-24c^5d^4eI_y^6-24c^3d^6eI_y^6-8cd^8eI_y^6+4c^8e^2I_y^6+12c^6d^2e^2I_y^6+12c^4d^4e^2I_y^6+4c^2d^6e^2I_y^6+4c^4d^4I_{xx}I_y^6+8c^2d^6I_{xx}I_y^6+4d^8I_{xx}I_y^6-8c^5d^2eI_{xx}I_y^6-$$

$$16c^3d^4eI_{xx}I_y^6-8cd^6eI_{xx}I_y^6+4c^6e^2I_{xx}I_y^6+8c^4d^2e^2I_{xx}I_y^6+4c^2d^4e^2I_{xx}I_y^6+2c^2d^{12}I_{yy}+2d^{14}I_{yy}-12c^3d^{10}eI_{yy}-12cd^{12}eI_{yy}+30c^4d^8e^2I_{yy}+30c^2d^{10}e^2I_{yy}-40c^5d^6e^3I_{yy}-$$

$$40c^3d^8e^3I_{yy}+30c^6d^4e^4I_{yy}+30c^4d^6e^4I_{yy}-12c^7d^2e^5I_{yy}-12c^5d^4e^5I_{yy}+2c^8e^6I_{yy}+2c^6d^2e^6I_{yy}+4c^2d^{10}I_x^2I_{yy}+14d^{12}I_x^2I_{yy}-16c^3d^8eI_x^2I_{yy}-76cd^{10}eI_x^2I_{yy}+24c^4d^6e^2I_x^2I_{yy}+$$

$$178c^2d^8e^2I_x^2I_{yy}+4d^{10}e^2I_x^2I_{yy}-16c^5d^4e^3I_x^2I_{yy}-232c^3d^6e^3I_x^2I_{yy}-16cd^8e^3I_x^2I_{yy}+4c^6d^2e^4I_x^2I_{yy}+178c^4d^4e^4I_x^2I_{yy}+24c^2d^6e^4I_x^2I_{yy}-76c^5d^2e^5I_x^2I_{yy}-16c^3d^4e^5I_x^2I_{yy}+$$

$$14c^6e^6I_x^2I_{yy}+4c^4d^2e^6I_x^2I_{yy}+2c^2d^8I_x^4I_{yy}+16d^{10}I_x^4I_{yy}-4c^3d^6eI_x^4I_{yy}-60cd^8eI_x^4I_{yy}+2c^4d^4e^2I_x^4I_{yy}+90c^2d^6e^2I_x^4I_{yy}+18d^8e^2I_x^4I_{yy}-64c^3d^4e^3I_x^4I_{yy}-64cd^6e^3I_x^4I_{yy}+$$

$$18c^4d^2e^4I_x^4I_{yy}+90c^2d^4e^4I_x^4I_{yy}+2d^6e^4I_x^4I_{yy}-60c^3d^2e^5I_x^4I_{yy}-4cd^4e^5I_x^4I_{yy}+16c^4e^6I_x^4I_{yy}+2c^2d^2e^6I_x^4I_{yy}+4d^8I_x^6I_{yy}-8cd^6eI_x^6I_{yy}+4c^2d^4e^2I_x^6I_{yy}+8d^6e^2I_x^6I_{yy}-$$

$$16cd^4e^3I_x^6I_{yy}+8c^2d^2e^4I_x^6I_{yy}+4d^4e^4I_x^6I_{yy}-8cd^2e^5I_x^6I_{yy}+4c^2e^6I_x^6I_{yy}+2c^2d^{10}I_{xx}I_{yy}+4d^{12}I_{xx}I_{yy}-8c^3d^8eI_{xx}I_{yy}-20cd^{10}eI_{xx}I_{yy}+12c^4d^6e^2I_{xx}I_{yy}+44c^2d^8e^2I_{xx}I_{yy}+$$

$$2d^{10}e^2I_{xx}I_{yy}-8c^5d^4e^3I_{xx}I_{yy}-56c^3d^6e^3I_{xx}I_{yy}-8cd^8e^3I_{xx}I_{yy}+2c^6d^2e^4I_{xx}I_{yy}+44c^4d^4e^4I_{xx}I_{yy}+12c^2d^6e^4I_{xx}I_{yy}-20c^5d^2e^5I_{xx}I_{yy}-8c^3d^4e^5I_{xx}I_{yy}+4c^6e^6I_{xx}I_{yy}+$$

$$2c^4d^2e^6I_{xx}I_{yy}+4c^2d^8I_x^2I_{xx}I_{yy}+4d^{10}I_x^2I_{xx}I_{yy}-8c^3d^6eI_x^2I_{xx}I_{yy}-8cd^8eI_x^2I_{xx}I_{yy}+4c^4d^4e^2I_x^2I_{xx}I_{yy}+12c^2d^6e^2I_x^2I_{xx}I_{yy}+8d^8e^2I_x^2I_{xx}I_{yy}-16c^3d^4e^3I_x^2I_{xx}I_{yy}-16cd^6e^3I_x^2I_{xx}I_{yy}+$$

$$8c^4d^2e^4I_x^2I_{xx}I_{yy}+12c^2d^4e^4I_x^2I_{xx}I_{yy}+4d^6e^4I_x^2I_{xx}I_{yy}-8c^3d^2e^5I_x^2I_{xx}I_{yy}-8cd^4e^5I_x^2I_{xx}I_{yy}+4c^4e^6I_x^2I_{xx}I_{yy}+4c^2d^2e^6I_x^2I_{xx}I_{yy}+2c^2d^6I_x^4I_{xx}I_{yy}+4cd^6eI_x^4I_{xx}I_{yy}+4c^2d^4e^2I_x^4I_{xx}I_{yy}+$$

$$2d^6e^2I_x^4I_{xx}I_{yy}+8cd^4e^3I_x^4I_{xx}I_{yy}+2c^2d^2e^4I_x^4I_{xx}I_{yy}+4d^4e^4I_x^4I_{xx}I_{yy}+4cd^2e^5I_x^4I_{xx}I_{yy}+2d^2e^6I_x^4I_{xx}I_{yy}+2d^{10}I_{xx}^2I_{yy}-8cd^8eI_{xx}^2I_{yy}+12c^2d^6e^2I_{xx}^2I_{yy}+2d^8e^2I_{xx}^2I_{yy}-8c^3d^4e^3I_{xx}^2I_{yy}-$$

$8cd^6e^3I_{xx}^2I_{yy} + 2c^4d^2e^4I_{xx}^2I_{yy} + 12c^2d^4e^4I_{xx}^2I_{yy} - 8c^3d^2e^5I_{xx}^2I_{yy} + 2c^4e^6I_{xx}^2I_{yy} + 2d^8I_x^2I_{xx}^2I_{yy} - 4cd^6eI_x^2I_{xx}^2I_{yy} + 2c^2d^4e^2I_x^2I_{xx}^2I_{yy} + 4d^6e^2I_x^2I_{xx}^2I_{yy} - 8cd^4e^3I_x^2I_{xx}^2I_{yy} +$

$4c^2d^2e^4I_x^2I_{xx}^2I_{yy} + 2d^4e^4I_x^2I_{xx}^2I_{yy} - 4cd^2e^5I_x^2I_{xx}^2I_{yy} + 2c^2e^6I_x^2I_{xx}^2I_{yy} - 4c^3d^9I_{xy}I_{yy} - 4cd^{11}I_{xy}I_{yy} + 16c^4d^7eI_{xy}I_{yy} + 12c^2d^9eI_{xy}I_{yy} - 4d^{11}eI_{xy}I_{yy} - 24c^5d^5e^2I_{xy}I_{yy} - 8c^3d^7e^2I_{xy}I_{yy} +$

$16cd^9e^2I_{xy}I_{yy} + 16c^6d^3e^3I_{xy}I_{yy} - 8c^4d^5e^3I_{xy}I_{yy} - 24c^2d^7e^3I_{xy}I_{yy} - 4c^7de^4I_{xy}I_{yy} + 12c^5d^3e^4I_{xy}I_{yy} + 16c^3d^5e^4I_{xy}I_{yy} - 4c^6de^5I_{xy}I_{yy} - 4c^4d^3e^5I_{xy}I_{yy} - 8c^3d^7I_x^2I_{xy}I_{yy} -$

$28cd^9I_x^2I_{xy}I_{yy} + 16c^4d^5eI_x^2I_{xy}I_{yy} + 88c^2d^7eI_x^2I_{xy}I_{yy} - 28d^9eI_x^2I_{xy}I_{yy} - 8c^5d^3e^2I_x^2I_{xy}I_{yy} - 120c^3d^5e^2I_x^2I_{xy}I_{yy} + 88cd^7e^2I_x^2I_{xy}I_{yy} + 88c^4d^3e^3I_x^2I_{xy}I_{yy} - 120c^2d^5e^3I_x^2I_{xy}I_{yy} -$

$8d^7e^3I_x^2I_{xy}I_{yy} - 28c^5de^4I_x^2I_{xy}I_{yy} + 88c^3d^3e^4I_x^2I_{xy}I_{yy} + 16cd^5e^4I_x^2I_{xy}I_{yy} - 28c^4de^5I_x^2I_{xy}I_{yy} - 8c^2d^3e^5I_x^2I_{xy}I_{yy} - 4c^3d^5I_x^4I_{xy}I_{yy} - 12c^2d^5eI_x^4I_{xy}I_{yy} - 4c^3d^3e^2I_x^4I_{xy}I_{yy} -$

$12cd^5e^2I_x^4I_{xy}I_{yy} - 12c^2d^3e^3I_x^4I_{xy}I_{yy} - 4d^5e^3I_x^4I_{xy}I_{yy} - 12cd^3e^4I_x^4I_{xy}I_{yy} - 4d^3e^5I_x^4I_{xy}I_{yy} - 4cd^9I_{xx}I_{xy}I_{yy} + 16c^2d^7eI_{xx}I_{xy}I_{yy} - 4d^9eI_{xx}I_{xy}I_{yy} - 24c^3d^5e^2I_{xx}I_{xy}I_{yy} +$

$16cd^7e^2I_{xx}I_{xy}I_{yy} + 16c^4d^3e^3I_{xx}I_{xy}I_{yy} - 24c^2d^5e^3I_{xx}I_{xy}I_{yy} - 4c^5de^4I_{xx}I_{xy}I_{yy} + 16c^3d^3e^4I_{xx}I_{xy}I_{yy} - 4c^4de^5I_{xx}I_{xy}I_{yy} - 12cd^7I_x^2I_{xx}I_{xy}I_{yy} + 24c^2d^5eI_x^2I_{xx}I_{xy}I_{yy} - 12d^7eI_x^2I_{xx}I_{xy}I_{yy} -$

$12c^3d^3e^2I_x^2I_{xx}I_{xy}I_{yy} + 12cd^5e^2I_x^2I_{xx}I_{xy}I_{yy} + 12c^2d^3e^3I_x^2I_{xx}I_{xy}I_{yy} - 12d^5e^3I_x^2I_{xx}I_{xy}I_{yy} - 12c^3de^4I_x^2I_{xx}I_{xy}I_{yy} + 24cd^3e^4I_x^2I_{xx}I_{xy}I_{yy} - 12c^2de^5I_x^2I_{xx}I_{xy}I_{yy} - 2c^2d^8I_{xy}^2I_{yy} - 2d^{10}I_{xy}^2I_{yy} +$

$8c^3d^6eI_{xy}^2I_{yy} + 8cd^8eI_{xy}^2I_{yy} - 12c^4d^4e^2I_{xy}^2I_{yy} - 12c^2d^6e^2I_{xy}^2I_{yy} + 8c^5d^2e^3I_{xy}^2I_{yy} + 8c^3d^4e^3I_{xy}^2I_{yy} - 2c^6e^4I_{xy}^2I_{yy} - 2c^4d^2e^4I_{xy}^2I_{yy} + 18c^2d^6I_x^2I_{xy}^2I_{yy} + 8d^8I_x^2I_{xy}^2I_{yy} - 36c^3d^4eI_x^2I_{xy}^2I_{yy} +$

$4cd^6eI_x^2I_{xy}^2I_{yy} + 18c^4d^2e^2I_x^2I_{xy}^2I_{yy} - 24c^2d^4e^2I_x^2I_{xy}^2I_{yy} + 18d^6e^2I_x^2I_{xy}^2I_{yy} + 4c^3d^2e^3I_x^2I_{xy}^2I_{yy} - 36cd^4e^3I_x^2I_{xy}^2I_{yy} + 8c^4e^4I_x^2I_{xy}^2I_{yy} + 18c^2d^2e^4I_x^2I_{xy}^2I_{yy} - 2d^8I_{xx}I_{xy}^2I_{yy} + 8cd^6eI_{xx}I_{xy}^2I_{yy} -$

$12c^2d^4e^2I_{xx}I_{xy}^2I_{yy} + 8c^3d^2e^3I_{xx}I_{xy}^2I_{yy} - 2c^4e^4I_{xx}I_{xy}^2I_{yy} - 8c^3d^9I_xI_yI_{yy} - 8cd^{11}I_xI_yI_{yy} + 32c^4d^7eI_xI_yI_{yy} + 24c^2d^9eI_xI_yI_{yy} - 8d^{11}eI_xI_yI_{yy} - 48c^5d^5e^2I_xI_yI_{yy} - 16c^3d^7e^2I_xI_yI_{yy} +$

$32cd^9e^2I_xI_yI_{yy} + 32c^6d^3e^3I_xI_yI_{yy} - 16c^4d^5e^3I_xI_yI_{yy} - 48c^2d^7e^3I_xI_yI_{yy} - 8c^7de^4I_xI_yI_{yy} + 24c^5d^3e^4I_xI_yI_{yy} + 32c^3d^5e^4I_xI_yI_{yy} - 8c^6de^5I_xI_yI_{yy} - 8c^4d^3e^5I_xI_yI_{yy} - 8c^3d^7I_x^3I_yI_{yy} -$

$36cd^9I_x^3I_yI_{yy} + 16c^4d^5eI_x^3I_yI_{yy} + 120c^2d^7eI_x^3I_yI_{yy} - 36d^9eI_x^3I_yI_{yy} - 8c^5d^3e^2I_x^3I_yI_{yy} - 168c^3d^5e^2I_x^3I_yI_{yy} + 120cd^7e^2I_x^3I_yI_{yy} + 120c^4d^3e^3I_x^3I_yI_{yy} - 168c^2d^5e^3I_x^3I_yI_{yy} - 8d^7e^3I_x^3I_yI_{yy} -$

$36c^5de^4I_x^3I_yI_{yy} + 120c^3d^3e^4I_x^3I_yI_{yy} + 16cd^5e^4I_x^3I_yI_{yy} - 36c^4de^5I_x^3I_yI_{yy} - 8c^2d^3e^5I_x^3I_yI_{yy} - 16cd^7I_x^5I_yI_{yy} + 32c^2d^5eI_x^5I_yI_{yy} - 16d^7eI_x^5I_yI_{yy} - 16c^3d^3e^2I_x^5I_yI_{yy} + 16cd^5e^2I_x^5I_yI_{yy} +$

$16c^2d^3e^3I_x^5I_yI_{yy} - 16d^5e^3I_x^5I_yI_{yy} - 16c^3de^4I_x^5I_yI_{yy} + 32cd^3e^4I_x^5I_yI_{yy} - 16c^2de^5I_x^5I_yI_{yy} - 8c^3d^7I_xI_{xx}I_yI_{yy} + 12cd^9I_xI_{xx}I_yI_{yy} + 16c^4d^5eI_xI_{xx}I_yI_{yy} - 72c^2d^7eI_xI_{xx}I_yI_{yy} +$

$12d^9eI_xI_{xx}I_yI_{yy} - 8c^5d^3e^2I_xI_{xx}I_yI_{yy} + 120c^3d^5e^2I_xI_{xx}I_yI_{yy} - 72cd^7e^2I_xI_{xx}I_yI_{yy} - 72c^4d^3e^3I_xI_{xx}I_yI_{yy} + 120c^2d^5e^3I_xI_{xx}I_yI_{yy} - 8d^7e^3I_xI_{xx}I_yI_{yy} + 12c^5de^4I_xI_{xx}I_yI_{yy} -$

$72c^3d^3e^4I_xI_{xx}I_yI_{yy} + 16cd^5e^4I_xI_{xx}I_yI_{yy} + 12c^4de^5I_xI_{xx}I_yI_{yy} - 8c^2d^3e^5I_xI_{xx}I_yI_{yy} - 8c^3d^5I_x^3I_{xx}I_yI_{yy} - 4cd^7I_x^3I_{xx}I_yI_{yy} - 16c^2d^5eI_x^3I_{xx}I_yI_{yy} - 4d^7eI_x^3I_{xx}I_yI_{yy} - 12c^3d^3e^2I_x^3I_{xx}I_yI_{yy} -$

$20cd^5e^2I_x^3I_{xx}I_yI_{yy} - 20c^2d^3e^3I_x^3I_{xx}I_yI_{yy} - 12d^5e^3I_x^3I_{xx}I_yI_{yy} - 4c^3de^4I_x^3I_{xx}I_yI_{yy} - 16cd^3e^4I_x^3I_{xx}I_yI_{yy} - 4c^2de^5I_x^3I_{xx}I_yI_{yy} - 8d^3e^5I_x^3I_{xx}I_yI_{yy} + 4cd^7I_xI_{xx}^2I_yI_{yy} - 8c^2d^5eI_xI_{xx}^2I_yI_{yy} +$

$4d^7eI_xI_{xx}^2I_yI_{yy} + 4c^3d^3e^2I_xI_{xx}^2I_yI_{yy} - 4cd^5e^2I_xI_{xx}^2I_yI_{yy} - 4c^2d^3e^3I_xI_{xx}^2I_yI_{yy} + 4d^5e^3I_xI_{xx}^2I_yI_{yy} + 4c^3de^4I_xI_{xx}^2I_yI_{yy} - 8cd^3e^4I_xI_{xx}^2I_yI_{yy} + 4c^2de^5I_xI_{xx}^2I_yI_{yy} + 16c^4d^6I_xI_{xy}I_yI_{yy} -$

$4c^2d^8I_xI_{xy}I_yI_{yy} - 20d^{10}I_xI_{xy}I_yI_{yy} - 32c^5d^4eI_xI_{xy}I_yI_{yy} + 80c^3d^6eI_xI_{xy}I_yI_{yy} + 112cd^8eI_xI_{xy}I_yI_{yy} + 16c^6d^2e^2I_xI_{xy}I_yI_{yy} - 168c^4d^4e^2I_xI_{xy}I_yI_{yy} - 168c^2d^6e^2I_xI_{xy}I_yI_{yy} +$

$16d^8e^2I_xI_{xy}I_yI_{yy} + 112c^5d^2e^3I_xI_{xy}I_yI_{yy} + 80c^3d^4e^3I_xI_{xy}I_yI_{yy} - 32cd^6e^3I_xI_{xy}I_yI_{yy} - 20c^6e^4I_xI_{xy}I_yI_{yy} - 4c^4d^2e^4I_xI_{xy}I_yI_{yy} + 16c^2d^4e^4I_xI_{xy}I_yI_{yy} + 16c^4d^4I_x^3I_{xy}I_yI_{yy} +$

$12c^2d^6I_x^3I_{xy}I_yI_{yy} - 32d^8I_x^3I_{xy}I_yI_{yy} + 40c^3d^4eI_x^3I_{xy}I_yI_{yy} + 152cd^6eI_x^3I_{xy}I_yI_{yy} + 12c^4d^2e^2I_x^3I_{xy}I_yI_{yy} - 144c^2d^4e^2I_x^3I_{xy}I_yI_{yy} + 12d^6e^2I_x^3I_{xy}I_yI_{yy} + 152c^3d^2e^3I_x^3I_{xy}I_yI_{yy} +$

$40cd^4e^3I_x^3I_{xy}I_yI_{yy} - 32c^4e^4I_x^3I_{xy}I_yI_{yy} + 12c^2d^2e^4I_x^3I_{xy}I_yI_{yy} + 16d^4e^4I_x^3I_{xy}I_yI_{yy} - 24c^2d^6I_xI_{xx}I_{xy}I_yI_{yy} - 4d^8I_xI_{xx}I_{xy}I_yI_{yy} + 48c^3d^4eI_xI_{xx}I_{xy}I_yI_{yy} - 32cd^6eI_xI_{xx}I_{xy}I_yI_{yy} -$

$24c^4d^2e^2I_xI_{xx}I_{xy}I_yI_{yy} + 72c^2d^4e^2I_xI_{xx}I_{xy}I_yI_{yy} - 24d^6e^2I_xI_{xx}I_{xy}I_yI_{yy} - 32c^3d^2e^3I_xI_{xx}I_{xy}I_yI_{yy} + 48cd^4e^3I_xI_{xx}I_{xy}I_yI_{yy} - 4c^4e^4I_xI_{xx}I_{xy}I_yI_{yy} - 24c^2d^2e^4I_xI_{xx}I_{xy}I_yI_{yy} +$

$28c^3d^5I_xI_{xy}^2I_yI_{yy} + 28cd^7I_xI_{xy}^2I_yI_{yy} - 56c^4d^3eI_xI_{xy}^2I_yI_{yy} - 28c^2d^5eI_xI_{xy}^2I_yI_{yy} + 28d^7eI_xI_{xy}^2I_yI_{yy} + 28c^5de^2I_xI_{xy}^2I_yI_{yy} - 28c^3d^3e^2I_xI_{xy}^2I_yI_{yy} - 56cd^5e^2I_xI_{xy}^2I_yI_{yy} +$

$28c^4de^3I_xI_{xy}^2I_yI_{yy} + 28c^2d^3e^3I_xI_{xy}^2I_yI_{yy} + 4c^4d^8I_y^2I_{yy} + 8c^2d^{10}I_y^2I_{yy} + 4d^{12}I_y^2I_{yy} - 16c^5d^6eI_y^2I_{yy} - 32c^3d^8eI_y^2I_{yy} - 16cd^{10}eI_y^2I_{yy} + 24c^6d^4e^2I_y^2I_{yy} + 48c^4d^6e^2I_y^2I_{yy} +$

$24c^2d^8e^2I_y^2I_{yy} - 16c^7d^2e^3I_y^2I_{yy} - 32c^5d^4e^3I_y^2I_{yy} - 16c^3d^6e^3I_y^2I_{yy} + 4c^8e^4I_y^2I_{yy} + 8c^6d^2e^4I_y^2I_{yy} + 4c^4d^4e^4I_y^2I_{yy} + 12c^4d^6I_x^2I_y^2I_{yy} + 30c^2d^8I_x^2I_y^2I_{yy} + 18d^{10}I_x^2I_y^2I_{yy} -$

$24c^5d^4eI_x^2I_y^2I_{yy} - 72c^3d^6eI_x^2I_y^2I_{yy} - 48cd^8eI_x^2I_y^2I_{yy} + 12c^6d^2e^2I_x^2I_y^2I_{yy} + 72c^4d^4e^2I_x^2I_y^2I_{yy} + 72c^2d^6e^2I_x^2I_y^2I_{yy} + 12d^8e^2I_x^2I_y^2I_{yy} - 48c^5d^2e^3I_x^2I_y^2I_{yy} - 72c^3d^4e^3I_x^2I_y^2I_{yy} -$

$24cd^6e^3I_x^2I_y^2I_{yy} + 18c^6e^4I_x^2I_y^2I_{yy} + 30c^4d^2e^4I_x^2I_y^2I_{yy} + 12c^2d^4e^4I_x^2I_y^2I_{yy} + 24c^2d^6I_x^4I_y^2I_{yy} + 8d^8I_x^4I_y^2I_{yy} - 48c^3d^4eI_x^4I_y^2I_{yy} + 16cd^6eI_x^4I_y^2I_{yy} + 24c^4d^2e^2I_x^4I_y^2I_{yy} - 48c^2d^4e^2I_x^4I_y^2I_{yy} +$

$24d^6e^2I_x^4I_y^2I_{yy} + 16c^3d^2e^3I_x^4I_y^2I_{yy} - 48cd^4e^3I_x^4I_y^2I_{yy} + 8c^4e^4I_x^4I_y^2I_{yy} + 24c^2d^2e^4I_x^4I_y^2I_{yy} + 4c^4d^6I_{xx}I_y^2I_{yy} + 8c^2d^8I_{xx}I_y^2I_{yy} + 4d^{10}I_{xx}I_y^2I_{yy} - 8c^5d^4eI_{xx}I_y^2I_{yy} - 16c^3d^6eI_{xx}I_y^2I_{yy} -$

$8cd^8eI_{xx}I_y^2I_{yy} + 4c^6d^2e^2I_{xx}I_y^2I_{yy} + 12c^4d^4e^2I_{xx}I_y^2I_{yy} + 12c^2d^6e^2I_{xx}I_y^2I_{yy} + 4d^8e^2I_{xx}I_y^2I_{yy} - 8c^5d^2e^3I_{xx}I_y^2I_{yy} - 16c^3d^4e^3I_{xx}I_y^2I_{yy} - 8cd^6e^3I_{xx}I_y^2I_{yy} + 4c^6e^4I_{xx}I_y^2I_{yy} +$

$8c^4d^2e^4I_{xx}I_y^2I_{yy} + 4c^2d^4e^4I_{xx}I_y^2I_{yy} + 12c^4d^4I_x^2I_{xx}I_y^2I_{yy} + 12c^2d^6I_x^2I_{xx}I_y^2I_{yy} + 18d^8I_x^2I_{xx}I_y^2I_{yy} + 24c^3d^4eI_x^2I_{xx}I_y^2I_{yy} - 48cd^6eI_x^2I_{xx}I_y^2I_{yy} + 12c^4d^2e^2I_x^2I_{xx}I_y^2I_{yy} + 132c^2d^4e^2I_x^2I_{xx}I_y^2I_{yy} +$

$12d^6e^2I_x^2I_{xx}I_y^2I_{yy} - 48c^3d^2e^3I_x^2I_{xx}I_y^2I_{yy} + 24cd^4e^3I_x^2I_{xx}I_y^2I_{yy} + 18c^4e^4I_x^2I_{xx}I_y^2I_{yy} + 12c^2d^2e^4I_x^2I_{xx}I_y^2I_{yy} + 12d^4e^4I_x^2I_{xx}I_y^2I_{yy} + 6c^2d^6I_{xx}^2I_y^2I_{yy} - 12c^3d^4eI_{xx}^2I_y^2I_{yy} + 12cd^6eI_{xx}^2I_y^2I_{yy} +$

$6c^4d^2e^2I_{xx}^2I_y^2I_{yy} - 24c^2d^4e^2I_{xx}^2I_y^2I_{yy} + 6d^6e^2I_{xx}^2I_y^2I_{yy} + 12c^3d^2e^3I_{xx}^2I_y^2I_{yy} - 12cd^4e^3I_{xx}^2I_y^2I_{yy} + 6c^2d^2e^4I_{xx}^2I_y^2I_{yy} - 8c^5d^5I_{xy}I_y^2I_{yy} - 16c^3d^7I_{xy}I_y^2I_{yy} - 8cd^9I_{xy}I_y^2I_{yy} + 16c^6d^3eI_{xy}I_y^2I_{yy} +$

$24c^4d^5eI_{xy}I_y^2I_{yy} - 8d^9eI_{xy}I_y^2I_{yy} - 8c^7de^2I_{xy}I_y^2I_{yy} + 24c^3d^5e^2I_{xy}I_y^2I_{yy} + 16cd^7e^2I_{xy}I_y^2I_{yy} - 8c^6de^3I_{xy}I_y^2I_{yy} - 16c^4d^3e^3I_{xy}I_y^2I_{yy} - 8c^2d^5e^3I_{xy}I_y^2I_{yy} - 24c^5d^3I_x^2I_{xy}I_y^2I_{yy} -$

$36c^3d^5I_x^2I_{xy}I_y^2I_{yy} - 12cd^7I_x^2I_{xy}I_y^2I_{yy} - 48c^4d^3eI_x^2I_{xy}I_y^2I_{yy} - 60c^2d^5eI_x^2I_{xy}I_y^2I_{yy} - 12d^7eI_x^2I_{xy}I_y^2I_{yy} - 12c^5de^2I_x^2I_{xy}I_y^2I_{yy} - 60c^3d^3e^2I_x^2I_{xy}I_y^2I_{yy} - 48cd^5e^2I_x^2I_{xy}I_y^2I_{yy} -$

$12c^4de^3I_x^2I_{xy}I_y^2I_{yy} - 36c^2d^3e^3I_x^2I_{xy}I_y^2I_{yy} - 24d^5e^3I_x^2I_{xy}I_y^2I_{yy} - 12c^3d^5I_{xx}I_{xy}I_y^2I_{yy} - 12cd^7I_{xx}I_{xy}I_y^2I_{yy} + 24c^4d^3eI_{xx}I_{xy}I_y^2I_{yy} + 12c^2d^5eI_{xx}I_{xy}I_y^2I_{yy} - 12d^7eI_{xx}I_{xy}I_y^2I_{yy} -$

$12c^5de^2I_{xx}I_{xy}I_y^2I_{yy} + 12c^3d^3e^2I_{xx}I_{xy}I_y^2I_{yy} + 24cd^5e^2I_{xx}I_{xy}I_y^2I_{yy} - 12c^4de^3I_{xx}I_{xy}I_y^2I_{yy} - 12c^2d^3e^3I_{xx}I_{xy}I_y^2I_{yy} + 2c^4d^4I_{xy}^2I_y^2I_{yy} + 4c^2d^6I_{xy}^2I_y^2I_{yy} + 2d^8I_{xy}^2I_y^2I_{yy} - 4c^5d^2eI_{xy}^2I_y^2I_{yy} -$

$8c^3d^4eI_{xy}^2I_y^2I_{yy} - 4cd^6eI_{xy}^2I_y^2I_{yy} + 2c^6e^2I_{xy}^2I_y^2I_{yy} + 4c^4d^2e^2I_{xy}^2I_y^2I_{yy} + 2c^2d^4e^2I_{xy}^2I_y^2I_{yy} - 8c^5d^5I_xI_y^3I_{yy} - 16c^3d^7I_xI_y^3I_{yy} - 8cd^9I_xI_y^3I_{yy} + 16c^6d^3eI_xI_y^3I_{yy} + 24c^4d^5eI_xI_y^3I_{yy} -$

$8d^9eI_xI_y^3I_{yy} - 8c^7de^2I_xI_y^3I_{yy} + 24c^3d^5e^2I_xI_y^3I_{yy} + 16cd^7e^2I_xI_y^3I_{yy} - 8c^6de^3I_xI_y^3I_{yy} - 16c^4d^3e^3I_xI_y^3I_{yy} - 8c^2d^5e^3I_xI_y^3I_{yy} - 16c^3d^5I_x^3I_y^3I_{yy} - 16cd^7I_x^3I_y^3I_{yy} + 32c^4d^3eI_x^3I_y^3I_{yy} +$

$16c^2d^5eI_x^3I_y^3I_{yy} - 16d^7eI_x^3I_y^3I_{yy} - 16c^5de^2I_x^3I_y^3I_{yy} + 16c^3d^3e^2I_x^3I_y^3I_{yy} + 32cd^5e^2I_x^3I_y^3I_{yy} - 16c^4de^3I_x^3I_y^3I_{yy} - 16c^2d^3e^3I_x^3I_y^3I_{yy} - 8c^5d^3I_xI_{xx}I_y^3I_{yy} - 12c^3d^5I_xI_{xx}I_y^3I_{yy} - 4cd^7I_xI_{xx}I_y^3I_{yy} -$

$16c^4d^3eI_xI_{xx}I_y^3I_{yy} - 20c^2d^5eI_xI_{xx}I_y^3I_{yy} - 4d^7eI_xI_{xx}I_y^3I_{yy} - 4c^5de^2I_xI_{xx}I_y^3I_{yy} - 20c^3d^3e^2I_xI_{xx}I_y^3I_{yy} - 16cd^5e^2I_xI_{xx}I_y^3I_{yy} - 4c^4de^3I_xI_{xx}I_y^3I_{yy} - 12c^2d^3e^3I_xI_{xx}I_y^3I_{yy} - 8d^5e^3I_xI_{xx}I_y^3I_{yy} +$

$16c^6d^2I_xI_{xy}I_y^3I_{yy} + 36c^4d^4I_xI_{xy}I_y^3I_{yy} + 24c^2d^6I_xI_{xy}I_y^3I_{yy} + 4d^8I_xI_{xy}I_y^3I_{yy} + 24c^5d^2eI_xI_{xy}I_y^3I_{yy} + 48c^3d^4eI_xI_{xy}I_y^3I_{yy} + 24cd^6eI_xI_{xy}I_y^3I_{yy} + 4c^6e^2I_xI_{xy}I_y^3I_{yy} + 24c^4d^2e^2I_xI_{xy}I_y^3I_{yy} +$

$36c^2d^4e^2I_xI_{xy}I_y^3I_{yy} + 16d^6e^2I_xI_{xy}I_y^3I_{yy} + 2c^6d^4I_y^4I_{yy} + 6c^4d^6I_y^4I_{yy} + 6c^2d^8I_y^4I_{yy} + 2d^{10}I_y^4I_{yy} - 4c^7d^2eI_y^4I_{yy} - 12c^5d^4eI_y^4I_{yy} - 12c^3d^6eI_y^4I_{yy} - 4cd^8eI_y^4I_{yy} + 2c^8e^2I_y^4I_{yy} +$

$$6c^6d^2e^2I_y^4I_{yy}+6c^4d^4e^2I_y^4I_{yy}+2c^2d^6e^2I_y^4I_{yy}+4c^4d^4I_x^2I_y^4I_{yy}+8c^2d^6I_x^2I_y^4I_{yy}+4d^8I_x^2I_y^4I_{yy}-8c^5d^2eI_x^2I_y^4I_{yy}-16c^3d^4eI_x^2I_y^4I_{yy}-8cd^6eI_x^2I_y^4I_{yy}+4c^6e^2I_x^2I_y^4I_{yy}+8c^4d^2e^2I_x^2I_y^4I_{yy}+$$

$$4c^2d^4e^2I_x^2I_y^4I_{yy}+2c^6d^2I_{xx}I_y^4I_{yy}+4c^4d^4I_{xx}I_y^4I_{yy}+2c^2d^6I_{xx}I_y^4I_{yy}+4c^5d^2eI_{xx}I_y^4I_{yy}+8c^3d^4eI_{xx}I_y^4I_{yy}+4cd^6eI_{xx}I_y^4I_{yy}+2c^4d^2e^2I_{xx}I_y^4I_{yy}+4c^2d^4e^2I_{xx}I_y^4I_{yy}+2d^6e^2I_{xx}I_y^4I_{yy}-$$

$$4c^7dI_{xy}I_y^4I_{yy}-12c^5d^3I_{xy}I_y^4I_{yy}-12c^3d^5I_{xy}I_y^4I_{yy}-4cd^7I_{xy}I_y^4I_{yy}-4c^6deI_{xy}I_y^4I_{yy}-12c^4d^3eI_{xy}I_y^4I_{yy}-12c^2d^5eI_{xy}I_y^4I_{yy}-4d^7eI_{xy}I_y^4I_{yy}+c^4d^8I_{yy}^2+2c^2d^{10}I_{yy}^2+d^{12}I_{yy}^2-4c^5d^6eI_{yy}^2-$$

$$8c^3d^8eI_{yy}^2-4cd^{10}eI_{yy}^2+6c^6d^4e^2I_{yy}^2+12c^4d^6e^2I_{yy}^2+6c^2d^8e^2I_{yy}^2-4c^7d^2e^3I_{yy}^2-8c^5d^4e^3I_{yy}^2-4c^3d^6e^3I_{yy}^2+c^8e^4I_{yy}^2+2c^6d^2e^4I_{yy}^2+c^4d^4e^4I_{yy}^2+2c^4d^6I_x^2I_{yy}^2+14c^2d^8I_x^2I_{yy}^2+$$

$$12d^{10}I_x^2I_{yy}^2-4c^5d^4eI_x^2I_{yy}^2-48c^3d^6eI_x^2I_{yy}^2-44cd^8eI_x^2I_{yy}^2+2c^6d^2e^2I_x^2I_{yy}^2+66c^4d^4e^2I_x^2I_{yy}^2+66c^2d^6e^2I_x^2I_{yy}^2+2d^8e^2I_x^2I_{yy}^2-44c^5d^2e^3I_x^2I_{yy}^2-48c^3d^4e^3I_x^2I_{yy}^2-4cd^6e^3I_x^2I_{yy}^2+$$

$$12c^6e^4I_x^2I_{yy}^2+14c^4d^2e^4I_x^2I_{yy}^2+2c^2d^4e^4I_x^2I_{yy}^2+c^4d^4I_x^4I_{yy}^2+8d^8I_x^4I_{yy}^2+4c^3d^4eI_x^4I_{yy}^2-32cd^6eI_x^4I_{yy}^2+54c^2d^4e^2I_x^4I_{yy}^2-32c^3d^2e^3I_x^4I_{yy}^2+4cd^4e^3I_x^4I_{yy}^2+8c^4e^4I_x^4I_{yy}^2+d^4e^4I_x^4I_{yy}^2+$$

$$2c^2d^8I_{xx}I_{yy}^2+2d^{10}I_{xx}I_{yy}^2-8c^3d^6eI_{xx}I_{yy}^2-8cd^8eI_{xx}I_{yy}^2+12c^4d^4e^2I_{xx}I_{yy}^2+12c^2d^6e^2I_{xx}I_{yy}^2-8c^5d^2e^3I_{xx}I_{yy}^2-8c^3d^4e^3I_{xx}I_{yy}^2+2c^6e^4I_{xx}I_{yy}^2+2c^4d^2e^4I_{xx}I_{yy}^2+6c^2d^6I_x^2I_{xx}I_{yy}^2-$$

$$12c^3d^4eI_x^2I_{xx}I_{yy}^2+12cd^6eI_x^2I_{xx}I_{yy}^2+6c^4d^2e^2I_x^2I_{xx}I_{yy}^2-24c^2d^4e^2I_x^2I_{xx}I_{yy}^2+6d^6e^2I_x^2I_{xx}I_{yy}^2+12c^3d^2e^3I_x^2I_{xx}I_{yy}^2-12cd^4e^3I_x^2I_{xx}I_{yy}^2+6c^2d^2e^4I_x^2I_{xx}I_{yy}^2+d^8I_{xx}^2I_{yy}^2-4cd^6eI_{xx}^2I_{yy}^2+$$

$$6c^2d^4e^2I_{xx}^2I_{yy}^2-4c^3d^2e^3I_{xx}^2I_{yy}^2+c^4e^4I_{xx}^2I_{yy}^2-16c^3d^5I_x^2I_{xy}I_{yy}^2-16cd^7I_x^2I_{xy}I_{yy}^2+32c^4d^3eI_x^2I_{xy}I_{yy}^2+16c^2d^5eI_x^2I_{xy}I_{yy}^2-16d^7eI_x^2I_{xy}I_{yy}^2-16c^5de^2I_x^2I_{xy}I_{yy}^2+16c^3d^3e^2I_x^2I_{xy}I_{yy}^2+$$

$$32cd^5e^2I_x^2I_{xy}I_{yy}^2-16c^4de^3I_x^2I_{xy}I_{yy}^2-16c^2d^3e^3I_x^2I_{xy}I_{yy}^2-4c^5d^5I_xI_yI_{yy}^2-8c^3d^7I_xI_yI_{yy}^2-4cd^9I_xI_yI_{yy}^2+8c^6d^3eI_xI_yI_{yy}^2+12c^4d^5eI_xI_yI_{yy}^2-4d^9eI_xI_yI_{yy}^2-4c^7de^2I_xI_yI_{yy}^2+$$

$$12c^3d^5e^2I_xI_yI_{yy}^2+8cd^7e^2I_xI_yI_{yy}^2-4c^6de^3I_xI_yI_{yy}^2-8c^4d^3e^3I_xI_yI_{yy}^2-4c^2d^5e^3I_xI_yI_{yy}^2-4c^5d^3I_x^3I_yI_{yy}^2-4c^3d^5I_x^3I_yI_{yy}^2-12c^4d^3eI_x^3I_yI_{yy}^2-12c^2d^5eI_x^3I_yI_{yy}^2-12c^3d^3e^2I_x^3I_yI_{yy}^2-$$

$$12cd^5e^2I_x^3I_yI_{yy}^2-4c^2d^3e^3I_x^3I_yI_{yy}^2-4d^5e^3I_x^3I_yI_{yy}^2+4c^3d^5I_xI_{xx}I_yI_{yy}^2+4cd^7I_xI_{xx}I_yI_{yy}^2-8c^4d^3eI_xI_{xx}I_yI_{yy}^2-4c^2d^5eI_xI_{xx}I_yI_{yy}^2+4d^7eI_xI_{xx}I_yI_{yy}^2+4c^5de^2I_xI_{xx}I_yI_{yy}^2-$$

$$4c^3d^3e^2I_xI_{xx}I_yI_{yy}^2-8cd^5e^2I_xI_{xx}I_yI_{yy}^2+4c^4de^3I_xI_{xx}I_yI_{yy}^2+4c^2d^3e^3I_xI_{xx}I_yI_{yy}^2-8c^4d^4I_xI_{xy}I_yI_{yy}^2-16c^2d^6I_xI_{xy}I_yI_{yy}^2-8d^8I_xI_{xy}I_yI_{yy}^2+16c^5d^2eI_xI_{xy}I_yI_{yy}^2+32c^3d^4eI_xI_{xy}I_yI_{yy}^2+$$

$$16cd^6eI_xI_{xy}I_yI_{yy}^2-8c^6e^2I_xI_{xy}I_yI_{yy}^2-16c^4d^2e^2I_xI_{xy}I_yI_{yy}^2-8c^2d^4e^2I_xI_{xy}I_yI_{yy}^2+2c^6d^4I_y^2I_{yy}^2+6c^4d^6I_y^2I_{yy}^2+6c^2d^8I_y^2I_{yy}^2+2d^{10}I_y^2I_{yy}^2-4c^7d^2eI_y^2I_{yy}^2-12c^5d^4eI_y^2I_{yy}^2-12c^3d^6eI_y^2I_{yy}^2-$$

$$4cd^8eI_y^2I_{yy}^2+2c^8e^2I_y^2I_{yy}^2+6c^6d^2e^2I_y^2I_{yy}^2+6c^4d^4e^2I_y^2I_{yy}^2+2c^2d^6e^2I_y^2I_{yy}^2+6c^6d^2I_x^2I_y^2I_{yy}^2+12c^4d^4I_x^2I_y^2I_{yy}^2+6c^2d^6I_x^2I_y^2I_{yy}^2+12c^5d^2eI_x^2I_y^2I_{yy}^2+24c^3d^4eI_x^2I_y^2I_{yy}^2+12cd^6eI_x^2I_y^2I_{yy}^2+$$

$$6c^4d^2e^2I_x^2I_y^2I_{yy}^2+12c^2d^4e^2I_x^2I_y^2I_{yy}^2+6d^6e^2I_x^2I_y^2I_{yy}^2+2c^4d^4I_{xx}I_y^2I_{yy}^2+4c^2d^6I_{xx}I_y^2I_{yy}^2+2d^8I_{xx}I_y^2I_{yy}^2-4c^5d^2eI_{xx}I_y^2I_{yy}^2-8c^3d^4eI_{xx}I_y^2I_{yy}^2-4cd^6eI_{xx}I_y^2I_{yy}^2+2c^6e^2I_{xx}I_y^2I_{yy}^2+$$

$$4c^4d^2e^2I_{xx}I_y^2I_{yy}^2+2c^2d^4e^2I_{xx}I_y^2I_{yy}^2-4c^7dI_xI_y^3I_{yy}^2-12c^5d^3I_xI_y^3I_{yy}^2-12c^3d^5I_xI_y^3I_{yy}^2-4cd^7I_xI_y^3I_{yy}^2-4c^6deI_xI_y^3I_{yy}^2-12c^4d^3eI_xI_y^3I_{yy}^2-12c^2d^5eI_xI_y^3I_{yy}^2-4d^7eI_xI_y^3I_{yy}^2+c^8I_y^4I_{yy}^2+$$

$$4c^6d^2I_y^4I_{yy}^2+6c^4d^4I_y^4I_{yy}^2+4c^2d^6I_y^4I_{yy}^2+d^8I_y^4I_{yy}^2+4c^4d^4I_x^2I_{yy}^3+8c^2d^6I_x^2I_{yy}^3+4d^8I_x^2I_{yy}^3-8c^5d^2eI_x^2I_{yy}^3-16c^3d^4eI_x^2I_{yy}^3-8cd^6eI_x^2I_{yy}^3+4c^6e^2I_x^2I_{yy}^3+8c^4d^2e^2I_x^2I_{yy}^3+4c^2d^4e^2I_x^2I_{yy}^3.$$

References

[1] Améndola, C. & Rodriguez, J. I. (2016). Solving parameterized polynomial systems with decomposable projections. *arXiv preprint arXiv:1612.08807*.

[2] Andoni, A., Panigrahy, R., Valiant, G., & Zhang, L. (2014). Learning polynomials with neural networks. In *International conference on machine learning* (pp. 1908–1916).

[3] Barron, J. T. & Malik, J. (2015). Shape, illumination, and reflectance from shading. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 37(8), 1670–1687.

[4] Basri, R., Jacobs, D., & Kemelmacher, I. (2007). Photometric stereo with general, unknown lighting. *International Journal of computer vision*, 72(3), 239–257.

[5] Basri, R. & Jacobs, D. W. (2003). Lambertian reflectance and linear subspaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, (2), 218–233.

[6] Basu, S., Pollack, R., & Roy, M.-F. (2006). *Algorithms in real algebraic geometry*, volume 10. Springer.

[7] Bates, D. J., Sommese, A. J., Hauenstein, J. D., & Wampler, C. W. (2013). *Numerically solving polynomial systems with Bertini*. SIAM.

[8] Bednarık, J., Fua, P., & Salzmann, M. (2018). Learning shape-from-shading for deformable surfaces. *arXiv preprint arXiv:1803.08908*.

[9] Belhumeur, P. N., Kriegman, D. J., & Yuille, A. L. (1999). The bas-relief ambiguity. *International Journal of Computer Vision (IJCV)*, 35(1), 33–44.

[10] Bochnak, J., Coste, M., & Roy, M.-F. (2013). *Real algebraic geometry*, volume 36. Springer Science & Business Media.

[11] Brand, M. (2003). Charting a manifold. In *Advances in neural information processing systems* (pp. 985–992).

[12] Breiding, P., Kališnik, S., Sturmfels, B., & Weinstein, M. (2018). Learning algebraic varieties from samples. *Revista Matemática Complutense*, 31(3), 545–593.

[13] Brooks, M. J. (1983). Two results concerning ambiguity in shape from shading. In *AAAI*.

[14] Cavanagh, P. (2005). The artist as neuroscientist. *Nature*, 434, 301–307.

[15] Chakraborty, R., Bouza, J., Manton, J., & Vemuri, B. C. (2018). Manifoldnet: A deep network framework for manifold-valued data. *arXiv preprint arXiv:1809.06211*.

[16] Chionh, E.-W., Goldman, R. N., & Miller, J. R. (1991). Using multivariate resultants to find the intersection of three quadric surfaces. *ACM Transactions on Graphics (TOG)*, 10(4), 378–400.

[17] Cox, D., Little, J., & O'Shea, D. (2006). *Using algebraic geometry*, volume 185. Springer Science & Business Media.

[18] Cox, D., Little, J., & O'Shea, D. (2013). *Ideals, varieties, and algorithms: an introduction to computational algebraic geometry and commutative algebra*. Springer Science & Business Media.

[19] Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4), 303–314.

[20] Drbohlav, O. & Chaniler, M. (2005). Can two specular pixels calibrate photometric stereo? In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 2 (pp. 1850–1857 Vol. 2).

[21] Ecker, A. & Jepson, A. D. (2010). Polynomial shape from shading. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (pp. 145–152).: IEEE.

[22] Eigen, D., Puhrsch, C., & Fergus, R. (2014). Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems (NeurIPS)* (pp. 2366–2374).

[23] Fan, J. & Wolff, L. B. (1997). Surface curvature and shape reconstruction from unknown multiple illumination and integrability. *Computer Vision and Image Understanding*, 65(2), 347–359.

[24] Farouki, R., Neff, C., & O'Conner, M. (1989). Automatic parsing of degenerate quadric-surface intersections. *ACM Transactions on Graphics (TOG)*, 8(3), 174–203.

[25] Finn, C., Abbeel, P., & Levine, S. (2017). Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning (ICML)*, volume 70 (pp. 1126–1135).: PMLR.

[26] Florack, L., ter Haar Romeny, B., Koenderink, J., & Viergever, M. (1992). *General Intensity Transformations and Second Order Invariants*, (pp. 22–29).

[27] Forsyth, D. A. (2011). Variable-source shading analysis. *International Journal of Computer Vision (IJCV)*, 91(3), 280–302.

[28] Gallian, J. (2012). *Contemporary abstract algebra*. Nelson Education.

[29] Genova, K., Cole, F., Sud, A., Sarna, A., & Funkhouser, T. (2020). Local deep implicit functions for 3d shape. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 4857–4866).

[30] Gordon, J., Bronskill, J., Bauer, M., Nowozin, S., & Turner, R. E. (2018). Meta-learning probabilistic inference for prediction. *arXiv preprint arXiv:1805.09921*.

[31] Groueix, T., Fisher, M., Kim, V. G., Russell, B. C., & Aubry, M. (2018). A papier-mâché approach to learning 3d surface generation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 216–224).

[32] Guillemin, V. & Pollack, A. (2010). *Differential topology*, volume 370. American Mathematical Soc.

[33] Harris, J. (2013). *Algebraic geometry: a first course*, volume 133. Springer Science & Business Media.

[34] Hartley, R. & Zisserman, A. (2003). *Multiple view geometry in computer vision*. Cambridge university press.

[35] Hartshorne, R. (1977). *Algebraic geometry*. New York: Springer-Verlag. Graduate Texts in Mathematics, No. 52.

[36] Hayakawa, H. (1994). Photometric stereo under a light source with arbitrary motion. *Journal of the Optical Society of America (JOSA) A*, 11(11), 3079–3089.

[37] Heal, K., Wang, J., Gortler, S. J., & Zickler, T. (2020). A lighting-invariant point processor for shading. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 94–102).

[38] Holtmann-Rice, D., Kunsberg, B., & Zucker, S. (2018). Tensors, differential geometry and statistical shading analysis. *Journal of Mathematical Imaging and Vision*, 60(6), 968–992.

[39] Horn, B. K. (1970). *Shape from shading: A method for obtaining the shape of a smooth opaque object from one view*. Technical Report AITR-232, MIT Artificial Intelligence Laboratory.

[40] Horn, B. K. (1977). Understanding image intensities. *Artificial intelligence*, 8(2), 201–231.

[41] Huo, X., Ni, X. S., & Smith, A. K. (2007). A survey of manifold-based learning methods. *Recent advances in data mining of enterprise data*, (pp. 691–745).

[42] Koenderink, J. & van Doorn, A. (2014). Shape, shading, brain and awareness. In *Neuromathematics of vision* (pp. 87–106). Springer.

[43] Koenderink, J. J. & van Doorn, A. J. (1987). Representation of local geometry in the visual system. *Biological cybernetics*, 55(6), 367–375.

[44] Koenderink, J. J., Van Doorn, A. J., Kappers, A. M., & Todd, J. T. (2001). Ambiguity and the 'mental eye' in pictorial relief. *Perception*, 30(4), 431–448.

[45] Kozera, R. (1997). Uniqueness in shape from shading revisited. *Journal of Mathematical Imaging and Vision*, 7(2), 123–138.

[46] Kukelova, Z., Heller, J., & Fitzgibbon, A. (2016a). Efficient intersection of three quadrics and applications in computer vision. In *Computer Vision and Pattern Recognition (CVPR)* (pp. 1799–1808).

[47] Kukelova, Z., Heller, J., & Fitzgibbon, A. (2016b). Efficient intersection of three quadrics and applications in computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2016- (pp. 1799–1808).: IEEE.

[48] Kukelova, Z., Kileel, J., Sturmfels, B., & Pajdla, T. (2017). A clever elimination strategy for efficient minimal solvers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4912–4921).

[49] Kunsberg, B. (2014). *A Differential Geometric Approach using Orientation Fields for Shape from Shading*. Yale University.

[50] Kunsberg, B. & Zucker, S. W. (2014). How shading constrains surface patches without knowledge of light sources. *SIAM Journal on Imaging Sciences*, 7(2), 641–668.

[51] Kunsberg, B. & Zucker, S. W. (2018). Critical contours: an invariant linking image flow with salient surface organization. *SIAM Journal on Imaging Sciences*, 11(3), 1849–1877.

[52] Levin, J. (1976). A parametric algorithm for drawing pictures of solid objects composed of quadric surfaces. *Communications of the ACM*, 19(10), 555–563.

[53] Levin, J. Z. (1979). Mathematical models for determining the intersections of quadric surfaces. *Computer Graphics and Image Processing*, 11(1), 73–87.

[54] Liang, S. & Srikant, R. (2017). Why deep neural networks for function approximation? *arXiv.org*.

[55] Marr, D. (1982). *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. New York, NY, USA: Henry Holt and Co., Inc.

[56] Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., & Geiger, A. (2019). Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4460–4470).

[57] Oliensis, J. (1991). Shape from shading as a partially well-constrained problem. *CVGIP: Image Understanding*, 54(2), 163 – 183.

[58] Penna, M. A. (1989). A shape from shading analysis for a single perspective image of a polyhedron. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6), 545–554.

[59] Pentland, A. (1984). Local shading analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pami-6, 170–187.

[60] Prados, E. & Faugeras, O. (2005). A generic and provably convergent shape-from-shading method for orthographic and pinhole cameras. *International Journal of Computer Vision*, 65(1-2), 97–125.

[61] Prados, E., Faugeras, O., & Camilli, F. (2004). *Shape from Shading: a well-posed problem?* Technical report.

[62] Ramamoorthi, R. & Hanrahan, P. (2001). An efficient representation for irradiance environment maps. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques* (pp. 497–500).: ACM.

[63] Richter, S. R. & Roth, S. (2015). Discriminative shape from shading in uncalibrated illumination. In *Computer Vision and Pattern Recognition (CVPR)* (pp. 1128–1136).

[64] Sommese, A. J., Verschelde, J., & Wampler, C. W. (2001). Numerical decomposition of the solution sets of polynomial systems into irreducible components. *SIAM Journal on Numerical Analysis*, 38(6), 2022–2046.

[65] Sommese, A. J. & Wampler, C. W. (2005). Coefficient-parameter homotopy. In *The Numerical Solution Of Systems Of Polynomials Arising In Engineering And Science* (pp. 91–116). World Scientific Publishing Co. Pte. Ltd.

[66] Szeliski, R. (2010). *Computer vision: algorithms and applications*. Springer Science & Business Media.

[67] Telgarsky, M. (2017). Neural networks and rational functions. *arXiv.org*.

[68] Todorović, D. (2014). How shape from contours affects shape from shading. *Vision Research*, 103, 1–10.

[69] Xiong, Y. (2015). Physics-based visual inference: Theory and applications.

[70] Xiong, Y., Chakrabarti, A., Basri, R., Gortler, S. J., Jacobs, D. W., & Zickler, T. (2015a). From shading to local shape. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 37(1), 67–79.

[71] Xiong, Y., Chakrabarti, A., Basri, R., Gortler, S. J., Jacobs, D. W., & Zickler, T. (2015b). From shading to local shape. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(1), 67–79.

[72] Yarotsky, D. (2017). Error bounds for approximations with deep relu networks. *Neural Networks*, 94, 103–114.

[73] Yuille, A. & Snow, D. (1997). Shape and albedo from multiple images using integrability. In *Computer Vision and Pattern Recognition (CVPR)* (pp. 158–164).

[74] Zhang, R., Tsai, P.-S., Cryer, J., & Shah, M. (1999). Shape-from-shading: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8), 690–706.

[75] Zhou, F., Wu, B., & Li, Z. (2018). Deep meta-learning: Learning to learn in the concept space. *CoRR*, abs/1802.03596.

[76] Zoph, B. & Le, Q. V. (2016). Neural architecture search with reinforcement learning. *arXiv preprint arXiv:1611.01578*.

[77] Zoran, D., Krishnan, D., Bento, J., & Freeman, B. (2014). Shape and illumination from shading using the generic viewpoint assumption. In *Advances in Neural Information Processing Systems* (pp. 226–234).