

**A GEOMETRIC VARIATIONAL APPROACH TO SHAPE INVERSION FOR
RADAR**

A Dissertation
Presented to
The Academic Faculty

By

Alper Yildirim

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology

May 2020

Copyright © Alper Yildirim 2020

A GEOMETRIC VARIATIONAL APPROACH TO SHAPE INVERSION FOR RADAR

Approved by:

Dr. Anthony J. Yezzi, Advisor
School of Electrical and Computer
Engineering
Georgia Institute of Technology

Dr. Christopher F. Barnes
School of Electrical and Computer
Engineering
Georgia Institute of Technology

Dr. Aaron Lanterman
School of Electrical Engineering
Georgia Institute of Technology

Dr. Patricio A. Vela
School of Computer Science
Georgia Institute of Technology

Dr. Thomas Orlando
School of Chemistry and Biochem-
istry
Georgia Institute of Technology

Date Approved: December 13, 2019

“The real voyage of discovery consists not in seeking new landscapes, but in having new eyes.”

Marcel Proust

I dedicate this thesis to,
*my mother **Firdes**,*
*my father **Ahmet**,*
*my sisters **Hilâl** and **Tuğçe**,*
*my brothers **Cengizhan** and **Metehan**,*
*and my niece **Umay**.*

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my advisor Anthony Yezzi for the endless guidance and support he provided during this hard and long journey. Having no prior exposure to radar literature and undertaking the challenge of making a rather unorthodox idea work, there were several moments where I had come to the conclusion that there was no way to crack this problem. Therefore, this work would not be possible without his insightful feedbacks and encouragement that kept me on trying every time I felt like giving up. I also would like to express my gratitude to the members of the Laboratory of Computational Computer Vision, Navdeep for helping me to get adapted to the lab when I first started and being a good friend and a supportive lab mate for the whole time and to Samuel, Robert, Albert, Avinash, Fareed, and all the former and current members of Laboratory of Computational Computer Vision.

I would like to thank my committee members, Chris Barnes, Aeron Lanterman, Patricio Vela and Thomas Orlando for serving on my dissertation committee.

I also would like to thank people which made me feel like home during my stay in Atlanta. I owe special thanks to Emre (Yilmaz), who has been a best friend to me since the beginning of this journey as, without him, Atlanta would not be half as good. The fun I had from our conversations, from the ordinary chit chat to the long discussions on science, math, art, history, literature and a bunch of other stuff, will always be memorable. I also want to thank Dogancan, who has been a very good friend since the beginning and who helped me a lot to get adapted to my new life here. Our tea sessions in front of the TSRB gave me the boost I needed everytime I got exhausted and also, our weekly Rio steakhouse visits as *three musketeers* was one of my all time favorite routines. I thank Abdurrahman for all the CS discussions, movie nights and best of all, for making the best *tarhana* soup. I also thank Erkam for all the coffee sessions, Sinan for picking me up from the airport when I first came to US and for the valuable friendship afterwards and Yusuf for the best camping

trip. I am indebted to Emre (Yigitoglu) and Caglar for being my housemates during my first year, helping me to get settled and also for numerous other things. I also would like to express my gratitude to other members of the Turkish community, including but not limited to Berkay, Bige, Can, Didem, Efe, Emre (Gursoy), Ezgi, Evren, Fatma, Gokcin, Melih, Nil, Selin. I also thank the members of the Motiwalla family, Shafi and Adria for their hospitality and giving me the opportunity to join their Sunday evening discussions where I got to meet with some amazing people.

I would like to thank my bro Cagkan who helped me about everything I could ask for and provided me endless moral support for the whole time despite to the fact that we were in different states. I am also indebted to Ammon family for their great hospitality and the warm welcome during my visits to Indiana.

Finally, I would like to express my graditute to my family whom I cannot thank enough, my mother Firdes and my father Ahmet for raising me in the best possible way and supporting me with my decision to come to US, my sisters Hilal and Tugce, my brothers Cengizhan and Metehan, my niece Umay who has become a great source of joy to our family the since day one.

TABLE OF CONTENTS

Acknowledgments	v
List of Tables	ix
List of Figures	x
Chapter 1: Introduction	1
Chapter 2: Physical Modelling	10
Chapter 3: Inversion	22
3.1 Choice of a waveform	22
3.2 Choice of the Cost Functional	24
3.2.1 First trial: Cost functional design using the time-domain representation	25
3.2.2 Second trial: Cost functional design using frequency-domain representation	28
3.2.3 Extraction of Electric Field Density Profile (over range)	32
Chapter 4: A 2D Discrete Shape Model	44
4.0.1 Forward Model	45
4.0.2 Inversion	47
4.0.3 Optimization	49

4.1	Results	53
4.1.1	Circular antenna placement.	55
4.1.2	Linear antenna placement.	57
4.1.3	Discussions	59
Chapter 5: 2D Level Set Based Parametrization		60
5.1	Level Set Methods	60
5.1.1	Narrow Band Methods	65
5.2	Our Approach	65
5.3	Inversion	66
5.3.1	Gradient Derivation	67
5.4	Results	77
5.4.1	Discussions	79
Chapter 6: Conclusions		81
Appendix A: Derivation of gradient flow		83
Appendix B: Derivation of Gradient Flow for Radar		85
References		90

LIST OF TABLES

4.1	Shape, Signal and RX/TX parameters	53
4.2	Three shape configurations we consider	54

LIST OF FIGURES

1.1	General structure of our shape estimation scheme. Φ denotes the parameter vector for a particular representation of the shape geometry. Signal attributes fed to the cost functional are any choice of information that can be extracted through a preprocessing.	9
2.1	Placement of the transmitter antenna with respect to the surface patch where $A_{x'}$ and $A_{y'}$ are size of aperture in the x' and y' dimensions, \mathbf{r}' is the vector connecting the aperture center to the surface patch, \mathbf{n} is surface unit normal and θ' is the angle between \mathbf{r}' and the aperture normal.	15
2.2	Placement of the receiver antenna with respect to the surface patch where A_x and A_y are size of aperture in the x and y dimensions, \mathbf{r} is the vector connecting the aperture center to the surface patch, \mathbf{n} is surface unit normal and θ is the angle between \mathbf{r} and the aperture normal.	16
2.3	Placement of the transmitter/receiver and the surface patch together.	20
3.1	Our subsampling of the received signal along the time instants	26
3.2	The trend of the cost functional value on a linear trajectory in the parameter space from initial shape model to the actual shape model. Horizontal axis denotes the point index at which the cost functional is computed. Image on the left is for a case where the wavelength is increase from 4mm to 25mm and for image on the right, it is increased from 3mm to 30mm.	28
3.3	Approximation of a linear chirp with a staircase frequency profile	30
3.4	Cost functional value as function of intermediate shapes (in the parameter space, on a linear trajectory) between the initial and the actual shape. Along the same row, distance between the initial and the actual shape increases. Distances are chosen to be the multiples (1,2,4) of the average wavelength (λ). Along the same column, number of the intermediate frequencies used in the cost function increases ($n = 40, 80, 160$)	31

3.5	Frequency spectrum of deramped signal for three scatterers lying at different range values. See how difference between the range of a scatterer and the predicted range value is mapped into frequency component. This allows us to obtain the decomposition of the total electric field strength along the range by looking at the frequency spectrum of deramped (stretch processed) signal.	34
3.6	Frequency (range) spectrum of deramped signal. It is given as a function of range value.	38
3.7	Frequency spectrum of deramped signal in 2D with its envelope $P_D(D)$. . .	39
4.1	Flow chart of our inversion algorithm.	45
4.2	Depiction of our forward model. Transmitter(TX) and Receiver(RX) are directional antennas. \mathbf{u}' and \mathbf{u} are unit ray directions and \mathbf{n} is the unit normal vector of a point on our shape. Domain of integration for i^{th} range is $S_i \cap \mathbf{e}$ that consists of two curve segments assuming both segments are fully visible to the antenna pair. ΔD is the difference of round trip distance values between the beginning and end of a range bin.	47
4.3	Curvature regularization for a polygonal object. We take the directions of consecutive edges of the polygon and penalize the norm of the difference between the directions of these two edges which is expected to canalize evolution towards smooth shapes.	49
4.4	Dependency graph showing how vertex coordinates are related to E_D . Subscript and superscripts of an edge (e) denote the vertices it connects to. Subscript and superscript of the electric field density values denote the range bin index and antenna index, respectively.	50
4.5	Possible placements of vertices with respect to a range bin when the vertex \mathbf{v}_k is included in the i^{th} range bin.	52
4.6	Antenna gain pattern illustrated on the left and as a polar plot on the left and the gain value as a function of angle shown on the right.	54
4.7	Shape evolution of Case-1 and Case-2 for a circular antenna pattern.	55
4.8	Shape evolution of Case-3 for a circular antenna pattern	56
4.9	Evolution plots of Case-1, Case-2, Case-3 for a circular antenna pattern . .	56
4.10	Shape evolution of Case-1 and Case-2 for a linear antenna pattern.	57

4.11	Shape evolution of Case-3 for a linear antenna pattern	58
4.12	Evolution plots of Case-1, Case-2, Case-3 for a linear antenna pattern . . .	58
5.1	Level set curves of a surface at different level values. Image is found online. <i>https://team.inria.fr/memphis/research/hierarchical-cartesian-schemes-for-pdes/level-set/</i>	61
5.2	With level-set methods, it is possible to start with an object on the left and reach to the one on the right.	64
5.3	A level set curve and the narrow band around the set. The level set function is updated only within the band boundaries.	66
5.4	Our weighting scheme as a function of range. Top shows the weighting used for the first range decomposition. The bottom shows the same for the second frequency spectrum. At the bottom, combined weight is shown. . . .	70
5.5	The depiction of how we can obtain our weighted average electric field density values from the radar signal. Our strategy is to obtain a dense range decomposition (bottom) from the preprocessing and then obtain the weighted range decomposition of our coarse weighted spectrum (top).	73
5.6	Our evolution results for a crescent shaped object. We initialize our shape model as an ellipse which eventually converges to our actual model. At the top, we see the snapshots of our evolving shape from initial shape to its final form. We obtained the convergence in 1500 iterations.	79
5.7	Our evolution results for a case where our initialization and the actual shape have different topology. Antennas are placed around the object on a circular pattern. We initialize our shape model as an ellipse which eventually splits up and converges to our actual model that consists of two different curves. At the top, we see the snapshots of our evolving shape from initial shape to its final form. We obtained the convergence in 2500 iterations.	80

SUMMARY

In this work, we develop a novel method for dense surface reconstruction of scenes using radar. For a given scene and a set of antennas looking towards to this scene, our method estimates the shape of the scene using the radar return signal. For this purpose, we use a deformable shape evolution approach which seeks to match the received signal to a computed forward model based on the evolving shape. Using such an approach comes with important advantages such as the ability to model the issues related to the object geometry which cannot be easily incorporated into the problem by the current radar based imaging techniques. As an example, we know that most scene surfaces have some level of smoothness. Exploiting such prior information can yield a more accurate estimation of the shape. It can also decrease the number of measurements required for an accurate estimation since the prior information limits the solution space to a subspace that favors surface smoothness. Another important geometric consideration is the self-occlusions present in the scene. We know that certain parts of the object surface are not visible for some antenna positions which can be very important to model, especially for close range applications in which the self-occlusions strongly change with the viewpoint. Iterations start with an initial shape which is gradually deformed until its image under the forward model gets sufficiently close to the actual measured signal. However, using an iterative inversion scheme for radar can be tricky as radar signals are highly oscillatory with respect to the surface shape which can introduce itself in the cost functional if cost function is not carefully designed. For this purpose, we employ the technique of stretch processing to extract geometric properties of the shape from radar return signal. This yields a smooth and purely geometric cost functional by which shape inversion can be robustly performed via gradient-based minimization algorithms. Employing such a cost functional, we test our approach on synthetic simulations where we use two different parametrizations. First, we use a polygonal shape model for our evolving shape where the set of parameters are chosen to be the vertices of the polyg-

onal model that are of finite dimension. Second, we use a level-set based approach where we have a continuous parametrization of the shape model. In this scenario, the shape is parameterized implicitly as a constant level-set of a function defined on the Cartesian grid. We obtain promising results using both cases where results show the promise of this type of an approach on some challenging scenarios.

CHAPTER 1

INTRODUCTION

Vision systems are becoming more and more essential in robotic systems because of the rich information they can provide about their environment. Especially for robots that are to navigate in cluttered environments, awareness of the scene structure is of great importance as it is usually the main limiting factor on robot motion. Inferring such structure using visual cues from camera images is a natural approach which mimics the way we sense the world with our eyes.

Shape sensing from is an established area of research in computer vision that is known as 3D reconstruction. This problem is studied under two main categories that are the bottom-up approaches and the top-down approaches. In bottom-up approaches, multiple images are used that are captured from different view points. Distinctive features are extracted from these images and feature matches are obtained after a cross-matching procedure. These feature matches are then used to jointly estimate the camera locations and a set of 3D point coordinates of the feature locations. Procedures can efficiently be done using the projective geometry tools where closed form expressions are available that yield one-shot solutions for the camera poses and the 3D feature coordinates. However, since these methods are usually sensitive to noises and calibration parameters, usually a nonlinear refinement procedure is performed over these initial results that yield more accurate results. Also, to obtain the scene geometry, a meshing can be performed over these sparse features. These methods also have some disadvantages. For the scenes where distinctive features are not abundant, it is not easy to obtain feature matches between images which makes the estimation process trickier. Another important consideration is that we cannot use the prior shape information we have about the scene with these methods which can be greatly benefited. These issues can be successfully addressed via using top-down approaches. In

these approaches, we start assuming a shape model, a reflectivity function attached to the shape and a camera model. Using all this information, we can generate the images for our assumed shape model which now can be compared to the actual images of the scene. Using the mismatch between the actual and the computed image, we can update our shape model aiming to decrease the discrepancy between two images. Running this procedure iteratively, we expect our shape model to converge to the actual scene. The main advantage of these approaches is that we obtain the scene geometry in a natural way (for bottom up approaches, usually a point cloud is obtained on which a meshing is needed to get geometry). Running the estimation using the shape model also comes with an opportunity of being able to introduce shape priors directly into the optimization problem. As a result, we can embed any prior geometric information available to us such as smoothness, continuity or the topology [1] [2].

Although camera based sensing methods are proven to be effective for robotics systems, these systems can be fragile to certain factors such as the presence of low ambient light or the obstructing factors for the visible light spectrum (rainy, foggy weather or smoke). As a result, alternative sensing modalities can often be required for the applications where these conditions are present. A good option to replace or complement the camera systems can be the radars. Radar systems are immune to many factors which challenge visual sensors as they have good penetration capabilities through certain mediums, air, water etc. [3] These systems are essential especially for airborne and spaceborne imaging applications where light rays can easily be blocked by the clouds or the thick layer of air between the antenna and ground scene. High resolution imaging can also be done with radar using an apparatus known as Synthetic aperture radar (SAR)[4][5][6][7]. A SAR system is usually composed of a small antenna or antennas attached to a moving platform which takes measurements of the scene from different viewpoints. These measurements are then used to synthesize a high resolution image of the scene. Using such an approach (taking measurements of the scene from different view points) is necessary since otherwise it would be impractical to

be able to get a high resolution in the along-track (azimuth) direction. For instance, for an X-band radar ($\lambda \approx 0.03m$) with an aperture length of $L = 1m$, beamwidth becomes:

$$\theta = \frac{\lambda}{L} = \frac{0.03}{1} = 0.03rad. \quad (1.1)$$

For an antenna scene distance of $r = 10000m$, resolution in the azimuth direction is obtained as:

$$\delta = \theta r = 300m \quad (1.2)$$

which means to be able to separate two scatters in a scene, distance between them should be greater or equal than δ . As a result, getting a reasonably high azimuth resolution for a usual scenario, either the wavelength should be impractically small or the antenna size should be very large. To overcome this limitation, Carl Wiley invented a new technique [8] that uses a coherent radar and the Doppler beam sharpening technique by which the achievable azimuth resolution was increased. In the following years, further developments were made and what we call the Synthetic aperture principle was invented [9]. Using this technique and certain algorithm classes, the achievable resolution for a side looking antenna is found as:

$$\delta = \frac{L}{2} \quad (1.3)$$

which is independent from the scene-antenna distance. Also, this equation yields a better resolution for a smaller antenna which may seem strange at first since the whole purpose of the synthetic aperture radar is to realize a very large antenna with a small one that is taking measurements at different viewpoints and synthesizing them. The reason why a small antenna gives a better resolution is because it has a larger beamwidth that covers a given scatterer for a larger angular dwell and as a result of this, the scatterer contributes to

a larger number of measurements which yields a better localization for the scatterer.

In the 50s and 60s, SAR systems were mostly considered for military applications where the purpose was to create images of man-made terrains and targets. In the 70s and 80s, civilian applications of SAR also started where SAR systems were developed to retrieve geological and bio-physical parameters of the earth surface. Use of the polarimetry for improved parameter retrieval [10], interferometry [11] to measure surface topography and differential interferometry to be able to detect surface displacements [12] were developed in 80s and 90s.

SAR has also been considered for 3D imaging. In this case, scene is modelled as a reflectivity function in 3D which is to be computed from measurements [13]. In [14], a true 3D tomographic formulation of spotlight mode SAR was developed which also led to a better interpretation of the layover phenomena that is caused by the out of plane scatterers in case of a 2D imaging. Besides the tomographic techniques, some near-field techniques was also developed for 3D SAR [15].

SAR has been proven to be a strong and informative tool for 2D and 3D imaging problems. Exploiting the use of frequency domain based techniques and low computational complexity of fast Fourier transform [16], SAR images can be synthesized in a surprisingly efficient manner. However, these methods lack in modelling certain kind of nonlinearities that can be present in imaging. For example, the well-known occlusion problem in computer vision is highly nonlinear and proven to be hard to deal with. When we look at a scene, our eyes capture the light rays reflected from the scene. Some of these rays may not be visible since they are blocked by other opaque objects or surfaces, or the object itself. When the point of view changes, set of scatterers in the scene also change and some of the previous parts of the scene we observe become invisible and some invisible parts become visible. Since characteristics of occlusion is a highly nonlinear function of the scene and the scene itself is to be estimated from the measurements, there is no easy way to incorporate this into SAR imaging. Although occlusion modelling may not be that critical in

the long range (spaceborne), the effect can be critical for close range applications where the visible parts of the object can drastically change with respect to the view-point. Another important consideration which is not available to the current SAR techniques is the notion of shape geometry since the raw output of these algorithms is a spatial reflectivity function defined on a scatterer grid. The imaging problem is to assign reflectivity values to each scatterer by using the radar measurements that are sampled from a moving antenna. Therefore, reflectivity value of each scatterer is estimated independently from others. Depending on the size of the cell, this approach comes with the phenomena called speckles which is caused by multiple scatterers in the same resolution cell that can constructively and destructively interfere with each other. Speckle manifests in the synthesized image as dark or bright spots.

We know that in a lot of real life applications, scenes have a certain level of surface regularity (continuity and smoothness) and these properties has been successfully exploited for computer vision problems with the use of the generative model based approaches that utilize a top-down approach for shape estimation. Considering the remarkable achievements of SAR in the last decades and the dramatic increase in the computational power available in today's world, one can naturally ask if these generative model based approaches can also be benefited in radar based shape sensing the way they have been in computer vision. **To this end, the main goal of this thesis is to develop a novel radar based shape sensing framework that adopts a generative model based shape estimation scheme. Since such approaches have been successfully used in computer vision, our main strategy will be based on transferring the available tools from computer vision to radar world which we think can be greatly benefited for a niche set of shape sensing problems in this domain.**

Our approach is based on utilizing a generative model through which radar measurements can be computed given an initial shape model, a reflectivity function of this model and antenna configuration (rotation, translation and a directional gain pattern). By using

such an approach, we will be able to naturally incorporate all kinds of geometric considerations such as occlusions, continuity, smoothness etc. into the estimation problem. The main structure of our estimation scheme will be adapted from stereo vision based shape estimation schemes where we will choose to modify or fully replace certain parts of these techniques when it becomes necessary. Although radar and camera systems are very different from one another in terms of how they work and the type of resolution they provide, there also exists a lot of similarities between them. To lay the bridge between computer vision and radar and understand which tools need to stay the same and which are to be adapted or replaced, it is important to understand both similarities and differences. A summary of the similarities can be listed as:

- Both can be used to detect and track objects or creating maps.
- Both acquire information through EM waves
- There is an inherent loss of information in both modalities.
- Recovering the information that is lost in the measurement process has to be recovered through multiple measurements acquired from different view points for both cameras and radar.

Differences can be listed as:

- Camera images are rich in directional (spatial) information. Radars signals are rich in temporal (or equivalently range) information.
- Raw measurement of the camera is the intensity of the light incident on the individual pixels (power). Considering a small exposure time, our measured quantity becomes time-invariant. Raw measurement of the radar is a complex sinusoid with time dependency.

- The governing equation that connects the scene shape to the image data for a pinhole camera is a simple division combined with an affine transformation (calibration matrix). For radar, the raw complex sinusoid measured from the receiver at any given instant is a function of all surface points (within the beam pattern) that needs to be computed from a 2D surface integral.
- In cameras, depth information of the 3D scene is lost due to the projection from 3D to 2D while information along the other spatial dimensions are preserved (directional information). Radar signals on the other hand, in its raw form, predicated on measuring the time delay information that can capture the range information. As a result, they cannot resolve two scatterers with same range value which lie at different directions unless combined with other measurements or a very small beamwidth antenna.

Considering the differences between two imaging modalities (cameras and radar), our first step is to replace the pinhole camera model with a simulation in which we implement the governing physical equations that compute the signal to be measured from receiver given a surface, its reflectivity function and the transmitted signal. As a result of this, we obtain the connection between the scene and the raw measurement induced by it which will be our source of geometric information. We derive the governing physical equations in Chapter 2.

A second modification to camera based shape reconstruction will be in the inversion domain. In the inversion part, we have a cost functional that measures the mismatch between the computed images and the actual ones. Computed images are the ones generated using the pinhole camera model (or a more complex camera model if necessary) that takes our evolving shape model and its reflectivity function as input. Actual images are on the other hand are the real measurements of the scene captured from different viewpoints. By defining the mismatch between these two sets of images using our cost functional and updating our shape model in the direction of decreasing mismatch at each step, we eventually expect our shape model to converge to the actual scene. Since this problem requires the

solution to a highly nonlinear optimization problem, we need to take certain considerations into account. For example, we previously mentioned that pixel intensity value is a measure of the average power incident on the particular camera array element. Assuming a smoothly changing reflectivity function on the surface with respect to the surface normal (a nonspecular reflectivity function such as Lambertian), we know that the intensity value of corresponding pixel smoothly changes with respect to the perturbations in the surface normal. The situation is similar for the perturbations in the location of the reflector. As a result, we can comfortably define our cost functional as a function of the pixel intensities since it is reasonable to assume that our cost functional will be reasonably well-behaving. By a well-behaving cost function, we mean that the cost functional is as local minima free as possible in the space of possible shape geometries. This is vital since we cannot have a closed form solution for shape estimation due to the cost functional being a highly nonlinear function of the shape geometry. As a result, we need an iterative scheme to update our shape model at each step where we are heavily dependent on the gradient of the cost functional. This factor should particularly be taken into consideration when designing a cost functional for a radar based shape estimation as raw radar measurements are by nature high frequency complex sinusoid signals where the high frequency oscillations can manifest themselves as local minima of the optimization problem. We will have a detailed discussion about how to develop a well behaving cost functional for radar based shape estimation through a proper choice of preprocessing in Chapter 3. In Chapter 4, we demonstrate the use of our framework for a 2D discrete parameter case. We also assume to have a 2D star-shaped objects which can be approximated by a polygonal shape. Finally in Chapter 5, we employ a level-set parametrization for our shape model which provides us the chance to address a much richer family of shapes when compared to the polygonal shape model. It should be noted that we demonstrate our approach for a 2D case merely because of its easiness of implementation where our framework is independent from the number of dimensions. General structure of the framework we will be using is depicted in

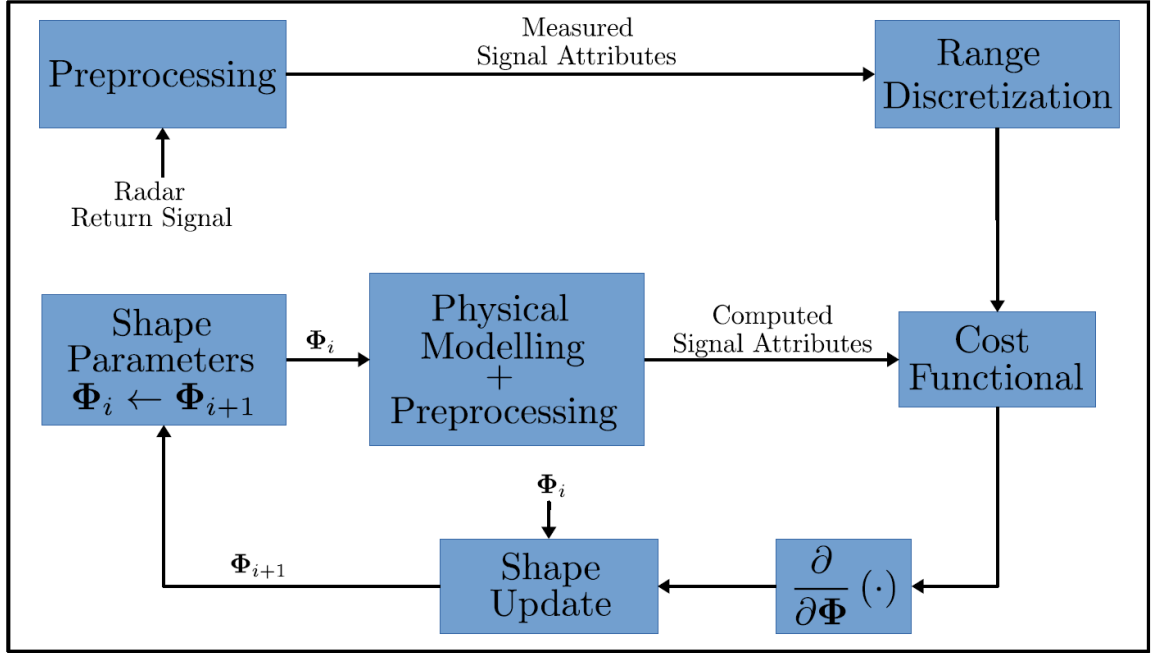


Figure 1.1: General structure of our shape estimation scheme. Φ denotes the parameter vector for a particular representation of the shape geometry. Signal attributes fed to the cost functional are any choice of information that can be extracted through a preprocessing.

Fig. 1.1.

CHAPTER 2

PHYSICAL MODELLING

Modelling of physical phenomena is required for understanding the relation between the scene shape and the received signal. To this end, we will develop a physical model by which we can generate measurements given a shape and antenna configuration. This will be important as our strategy for the inversion will be formulated on the mismatch between the actual measurement and the computed measurement that requires the simulation of the physical phenomena. We will call such simulation as our forward model.

We will derive our forward model at three steps which, when combined, yields the governing equations of the whole physical phenomenon. These steps are:

- Computation of the measurements of the wave field generated by the transmitter at a specified point on the object surface given its relative position with respect to the transmitter (TX) aperture .
- Modelling the interaction between the transmitted waveform and the object surface. This includes both how the transmitted signal interacting with the object surface and how the surface radiates back. We will use first Born approximation which means we won't be modelling multi path interactions.
- Computation of the signal induced by surface radiation on the receiver (TX) aperture.

This process can be modelled as a transfer of energy from the transmitter to the receiver which we will derive using radiometric principles. For simplicity, we will use the ray optics to model the wave propagation. Our model will be based on the following assumptions:

- Transmitter antenna consists of a rectangular surface where each point on the surface behaves as an infinitesimal ideal diffuse radiator. This means that the radiance (per-

ceived intensity) is constant along all directions in the upper hemisphere defined by the radiator surface.

- The scene behaves as an ideal diffuse reflector where the surface radiance is constant along all directions (Lambertian).

Our strategy is to derive the resultant electric field incident on the receiver aperture for a given transmitted signal.

Assuming each point on the transmitter aperture emits a power density of $1 \frac{Watt}{m^2}$, we can compute the radiance from conservation of energy. The relation between the power density and the radiance $L(\frac{Watt}{m^2 sr})$ (radiant flux emitted per solid angle per unit projected area) is given as:

$$1 \frac{Watt}{m^2} = \int_{\Omega_0} L^{TX}(\theta) \cos(\theta) d\Omega \quad (2.1)$$

where θ is the angle between the ray and the antenna normal and we integrate over the rays transmitted along all directions ($d\Omega$ is a solid angle measure and Ω_0 is the upper hemisphere defined by antenna normal). Expressing the integral over two angle parameters θ and ϕ), we have:

$$1 \frac{Watt}{m^2} = \int_0^{\pi/2} \int_0^{2\pi} L^{TX} \cos(\theta) \sin(\theta) d\phi d\theta \quad (2.2)$$

$$= \int_0^{\pi/2} \int_0^{2\pi} L^{TX} \frac{\sin 2\theta}{2} d\phi d\theta \quad (2.3)$$

$$= \int_0^{\pi/2} L^{TX} \frac{\sin 2\theta}{2} \int_0^{2\pi} d\phi d\theta \quad (2.4)$$

$$= \int_0^{\pi/2} L^{TX} \frac{\sin 2\theta}{2} (2\pi) d\theta \quad (2.5)$$

$$= \pi L^{TX} \left(\frac{-\cos(\pi)}{2} + \frac{\cos(0)}{2} \right) \quad (2.6)$$

$$= \pi L^{TX} \quad (2.7)$$

$$L^{TX} = \frac{1}{\pi}. \quad (2.8)$$

Using this, we can compute how much power an infinitesimal radiator emits in a certain direction. For a point in the space (this will be a point on the object surface) where the ray vector connecting the radiator to the surface is \mathbf{q}' , density of the average transmitted power is given as:

$$I_{dd}^{TX}(\mathbf{q}') = L^{TX} \frac{q'_{bz}}{\|\mathbf{q}_b'\|} \frac{1}{\|\mathbf{q}'\|^2} \quad (2.9)$$

$$\frac{q'_{bz}}{\pi\|\mathbf{q}_b'\|} \frac{1}{\|\mathbf{q}'\|^2} \quad (2.10)$$

where the term $\frac{q'_{bz}}{\|\mathbf{q}_b'\|}$ is the projected area of radiator in \mathbf{q}' direction where \mathbf{q}_b' is \mathbf{q}' with respect to the antenna frame and $\frac{1}{\|\mathbf{q}'\|^2}$ is the dilution of power with distance. As the unit of radiance L^{TX} is $\frac{W_{att}}{m^2 \cdot sr}$ and the unit of $\frac{1}{\|\mathbf{q}'\|^2}$ is $\frac{sr}{m^2}$, unit of I_{dd} becomes $\frac{W_{att}}{m^2 \cdot m^2}$ which is a double power density that is over both the scene surface and the antenna aperture. Electric field induced by a single ray can be computed through the relation between the electric field strength and the average power:

$$P = \frac{c\epsilon_0 E_0^2}{2} \quad (2.11)$$

where ϵ_0 is the permittivity of the free space and c is the speed of the light. In this case, electric field created by infinitesimal radiator is given as:

$$E = E_0 \exp \left[i2\pi f \left(t - \frac{\|\mathbf{q}'\|}{c} \right) \right] \quad (2.12)$$

where f is the frequency of our complex sinusoid. In our case, we have an infinitesimal radiator which creates an electric field density (E_d) for which the relation is given as:

$$I_{dd}^{TX}(\mathbf{q}') = \frac{c\epsilon_0 (E_{d0}^{TX})^2}{2} \quad (2.13)$$

Using this relation, the electric field density is given as:

$$E_d^{TX}(\mathbf{q}') = \sqrt{\frac{2I_{dd}^{TX}(\mathbf{q}')}{c\epsilon_0}} \exp \left[i2\pi f \left(t - \frac{\|\mathbf{q}'\|}{c} \right) \right] \quad (2.14)$$

$$= \left(\sqrt{\frac{2q'_z}{c\epsilon_0\pi\|\mathbf{q}'\|}} \right) \frac{1}{\|\mathbf{q}'\|} \exp \left[i2\pi f \left(t - \frac{\|\mathbf{q}'\|}{c} \right) \right]. \quad (2.15)$$

To compute the resultant electric field, we sum over all rays incident on our surface point (emanated from the antenna aperture) which is given as:

$$E = \int_{-A_y/2}^{A_y/2} \int_{-A_x/2}^{A_x/2} E_d^{TX}(\mathbf{q}') dx dy \quad (2.16)$$

$$= \int_{-A_y/2}^{A_y/2} \int_{-A_x/2}^{A_x/2} \left(\sqrt{\frac{2q'_z}{c\epsilon_0\pi\|\mathbf{q}'\|}} \right) \frac{1}{\|\mathbf{q}'\|} \exp \left[i2\pi f \left(t - \frac{\|\mathbf{q}'\|}{c} \right) \right] dx' dy'. \quad (2.17)$$

Since antenna aperture is small when compared to the antenna point distance, we will make $\|\mathbf{q}'\| \approx \|\mathbf{r}'\| = R'$ assumption for the terms outside the complex sinusoid where \mathbf{r}' is the ray emanates from the center of the aperture. For the phase term, to be able to get an analytical expression, we use Fraunhofer diffraction where a first-order approximation of $\|\mathbf{q}'\|$ is used. Our expression then becomes:

$$E = \int_{-A_y/2}^{A_y/2} \int_{-A_x/2}^{A_x/2} \left(\sqrt{\frac{2q'_z}{c\epsilon_0\pi\|\mathbf{q}'\|}} \right) \frac{1}{\|\mathbf{q}'\|} \exp \left[i2\pi f \left(t - \frac{\|\mathbf{q}'\|}{c} \right) \right] dx' dy' \quad (2.18)$$

$$= \left(\sqrt{\frac{2 \cos(\theta')}{c\epsilon_0\pi}} \frac{1}{R'} \right) \iint \exp \left[i2\pi f \left(t - \frac{R' - x'u'_{bx} - y'u'_{by}}{c} \right) \right] dx' dy' \quad (2.19)$$

$$= \underbrace{\left(\sqrt{\frac{2 \cos(\theta')}{c\epsilon_0\pi}} \frac{1}{R'} \right) \exp \left[i2\pi f \left(t - \frac{R'}{c} \right) \right]}_{K'} \iint \exp \left[i2\pi f \left(\frac{x'u'_{bx} + y'u'_{by}}{c} \right) \right] dx' dy' \quad (2.20)$$

we have $\mathbf{u}' = \mathbf{r}'/\|\mathbf{r}'\|$, $\cos(\theta') = u_{bz}$ and $\mathbf{u}'_b = [u'_{bx}, u'_{by}, u'_{bz}]$ where \mathbf{u}'_b is the unit ray direction of \mathbf{r}' with respect to the antenna frame. Expressing \mathbf{u}'_b in terms of angular coor-

dinates:

$$= K' \iint \exp \left[i2\pi f \left(\frac{x' \sin \theta' \cos \phi' + y' \sin \theta' \sin \phi'}{c} \right) \right] dx' dy' \quad (2.21)$$

$$= K' \iint \exp \left(i2\pi f \left(\frac{\sin \theta' (x' \cos \phi' + y' \sin \phi')}{c} \right) \right) dx' dy' \quad (2.22)$$

$$= K' \iint \exp \left(\frac{i2\pi f x' \sin \theta' \cos \phi'}{c} \right) \exp \left(\frac{i2\pi f y' \sin \theta' \sin \phi'}{c} \right) dx' dy' \quad (2.23)$$

$$= K' \int \exp \left(\frac{i2\pi f x' \sin \theta' \cos \phi'}{c} \right) dx' \int \exp \left(\frac{i2\pi f y' \sin \theta' \sin \phi'}{c} \right) dy' \quad (2.24)$$

$$= K' \left(\frac{\exp \left(\frac{i2\pi f x' \sin \theta' \cos \phi'}{c} \right)}{\frac{i2\pi f \sin \theta' \cos \phi'}{c}} \right) \Big|_{\frac{-A_{x'}}{2}}^{\frac{A_{x'}}{2}} \left(\frac{\exp \left(\frac{i2\pi f y' \sin \theta' \sin \phi'}{c} \right)}{\frac{i2\pi \sin \theta' \sin \phi'}{c}} \right) \Big|_{\frac{-A_{y'}}{2}}^{\frac{A_{y'}}{2}} \quad (2.25)$$

$$= K' A_{x'} A_{y'} \operatorname{sinc} \left(\underbrace{\frac{f A_{x'} \sin \theta' \cos \phi'}{c}}_{Q'_x} \right) \operatorname{sinc} \left(\underbrace{\frac{f A_{y'} \sin \theta' \sin \phi'}{c}}_{Q'_y} \right) \quad (2.26)$$

$$= \left(\sqrt{\frac{2 \cos(\theta')}{c \epsilon_0 \pi}} \frac{1}{R'} \right) \exp \left(t - \frac{R'}{c} \right) A_{x'} A_{y'} \operatorname{sinc}(Q'_x) \operatorname{sinc}(Q'_y) \quad (2.27)$$

$$= \underbrace{A_{x'} A_{y'} \left(\sqrt{\frac{2 \cos(\theta')}{c \epsilon_0 \pi}} \right) \operatorname{sinc}(Q'_x) \operatorname{sinc}(Q'_y) \frac{1}{R'}}_{E_0} \exp \left(t - \frac{R'}{c} \right). \quad (2.28)$$

We use the resultant electric field to obtain the incident power on an infinitesimal surface patch (with a unit normal of \mathbf{n}) in \mathbf{u}' direction. Transmitter and surface patch configuration is depicted in Fig. 2.1. Using the relation between the electric field and average power,

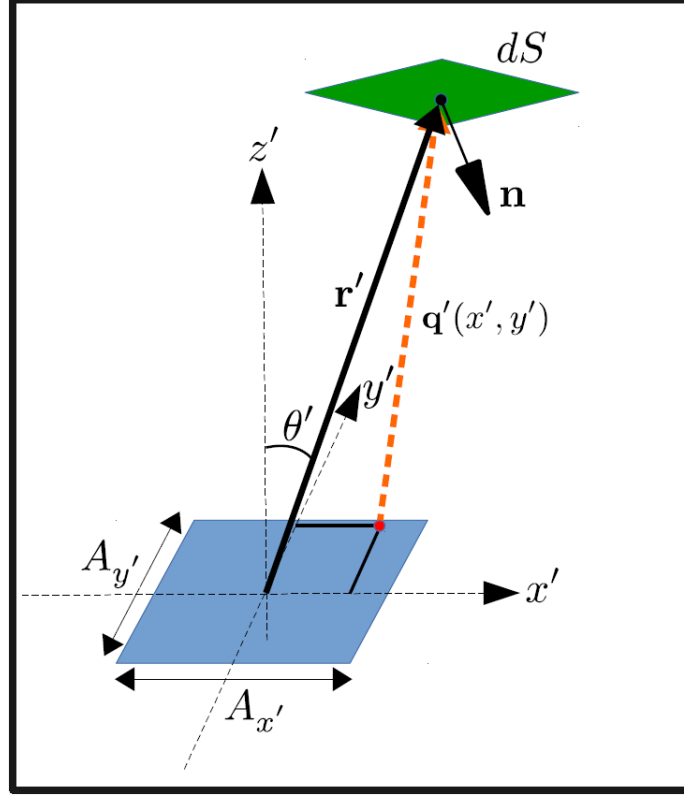


Figure 2.1: Placement of the transmitter antenna with respect to the surface patch where $A_{x'}$ and $A_{y'}$ are size of aperture in the x' and y' dimensions, \mathbf{r}' is the vector connecting the aperture center to the surface patch, \mathbf{n} is surface unit normal and θ' is the angle between \mathbf{r}' and the aperture normal.

irradiance incident on our point (infinitesimal surface patch) becomes:

$$P_{dn}(\mathbf{u}') = \frac{c\epsilon_0 \left[A_{x'} A_{y'} \left(\sqrt{\frac{2 \cos(\theta')}{c\epsilon_0 \pi}} \right) \text{sinc}(Q'_x) \text{sinc}\left(Q'_y \frac{1}{R'}\right) \right]^2}{2} (-\mathbf{u}' \cdot \mathbf{n}) \quad (2.29)$$

$$= \frac{c\epsilon_0 A_x'^2 A_y'^2 2 \cos(\theta') \text{sinc}^2(Q'_x) \text{sinc}^2(Q'_y)}{2c\epsilon_0 \pi R'^2} (-\mathbf{u}' \cdot \mathbf{n}) \quad (2.30)$$

$$= \frac{1}{\pi R'^2} \left(\underbrace{A'_x A'_y \sqrt{\cos(\theta')} \text{sinc}(Q'_x) \text{sinc}(Q'_y)}_{G'} \right)^2 (-\mathbf{u}' \cdot \mathbf{n}) \quad (2.31)$$

$$= \frac{1}{\pi} \frac{G'^2}{R'^2} (-\mathbf{u}' \cdot \mathbf{n}). \quad (2.32)$$

where we have a scale of $(-\mathbf{u}' \cdot \mathbf{n})$ from the fact that \mathbf{u}' is not perpendicular to our surface patch (point). Therefore unit solid angle (with respect to antenna aperture that is now

modelled as a point) covers a larger area on the surface that dilutes the power by the cosine between the ray and the surface normal. Assuming no power loss on our surface patch, we can assume the incident power will be fully radiated back. In our case, we assume our scene surface to be an ideal diffuse reflector in which case the radiance is constant along all directions. From the conservation of energy (as in Eq.2.2) we compute the surface radiance as:

$$L_s = \frac{1}{\pi} (P_{dn}(\mathbf{u}')) \quad (2.33)$$

$$= \frac{1}{\pi^2} \frac{G'^2}{R'^2} (-\mathbf{u}' \cdot \mathbf{n}) . \quad (2.34)$$

Using the surface radiance, we compute how much irradiance is incident on our receiver aperture created by our infinitesimal surface patch. The surface patch-receiver configuration is depicted in Fig. 2.2. Note that unit of irradiance is $Watt/m^2$ where contribu-

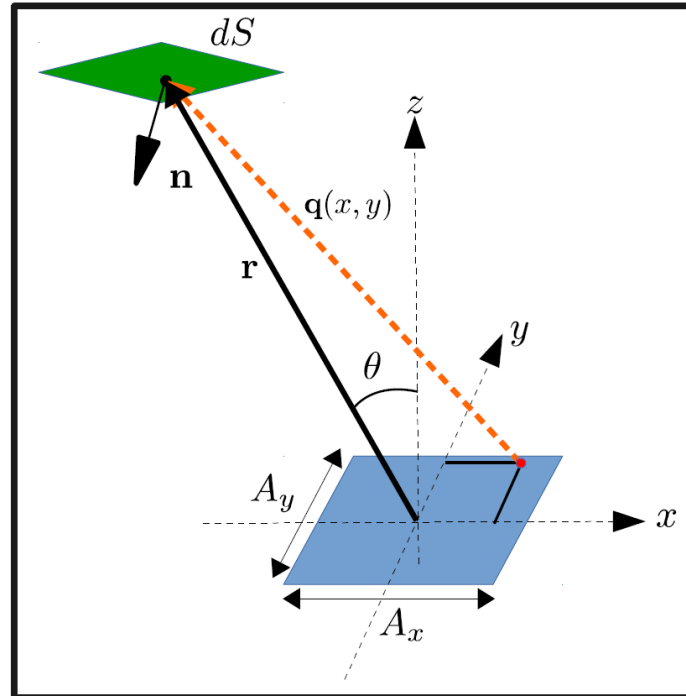


Figure 2.2: Placement of the receiver antenna with respect to the surface patch where A_x and A_y are size of aperture in the x and y dimensions, \mathbf{r} is the vector connecting the aperture center to the surface patch, \mathbf{n} is surface unit normal and θ is the angle between \mathbf{r} and the aperture normal.

tion provided by an infinitesimal surface needs to be a density quantity that has a unit of $Watt/m^2 \cdot m^2$ since total irradiance is the integration of this density over the scene surface. Receiver antenna has a rectangular aperture where the antenna-surface distance is a function of the aperture coordinates that creates a constructive/destructive interference on the aperture. For a ray (\mathbf{q}) connecting a point on the receiver aperture to the surface patch, radiated power (from the surface) is given as:

$$I_{dd}^S(\mathbf{q}) = L_s \frac{(-\mathbf{q} \cdot \mathbf{n})}{\|\mathbf{q}\|} \frac{1}{\|\mathbf{q}\|^2} \frac{q_{bz}}{\|\mathbf{q}\|} \quad (2.35)$$

$$= \frac{G'^2}{\pi^2 R'^2} \frac{(-\mathbf{u}' \cdot \mathbf{n})}{\|\mathbf{q}\|} \frac{(-\mathbf{q} \cdot \mathbf{n})}{\|\mathbf{q}\|} \frac{1}{\|\mathbf{q}\|^2} \frac{q_{bz}}{\|\mathbf{q}_b\|} \quad (2.36)$$

where we multiply the radiance by a cosine term $\frac{(-\mathbf{q} \cdot \mathbf{n})}{\|\mathbf{q}\|}$ that comes from cosine power law (so that perceived power is invariant with respect to the direction of measurement), a power decay factor $\frac{1}{\|\mathbf{q}\|^2} \left[\frac{sr}{m^2} \right]$ (radiance is defined as power emitted per unit solid angle per unit projected area) and another cosine factor $\frac{q_{bz}}{\|\mathbf{q}_b\|}$ diluting the power as the ray is not perpendicular to antenna aperture so power is distributed to a larger area of $\frac{dxdy}{q_{bz}/\|\mathbf{q}_b\|}$ where $\mathbf{q}_b = [q_{bx}, q_{by}, q_{bz}]$ is the ray \mathbf{q} with respect to the receiver frame. Amplitude of the electric field density (a double density since it is a density over both the scene surface and the receiver aperture) induced by a ray is given as (from Eq. 2.11):

$$E_{dd0}^{RX}(\mathbf{q}) = \sqrt{\frac{2I_{dd}^S(\mathbf{q})}{c\epsilon_0}} \quad (2.37)$$

$$= \sqrt{\frac{2}{c\epsilon_0} L_s \frac{(-\mathbf{q} \cdot \mathbf{n})}{\|\mathbf{q}\|} \frac{1}{\|\mathbf{q}\|^2} \frac{q_{bz}}{\|\mathbf{q}_b\|}} = \sqrt{\frac{2}{c\epsilon_0} \frac{G'^2}{\pi^2 R'^2} \frac{(-\mathbf{u}' \cdot \mathbf{n})}{\|\mathbf{q}\|} \frac{(-\mathbf{q} \cdot \mathbf{n})}{\|\mathbf{q}\|} \frac{1}{\|\mathbf{q}\|^2} \frac{q_{bz}}{\|\mathbf{q}_b\|}}. \quad (2.38)$$

Electric field then becomes:

$$E_{dd}^{RX}(t, \mathbf{q}) = E_{dd0}^{RX}(\mathbf{q}) \exp \left[i2\pi f \left(t - \frac{R' + \|\mathbf{q}\|}{c} \right) \right] \quad (2.39)$$

$$= \sqrt{\frac{2}{c\epsilon_0} \frac{G'^2}{\pi^2 R'^2} \frac{(-\mathbf{u}' \cdot \mathbf{n})}{\|\mathbf{q}\|} \frac{(-\mathbf{q} \cdot \mathbf{n})}{\|\mathbf{q}\|^2} \frac{q_{bz}}{\|\mathbf{q}_b\|}} \exp \left[i2\pi f \left(t - \frac{R' + \|\mathbf{q}\|}{c} \right) \right] \quad (2.40)$$

$$= \underbrace{\sqrt{\frac{2G'^2}{c\epsilon_0\pi^2 R'^2} \exp \left[i2\pi f \left(t - \frac{R'}{c} \right) \right]}}_C \sqrt{\frac{(-\mathbf{q} \cdot \mathbf{n})}{\|\mathbf{q}\|} \frac{q_{bz}}{\|\mathbf{q}_b\|} \frac{1}{\|\mathbf{q}\|}} \exp \left[i2\pi f \left(-\frac{\|\mathbf{q}\|}{c} \right) \right] \quad (2.41)$$

$$= C \sqrt{\frac{(-\mathbf{q} \cdot \mathbf{n})}{\|\mathbf{q}\|} \frac{q_{bz}}{\|\mathbf{q}_b\|} \frac{1}{\|\mathbf{q}\|}} \exp \left[i2\pi f \left(-\frac{\|\mathbf{q}\|}{c} \right) \right]. \quad (2.42)$$

To compute the resultant electric field density (single density) on the receiver aperture induced by our infinitesimal surface patch, we sum the electric field density contributions of all rays emanate from the surface patch that are incident on the receiver aperture. As a result, resultant electric field density is given as:

$$E_d^{RX}(t) = \int_{-A_y/2}^{A_y/2} \int_{-A_x/2}^{A_x/2} C \sqrt{\frac{(-\mathbf{q} \cdot \mathbf{n})}{\|\mathbf{q}\|} \frac{q_{bz}}{\|\mathbf{q}_b\|} \frac{1}{\|\mathbf{q}\|}} \exp \left[i2\pi f \left(-\frac{\|\mathbf{q}\|}{c} \right) \right] dx dy. \quad (2.43)$$

Assuming aperture size is small compared to surface-receiver distance, we again use Fraunhofer diffraction to compute a closed form expression where the receiver can be modelled as a point antenna with directional gain value. We use the approximations $\|\mathbf{q}\| \approx \|\mathbf{r}\| = R$, $\mathbf{u} = \mathbf{r}/R$, $\cos(\theta) = u_{bz}$ for non phase terms where $\mathbf{u}_b = [u_{bx}, u_{by}, u_{bz}]$ and \mathbf{u}_b is the unit ray direction with respect to the receiver frame. For the phase terms, we again use a first order approximation of $\|\mathbf{q}\| = R - x u_{bx} - y u_{by}$ that yields:

$$E_d^{RX}(t) = \int_{-A_y/2}^{A_y/2} \int_{-A_x/2}^{A_x/2} C \sqrt{(-\mathbf{u} \cdot \mathbf{n}) \cos(\theta)} \frac{1}{R} \exp \left[i2\pi f \left(-\frac{R - x u_{bx} - y u_{by}}{c} \right) \right] dx dy. \quad (2.44)$$

Grouping the constant terms as:

$$K = C \sqrt{(-\mathbf{u} \cdot \mathbf{n}) \cos(\theta)} \frac{1}{R} \exp \left[i2\pi f \left(-\frac{R}{c} \right) \right] \quad (2.45)$$

and expressing \mathbf{u}_b in terms of angular coordinates we have:

$$E_d^{RX}(t) = K \int_{-A_y/2}^{A_y/2} \int_{-A_x/2}^{A_x/2} \exp \left[i2\pi f \frac{xu_{bx} + yu_{by}}{c} \right] dx dy \quad (2.46)$$

$$= K \int_{-A_y/2}^{A_y/2} \int_{-A_x/2}^{A_x/2} \exp \left[i2\pi f \left(\frac{x \sin \theta \cos \phi + y \sin \theta \sin \phi}{c} \right) \right] dx dy \quad (2.47)$$

$$= K \int_{-A_y/2}^{A_y/2} \int_{-A_x/2}^{A_x/2} \exp \left(i2\pi f \left(\frac{\sin \theta (x \cos \phi + y \sin \phi)}{c} \right) \right) dx dy \quad (2.48)$$

$$= K \iint \exp \left(\frac{i2\pi f x \sin \theta \cos \phi}{c} \right) \exp \left(\frac{i2\pi f y \sin \theta \sin \phi}{c} \right) dx dy \quad (2.49)$$

$$= K \int \exp \left(\frac{i2\pi f x \sin \theta \cos \phi}{c} \right) dx \int \exp \left(\frac{i2\pi f y \sin \theta \sin \phi}{c} \right) dy \quad (2.50)$$

$$= K \left(\exp \left(\frac{i2\pi f x \sin \theta \cos \phi}{c} \right) \right) \Big|_{-\frac{A_x}{2}}^{\frac{A_x}{2}} \left(\exp \left(\frac{i2\pi f y \sin \theta \sin \phi}{c} \right) \right) \Big|_{-\frac{A_y}{2}}^{\frac{A_y}{2}} \quad (2.51)$$

$$= K A_x A_y \operatorname{sinc} \left(\underbrace{\frac{f A_x \sin \theta \cos \phi}{c}}_{Q_x} \right) \operatorname{sinc} \left(\underbrace{\frac{f A_y \sin \theta \sin \phi}{c}}_{Q_y} \right) \quad (2.52)$$

$$= C \sqrt{(-\mathbf{u} \cdot \mathbf{n}) \cos(\theta)} \frac{1}{R} \exp \left[i2\pi f \left(-\frac{R}{c} \right) \right] A_x A_y \operatorname{sinc}(Q_x) \operatorname{sinc}(Q_y) \quad (2.53)$$

$$= C \underbrace{A_x A_y \operatorname{sinc}(Q_x) \operatorname{sinc}(Q_y)}_G \sqrt{\cos(\theta)} \sqrt{(-\mathbf{u} \cdot \mathbf{n})} \frac{1}{R} \exp \left[i2\pi f \left(-\frac{R}{c} \right) \right] \quad (2.54)$$

$$= C G \sqrt{(-\mathbf{u} \cdot \mathbf{n})} \frac{1}{R} \exp \left[i2\pi f \left(-\frac{R}{c} \right) \right] \quad (2.55)$$

$$= \sqrt{\frac{2G'^2 (-\mathbf{u}' \cdot \mathbf{n})}{c\epsilon_0 \pi^2 R'^2}} \exp \left[i2\pi f \left(t - \frac{R'}{c} \right) \right] G \sqrt{(-\mathbf{u} \cdot \mathbf{n})} \frac{1}{R} \exp \left[i2\pi f \left(-\frac{R}{c} \right) \right] \quad (2.56)$$

$$= \sqrt{\frac{2G'^2 G^2 (-\mathbf{u}' \cdot \mathbf{n}) (-\mathbf{u} \cdot \mathbf{n})}{c\epsilon_0 \pi^2 R'^2 R^2}} \exp \left[i2\pi f \left(t - \frac{R' + R}{c} \right) \right] \quad (2.57)$$

$$= \sqrt{\frac{2}{c\epsilon_0 \pi^2}} \left[\frac{G' G \sqrt{(-\mathbf{u}' \cdot \mathbf{n})} \sqrt{(-\mathbf{u} \cdot \mathbf{n})}}{R' R} \right] \exp \left[i2\pi f \left(t - \frac{R' + R}{c} \right) \right] \left[\frac{W}{m^2} \right] \quad (2.58)$$

where this result is the electric field density induced on receiver aperture by an infinitesimal surface patch where transmitter, surface patch and the receiver is depicted in Fig. 2.3.

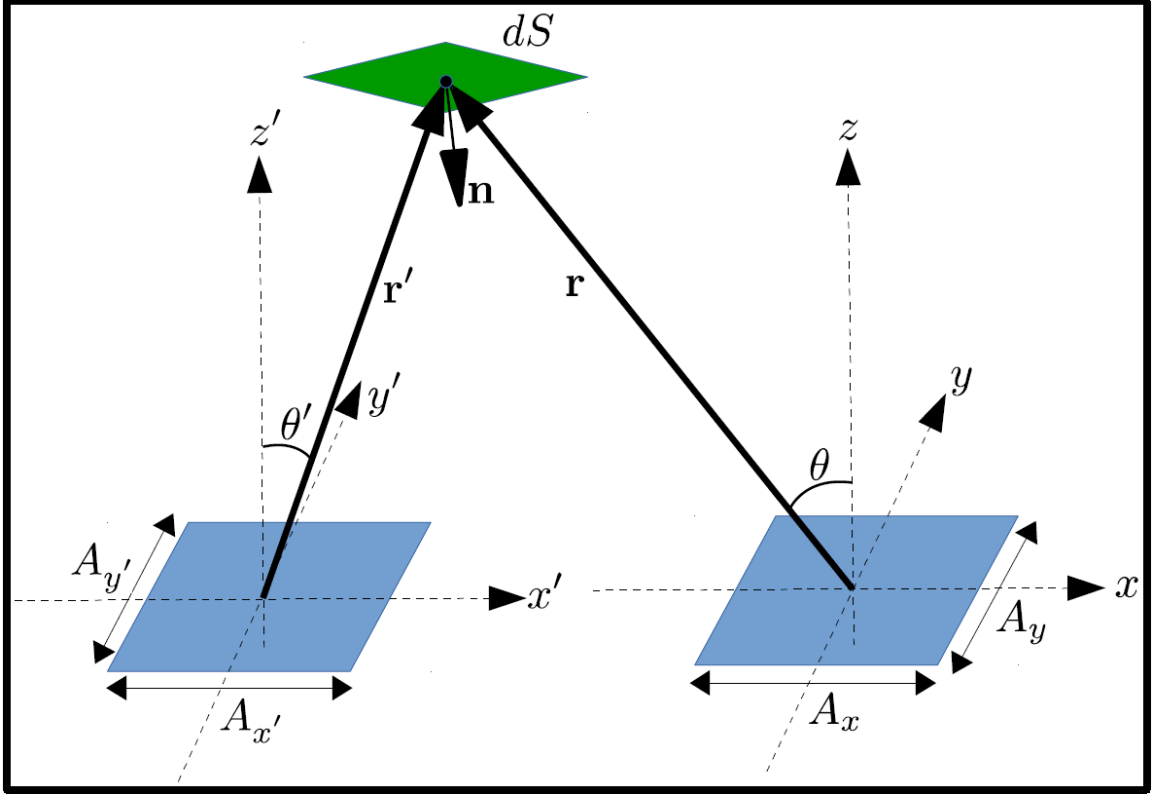


Figure 2.3: Placement of the transmitter/receiver and the surface patch together.

Computing the resultant electric field requires integration of this expression over visible scene surface which is given as:

$$E^{RX}(t) = \int_{S_0} E_d^{RX} dS \quad (2.59)$$

$$= \int_{S_0} \sqrt{\frac{2}{c\epsilon_0\pi^2}} \left[\frac{G'G\sqrt{(-\mathbf{u}' \cdot \mathbf{n})}\sqrt{(-\mathbf{u} \cdot \mathbf{n})}}{R'R} \right] \exp \left[i2\pi f \left(t - \frac{R' + R}{c} \right) \right] dS \quad (2.60)$$

$$= \sqrt{\frac{2}{c\epsilon_0\pi^2}} \int_{S_0} \left[\frac{G'G\sqrt{(-\mathbf{u}' \cdot \mathbf{n})}\sqrt{(-\mathbf{u} \cdot \mathbf{n})}}{R'R} \right] \exp \left[i2\pi f \left(t - \frac{R' + R}{c} \right) \right] dS. \quad (2.61)$$

Since for the scope of this thesis, we will only present simulation results where our mea-

surements will also be computed from our forward model, we can safely ignore the constant term that multiplies the integral as such a factor does not have any effect in the shape evolution due to the homogeneity. As a result, the final expression for our forward model is given as:

$$E^{RX}(t) = \int_{S_0} \left[\frac{G'G\sqrt{(-\mathbf{u}' \cdot \mathbf{n})}\sqrt{(-\mathbf{u} \cdot \mathbf{n})}}{R'R} \right] \exp \left[i2\pi f \left(t - \frac{R' + R}{c} \right) \right] dS. \quad (2.62)$$

It should be noted that this equations are derived assuming a constant frequency sinusoid whereas our framework assumes an LFM (linear frequency modulated) signal in which case the gain terms (G and G' becomes time dependent). As our simulations will be conducted for a very high carrier frequency of ($79GHz$) and a relatively small bandwidth ($4GHz$) compared to the it, we will assume these terms as constant.

CHAPTER 3

INVERSION

Our forward model (physical modelling) is an integral over the scene surface. It yields a highly oscillatory time dependent signal. Since our formulation uses a deformable model that is to be evolved with iterations, we substitute our current estimate of the shape into the forward model at each iteration and compute what would be the received signal for this estimate. This shape is then evolved in such a way that its image under the forward model gets closer to the actual measured signal in the next iteration. For this purpose, we need to define a cost functional by which we can score the mismatch between the actual received signal and the signal we compute from the forward model. To this end, we will have two main design steps for our inversion that are:

- First is the choice/design of a waveform that can provide the maximum amount of geometric information. It should be noted that the hardware capabilities need to be considered at this point as it might not be possible to generate every possible waveform.
- Second is the design of a cost functional that measures the mismatch between two given signal. As we will see, this will play the key role to a successful estimation.

3.1 Choice of a waveform

It is well known that waveform of the transmitted signal is a key factor for radar detection/tracking/estimation problems since resolution of radar system heavily depends on it. The simplest possible choice for a radar application could be a constant frequency sinusoid. Using the forward model we derived in the previous chapter, we can have a sense of how much information we can gather from using a constant frequency sinusoid. For a constant

frequency sinusoid, received signal is given as:

$$E^{RX}(t) = \int_{S_0} \frac{G'G\sqrt{(-\mathbf{u}' \cdot \mathbf{n})}\sqrt{(-\mathbf{u} \cdot \mathbf{n})}}{R'R} \exp \left[i2\pi f \left(t - \frac{R' + R}{c} \right) \right] dS. \quad (3.1)$$

All of terms in the integral is a function of the integration variable S , except the time parameter t . Taking it outside the integral, we have:

$$E^{RX}(t) = \exp(i2\pi ft) \overbrace{\int_{S_0} \frac{G'G\sqrt{(-\mathbf{u}' \cdot \mathbf{n})}\sqrt{(-\mathbf{u} \cdot \mathbf{n})}}{R'R} \exp \left(-i2\pi f \frac{R' + R}{c} \right) dS}^{\text{Shape information is embedded in a single complex factor}}. \quad (3.2)$$

As a result, our received signal becomes another constant frequency sinusoid that is our transmitted signal multiplied with a complex number. This implies that all of the geometric information is encoded in a complex number which obviously makes it impossible to be able to recover the scene geometry from such measurement. This is analogous to the LTI (Linear, time invariant) systems where the transmitted signal can be thought as the input, the scene geometry is the system model and the output is the received signal. One way to understand the dynamic characteristic of an LTI system is to feed the system with different frequency sinusoids and measure the steady-state response at the output where the output/input amplitude ratio (modulus) and the amount of the phase delay values (argument) are recorded. Bode plots are then created by using this recorded data by which the locations of the poles and zeros of the system can be identified (except for some edge cases such as pole/zero cancellation). The key take away is that to identify the underlying system, frequency response to multiple frequencies is a necessity.

Using multiple frequencies can also be adopted for radar where we transmit multiple constant frequency sinusoids and listening to the responses to each frequency. The more the number of frequencies are, the more our information about the scene geometry becomes for which stepped frequency radars [17] can be employed. However, FMCW (Frequency-modulated continuous wave) radars can also be considered for this purpose since the fre-

quency spectrum of such radar signals consists of multiple frequencies in a certain bandwidth. As a result, we can collect the frequency response to multiple frequencies by using a single pulse. It should be noted that in such case, we implicitly use the fact that for each frequency, frequency response to that individual frequency has also the same frequency. This theoretically makes it possible to decompose the received signal into the frequency responses of individual frequencies. We should also note that, in practice, we are bound to use the pulsed signals which limits our ability to resolve the frequency responses of close frequency values (the longer the pulse is, the better the resolving power gets between two close frequencies). Among the FMCW radars, linear frequency-modulated (LFM) radars are very popular and off-the-shelf solutions (mm-wave radars) are also available for such systems. They also yield received signals that are rich in information as we previously mentioned (frequency response to multiple frequencies). Therefore, LFM signals will be our choice for the waveform design. A pulsed LFM signal is mathematically given as:

$$f_0(t) = A\Pi\left(\frac{t-t_0}{\tau_0}\right) \exp\left[i2\pi \overbrace{(f_c + \alpha(t-t_0))}^{\text{instantaneous frequency}}(t-t_0)\right] \quad (3.3)$$

where t_0 is time center of pulse, τ_0 is the pulse length, α is the slope of the time-frequency curve, A is the amplitude of the pulse, f_c is the carrier frequency and $\Pi(\cdot)$ is the standard rectangular window function.

3.2 Choice of the Cost Functional

The second design step of our inversion process is the design of the cost functional that measures the mismatch between two signal that are the actual and the computed return signal. By measuring the distance between these two signals, we decide how our shape model should be updated so that the mismatch becomes smaller in the next iteration. As a result, the design of the cost functional becomes the most crucial component of our inversion scheme since our decision on how to update our shape model is based on how the cost func-

tional value is changing with respect to the shape model. In the stereo vision case, forward model (camera model) generates images using shape model (shape and the reflectivity) of the scene. Generated images are then compared to the actual images of the scene using the cost functional and shape model is updated accordingly (in the direction of decreasing cost functional value). For camera systems, this is relatively straightforward. We can formulate our cost functional directly in terms of the raw image data that are the pixel brightness values. With the help of shape priors and/or other regularizers, initial shape model can be successfully evolved into the actual shape of the scene through iterations. Since such an approach is effective in stereo vision systems, one can think such approach should also be applicable to a radar based shape estimation. In the next two sections, we will discuss our two failed attempts and in the third section, we will discuss our last attempt that has become successful. We think it is important to understand the failed attempts as the method we will follow in the third section is significantly motivated by the lessons we learn from these failures and how to avoid the causal agents contributing to them.

3.2.1 First trial: Cost functional design using the time-domain representation

In its rawest form, received signal of a radar system is a sampled version of the received signal. To make sure we have the perfect representation of the received signal and there is no data loss, our sampling rate needs to be large enough. Depending on how big the bandwidth of the transmitted signal is, number of the samples can be very large. As an example, for a mm-wave radar with a bandwidth of 4GHz that operates between 77GHz-81GHz, we can move the center frequency to the origin by performing a demodulation procedure. To be able to have a fully represent the signal as a discrete set of values, we still need to sample the it with a minimum sampling frequency of 4GHz ¹ Assuming a signal

¹Author is aware of the fact that such a high sampling rate is not typical if not impossible and a possible solution would require expensive hardware. For the sake of our argument here, we assume this is not an issue.

duration of $100\mu m$, number of samples we have become:

$$N_s = (4 \times 10^9 Hz)(1 \times 10^{-4} sec) \quad (3.4)$$

$$= 4 \times 10^5. \quad (3.5)$$

We have 4×10^5 samples recorded from the receiver. We will also need this many samples where we need to compute our that are computed from our forward model using our evolving shape model. This requires computing the value of Eq. 2.62 for 4×10^5 that is 2D surface integral. As a result, it is not a computationally feasible operation for our case (although it might be feasible for small bandwidth applications). Although, using all time samples are impractical, this problem can be overcome by formulating the cost functional in terms the subsamples of the received signal. To this end, we propose to design our cost functional in terms of equally spaced slices of time intervals sampled from the received signal. Our subsampling is depicted in Fig 3.1. Our motivation for such sampling has two

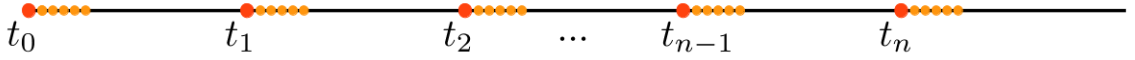


Figure 3.1: Our subsampling of the received signal along the time instants

reasons:

- We take equally spaced intervals of time slices that corresponds to different frequency values since we use an LFM pulse as the transmitted signal. Assuming the size of a time interval is small when compared to slice-to-slice distance, this can be interpreted as having a constant frequency within each interval where each time interval itself is responsible of capturing the frequency response of the shape geometry to a different frequency.
- Within each interval, we still sample the signal above the Nyquist rate as to prevent aliasing.

Our cost functional is given as:

$$E(\Phi) = \sum_{i=0}^{n-1} \sum_{j=0}^{k-1} (|E^{RX}(t_i + j\Delta t, \Phi) - E^{RXm}(t_i + j\Delta t)|)^2 \quad (3.6)$$

where Φ is the parameter set we estimate, E^{RX} is the output of our forward model and E^{RXm} is the actual measured signal. It should be noted that for all of our simulations that use this cost functional, we generate both computed return signal E^{RX} and the actual measurement E^{RXm} from our forward model since we do not have any real radar measurements.

Unfortunately, simulations we run with this cost functional is not successful even for the the simplest possible cases. We try with very simple objects with small number of parameters and start our initial shape model very close to the actual shape but the evolution always gets stuck at a local minima between the initial and the actual shape. Such situation implies that our cost functional is not well behaving with respect to shape parameters which means when our cost functional is computed on a path connecting two shapes in the parameter space, cost functional value makes ups and downs in between. Such behaviour makes the minimization impractical as our shape estimation scheme would stop at the closest local minima. To test if this really is the case, we choose two different parameter set that are our initial shape Φ_i and the actual one Φ_a . We then create a linear path in the parameter space connecting Φ_i to Φ_a and compute our cost functional in between. We compute the in between points from the convex combination of Φ_i and (Φ_i) that is given as:

$$\Phi(\lambda) = \Phi_i + \lambda(\Phi_a - \Phi_i) \quad (3.7)$$

. We start start from $\lambda = 0$ and move to $\lambda = 1$ and compute the cost functional at 1000 different Φ values. A snippet from some of the results we get is shown in Fig. 3.2. Results confirm our diagnosis about the presence of the local minima between two shape configurations. This clearly shows that our cost functional design in the time-domain is impractical

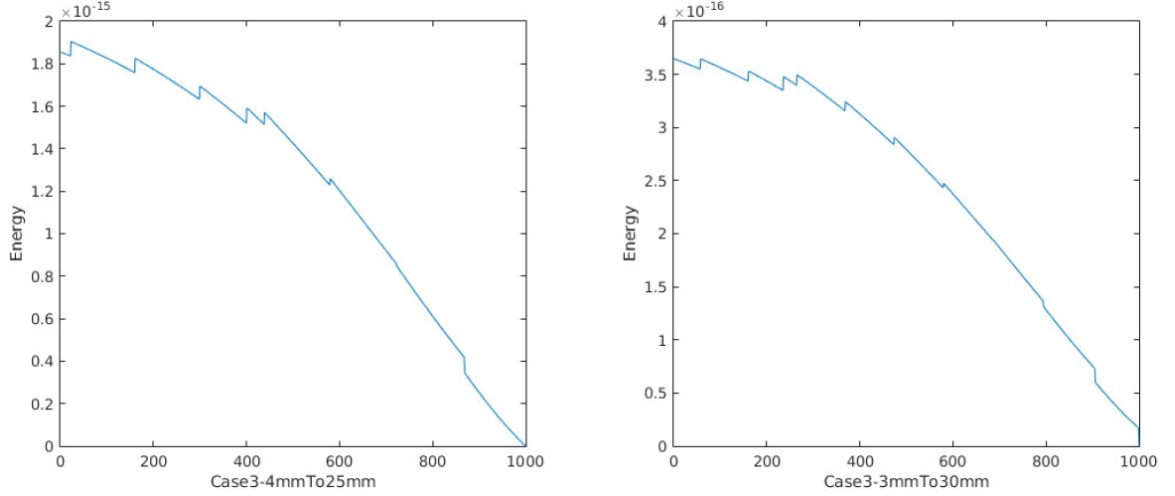


Figure 3.2: The trend of the cost functional value on a linear trajectory in the parameter space from initial shape model to the actual shape model. Horizontal axis denotes the point index at which the cost functional is computed. Image on the left is for a case where the wavelength is increase from 4mm to 25mm and for image on the right, it is increased from 3mm to 30mm.

to be used with an iterative shape estimation scheme due to the existence of local minima between two shape configurations.

Since we now know that using raw radar measurements to formulate our cost functional is impractical for our purposes, it is beneficial to take another look to the stereo vision case. Pixel intensity value is a measure of the power incident on the the corresponding element of the CCD array and we use the pixel intensity values of these array of elements in our cost functional for vision problems. Note that both for cameras and radar, we acquire our measurements through EM waves. The difference is that, with cameras, our raw measurement is the intensity of the EM wave whereas with radar, we measure the waveform itself. A natural question at this point is if it can be possible to express our cost functional in terms of the magnitude of EM wave for radar as well which is the basis of our second trial.

3.2.2 Second trial: Cost functional design using frequency-domain representation

We now develop a frequency-domain based cost functional that could possibly help with the local minima issue we have in a time-domain based solution. Our main motivation for

such an attempt is what we know from Eq. 3.2. For a constant frequency transmitted signal, all of the information we can get from the received signal is embedded in the complex factor between transmitted and received signal. This factor naturally lives in the frequency domain as it corresponds to the ratio between the Fourier transforms of transmitted received signals computed at that the frequency we use. Having multiple frequencies, we can formulate our cost functional in terms of the mismatch between the sets of complex factors, one from the computed return signal and the other from the actual measurement. Such representation comes with two advantages:

- The first advantage is that we our geometric information is packaged in a compact manner when compared to a time-domain based one. As a result, we do not have to use subsampling that we used before.
- We also do not explicitly introduce oscillations to our cost functional that could manifest as local minima whereas in the in the time-domain approach, we explicitly feed the cost functional with the signal itself which is oscillatory.

As a result, we could expect this design of the cost functional should help. Such an estimation scheme requires a stepped frequency radar where for each frequency, we take the complex factor relating the transmitted signal to the received signal. It should be noted that this does not have to contradict with our choice of the waveform (LFM). A linear chirp can also be approximated as a sequence of intervals where frequency is constant within each interval so that we can collect multiple frequency data using a single chirp. such an approximation is depicted in Fig. 3.3. Our cost functional in terms of frequency domain representation is given as:

$$E(\Phi) = \sum_{i=0}^{n-1} (|J(f_i, \Phi) - J^m(f_i)|)^2 \quad (3.8)$$

where $J(f_i, \Phi)$ and $J^m(f_i)$ are the complex factors relating the transmitted signal to the received signal (at frequency f_i) for computed and the measured signals, respectively. J is

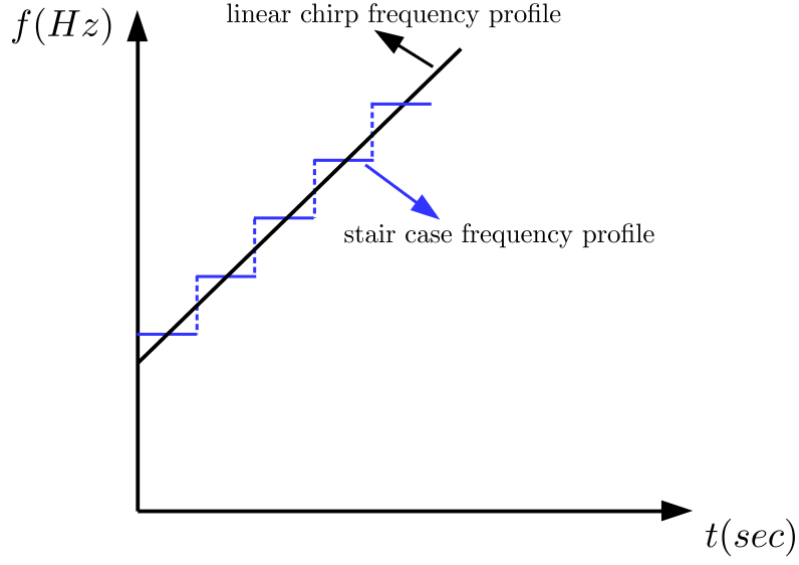


Figure 3.3: Approximation of a linear chirp with a staircase frequency profile

(from Eq. 3.2) given as:

$$J(f_i, \Phi) = \int_{S_0} \frac{G' G \sqrt{(-\mathbf{u}' \cdot \mathbf{n})} \sqrt{(-\mathbf{u} \cdot \mathbf{n})}}{R' R} \exp \left(-i 2\pi f_i \frac{R' + R}{c} \right) dS. \quad (3.9)$$

When we ran simulations by trying to minimize using this cost functional, we come across the similar problems as we have with the time-domain representation of the signal. We do not get convergence even when we start with a close initialization and choose a simple object. Taking a look at the cost functional value with respect to a linear trajectory in the parameter space, it becomes apparent that we still have issues related to the local minima. A snippet showing the cost functional value with respect to the path from the initial object to the actual object is depicted in Fig. 3.4. We can see why this cost functional cannot be effectively used for our framework which is due to the presence of local minima between the initial and the actual objects even when these objects are chosen to be very close to each other. We can also observe a correlation between the number of ups and downs and the initial-actual object distance which tells us that the frequency of the local minima is inversely proportional with the wavelength being used. As a result, we conclude that using

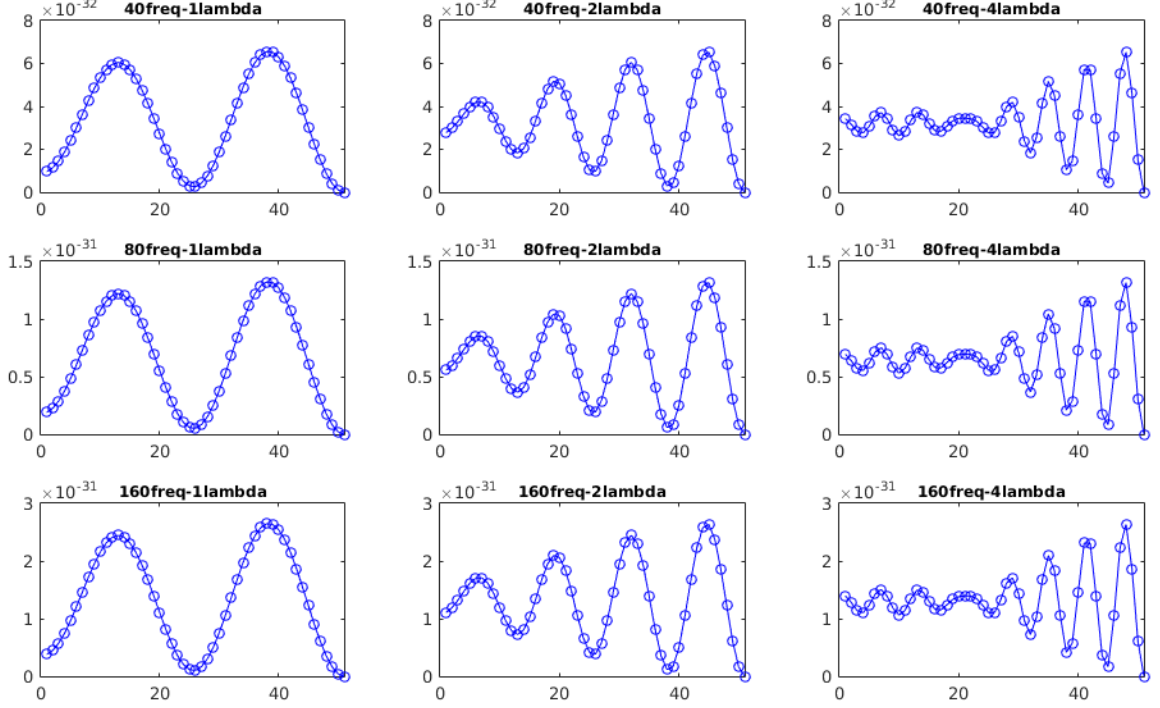


Figure 3.4: Cost functional value as function of intermediate shapes (in the parameter space, on a linear trajectory) between the initial and the actual shape. Along the same row, distance between the initial and the actual shape increases. Distances are chosen to be the multiples (1,2,4) of the average wavelength (λ). Along the same column, number of the intermediate frequencies used in the cost function increases ($n = 40, 80, 160$)

this cost function cannot be feasible for iterative shape estimation framework unless the wavelength that is being used is large. We can also make another observation which is: **Our local minima problem is caused by the waveform structure being introduced in our cost functional.**

At this point, we realize that trying to make use of the raw radar signal is impractical for our purposes and some kind of preprocessing is required. We expect our preprocessing to yield some signal attributes such that when we formulate our cost functional in terms of these attributes, local minima that are created by the manifestation of waveform structure in the cost functional can be avoided. Ideally, such attributes should provide three main traits:

- It should preserve as much geometric information as possible that is available in the

return signal.

- It should be as free as possible from the high frequency waveform structure of return signals.
- It should be smoothly changing with respect to the shape geometry, as we will rely on the gradient of the cost functional that will be formulated in terms of extracted information.

In the next section, we will develop a method by which these criteria are achieved.

3.2.3 Extraction of Electric Field Density Profile (over range)

We know that radar measurement consists of the returns reflected from the individual scatterers in the scene that constructively and destructively interfere each other and yields our measurement at each time instant. As a result, the shape information of the scene is embedded in the radar return signal in a very convoluted way in its raw form which makes it hard to use. There are also hardware limitations that need to be overcome since most of the time, radars operate at frequencies much higher than the sampling limits of available state-of-the-art analog to digital converters(ADC)[18]. Sampling requirements can partially be relaxed using demodulation but problem still remains when the signal is of a high bandwidth. We know that the main information radars use is the time delay between the transmitted and the received signal. We also know that a geometric object in front of the radar occupies a continuum of range values and as a result of this, radar return signal will be the composition of return signals with different time delay values. To this end, our preprocessing will aim to decompose the radar return signal into its components with different time delay values (ranges) where we can utilize the strength of these components in our energy functional. Since we have a continuum of range values, we express the strength of each component as a density quantity over the range. We propose a two step approach to this problem that will show the feasibility of extracting such information from the return signal. Our algorithm

consist of two main steps:

- In the first step, we employ the stretch processing algorithm on the return signal by which we will obtain an intermediate signal. We will show mathematically that the strength of every frequency component in this intermediate signal reveals us the information of how much of the resultant electric field comes from a unique range value. Since scenes usually cover a continuum of range values, for each range, we get an electric field density value.
- In the second part, we will develop a way to extract the electric field density profile as a function of range by using the time samples of the intermediate signal we obtain from stretch processed (deramped) signal.

Stretch Processing

First, we use a well known technique called stretch processing [19] which both makes sampling at lower ADC frequencies possible and also makes the range decomposition of a signal possible. Stretch processing is done by mixing the radar return signal with a heterodyne signal that is a time-delayed replica of the transmitted signal. Result of this process (deramped signal) is a new signal with a much lower frequency components which significantly relaxes the sampling requirements. However, the key property of this new signal that is useful for our purposes is that it gives a direct mapping from frequencies to ranges when the transmitted signal is an LFM waveform. Stretch processing is depicted in Fig. 3.5. Assuming a unit amplitude LFM pulse centered at the origin for our transmitted signal ($f(t)$) which is given as:

$$f(t) = \Pi\left(\frac{t}{\tau_p}\right) \exp(i2\pi(f_c + \alpha t)t) \quad (3.10)$$

and a point object in the space that has a round-trip distance value of D with respect to a given transmitter and receiver pair, received signal($f_r(t)$) becomes a time-shifted and

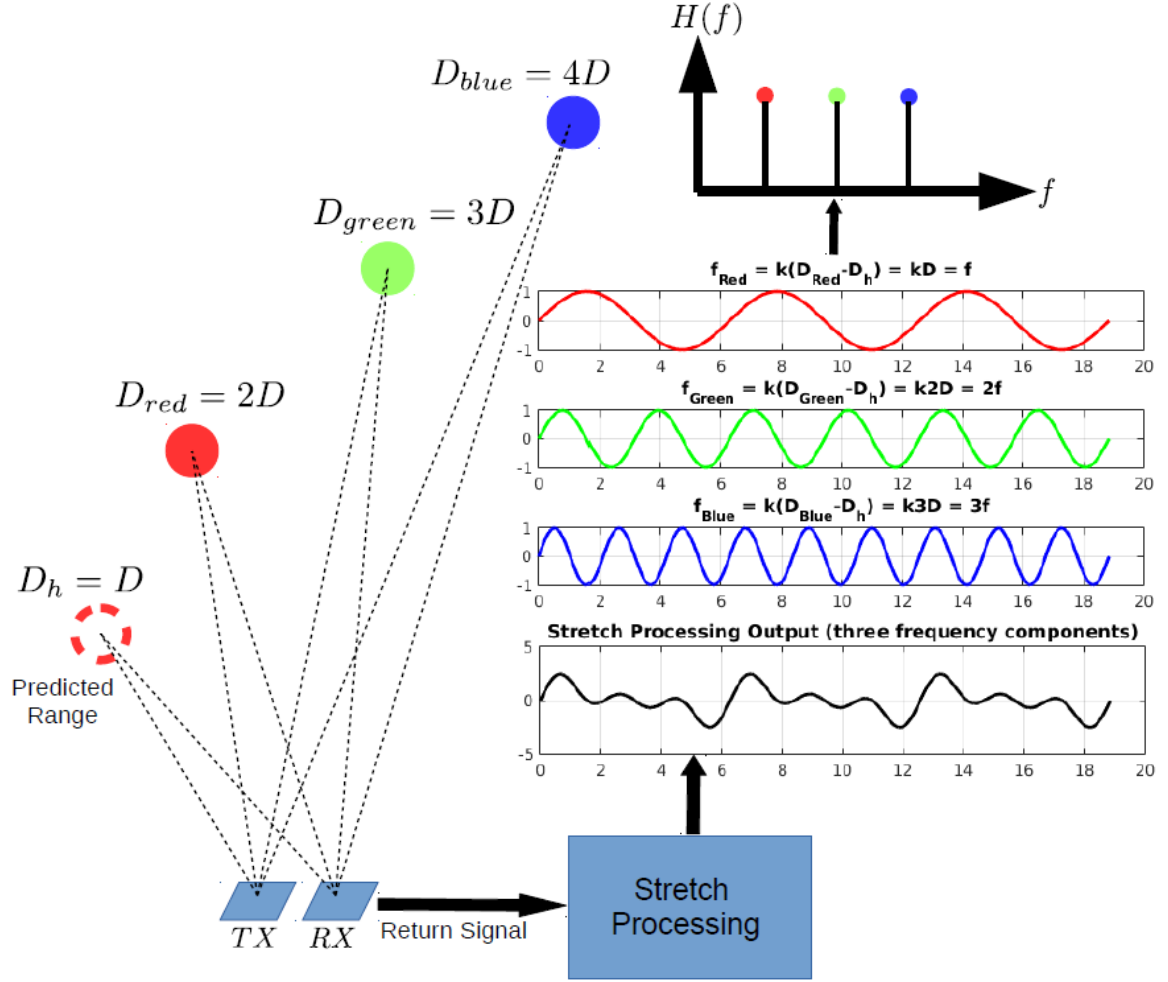


Figure 3.5: Frequency spectrum of deramped signal for three scatterers lying at different range values. See how difference between the range of a scatterer and the predicted range value is mapped into frequency component. This allows us to obtain the decomposition of the total electric field strength along the range by looking at the frequency spectrum of deramped (stretch processed) signal.

scaled version of the transmitted pulse. It is given as:

$$f_r(t) = C\Pi_r(t) \exp[i2\pi(f_c + \alpha(t - t_r))(t - t_r)] \quad (3.11)$$

$$\Pi_r(t) = \Pi\left(\frac{t - t_r}{\tau_r}\right) \quad (3.12)$$

$$\tau_r = \tau_p \quad (3.13)$$

$$t_r = \frac{D}{c} = \frac{R' + R}{c} \quad (3.14)$$

We also keep a replica of the transmitted signal within the stretch processor with a pre-designed amount of delay (t_h) and possibly with a different pulse length ($\tau_h > \tau_r$). It is given as:

$$f_h(t) = \Pi_h(t) \exp [i2\pi (f_c + \alpha (t - t_h)) (t - t_h)] \quad (3.15)$$

$$\Pi_h(t) = \Pi\left(\frac{t - t_h}{\tau_h}\right) \quad (3.16)$$

When the received signal ($f_r(t)$) is fed to the stretch processor where its conjugate is mixed with the heterodyne signal ($f_h(t)$), at the mixer output we obtain:

$$f_{r^*h}(t) = f_r^*(t) f_h(t) \quad (3.17)$$

$$= C \Pi_r(t) \Pi_h(t) e^{-i2\pi[f_c + \alpha(t - t_r)](t - t_r)} e^{i2\pi[f_c + \alpha(t - t_h)](t - t_h)} \quad (3.18)$$

$$= C \Pi_r(t) \Pi_h(t) e^{i2\pi\{-[f_c + \alpha(t - t_r)](t - t_r) + [f_c + \alpha(t - t_h)](t - t_h)\}} \quad (3.19)$$

$$= C \Pi_r(t) \Pi_h(t) e^{i2\pi f_c(t_r - t_h)} e^{-i2\pi\alpha(t_r^2 - t_h^2)} e^{i4\pi\alpha(t_r - t_h)t}. \quad (3.20)$$

We have two window functions in the equation. Multiplication of these give another window function that takes the value of one where both functions are one. In a more concise form we obtain:

$$\Pi_{rh}(t) = \Pi_r(t) \Pi_h(t) \quad (3.21)$$

$$f_{r^*h}(t) = C \Pi_{rh}(t) e^{i2\pi f_c(t_r - t_h)} e^{-i2\pi\alpha(t_r^2 - t_h^2)} e^{i4\pi\alpha(t_r - t_h)t} \quad (3.22)$$

As a result, we can see that for a point object in the scene, the signal at the mixer output becomes a constant frequency sinusoid with a frequency value of $2\alpha(t_r - t_h)$. It should be noted that the first two exponential terms are phase terms and do not have any t dependency. Since t_h is a fixed parameter, we obtain a linear relation between the frequency of the signal at the mixer output and the range of the point object. Therefore, if we assume an infinitely

long pulse, a point object manifests itself in the frequency domain as a weighted delta function. Under rectangular windowing, this delta function is mapped to a sinc function. Therefore, we obtain a sinc shaped frequency spectrum when we use an LFM pulse. In this case, object range can be estimated from the peak frequency component of the discrete Fourier transform (DFT) of the signal. We will shortly see that this is not practical in our case since, for our problem, we have scatterers lying on a continuous range interval which creates a continuous spectrum of frequencies.

A critical consideration at this point is to design t_h . We know that if the discrepancy between t_h and t_r is large, so will be the frequency of the signal at the mixer output. It also comes into the picture at the intersection of two window functions where a sloppy choice of t_h can result in a zero intersection of t_r and t_h . Because of these reasons, we need a rough estimate of the object range by which we can design t_h . A smart choice of t_h should both minimize the upper bound of the set of possible frequencies and at the same time should yield a maximum possible amount of intersection between $\Pi_r(t)$ and $\Pi_h(t)$. When we have a continuous surface instead of point scatters, the process becomes tricky. A continuous surface is a composition of infinitely many scatterers that occupy a continuous range in space with respect to a chosen transmitter and receiver pair. In the case of an LFM pulse, the mixer output is mathematically given as (in terms of a given range value D):

$$t_r = \frac{D}{c} \quad (3.23)$$

$$M_o(t) = \int_{D_{min}}^{D_{max}} P_D(D) \Pi_{rh}(t) e^{i2\pi f_c(t_r - t_h)} e^{-i2\pi\alpha(t_r^2 - t_h^2)} e^{i4\pi\alpha(t_r - t_h)t} dD \quad (3.24)$$

$$= \int_{D_{min}}^{D_{max}} P_D(D) \Pi_{rh}(t) \underbrace{e^{i2\pi f_c(\frac{D}{c} - t_h)} e^{-i2\pi\alpha((\frac{D}{c})^2 - t_h^2)}}_{\text{Time independent terms}} e^{i4\pi\alpha(\frac{D}{c} - t_h)t} dD \quad (3.25)$$

where M_o denotes the signal at the mixer output, $P_D(D)$ denotes the resultant electric field density coming from the range value (round-trip distance) of D . As we can see from the second exponential, we have a continuum of frequencies in the integral. We will estimate a discrete version of $P_D(\cdot)$ for our method which is to be used as our feature set. Since

we will have a finite number of discrete samples of $M_0(\cdot)$ at the mixer output, we need to extract $P_D(\cdot)$ from this discrete signal. This estimation will then be used as our feature set.

From Time Samples to Average Electric Field Densities

Extracting $P_D(\cdot)$ from $M_0(\cdot)$ is not straightforward as it is composed of a continuum of frequencies and we only have a finite number of discrete samples of $M_0(\cdot)$. We know that assuming a certain level of shape regularity for the scene surface, $P_D(\cdot)$ will have a smooth profile. However, extracting the frequency components of $M_0(\cdot)$ does not give us $P_D(\cdot)$ directly. Instead what we obtain is $P_D(\cdot)$ multiplied by the first two exponential phase terms in the integral that are independent of time. From Eq. 3.25, we can see that depending on how high the carrier frequency is, these phase terms can be very oscillatory with respect to the range value. As a result, estimation of $P_D(\cdot)$ using a discrete Fourier transform may not yield a good estimation of the electric field density profile as such an operation does not compensate for the phase terms which we do not want. It should also be noted that although the frequency spectrum is oscillatory, we know the structure of this oscillation which can help us when correctly incorporated into the problem. That is why we propose a different technique that considers the prior information we have about the structure of the frequency spectrum. The oscillatory nature of the frequency spectrum of a stretch processed signal is depicted in Fig. 3.6 and 3.7.

We start by simplifying our integral expression in Eq. 3.25. Since the window function $\Pi_{rh}(\cdot)$ depends on t_r and t_r depends on the range(round-trip distance), we have a different $\Pi_{rh}(\cdot)$ profile for each range value. We will limit ourselves to the time range where all possible window functions take the value of one. It should be noted that this is barely a restriction unless we have a scene that occupies a very large range interval. For instance, let's assume the difference between the largest and the smallest range value of the scene is $100m$ and the pulse length is $100\mu sec$. Multiplying the pulse duration with the speed of the

Frequency Spectrum of the Stretch Processor Output($M_0(t)$)

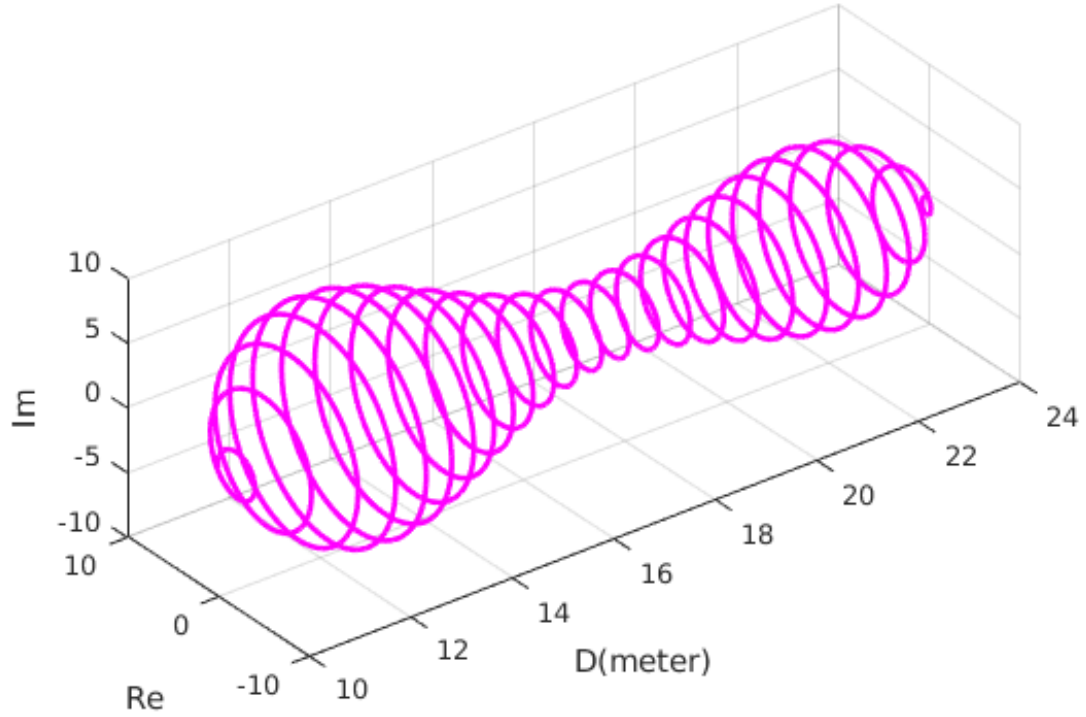


Figure 3.6: Frequency (range) spectrum of deramped signal. It is given as a function of range value.

light, we compute the length of the transmitted pulse in space as:

$$= 299792458m/sec * 100 \times 10^{-6}sec = 29979.2458m \quad (3.26)$$

Obviously for the points with minimum and maximum range values on the surface, the shift between the associated window functions will be equal to the 100m. As a result, we crop the first and the last 100m sections of the received signal (%0.667 of the total pulse duration) and for the remaining part we know window functions of all range values will return the value of one. This allows us to remove the window function from the equation

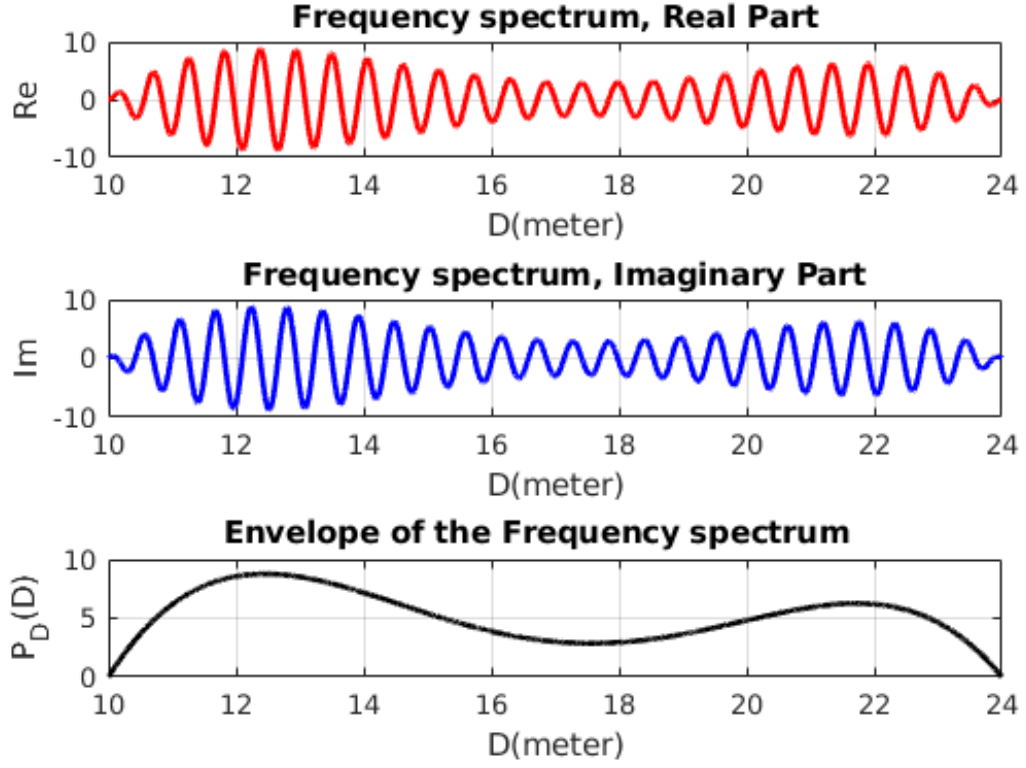


Figure 3.7: Frequency spectrum of deramped signal in 2D with its envelope $P_D(D)$.

after which our equation becomes:

$$M_o(t) = \int_{D_{min}}^{D_{max}} P_D(D) e^{i2\pi f_c \left(\frac{D}{c} - t_h\right)} e^{-i2\pi\alpha \left(\left(\frac{D}{c}\right)^2 - t_h^2\right)} e^{i4\pi\alpha \left(\frac{D}{c} - t_h\right)t} dD \quad (3.27)$$

Since we want to estimate the frequency spectrum of the signal, we will find it more convenient to express the integral in terms of frequency. From the time dependent exponential term in Eq. 3.27, we have frequency in terms of the range (D) as:

$$f = 2\alpha \left(\frac{D}{c} - t_h \right) \quad (3.28)$$

range then becomes:

$$D = \left(\frac{f}{2\alpha} + t_h \right) c \quad (3.29)$$

$$dD = \frac{c}{2\alpha} df \quad (3.30)$$

Substituting this expression in Eq. 3.27, we obtain:

$$P_f(f) = P_D(D(f)) \quad (3.31)$$

$$M_o(t) = \int_{f_{min}}^{f_{max}} P_f(f) \frac{c}{2\alpha} e^{i2\pi \left(\frac{f_c f}{2\alpha} - \frac{f^2}{4\alpha} + (t-t_h)f \right)} df \quad (3.32)$$

where $D(f)$ denotes the range value corresponding to a given frequency f . Let's assume the number of the time samples we have is N_t . In a similar manner, we will also divide the integral expression into pieces by which we can assume $P_f(f)$ is approximately constant within each interval. At a given time instant t_i , our expression is given by:

$$M_o^i = \sum_{j=1}^{N_f} \int_{f_{j-1}}^{f_j} P_f(f) \frac{c}{2\alpha} e^{i2\pi \left(\frac{f_c f}{2\alpha} - \frac{f^2}{4\alpha} + (t_i-t_h)f \right)} df \quad (3.33)$$

For a fine enough partition of the integral, it can be assumed that $P_f(f)$ is approximately constant for each interval since we assume a certain amount of regularity of the surface shape with respect to the range value that yields a smooth $P_f(f)$. Thus we can take $P_f(f)$ outside of the integral which gives us:

$$M_o^i = \sum_{j=1}^{N_f} P_f^j \int_{f_{j-1}}^{f_j} \frac{c}{2\alpha} e^{i2\pi \left(\frac{f_c f}{2\alpha} - \frac{f^2}{4\alpha} + (t_i-t_h)f \right)} df \quad (3.34)$$

Using this decomposition for each time instant, we obtain a linear systems of equations that

can be expressed compactly as:

$$\begin{bmatrix} M_o^1 \\ M_o^2 \\ M_o^3 \\ \vdots \\ M_o^{N_t} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{13} & \cdots & A_{1N_f} \\ A_{21} & A_{22} & A_{23} & \cdots & A_{2N_f} \\ A_{31} & A_{32} & A_{33} & \cdots & A_{3N_f} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_{N_t1} & A_{N_t2} & A_{N_t3} & \cdots & A_{N_tN_f} \end{bmatrix} \begin{bmatrix} P_f^1 \\ P_f^2 \\ P_f^3 \\ \vdots \\ P_f^{N_f} \end{bmatrix} \quad (3.35)$$

$$\mathbf{M}_0 = \mathbf{A} \mathbf{P}_f \quad (3.36)$$

where A_{ij} is given as:

$$A_{ij} = \int_{f_{j-1}}^{f_j} \frac{c}{2\alpha} e^{i2\pi \left(\frac{f_c f}{2\alpha} - \frac{f^2}{4\alpha} + (t_i - t_h) f \right)} df \quad (3.37)$$

The integrand is an exponential of a quadratic polynomial with respect to the integral variable f . This can be expressed in terms of special functions. Completing the square

$$A_{ij} = \int_{f_{j-1}}^{f_j} \frac{c}{2\alpha} e^{i2\pi \left(\frac{f_c f}{2\alpha} - \frac{f^2}{4\alpha} + (t_i - t_h) f \right)} df \quad (3.38)$$

$$= \int_{f_{j-1}}^{f_j} \frac{c}{2\alpha} e^{-i2\pi \left(\frac{1}{4\alpha} f^2 - \left(\frac{f_c}{2\alpha} + t_i - t_h \right) f \right)} df \quad (3.39)$$

$$= \int_{f_{j-1}}^{f_j} \frac{c}{2\alpha} e^{-i2\pi \left[\frac{f}{2\sqrt{\alpha}} - \sqrt{\alpha} \left(\frac{f_c}{2\alpha} + t_i - t_h \right) \right]^2} e^{i2\pi \alpha \left(\frac{f_c}{2\alpha} + t_i - t_h \right)^2} df \quad (3.40)$$

$$= \frac{c}{2\alpha} e^{i2\pi \alpha \left(\frac{f_c}{2\alpha} + t_i - t_h \right)^2} \int_{f_{j-1}}^{f_j} e^{-i2\pi \left[\frac{f}{2\sqrt{\alpha}} - \sqrt{\alpha} \left(\frac{f_c}{2\alpha} + t_i - t_h \right) \right]^2} df \quad (3.41)$$

A change of variables yields:

$$x = -\sqrt{\frac{\pi}{2\alpha}} f + \sqrt{2\pi\alpha} \left(\frac{f_c}{2\alpha} + t_i - t_h \right) \quad (3.42)$$

$$dx = -\sqrt{\frac{\pi}{2\alpha}} df \quad (3.43)$$

Using the new integration variable, A_{ij} is expressed as:

$$A_{ij} = -\frac{c}{\sqrt{2\alpha\pi}} e^{i2\pi\alpha\left(\frac{f_c}{2\alpha}+t_i-t_h\right)^2} \int_{x_{j-1}}^{x_j} e^{ix^2} dx \quad (3.44)$$

$$= -\frac{c(1-i)}{4\sqrt{\alpha}} e^{i2\pi\alpha\left(\frac{f_c}{2\alpha}+t_i-t_h\right)^2} \left(\operatorname{erfi} \left[\left(\frac{1+i}{\sqrt{2}} \right) x_j \right] - \operatorname{erfi} \left[\left(\frac{1+i}{\sqrt{2}} \right) x_{j-1} \right] \right) \quad (3.45)$$

Computing all A_{ij} , we can form the matrix in Eq. 3.35. Since we also know the time samples (\mathbf{M}_0), the vector of the signal strength values (\mathbf{P}_f) can be obtained by solving the linear system of equations.

Physically, the value of P_f^j corresponds to an average electric field density value coming from the subset of the object surface within the range interval of $[D_{j-1}, D_j]$. However, in this form it is an average over the frequency. We will find it useful to express it with respect as an average over range value which becomes (from Eq. 3.28):

$$P_D^j = P_f^j \frac{2\alpha}{c}. \quad (3.46)$$

As a result, the vector of P_D^j gives us a discrete electric field density profile over the range value. It should be noted that P_D^j is completely defined by the object geometry and is not effected by the highly oscillatory nature of the radar signals. Since we use gradient-based minimization algorithms, defining our cost functional in terms of these average electric field density values gives us the chance to avoid the local minima that could be introduced by oscillations of the radar return signal. To keep the notation simpler, for the following discussion, we will use H_j to denote the average electric field density where the subscript denotes the frequency bin index.

We can also relate these average electric field density values to shape geometry in a direct fashion. Since each of these is related to a specific range interval, for a given shape, we can express its value as an integration over the set of scatterers within that range bin.

We have H_j (from our forward model) as:

$$H_j = \frac{1}{\Delta D} \int_{D_{j-1}}^{D_j} \frac{G'G\sqrt{(-\mathbf{u}' \cdot \mathbf{n})}\sqrt{(-\mathbf{u} \cdot \mathbf{n})}}{R'R} dS \quad (3.47)$$

where we integrate over the part of the object that lies within the range value of D_{j-1} and D_j . We can see that H_j is a purely geometric quantity as its integrand only depends on the distance of the infinitesimal radiator and its surface normal. Another observation we can make is that integrand does not include any sinusoidal that can manifest as local minima in our cost functional which is, as we previously discuss, one of the main motivations of performing this preprocessing. **We should note that, we do not actually perform this preprocessing in any of the results presented in this thesis. Instead we choose to generate these average electric field density values directly from Eq. 3.47 as all of our results are based on simulation where we have the knowledge of both our evolving and the actual shape. This section is to show that the information (electric field density profile over range) we build our inversion algorithm on is possible to extract from actual radar measurements.**

CHAPTER 4

A 2D DISCRETE SHAPE MODEL

In this chapter, we will consider a tailored version of our inversion algorithm and will present simulation results. We describe the specific case we consider as:

- We work in 2D instead of 3D. We will still be using the forward model we use for 3D where we will discard the changes in the third dimension. A physical interpretation of this can be thought as having the shape as thin plate where the changes in G' , G , \mathbf{u}' , \mathbf{u} , \mathbf{n} and the range are negligible in the direction of thickness and can be thought as constant. We will also assume that antenna normal and the plate lies in the same plane.
- Our shape model is restricted to be closed curve which is also a star-shaped object (all of the points on the curve is visible from a point inside the curve).
- We use a discrete shape model where we approximate our shape model as a polygonal shape which is parameterized with the vertex coordinates of the polygonal model.

We use an iterative inversion scheme where we start with an initial shape and evolve it along the iterations trying to minimize a cost functional. We assume that given a radar return signal, we can extract the electric field density profile as a function of a range continuum on which our inversion will be based (as discussed in Chapter 3). In explanation, we assume to have the knowledge of how strong the total reflection is at a given range in space which is obtained from the radar return signal through a proper preprocessing. Our inversion scheme is depicted in the Fig. 4.1.

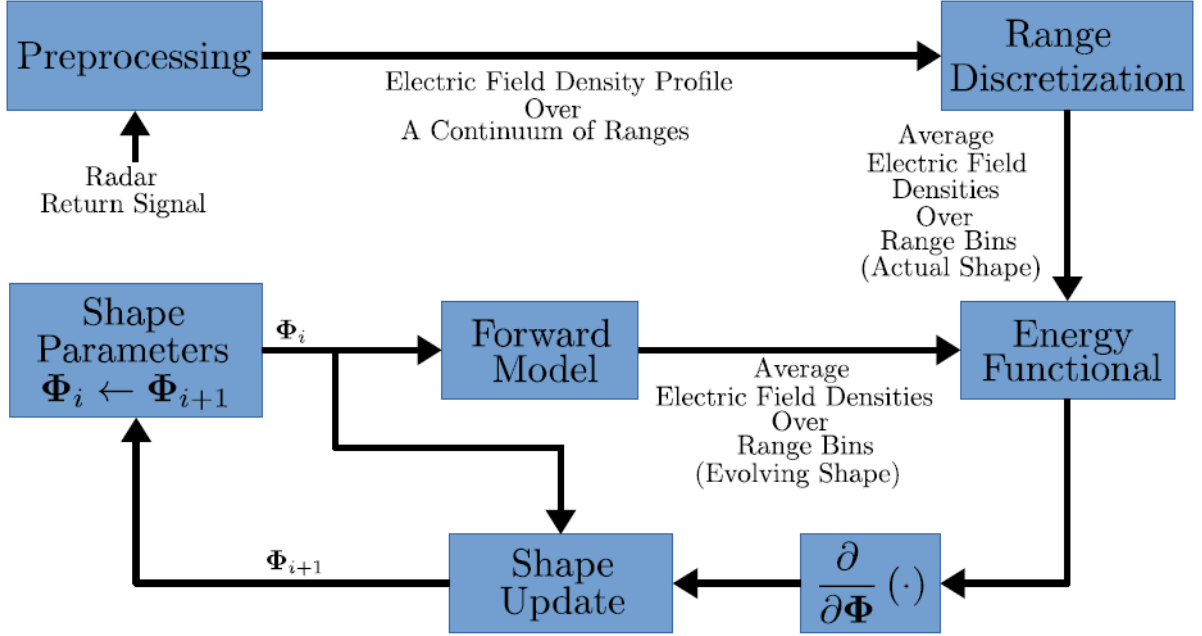


Figure 4.1: Flow chart of our inversion algorithm.

4.0.1 Forward Model

The input to our forward model are a shape and an antenna pair (TX/RX) for which we compute a set of average electric field values through preprocessing we discuss in Chapter 3. We compute these values by chopping off the range continuum into finite number of bins and computing the average reflection over the part of our shape that lies within each range bin. Our forward mode is built on two assumptions that can be listed as:

- Transmitter and receiver are directional antennas where the transmitted and received power is a function of the ray direction.
- The governing reflection model on the surface is Lambertian where we assume that the shape behaves as an ideal diffuse reflector.

Under these assumptions, an infinitely small reflector within the i^{th} range bin, contributes to the average electric field density by:

$$dH_i = \frac{1}{\Delta D} Q(\mathbf{x}, \mathbf{n}) ds. \quad (4.1)$$

where $Q(\mathbf{x}, \mathbf{n})$ is the amplitude of the electric field density created by infinitesimal reflector with a unit normal vector of \mathbf{n} that is located at \mathbf{x} and ΔD is the range width. Using the assumptions we have of antennas and the surface reflectivity, $Q(\mathbf{x}, \mathbf{n})$ is given as:

$$Q(\mathbf{x}, \mathbf{n}) = \frac{G'(\mathbf{x}) G(\mathbf{x}) \sqrt{-\mathbf{u}'(\mathbf{x}) \cdot \mathbf{n}} \sqrt{-\mathbf{u}(\mathbf{x}) \cdot \mathbf{n}}}{R'(\mathbf{x}) R(\mathbf{x})} \quad (4.2)$$

where G' and G are the antenna gains of the transmitter and receiver antennas, \mathbf{u}' and R' are the unit ray direction and ray length for the incident power on the point reflector, \mathbf{u} and R are the unit ray direction and the ray length for radiated power measured by the receiver and \mathbf{n} is the surface normal vector of the point reflector located at \mathbf{x} . It should be noted for a fixed antenna pose (rotation and translation) in space, that G' , G , \mathbf{u}' , \mathbf{u} are fully defined given \mathbf{x} which we will use as the only independent variable for these functions. To compute the average electric field, we integrate the electric field density over the shape boundaries that lies within a range bin and divide it by the thickness of the corresponding range bin. However, we also need to consider the fact we may not get any reflection from some parts of the object within the same bin due to the self occlusions as some parts of the shape may block other parts from getting the transmitted signal. As a result, we need to exclude the occluded parts of the object from the integration to be able to correctly model the physical phenomena. Luckily, our inversion scheme can naturally handle this matter as we have an evolving shape on which we can perform a visibility analysis. We then compute the average electric field density within the i^{th} range bin with respect to j^{th} antenna as:

$$H_i^j = \frac{1}{\Delta D} \int_{S_i^j \cap V_j \cap e} Q(\mathbf{x}, \mathbf{n}) ds \quad (4.3)$$

where S_i^j set shape of all points that lie within the i^{th} range bin with respect to j^{th} antenna pair, e is the set of the visible points that defines the shape boundaries from which the transmitted signal is reflected, V_j is the set of points visible to the j^{th} antenna and (ΔD)

is the size of the range bin which we define as the difference between the maximum and minimum round trip distances within the bin. For a general placement of the antenna pair where the transmitter(TX) and receiver(RX) are not collocated, the borders of the range bins are defined by a set of ellipses where foci of these ellipses are being the transmitter and receiver locations. Our forward model is depicted in the Fig. 4.2.

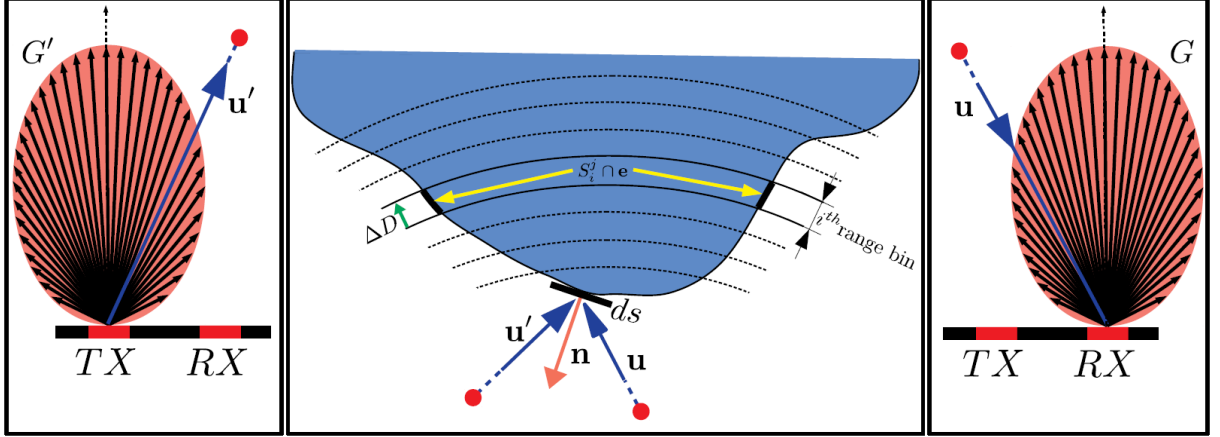


Figure 4.2: Depiction of our forward model. Transmitter(TX) and Receiver(RX) are directional antennas. \mathbf{u}' and \mathbf{u} are unit ray directions and \mathbf{n} is the unit normal vector of a point on our shape. Domain of integration for i^{th} range is $S_i^r \cap \mathbf{e}$ that consists of two curve segments assuming both segments are fully visible to the antenna pair. ΔD is the difference of round trip distance values between the beginning and end of a range bin.

4.0.2 Inversion

The input to our inversion are the average electric field density values computed from forward model and those computed from the discretization of electric field density profile obtained through preprocessing of the radar return signal. Our inversion process aim to evolve our initial shape through iterations such that these two sets of information get close to each other. To this end, we design a energy functional that measures the discrepancy between two sets. Since there can be multiple shapes that yields the same radar measurement, we will also find useful to add a regularizer term to our energy functional. Assuming the general case where we have multiple antennas or a single moving antenna, data fidelity

term of out cost functional is expressed as:

$$E_D = \frac{1}{2} \sum_{j=1}^N [\mathbf{D}_j^T (\mathbf{H}_j - \mathbf{H}_j^m)]^2 \quad (4.4)$$

where \mathbf{H}_j is the vector of average electric field density values of evolving object and \mathbf{H}_j^m is that of radar measurement and \mathbf{D}_j is the vector of average range values for the corresponding ranges all with respect to j^{th} antenna. We weight each electric field density value with the average range value of the corresponding bin to balance the decay in the electric field strength over range so that our cost function can treat uniformly to all parts of the object as otherwise parts that are close to the antennas could be favored due to the strong reflection.

For the regularization term, we have the flexibility to incorporate any kind shape priors. For the scope of this work, we will choose to define it as the curvature of the shape, which in result, will favor smooth shapes. At this point, our choice for the parametrization of the shape becomes important as we need to tailor our curvature regularizer for our specific parametrization. For the easiness of the implementation, we will consider star-shaped objects that can be expressed in polar coordinates where every point on the object has a unique angular position. We will additionally choose a discrete representation for the curve where it will be represent as a polygonal object. We will then evolve the curve by updating the vertex coordinates. Our curvature regularizer is given as:

$$E_R = \frac{1}{2} \sum_{k=1}^{N_v} \left\| \frac{\mathbf{v}_{k+1} - \mathbf{v}_k}{\|\mathbf{v}_{k+1} - \mathbf{v}_k\|} - \frac{\mathbf{v}_k - \mathbf{v}_{k-1}}{\|\mathbf{v}_k - \mathbf{v}_{k-1}\|} \right\|^2 \quad (4.5)$$

where N_v is the number of the vertices and we have $\mathbf{v}_{-1} = \mathbf{v}_{N_v}$ and $\mathbf{v}_{N_v+1} = \mathbf{v}_1$ due to our shape being a closed curve.

Adding two terms together, we have our energy functional as:

$$E = E_D + \lambda E_R \quad (4.6)$$

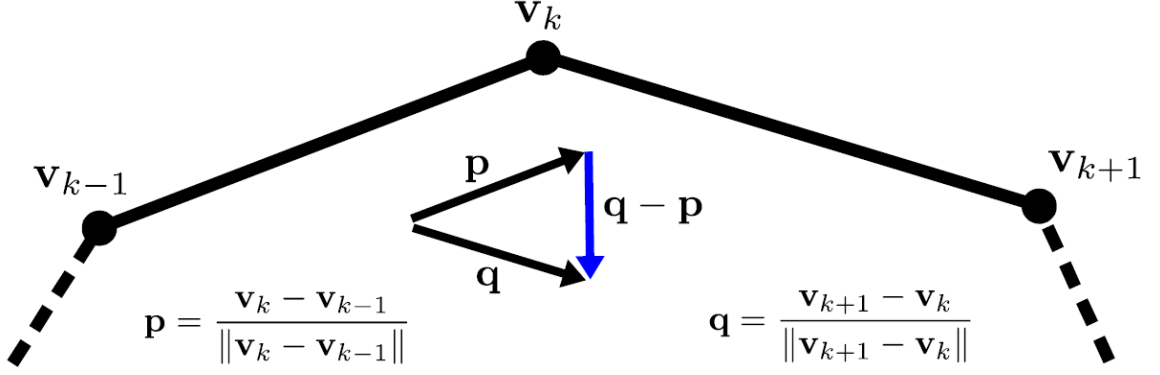


Figure 4.3: Curvature regularization for a polygonal object. We take the directions of consecutive edges of the polygon and penalize the norm of the difference between the directions of these two edges which is expected to canalize evolution towards smooth shapes.

where E_D favoring decreasing the discrepancy between the evolving shapes and the actual shape and E_R favoring smoothness of the shape model. λ is the regularization coefficient.

4.0.3 Optimization

Given an initial shape, our method requires computation of the gradient of the energy functional with respect to the parameter set we will use for shape evolution. Since we consider star-shaped polygonal objects where the vertices are the control points, we will find it useful to derive the derivatives of the energy functional with respect to the control points of the polygonal shape which will then be used to compute the gradient. As the inputs of energy functional are integral expressions, derivatives are functions of both the integrand and the domain of integration which requires us to perform the visibility analysis.

Since we use a polygonal shape model in the simulations we will present, we use a visibility analysis method for such shapes. For the polygonal representation, we use a simple algorithm that takes the evolving shape and the antenna pair as an input and returns the visible edges of the shape with respect to the antenna pair. Steps of our algorithm is given in Alg.1. It should be noted that the algorithm we use has $O(n^2)$ complexity with respect to the number of vertices which will suffice for our purposes. It can as well be replaced with more efficient visibility analysis tools available in computational geometry

Algorithm 1 Computation of the visible edges of the polygon.

```

 $\mathbf{S}^c \leftarrow \{\}$ 
for  $e_i^{i+1} \in \{e_1^2, e_2^3, \dots, e_{N_v}^1\}$  do
   $p_i^c \leftarrow$  Center point of  $e_i^{i+1}$ 
   $p_i^a \leftarrow$  Location of the RX/TX antenna pair
   $l_i \leftarrow$  Line segment that connects  $p_i^c$  to  $p_i^a$ 
  for  $e_j^{j+1} \in \{e_1^2, e_2^3, \dots, e_{N_v}^1\} \setminus e_i^{i+1}$  do
    if  $l_i$  intersects  $e_j^{j+1}$  then
       $\mathbf{S}^c \leftarrow \mathbf{S}^c + e_i^{i+1}$ 
      break
    end if
  end for
end for
return  $\{e_1^2, e_2^3, \dots, e_{N_v}^1\} \setminus \mathbf{S}^c$ 

```

literature [20] [21] [22].

To compute the derivatives of the energy functional with respect to the vertex coordinates, we find it useful to use the chain rule since the dependency of the energy functional to the vertex coordinates can be concisely expressed using function composition. A perturbation in the vertex coordinate changes the placement of two edges that is connected to it and a change in the placement of the edges changes the average electric field density values of the range bins that have an intersection with this edges. Dependency graph of the data fidelity term of energy functional (E_D) to a vertex coordinates \mathbf{v}_k is shown in the Fig. 4.4. It should be noted that computation of the partial derivative of E_D with respect to \mathbf{H}_j and

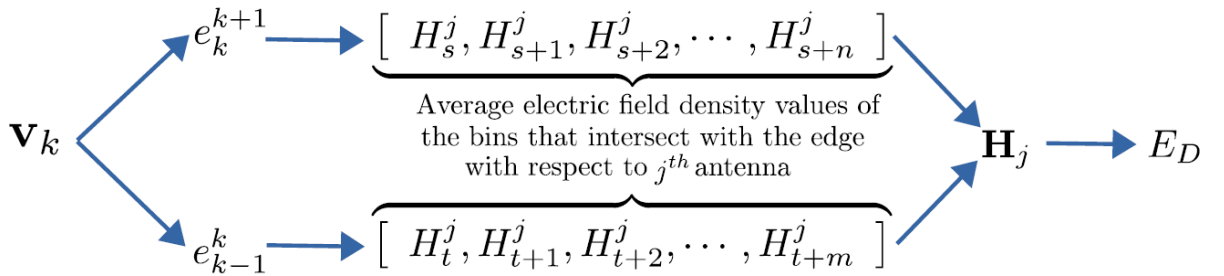


Figure 4.4: Dependency graph showing how vertex coordinates are related to E_D . Subscript and superscripts of an edge (e) denote the vertices it connects to. Subscript and superscript of the electric field density values denote the range bin index and antenna index, respectively.

that of \mathbf{H}_j with respect to its i^{th} frequency bin (H_i^j) is straightforward from the definition of the energy functional in Eq. 4.4. However, partial derivative of H_i^j with respect to \mathbf{v}_k can be tricky depending on how the edges of the polygon are placed with respect to the range bins. This is due to the fact that some of the edges will lie on multiple range bins that causes the integration borders to be a function of the polygon vertices since a perturbation in the vertex coordinate will change where the connected edge will intersect the range border. Some of the possible scenarios are shown in Fig. 4.5. Partial derivative of H_i^j with respect to \mathbf{v}_k is given as:

$$\frac{\partial H_i^j}{\partial \mathbf{v}_k} = \frac{1}{\Delta D} \frac{\partial}{\partial \mathbf{v}_k} \left(\int_{S_i^j \cap e_{k-1}^k} Q(\mathbf{x}, \mathbf{n}) ds + \int_{S_i^j \cap e_k^{k+1}} Q(\mathbf{x}, \mathbf{n}) ds \right) \quad (4.7)$$

where S_i^j is the set of points contained in i^{th} range bin with respect to the j^{th} antenna pair. Domain of integration then becomes the part of the edge contained in the range bin. Taking the derivative, we get:

$$\frac{\partial H_i^j}{\partial \mathbf{v}_k} = \frac{1}{\Delta D} \left(\int_{S_i^j \cap e_{k-1}^k} \frac{\partial}{\partial \mathbf{v}_k} Q(\mathbf{x}, \mathbf{n}) ds + \int_{S_i^j \cap e_k^{k+1}} \frac{\partial}{\partial \mathbf{v}_k} Q(\mathbf{x}, \mathbf{n}) ds + B_{up} + B_{lo} \right) \quad (4.8)$$

where B_{up} and B_{lo} are the boundary terms that are given as:

$$B_{up} = Q(\mathbf{p}_k^{k+1}, \mathbf{n}) \frac{\partial \|\mathbf{p}_k^{k+1} - \mathbf{v}_k\|}{\partial \mathbf{p}_k^{k+1}} \frac{\partial \mathbf{p}_k^{k+1}}{\partial \mathbf{v}_k}, \quad \text{for } \mathbf{p}_k^{k+1} \neq \mathbf{v}_{k+1} \quad (4.9)$$

$$B_{lo} = Q(\mathbf{p}_{k-1}^k, \mathbf{n}) \frac{\partial \|\mathbf{p}_k^{k+1} - \mathbf{v}_k\|}{\partial \mathbf{p}_{k-1}^k} \frac{\partial \mathbf{p}_{k-1}^k}{\partial \mathbf{v}_k}, \quad \text{for } \mathbf{p}_{k-1}^k \neq \mathbf{v}_{k-1} \quad (4.10)$$

where we have the $B_{up} = 0$ when \mathbf{v}_{k+1} and \mathbf{v}_k is in the same range bin or $B_{lo} = 0$ when \mathbf{v}_{k+1} and \mathbf{v}_k are in the same range bin as the dependency of the domain of integration with respect to \mathbf{v}_k drops in these cases. Similarly, in Eq. 4.8, one or both of integral terms drop if the edge corresponding to the expression is not visible.

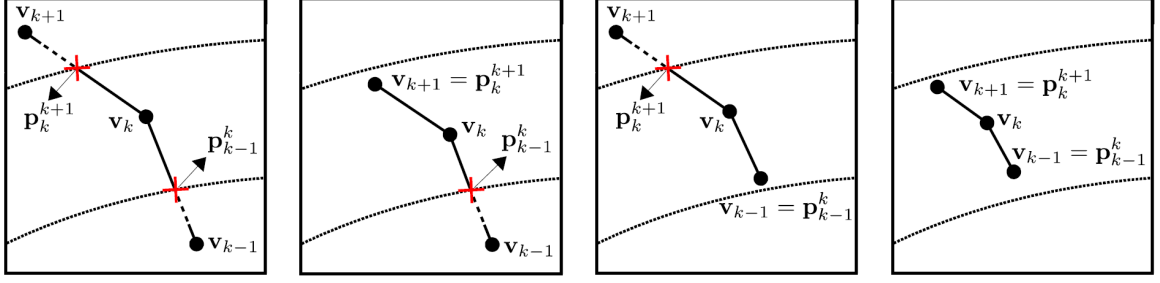


Figure 4.5: Possible placements of vertices with respect to a range bin when the vertex v_k is included in the i^{th} range bin.

For the curve evolution, we consider star shaped objects where we have a unique angular position for every point on the shape. This allows us to parameterize the shape with a polar representation where we will constraint the evolution of the vertices to stay on the same angular position they lie on. Thus, we will use the radii of the vertices as our the parameter set for the shape evolution. To this end, our gradient vector consists of the component of the the derivative of energy functional with respect to the vertex coordinates in the direction of angular position of the vector. For k^{th} vertex, our gradient component becomes:

$$\frac{\partial E}{\partial \phi_k} = \left\langle \frac{\partial E}{\partial \mathbf{v}_k}, \frac{\mathbf{v}_k}{\|\mathbf{v}_k\|} \right\rangle. \quad (4.11)$$

We can then update the shape by perturbing it in the negative gradient direction and expect the energy functional take smaller values at each iteration. However, we will instead use a momentum based gradient descend scheme that is shown to provide robustness for shallow local minima in the energy surface and also faster convergence rates. Our update equations of the shape is then given as:

$$\mathbf{V}_{i+1} = \alpha \mathbf{V}_i - \beta \frac{\partial E}{\partial \Phi} \quad (4.12)$$

$$\Phi_{i+1} = \Phi_i + \mathbf{V}_{i+1} \quad (4.13)$$

where \mathbf{V} is the velocity, Φ is the radii of the vertices we update, α is the momentum

coefficient and β is the step size of the algorithm.

4.1 Results

We conduct simulations for three different scenarios that showcase how our method handles different cases. For each case, we use multiple antenna pairs to collect data. Since we consider a 2D case, we assume the antenna aperture for both the transmitter (TX) and receiver (RX) antennas are of shape of a line segment that are located side by side and have the same surface normal. The common parameter set we use for all simulations are given in the Table 4.1. We have the antenna gain terms as:

Table 4.1: Shape, Signal and RX/TX parameters

Number of Antennas	20
Aperture Length	2mm
Offset between TX and RX	6mm
Center Frequency of the Carrier Signal	79 GHz
Bandwidth	4 GHz
Chirp Rate	$4 \times 10^{13} \text{ Hz/sec}$
Number of Range bins	50 sec
Minimum/Maximum Range	2 meter / 21 meter
Number of Control Points of Evolving Shape	120

$$G' = A_{x'} \sqrt{\cos(\theta')} \operatorname{sinc} \left(\frac{f A_{x'} \sin(\theta')}{c} \right) \quad (4.14)$$

$$G = A_x \sqrt{\cos(\theta)} \operatorname{sinc} \left(\frac{f A_x \sin(\theta)}{c} \right) \quad (4.15)$$

where A_x and $A_{x'}$ are the aperture lengths for transmitter and receiver antennas which are the same in our case. θ and θ' are the angle of the transmitted and received rays from the antenna surface normal and the c is the speed of the light. Using the parameters in Table 4.1, we compute the antenna gain pattern and the gain value as a function of the angle that is illustrated in Figure 4.6. For our simulations, we use three different shapes configuration (initial object and the actual object). For each configuration, we use two different antenna placements. First, we use a circular pattern of antennas where each antenna pair looks to-

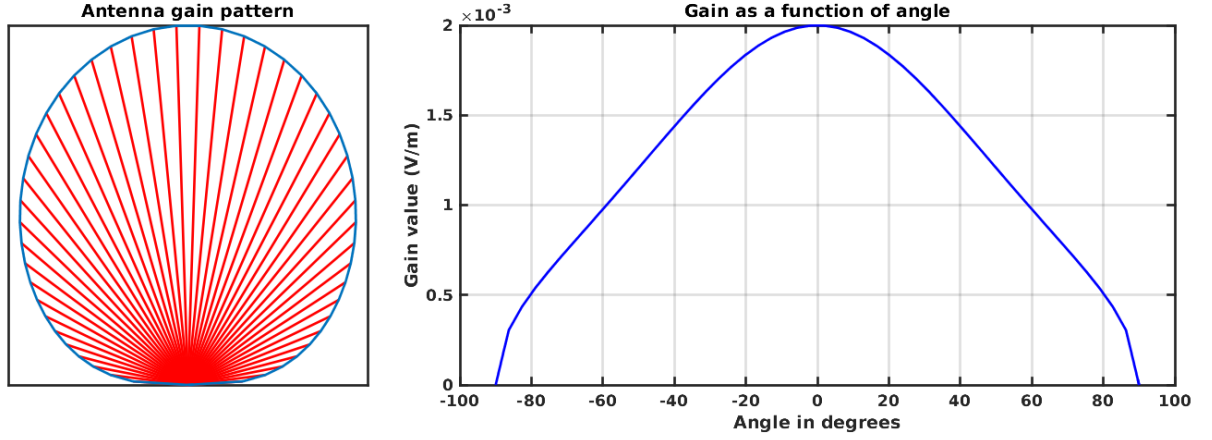


Figure 4.6: Antenna gain pattern illustrated on the left and as a polar plot on the left and the gain value as a function of angle shown on the right.

ward the center of the pattern so that each subsection of the shape profile is visible to at least one antenna pair. In our second case, we use an antenna configuration where the antennas are placed on a linear path which is a more common case for real world applications. For each case, we start with a large regularization coefficient which is manually decreased during the evolution by which we are able to capture the fine details of the geometry which would not be possible with a strong regularizer. Our shape configurations are given in the Table 4.2. For each case, small subplots at the top of the plot show the evolution of the

Table 4.2: Three shape configurations we consider

Name	Actual Shape	Initial Shape
Case-1	Triangle	Hexagon
Case-2	Quadrilateral	Hexagon
Case-3	Free form	Circle

shape model from left to right. Under these subplots, on the left, we have the actual shape and the evolving shape shown together the first iteration with the transmitter and receiver antennas around them. On the right, we again have the actual and evolving shapes at the last iteration. We also provide the optimization parameters as a function of the iteration index for each case.

4.1.1 Circular antenna placement.

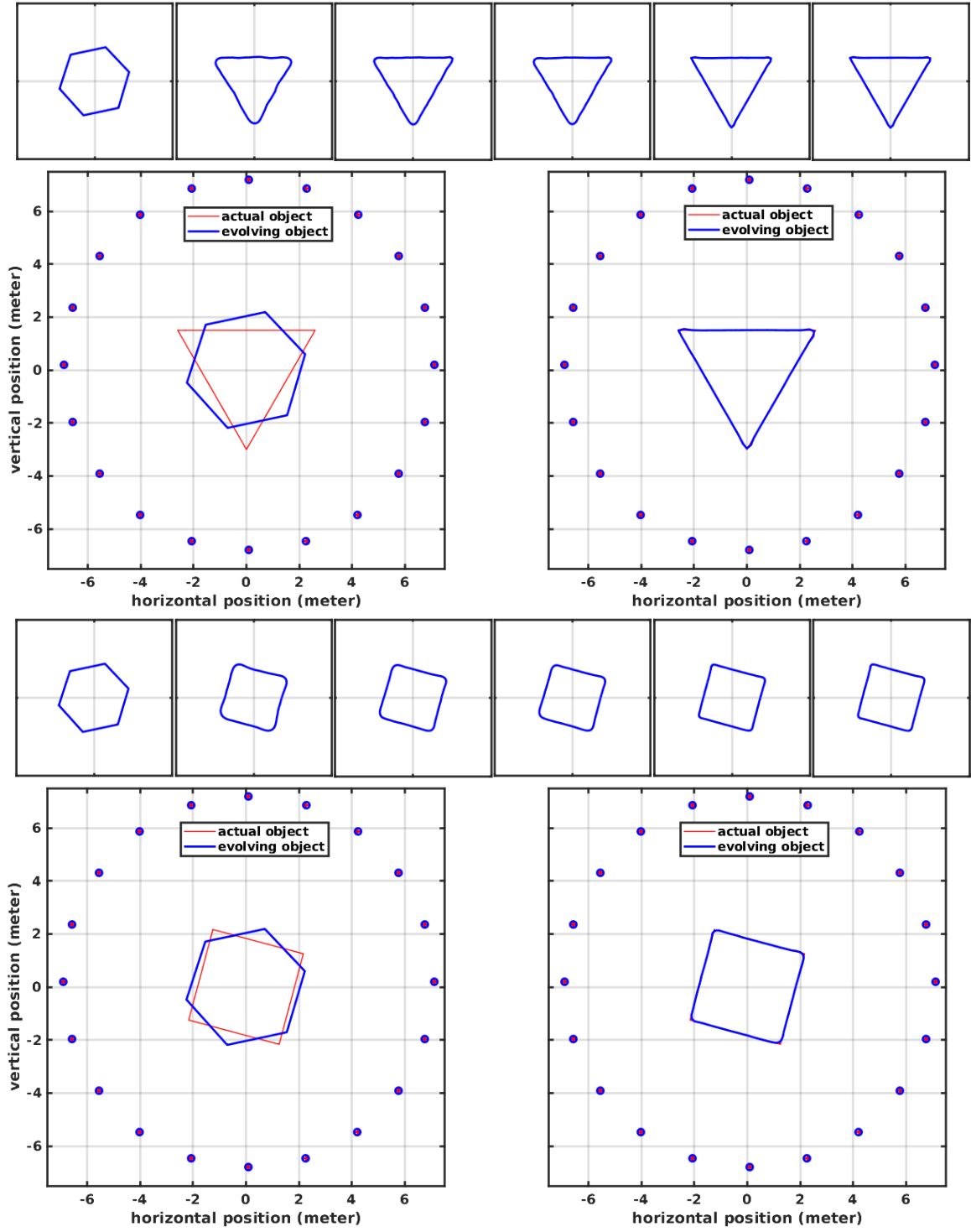


Figure 4.7: Shape evolution of Case-1 and Case-2 for a circular antenna pattern.

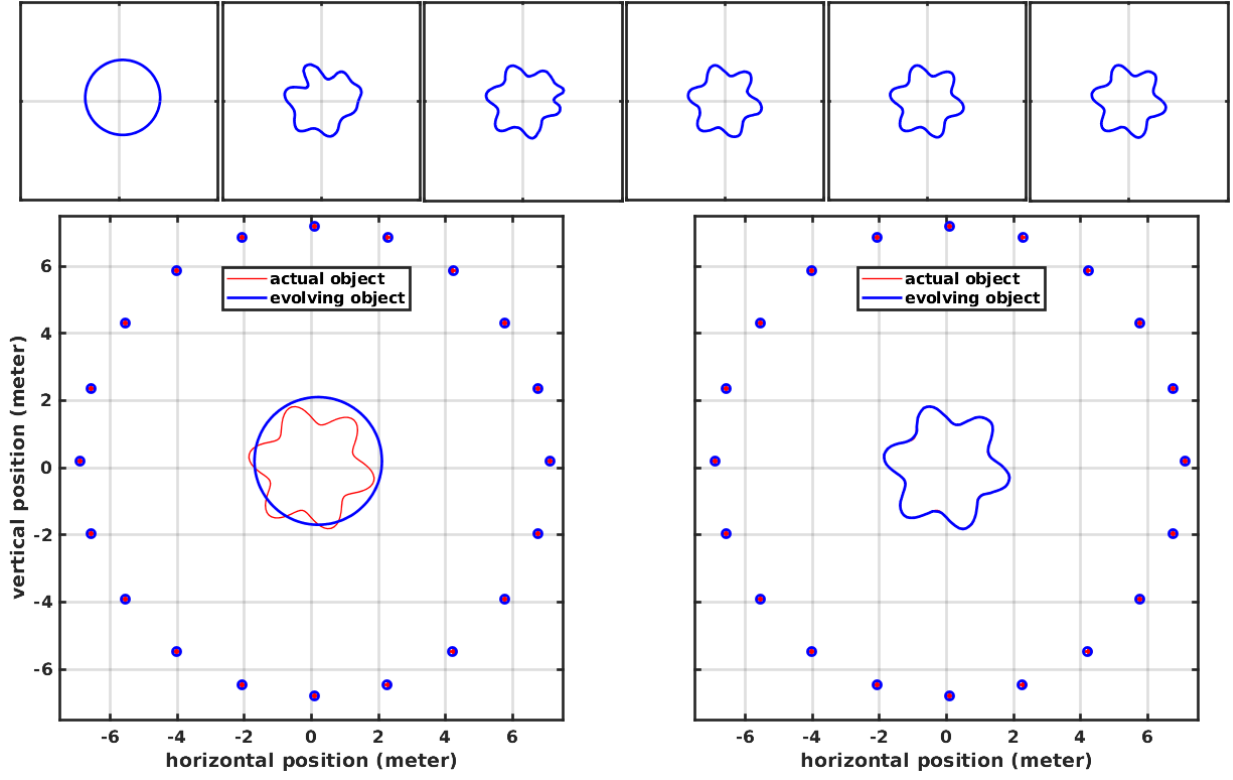


Figure 4.8: Shape evolution of Case-3 for a circular antenna pattern

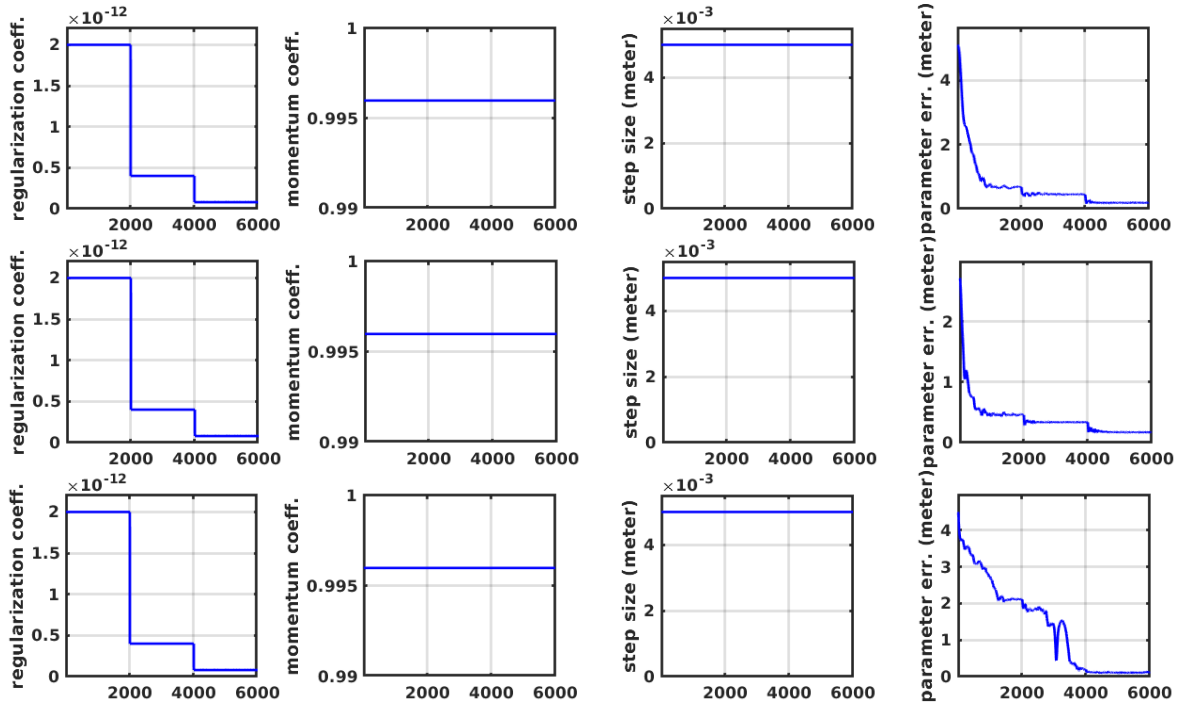


Figure 4.9: Evolution plots of Case-1, Case-2, Case-3 for a circular antenna pattern

4.1.2 Linear antenna placement.

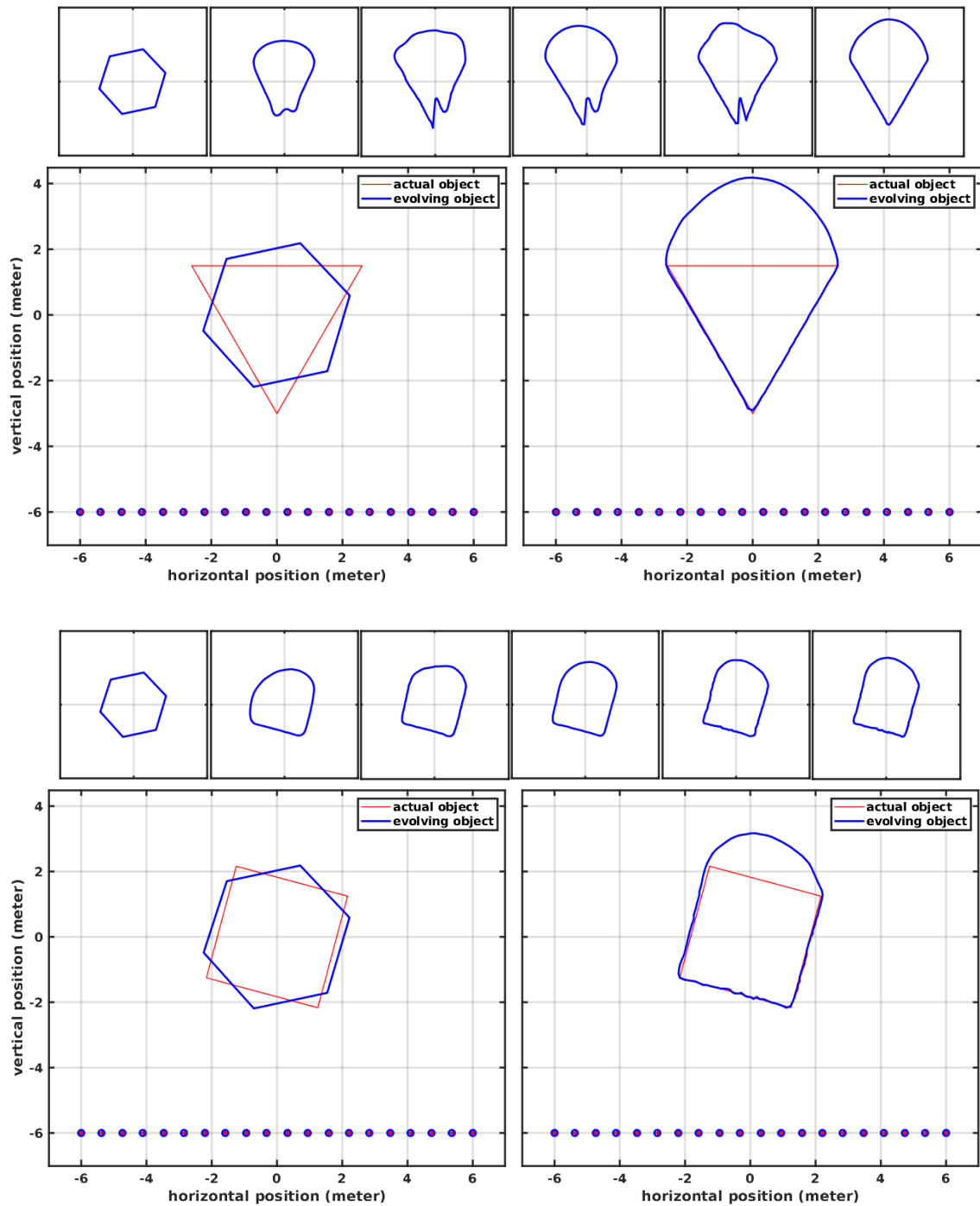


Figure 4.10: Shape evolution of Case-1 and Case-2 for a linear antenna pattern.

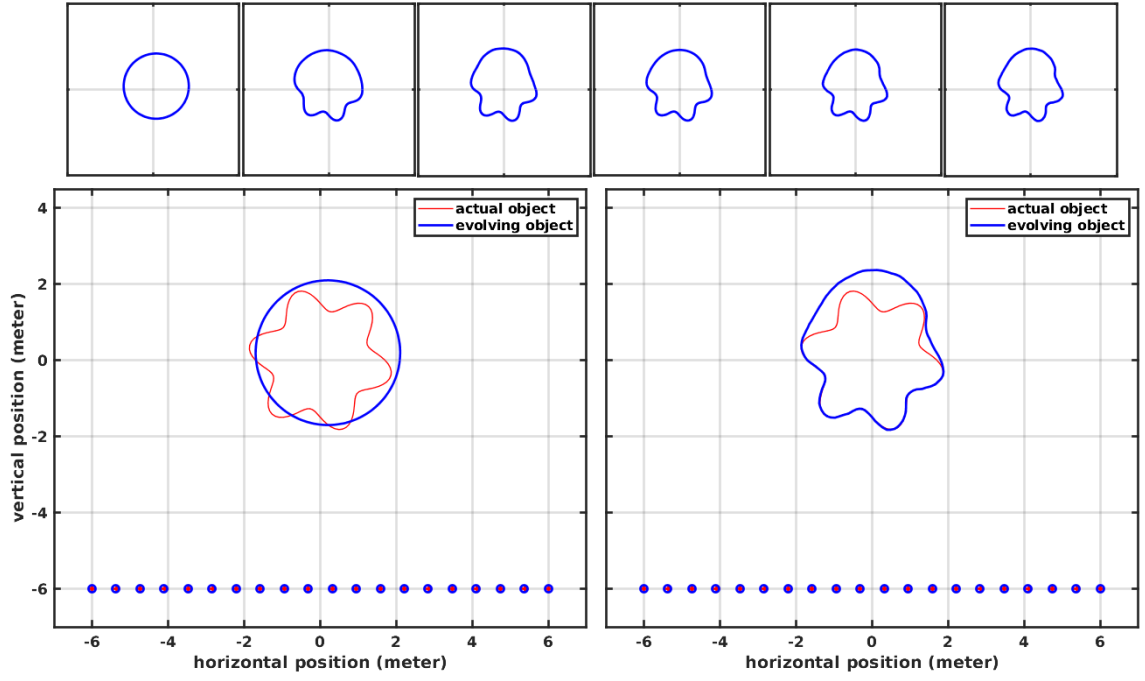


Figure 4.11: Shape evolution of Case-3 for a linear antenna pattern

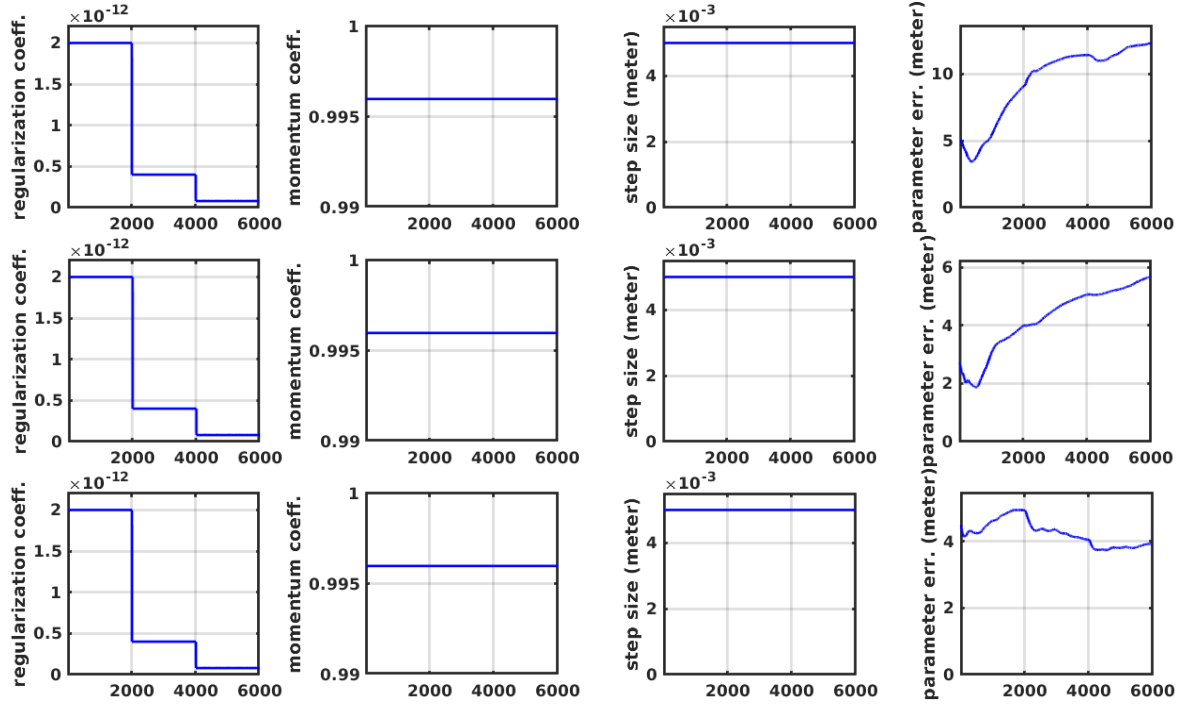


Figure 4.12: Evolution plots of Case-1, Case-2, Case-3 for a linear antenna pattern

4.1.3 Discussions

In Section 4.1.1, we consider a shape surrounded by antennas where each part of the shape profile is visible to at least one antenna pair. As a result, we expect our initial object to fully evolve to the actual shape which is successfully obtained in all three cases, except around the corners of Case-1 and Case-2 where we have a triangle and a quadrilateral (see Fig. 4.7). This is due to the curvature regularizer favoring the smoothness and thus causing the corners of the shape model to be rounded. In the Case-3, we do not observe such behaviour as our shape does not have any corners that are heavily penalized by the regularizer.

In the Section 4.1.2, instead of enclosing the shape with antennas, we use a linear antenna pattern. This causes certain parts of the shape to become invisible where we expect the notion of visibility to come into the picture. Results confirms our prediction since in all cases, our initial shape successfully evolved to the actual shape for the subsections of the object that are visible that is shown in Fig. 4.10 and Fig. 4.11. For the invisible parts, data term of our error functional becomes zero as we do not get any measurements from these parts and the shape evolution is dominated by the regularizer which favors the smoothest possible shape profile. This also causes the increase in the parameter error in all cases that is shown in the third column of Fig. 4.12.

CHAPTER 5

2D LEVEL SET BASED PARAMETRIZATION

In this chapter, we will again consider a tailored version of our algorithm to a 2D case as we did in the previous chapter. However, we will choose to relax two of our previous assumptions related to our shape model which are:

- We have a free form objects instead of star-shaped ones. This will allow us to greatly generalize the set of possible shapes that can be captured in our framework.
- We also will not be limiting ourselves to single shape topology. In the previous chapter, we use an explicit parametrization for our shape (a closed curve) model where it is not possible to capture objects with different topology. In this chapter, we will relax this constraint and will be able to capture shapes that have a different topology from our initial shape model.
- We now use a continuous parametrization (infinite dimensional) for the shape model as opposed to the previous chapter where we model a given shape with a polygonal shape model.

As a result, these relaxations will allow us to be able to handle a much more general set of shapes. This will be possible through the use of an implicit parametrization of the shape where we will employ a level-set parametrization to represent our scene shape.

5.1 Level Set Methods

Level-set method is developed by Osher and Sethian[23]. This method is based on representing the the shape of an object (a curve or a surface) through a function in which the function returns a constant value for the points defining a certain shape. Mathematically

this is expressed as:

$$\phi(\mathbf{x}(t), t) = c \quad (5.1)$$

where we can extract our shapes by finding the points that gives the value of c under the level-set function ϕ for a time instant t . Level sets are depicted in Fig. 5.1. There is

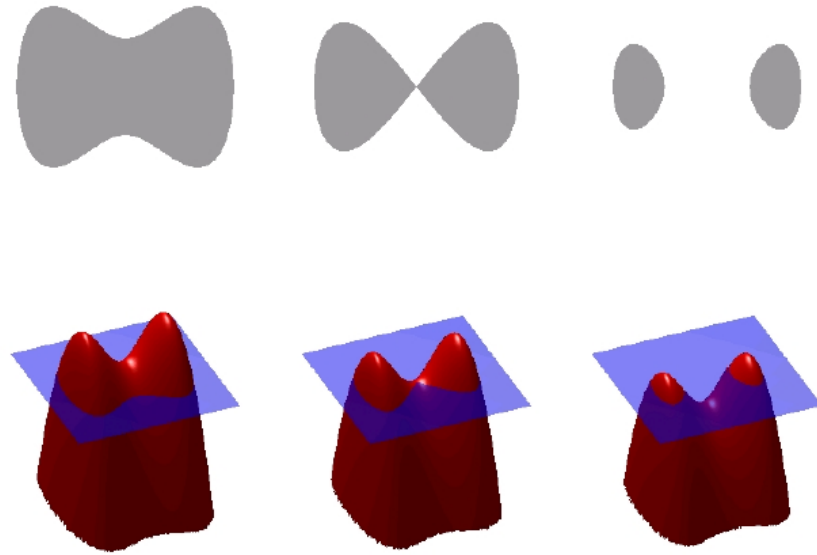


Figure 5.1: Level set curves of a surface at different level values. Image is found online. <https://team.inria.fr/memphis/research/hierarchical-cartesian-schemes-for-pdes/level-set/>

also parameter t in the function by which the evolution of the level set function is possible. Such an embedding of the shape allows us to implicitly parameterize any kind of shape on a fixed Cartesian grid where the shape is evolved through updating the value ϕ on the grid. It should be noted that this is fundamentally different than the method we use in the previous chapter where we adopt a particle based approach where our main interest is to evolve the shape through updating the coordinates of a discrete number of particles (control points). These two approaches are analogous to the different ways of solving computational fluid dynamics problems. In one approach, we set a set of initial particles (parcels) and express

the equations in such a way that we follow how these individual particles move in time. This is known as the Lagrangian specification of fluid motion. Other way is to define a fixed Cartesian grid where instead of following individual particles, we are interested in what goes inside and what goes out from grid points. This is known as the Eulerian specification of the flow. As a result, the approach we use in the previous chapter corresponds to a Lagrangian approach and the one we will use in this chapter corresponds to an Eulerian approach.

Since level-set methods are often used to model some phenomena propagating in the space with time, it is insightful to look at its gradient. Taking derivative of both sides of the equation in Eq. 5.1 with respect to time, we have:

$$\frac{d\phi}{dt} = \frac{\partial\phi}{\partial\mathbf{x}} \frac{\partial\mathbf{x}}{\partial t} + \frac{\partial\phi}{\partial t} = 0. \quad (5.2)$$

We know that the first partial derivative term (derivative with respect to the point coordinate) is the gradient of level-set function. As a result, we can write:

$$\phi_t = -\nabla\phi \frac{d\mathbf{x}}{dt}. \quad (5.3)$$

An important insight to realize is that that by the very definition of the level-set function, perturbation of \mathbf{x} along the tangent of the curve does not change the level-set function value as such perturbations keep the the point on the same level-set where ϕ is invariant. Gradient being the direction in which the function value increases most dramatically, we can conclude that the gradient of level-set function is in the normal direction as any other direction corresponds to a lesser change in the function value (given the size of the perturbation is

constant for all directions). From this observation, we have

$$\phi_t = -\|\nabla\phi\|\mathbf{N}^T \frac{d\mathbf{x}}{dt} \quad (5.4)$$

$$= -F\|\nabla\phi\| \quad (5.5)$$

for $F = \mathbf{N}^T \mathbf{x}_t$ which is the speed of the point in the normal direction (also called the force). F needs to be designed according to the phenomena that is being modelled. We will derive our expressions for F from our cost functional such that the mismatch between the computed radar measurement (computed for our evolving shape) and the actual one decreases with time. Another contribution to F will be provided by the regularizer we use so that we can keep the shape smooth during the evolution. After designing F , our final expression is a partial differential equation to be solved. Depending on the structure of F , our equations, most of the time, yield a highly nonlinear PDE that requires a numerical solution. Thus, we need to discretize this PDE so that we have algebraic expressions by which the level-set function is updated at each time step. Advantages of using a level-set representation are numerous. First, as a result of the implicit parametrization we use in the evolution equation (Eq. 5.5) is given in terms of fully geometric quantities that are independent from any specific parametrization. Working on a fixed Cartesian grid and evolving the interface through updating the the level-set function values on the grid, we naturally avoid a lot of implications that can be present with particle tracking based approaches (our approach in the previous chapter). These can be listed as:

- In particle tracking based approaches, we can have the spacing between the particles become very tight or wide along iterations that causes certain part of the shape model to be over sampled (tight) or under sampled (wide). Because of this considerations, in the last chapter, we limit the motion of each particle to stay on the same angular direction so that we make sure all of the vertex coordinates have equal angular spacing.

- Particle tracking based approaches can also suffer from self intersection problem. Since we update the locations of our shape model at each step, it is possible some part of the shape to entangle with other parts of itself. Preventing such behaviour has been studied and possible for 2D though it requires a careful effort to handle all possible exceptions and edge cases. The reason we only consider the star-shaped objects in the last chapter is to avoid these implication. With a level-set based method, we also bypass this issue as we formulate our evolution as updating the values of the level-set function on a fixed Cartesian grid.
- The most important consideration to avoid a particle tracking based approach is to be able to naturally handle the topological changes which was one of the biggest motivations during the development of level-set based methods. For example, in the last chapter, we choose our shape model to be a closed curve. For a case where the actual scene consists of two distinct objects, it is not possible to capture the geometry of the actual scene as our evolving shape model and the actual scene shape have different topology. For such a case, a level-set based representation of our shape (interface) can greatly helps as these methods allow the topological changes in the shape interface. A depiction of a topological change is depicted in Fig. 5.2.

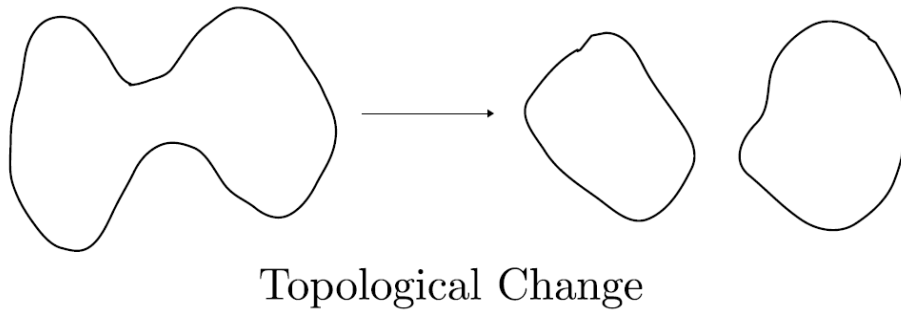


Figure 5.2: With level-set methods, it is possible to start with an object on the left and reach to the one on the right.

As a result, a level-set based representation of our shape allows us to handle many more different shapes without having much prior information about the number of the objects

in the scene or their topology. Another important aspect of such representation is that it requires relatively less effort to extend it to 3D applications.

5.1.1 Narrow Band Methods

It should be noted that in the direct implementation of the level-set method, we track not only the level-set we are interested in, but all of them. This requires a significantly more computational power when compared to our case in the previous chapter where we only update a set of particles in space. As a result, in 2D, update process requires $O(kl)$ for level-set implementation where k and l are the size of our fixed Cartesian grids in vertical and horizontal axes (since at this point, PDE is discretized) whereas the algorithm we use in the previous chapter requires update that has a complexity of $O(n)$ where n is the number of particles we use (vertices of the polygonal shape model). However, a computationally more efficient algorithms are available [24] by which this complexity can be reduced to the same level as the particle based methods. This method relies on the not computing the level set function values unless they are nor necessary. To this end, we take level-set that corresponds to our evolving interface (shape model) and choose a narrow band around the interface with a fixed thickness on which the level-set function is computed. Remaining grid elements are only computed when they are within the chosen threshold so that we can decrease the number of computations significantly. A level set curve and the narrow band around it is depicted in Fig. 5.3. For a 3D case where we have a 3D grid and a 2D level surface, complexity reduces from $O(n^3)$ to $O(n^2)$ using narrow band methods.

5.2 Our Approach

We will use the same forward model to what we use in the last chapter. To summarize the steps of our forward model:

- For each antenna pair, we measure we transmit an LFM waveform

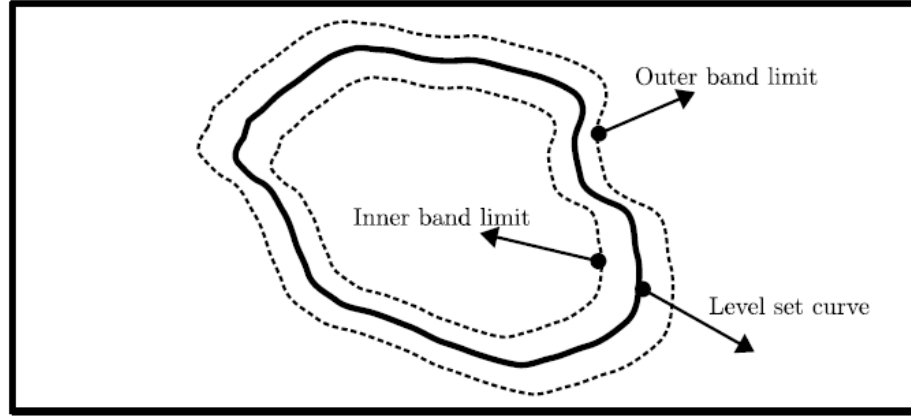


Figure 5.3: A level set curve and the narrow band around the set. The level set function is updated only within the band boundaries.

- The echo measured by the receiver antenna is deramped to yield a signal where the strength of each frequency components corresponds to the resultant reflection of a unique range value.
- Through a preprocessing similar to what we use in the previous chapter, we can extract the average electric field density values of a set of discrete range bins (range decomposition).
- These average electric field density values will form the basis of our inversion algorithm as they will be used to define the cost functional we will use.

5.3 Inversion

As we do with our forward model, we can also use the same energy functional for this section. We will shortly see that this is not a good idea due to the complications that can arise because of the continuous representation of our shape model. However, to motivate the final form of the cost functional, it is necessary to understand what we have to go through if we use some other cost functional. Let our cost functional given be (without the term we add in the last chapter to compensate for the decay in the electric field with the

range):

$$E = \frac{1}{2N_A N_B} \sum_{i=1}^{N_A} \sum_{j=1}^{N_B} \left[\left(\frac{1}{\Delta D} \int_{S_i^j \cap V_j \cap \mathbf{e}} Q(\mathbf{x}, \mathbf{n}) ds \right) - H_j^{i,m} \right]^2 \quad (5.6)$$

$$= \frac{1}{2N_A N_B} \sum_{i=1}^{N_A} \sum_{j=1}^{N_B} [H_j^i - H_j^{i,m}]^2 \quad (5.7)$$

where N_A and N_B are the number of antennas and number of range bins for each range bin. We normalize the cost functional value with N_A and N_B so that increasing the number of antennas do not require changing how strong the regularization is with respect to the number of antennas or range bins. H_j^i denotes the average electric field density computed from the forward model using the evolving shape at the j^{th} frequency bin of i^{th} antenna. Similarly, $H_j^{i,m}$ is the average electric field density value that is obtained from the radar measurements.

We parameterize our shape as a level set function where all of the points on our shape model is evolved iteratively so that eventually we expect our shape to converge to the actual shape. Since we use a continuous parametrization for our shape model, minimizing the cost functional requires to solve a variational optimization problem where gradient is infinite dimensional. In the next section, we derive the gradient flow equations for our cost functional.

5.3.1 Gradient Derivation

First, it should be observed that in Eq. 5.7, our cost function is sum of geometric integrals (since we use a geometric measure) over our shape model which is analogous to the family of methods popular in image processing that are called geometric active contours [25][26]. These methods are often used to segment features from images by evolving a given curve. Depending on a weight value designed on the image (in terms of intensities, gradient of the intensities, etc.) that can be computed at any point in the image, the problem is to evolve a curve (or a surface in 3D) such that the integration of this weight over the curve is

minimized. A classic application for the use of geodesic active contours is to segment the edge boundaries of an object in an image. In such case (ignoring regularization), we need to minimize the following expression:

$$E_{image} = \int g(\mathbf{x}) ds \quad (5.8)$$

where $g(\mathbf{x})$ is typically (for edge segmentation) given as:

$$g(\mathbf{x}) = \frac{1}{1 + \|\nabla I(x)\|} \quad (5.9)$$

where the $\nabla I(x)$ is the image gradient at the point \mathbf{x} . Such a weight function $g(\mathbf{x})$ takes small value at the edge points since gradient will be large so when we minimize E_{image} , we expect our shape model to converge to the edge boundaries of the object we are looking for. Our problem is not an imaging problem but this idea is also applicable to our problem as our cost functional E is also defined as a geometric integral over the shape boundaries. It should be noted that in its current version, for any point in the space, our integrand $Q(\mathbf{x}, \mathbf{n})$ is a function of both point coordinates \mathbf{x} and \mathbf{n} . Since we use a continuous representation for our shape model where there are infinitely many number of parameters, we need to formulate our problem as a variational optimization problem in which case gradient also becomes infinite dimensional. To compute the gradient, we will find it useful to compute the derivative of our cost functional with respect an evolution parameter (τ) that can be thought as time. Taking the derivative we have:

$$\frac{\partial E}{\partial \tau} = \frac{1}{N_A N_B} \sum_{i=1}^{N_A} \sum_{j=1}^{N_B} [H_j^i - H_j^{i,m}] \frac{\partial H_j^i}{\partial \tau}. \quad (5.10)$$

where we obtain a residual term of $[H_j^i - H_j^{i,m}]$ and the partial derivative of H_j^i which is given as:

$$\frac{\partial H_j^i}{\partial \tau} = \frac{\partial}{\partial \tau} \left(\frac{1}{\Delta D} \int_{S_i^j \cap V_j \cap \mathbf{e}} Q(\mathbf{x}, \mathbf{n}) ds \right). \quad (5.11)$$

Looking carefully to the integral, we see that it is defined over some part of our shape that lies in a specific range bin. Since our integral is over the arc length parameter ds and (as a result of this) borders of integration is not fixed, a perturbation to the curve also effects the borders of integration. Thus, we need to take the boundary terms into account when we compute the derivative as the integrand does not necessarily take zero values at the boundaries. Ideally, for simplicity of the final expression and to have a numerically more stable algorithm, we want these boundary terms to vanish at the boundary points. In what follows, we develop a new cost functional in which this is accomplished.

Our strategy will be based on a weighting scheme where, within a range bin, we will compute the average electric field density value of that bin by a continuous weight function. In explanation, for j^{th} range bin of a given antenna that covers the range interval of $[D_l^j, D_k^j]$ with a mid range value of D_{mid}^j , we weight the electric field density values in between with the following weight function:

$$W_j(D) = \begin{cases} \frac{D - D_l^j}{D_{mid}^j - D_l^j} & D \leq D_{mid} \\ 1 - \frac{D - D_{mid}^j}{D_k^j - D_{mid}^j} & D > D_{mid} \end{cases}. \quad (5.12)$$

This is a triangle shape weight function that takes zero value at the range boundaries and takes the value of 1 in the middle. By using such weighting, we ensure that our integrand value vanishes at the range boundaries (due to zero weighting) by which boundary terms we previously mention can be eliminated. However, employing such weighting also causes our shape evolution to be insensitive to the regions of our shape that lies close to the range

boundaries. To solve this issue, we propose to create a new range decomposition of scene by shifting each range bin by half the range size. Using the same weighting scheme on the shifted range bins, this time we have maximum weight value of 1 at the regions that are weighted as zero in our original range decomposition. Formulating the cost functional in terms the weighted averages of these two range decomposition, we give the same weight value to the contributions coming from each range value. Our weighting scheme for two different range decomposition and how they add up to an equal weighting at the end is depicted in the Fig. 5.4 As a result, we see that by employing the same weighting scheme

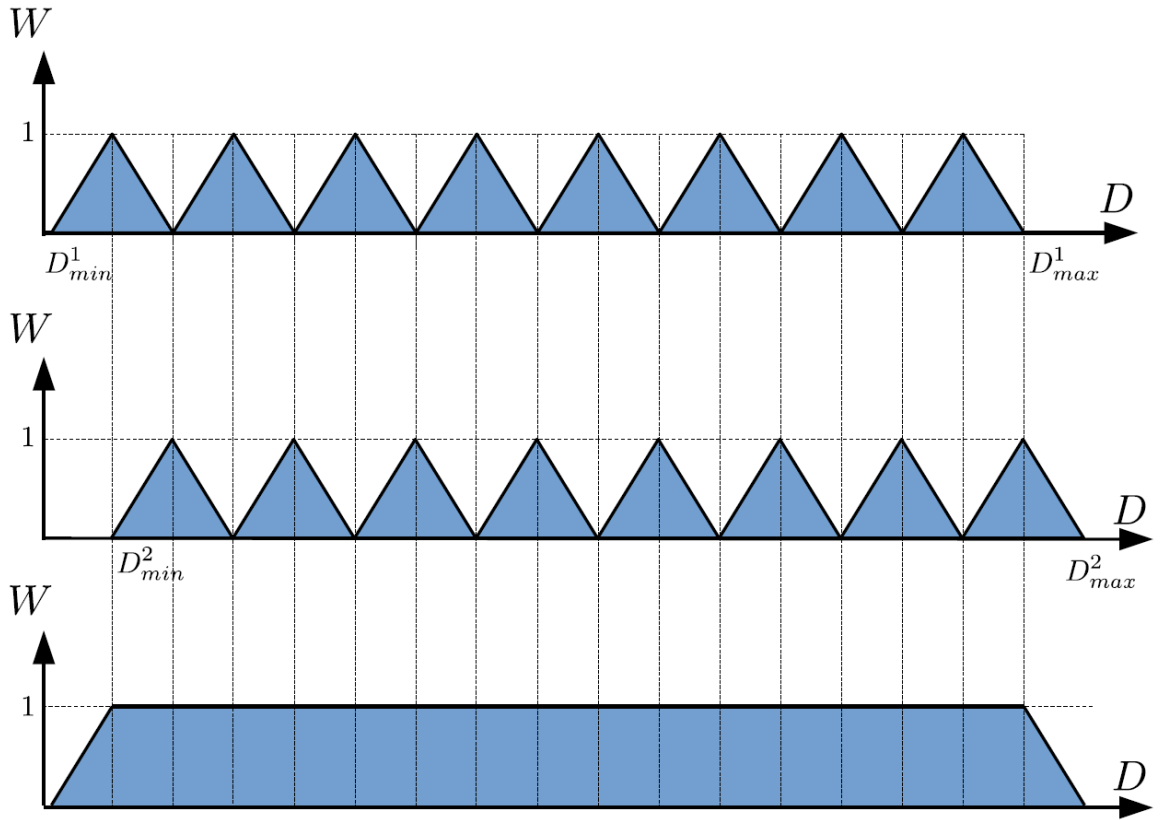


Figure 5.4: Our weighting scheme as a function of range. Top shows the weighting used for the first range decomposition. The bottom shows the same for the second frequency spectrum. At the bottom, combined weight is shown.

for two range decomposition where one is shifted by half a bin size, we can (in some sense) give equal weighting to each range value in our cost functional as our cost functional will be sum of these two weighted averages. Using these two terms, our new cost functional

now becomes:

$$E = \frac{1}{2N_A N_B} \sum_{i=1}^{N_A} \sum_{j=1}^{N_B} [H_j^{i,w,1} - H_j^{i,w,1,m}]^2 + [H_j^{i,w,2} - H_j^{i,w,2,m}]^2 \quad (5.13)$$

where $H_j^{i,w,1}$ denotes the weighted average electric field density value of j^{th} frequency bin of the i^{th} antenna for the first range decomposition and $H_j^{i,w,2}$ is the same for the second (shifted) range decomposition. The superscript m added to these term express the fact that these are the weighted average electric field densities that are obtained from the radar measurements (from the actual shape we are trying to estimate). Using weighting, $H_j^{i,w,1}$ and $H_j^{i,w,2}$ are given as:

$$H_j^{i,w,1} = \frac{1}{\Delta D} \int_{S_i^j \cap V_j \cap \mathbf{e}} w_1(\mathbf{x}) Q(\mathbf{x}, \mathbf{n}) ds \quad (5.14)$$

$$H_j^{i,w,2} = \frac{1}{\Delta D} \int_{S_i^j \cap V_j \cap \mathbf{e}} w_2(\mathbf{x}) Q(\mathbf{x}, \mathbf{n}) ds \quad (5.15)$$

where w_1 and w_2 are the weight functions for first and second range decompositions. For a given \mathbf{x} value, we have the equality:

$$w_1(\mathbf{x}) + w_2(\mathbf{x}) = 1. \quad (5.16)$$

Using the cost functional in Eq.5.13, we get rid of the boundary terms when taking derivative of the cost functional and also give equal weights to each range values at the same time by using the shifted range decomposition. However, an important question also arises with this new cost function. We know that we can do any kind of weighting over the range values in our forward model since we have our continuous shape model (evolving shape model) on which we can compute weighted average of electric field density for each range. However, we know that the average electric field densities obtained from the preprocessing of the radar signal (see the discussion in Chapter 3) naturally assumes equal weighting as

the our weighting does not corresponds to anything physical. As a result, our question is if it is possible to estimate $H_j^{i,w,1,a}$ and $H_j^{i,w,2,a}$ from a radar measurement. Luckily, there is a way to do such estimation. In chapter 3, we develop a method to extract range decomposition of our deramped signal through a preprocessing. Assuming a $50\mu sec$ pulse and a modest sampling frequency of $10MHz$, the number of time samples we have becomes 50000. The number of range bins we use in the previous chapter on the other hand is chosen as 50. This means our system of equations is highly over determined where we estimate 50 unknowns from 50000 measurements. This implies that there should be room to choose the number of range bins much larger that they currently are. This fact can be exploited to estimate $H_j^{i,w,1,a}$ and $H_j^{i,w,2,a}$.

During the preprocessing, we can always choose the number of range bins much more than what we will actually use and then can use this densely sampled range decomposition to estimate the weighted average of electric field densities over a coarsely sampled range decomposition that is used in our cost functional. This is depicted in Fig. 5.5. $H_j^{i,w,1,a}$ is than given as:

$$H_j^{i,w,1,a} = \frac{1}{N_{fb}} \sum_{i=1}^{N_{fb}} W_j(D_i) H_j^{i,w,1,f,a}(D_i) \quad (5.17)$$

N_{fb} is the number of fine range bins in a coarse range bin, W_j is defined in Eq. 5.12, D_i is the mid range value of fine range bin and $H_j^{i,w,1,f,a}$ is the average electric field density of the corresponding fine range bin that is obtained through preprocessing. As a result, we now can estimate the weighted average electric field densities from the radar measurements. **Note that we again skip such procedure in our implementation (as we did in preprocessing) and compute the weighted spectrum directly using the geometry as our results are all in simulation where we have the geometry information of both evolving and actual shape. Therefore, we directly compute the weighted integral expression given in 5.14 and 5.15 in our implementation to obtain the weighted spectrums.**

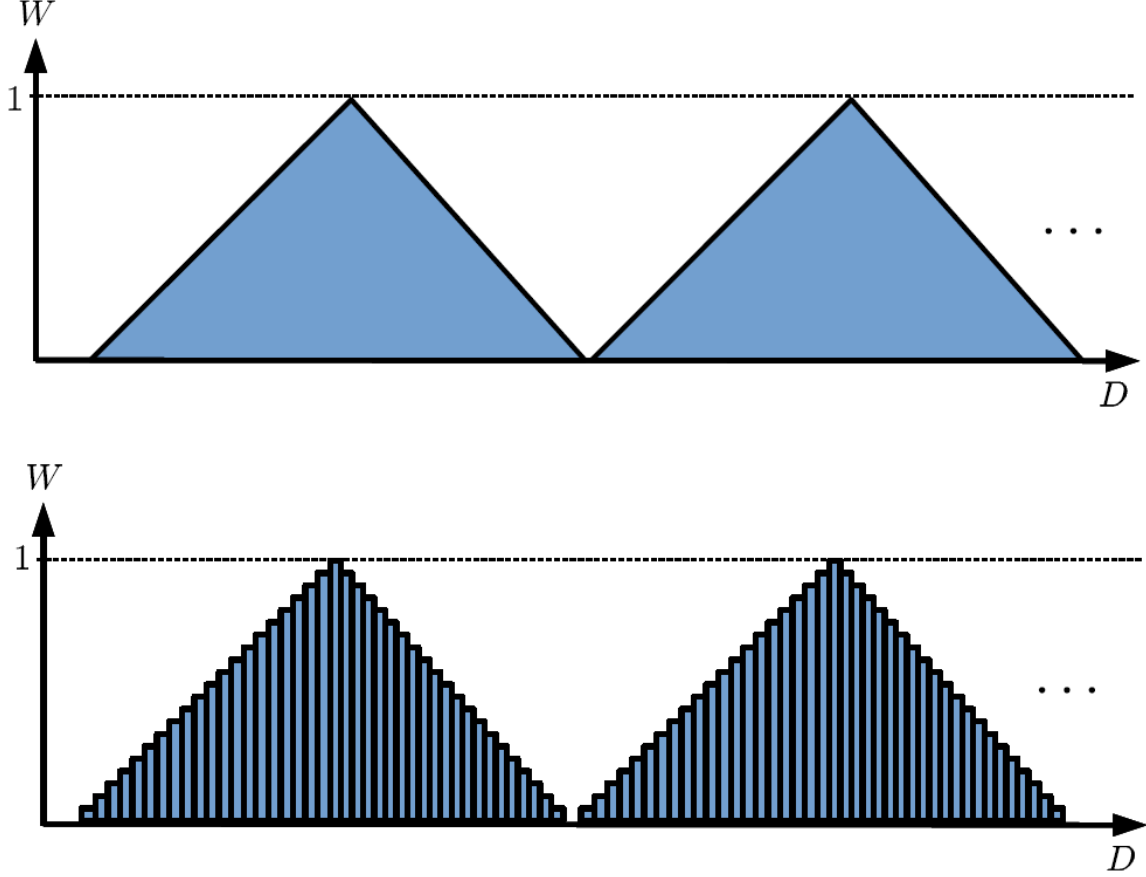


Figure 5.5: The depiction of how we can obtain our weighted average electric field density values from the radar signal. Our strategy is to obtain a dense range decomposition (bottom) from the preprocessing and then obtain the weighted range decomposition of our coarse weighted spectrum (top).

Going back to the gradient, for simplicity, we will only compute the gradient for only one term (first range decomposition) of our cost functional as the expression will also be the same for the other. Taking the derivative, we obtain:

$$\frac{\partial E_1}{\partial \tau} = \frac{\partial}{\partial \tau} \left(\frac{1}{2N_A N_B} \sum_{i=1}^{N_A} \sum_{j=1}^{N_B} [H_j^{i,w,1} - H_j^{i,w,1,m}]^2 \right) \quad (5.18)$$

$$= \frac{\partial}{\partial \tau} \left(\underbrace{\frac{1}{N_A N_B} \sum_{i=1}^{N_A} \sum_{j=1}^{N_B} [H_j^{i,w,1} - H_j^{i,w,1,m}]}_{\text{residual term}} \right) \frac{\partial H_j^{i,w,1}}{\partial \tau}. \quad (5.19)$$

Note that the only term that changes with the evolving shape is $H_j^{i,w,1}$. Expanding this

term, we obtain:

$$\frac{\partial H_j^{i,w,1}}{\partial \tau} = \frac{\partial}{\partial \tau} \left(\frac{1}{\Delta D} \int w_1(\mathbf{x}) Q(\mathbf{x}, \mathbf{n}) \right). \quad (5.20)$$

We will assume collocated antennas for simplicity in which case our expression then becomes:

$$\frac{\partial H_j^{i,w,1}}{\partial \tau} = \frac{\partial}{\partial \tau} \left(\frac{1}{\Delta D} \int w_1(\mathbf{x}) \frac{G^2(-\mathbf{u} \cdot \mathbf{n})}{R^2} ds \right). \quad (5.21)$$

At this point, to compute the gradient of our expression, we will derive a more general formula that computes the gradient for a general integrand ($f(\mathbf{x}, \mathbf{n})$) that has an arbitrary dependence to \mathbf{x} (point) and \mathbf{n} (normal). After algebraic manipulations (see the Appendix B for derivation), we obtain:

$$\frac{\partial}{\partial \tau} \left(\int f(\mathbf{x}, \mathbf{n}) ds \right) = \int \left(\frac{\partial f}{\partial \mathbf{x}} \mathbf{n} - \kappa \frac{\partial f}{\partial \mathbf{n}} \mathbf{n} + \mathbf{t}^\top \frac{\partial^2 f}{\partial \mathbf{n} \partial \mathbf{x}} \mathbf{t} + \kappa \mathbf{t}^\top \frac{\partial^2 f}{\partial \mathbf{n}^2} \mathbf{t} + \kappa f \right) \mathbf{n}^T \mathbf{x}_\tau ds \quad (5.22)$$

for \mathbf{t} is the unit tangent of the curve at point \mathbf{x} and κ is the curvature of the curve at the same point. When we look at the expression carefully, we see that this is an inner product in the geometric L^2 function space where geometric part comes from the fact that this is an integration over the arc length parameter. The derivative expression with respect to τ gives us how much the total integral changes at a given instant on our curve evolution. Interpreting \mathbf{x}_τ as our perturbation direction along the curve, our gradient becomes the rest of the terms in the integrand as this is the definition of gradient. Gradient is a vector that is, when dotted (inner product) with a direction, yields the directional derivative in that

direction. Writing more clearly,

$$\frac{\partial}{\partial \tau} \left(\int f(\mathbf{x}, \mathbf{n}) ds \right) = \langle \nabla_{\mathbf{x}} f, \mathbf{x}_{\tau} \rangle \quad (5.23)$$

$$= \int (\nabla_{\mathbf{x}} f) \mathbf{x}_{\tau} ds. \quad (5.24)$$

We then find our gradient as:

$$\nabla_{\mathbf{x}} f = \int \left(\frac{\partial f}{\partial \mathbf{x}} \mathbf{n} - \kappa \frac{\partial f}{\partial \mathbf{n}} \mathbf{n} + \mathbf{t}^{\top} \frac{\partial^2 f}{\partial \mathbf{n} \partial \mathbf{x}} \mathbf{t} + \kappa \mathbf{t}^{\top} \frac{\partial^2 f}{\partial \mathbf{n}^2} \mathbf{t} + \kappa f \right) \mathbf{n}. \quad (5.25)$$

Plugging our integrand in f , we can compute the gradient of the expression in Eq. 5.21. However, note that there are curvature dependent terms in the gradient expression. For the partial differential equation we will be solving to obtain our shape evolution, curvature dependent terms should be given special attention as a negative factor multiplying these terms can cause the corresponding PDE to be unstable due to an effect known as backward heat flux. We know that both curvature dependent terms in our expression $\kappa \mathbf{t}^{\top} \frac{\partial^2 f}{\partial \mathbf{n}^2} \mathbf{t}$ and $\kappa f \mathbf{n}$ are positive (first being a quadratic term and the second one due to the f being positive by nature in our cost functional). However, it should be noted that this gradient is also multiplied with the residual term in Eq. 5.19 which can be both positive or negative and as a result of this, so does the sign of the curvature dependent terms. Luckily, there is a special case in which these curvature terms vanish and we do not suffer from this issue. It is a well known fact that if can write our expression as in the form of:

$$f(\mathbf{x}, \mathbf{n}) = \mathbf{F}(\mathbf{x}) \cdot \mathbf{n} \quad (5.26)$$

where $\mathbf{F}(\mathbf{x})$ is a vector field over \mathbf{x} , curvature terms cancel each other. Trying to write our

integrand in this form results in:

$$w_1(\mathbf{x})Q(\mathbf{x}, \mathbf{n}) = \underbrace{\left[\frac{w_1 G^2(-\mathbf{u})}{R^2} \right]}_{\mathbf{F}} \cdot \mathbf{n}. \quad (5.27)$$

We can see that our integrand satisfies this condition as our \mathbf{F} does not have extra \mathbf{n} dependency and is only a function of \mathbf{x} . Plugging our integrand $w_1(\mathbf{x})Q(\mathbf{x}, \mathbf{n})$ in Eq. 5.25, our final expression becomes (see the Appendix B for derivation):

$$\begin{aligned} \nabla_{\mathbf{x}}(w_1(\mathbf{x})Q(\mathbf{x}, \mathbf{n})) &= \left[\frac{\partial w_1(\mathbf{x})}{\partial \mathbf{x}} \left(-\frac{G^2}{R^2} \mathbf{u} \right) + w_1(\mathbf{x}) \left(\nabla_{\mathbf{x}} \left[-\frac{G^2}{R^2} \mathbf{u} \right] \right) \right] \mathbf{n} \quad (5.28) \\ &= \left[\frac{\partial w_1(\mathbf{x})}{\partial \mathbf{x}} \left(-\frac{G^2}{R^2} \mathbf{u} \right) + w_1(\mathbf{x}) \left(\frac{G}{R^2} \left[\frac{G}{R} - 2(\mathbf{u} \cdot \mathbf{G}_{\mathbf{x}}) \right] \right) \right] \mathbf{n} \quad (5.29) \end{aligned}$$

where we have:

$$\frac{\partial w_1(\mathbf{x})}{\partial \mathbf{x}} = \frac{\partial w_1}{\partial D} \frac{\partial D}{\partial R} \frac{\partial R}{\partial \mathbf{x}} \quad (5.30)$$

$$= \frac{\partial w_1}{\partial D} 2\mathbf{u} \quad (5.31)$$

and $\frac{\partial w_1}{\partial D}$ is given as:

$$\frac{\partial w_1}{\partial D} = \begin{cases} \frac{2}{\Delta D} & D \leq D_{mid} \\ -\frac{2}{\Delta D} & D > D_{mid} \end{cases}. \quad (5.32)$$

It should be noted that the gradient we derive is in the normal direction (Eq. 5.29) which is compatible with our previous observation where we conclude that gradient must be in the

normal direction. Our gradient for two different range decomposition is given as:

$$\nabla E_1 = \frac{1}{N_A N_B \Delta D} \sum_{i=1}^{N_A} \sum_{j=1}^{N_B} [H_j^{i,w,1} - H_j^{i,w,1,m}] \nabla_{\mathbf{x}} (w_1(\mathbf{x}) Q(\mathbf{x}, \mathbf{n})) \quad (5.33)$$

$$\nabla E_2 = \frac{1}{N_A N_B \Delta D} \sum_{i=1}^{N_A} \sum_{j=1}^{N_B} [H_j^{i,w,2} - H_j^{i,w,2,m}] \nabla_{\mathbf{x}} (w_2(\mathbf{x}) Q(\mathbf{x}, \mathbf{n})) \quad (5.34)$$

where both gradients are in the normal direction. The resultant force value (see Eq. 5.5) that applies to a point on our level set is given as the sum of these gradients:

$$F = \|\nabla E_1 + \nabla E_2\|. \quad (5.35)$$

5.4 Results

In this section, we present our simulation results using the equations derived in the previous chapter. This requires a discretization of the PDE we obtain. It should be noted that we skipped the visibility considerations in the previous section. The reason is that we wanted to consider visibility considerations after discretization as otherwise, derivation of the gradient would be unnecessarily complex. It is much easier to apply visibility after the discretization as this way, we can use the algorithm we develop in the previous chapter. Visibility analysis is applied to our shape estimation as follows:

- Our gradients E_1 and E_2 includes a sum over both range bins and the antennas. Considering the fact that spatial discretization yields a set of line segments, we find which line segments are visible for each antenna.
- As a result, we also have the knowledge of antennas from which a given line segment is visible. Since our force calculation is implemented over a discrete sum over antennas (Eq. 5.33 and Eq. 5.34), we add the force contribution of an antenna to a line segment only if that line segment is visible from the antenna.

Since we previously mentioned there are several advantages of using a level-set based

framework when compared to the approach we use in the previous chapter, we test our algorithm on cases that would be very challenging to handle with our previous approach. To this end, we first want to try our algorithm on an object that is not star-shaped for which our previous method does not work. This is because our approach in the previous chapter uses a polar representation of the shape where we parameterize our shape as a set of radii each of which lies on a different angular location.

We will use the same antenna configurations as we use in the previous chapter where the only difference is that this time, we assume our antennas to be collocated (in the previous chapter, the distance between the transmitter and the receiver was $6mm$). In the previous chapter, we also have a regularizer that penalizes the discrete curvature. In this chapter, instead of using a curvature based regularizer, we choose to use a simpler arc-length penalizer. We also use an accelerated gradient flow which is the continuous analog of the momentum based approach we use in the previous chapter [27]. We choose our initial shape as an ellipse and run simulations which eventually converged to the actual shape. The evolution results are shown in Fig. 5.6. Antennas are placed on a circular trajectory around the object. Another shape configuration we try which would not be possible to handle with the method we develop in previous chapter. This case requires a topological change as we will choose our actual shape as two distinct object whereas our initialization will still be a single ellipse. Since we use the level-set based parametrization, we expect such a topological change is automatically captured. We use the same antenna configuration we use for the crescent-shaped object. Our results are shown in Fig. 5.7.

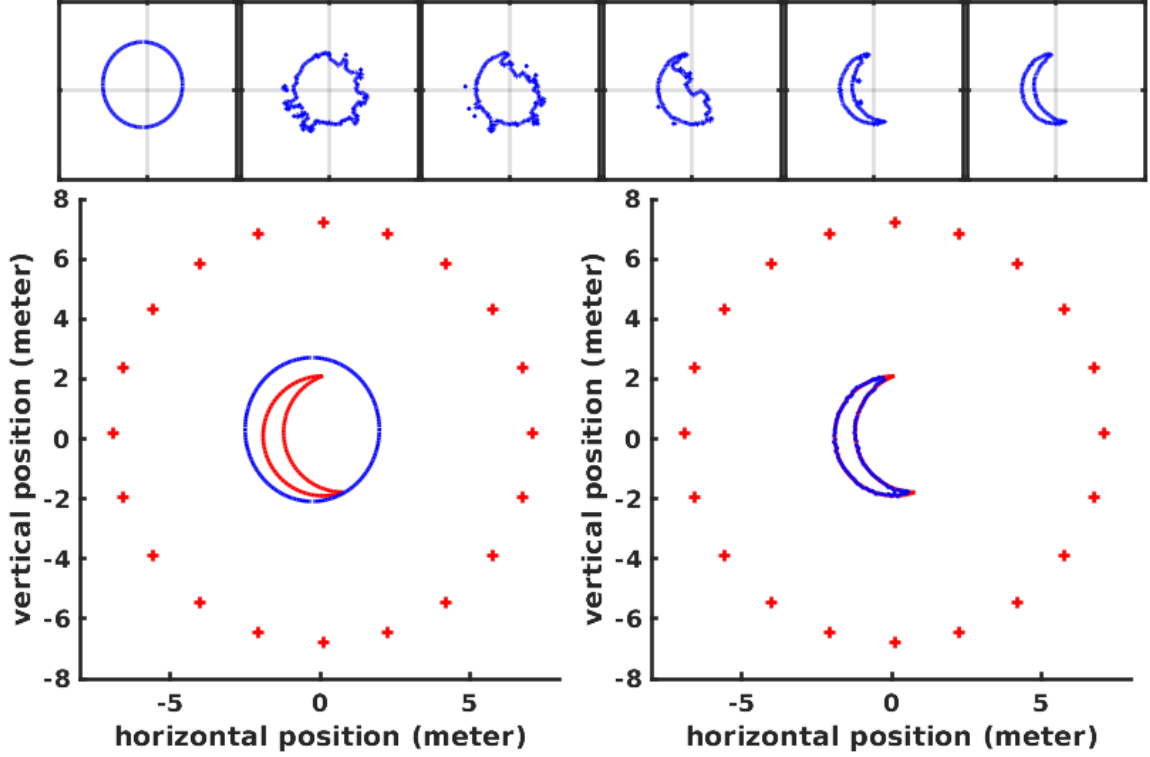


Figure 5.6: Our evolution results for a crescent shaped object. We initialize our shape model as an ellipse which eventually converges to our actual model. At the top, we see the snapshots of our evolving shape from initial shape to its final form. We obtained the convergence in 1500 iterations.

5.4.1 Discussions

We develop a variational technique for shape estimation where we employ a level-set parametrization of the object. After developing the mathematical framework and expressing the optimization problem as a PDE, we are able to capture some complex shapes that would not be possible with our previous method. This has been possible by the use of an implicit parametrization that is called the level-set parametrization where we could relax our previous constraints that are needed to be used with an explicit parametrization.

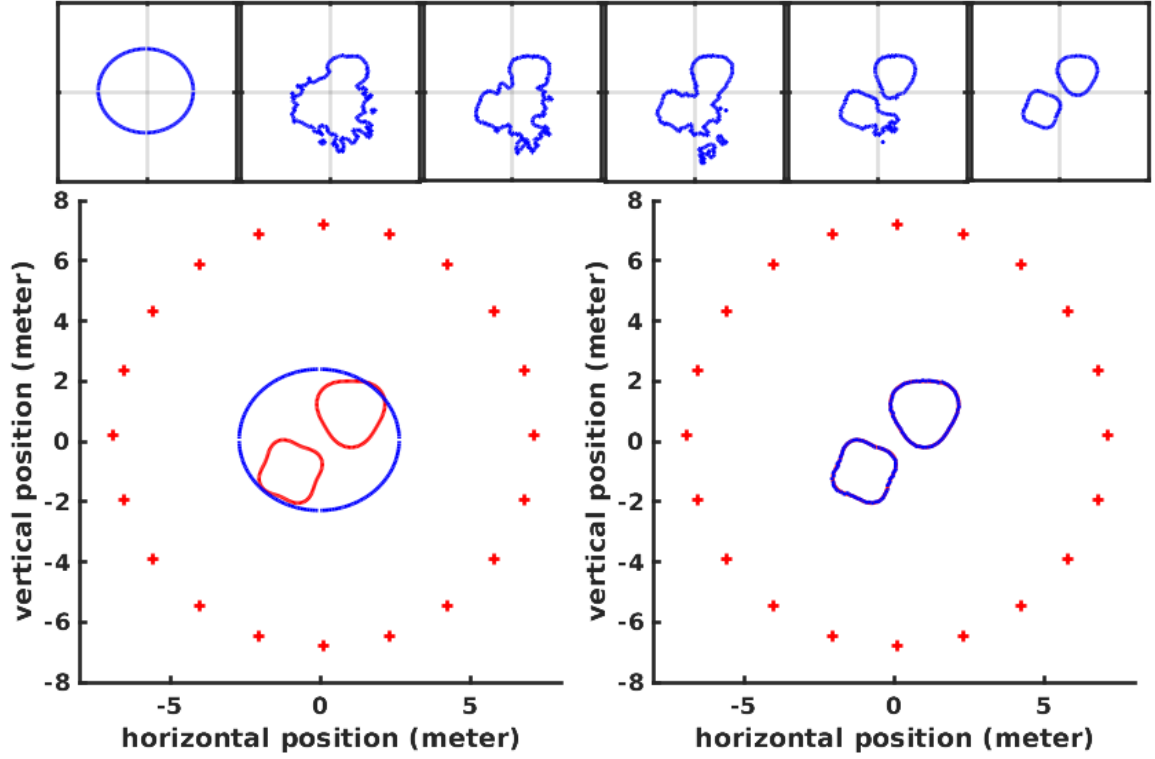


Figure 5.7: Our evolution results for a case where our initialization and the actual shape have different topology. Antennas are placed around the object on a circular pattern. We initialize our shape model as an ellipse which eventually splits up and converges to our actual model that consists of two different curves. At the top, we see the snapshots of our evolving shape from initial shape to its final form. We obtained the convergence in 2500 iterations.

CHAPTER 6

CONCLUSIONS

In this thesis, we propose a geometric method for radar based shape inversion. Our approach differs from the classical radar imaging techniques since it uses an evolving geometric model that tries to capture the scene shape by minimizing an error functional along iterations. By using such model, we are able to introduce any type of shape priors directly into the estimation process. In our case, we do this by adding a regularizer term to our cost functional by which we decrease the degrees of freedom for our parameter set that helps us to make the problem well-posed. However, it should be noted that any prior information about the structure or the silhouette of the shape can be incorporated as convenient as our curvature regularizer. Another advantage of using a geometric model would be its ability to address the occlusions. In our simulations, we add a visibility analysis step that takes the antenna pair location and the evolving shape and compute which parts of the object does contribute to the signal measured by that antenna pair. We observe the effects the visibility analysis where the parts of the shapes visible to the antennas successfully evolve to the actual object where the invisible parts fail to do so due to the lack of measurements from those parts. To accomplish this, we first develop a method for scenes that can be modelled as a star shape objects that can be uniquely parameterized with a polar representation and can be approximated by a polygonal shape. We use such a model due to the simplicity of implementation by which we show a geometric model based shape inversion from radar signals is possible. Finally, we extend our work to an active contour based framework where represent our shapes using level-sets. This allowed us to handle scenes with arbitrary shapes and topologies which greatly generalizes our initial approach.

Appendices

APPENDIX A

DERIVATION OF GRADIENT FLOW

We will derive the gradient flow equations for a general integrand f_s that is a function of both the point coordinates \mathbf{x} and outward normal vector of the curve \mathbf{n} at that point. It is given as:

$$\frac{\partial e}{\partial \tau} = \frac{\partial}{\partial \tau} \left[\int_{s_0}^{s_1} f_s(\mathbf{x}, \mathbf{n}) ds \right]. \quad (\text{A.1})$$

We reparameterize the integral so that the borders are fixed. Reparameterized integral is given as:

$$= \int_0^1 \frac{\partial}{\partial \tau} (f_s \|\mathbf{x}_p\|) dp \quad (\text{A.2})$$

$$= \int_0^1 \left(\frac{\partial f_s}{\partial \tau} \|\mathbf{x}_p\| + f_s \frac{\partial}{\partial \tau} \|\mathbf{x}_p\| \right) dp \quad (\text{A.3})$$

$$= \int_0^1 \left[\left(\frac{\partial f_s}{\partial \mathbf{x}} \mathbf{x}_\tau + \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{n}_\tau \right) \|\mathbf{x}_p\| + f_s \frac{\mathbf{x}_p \cdot \mathbf{x}_{p\tau}}{\|\mathbf{x}_p\|} \right] dp \quad (\text{A.4})$$

$$= \int_0^1 \left[\left(\frac{\partial f_s}{\partial \mathbf{x}} \mathbf{x}_\tau + \frac{\partial f_s}{\partial \mathbf{n}} (-\mathbf{x}_{s\tau} \cdot \mathbf{n}) \mathbf{t} \right) \|\mathbf{x}_p\| + f_s \frac{\mathbf{x}_p \cdot \mathbf{x}_{p\tau}}{\|\mathbf{x}_p\|} \right] dp \quad (\text{A.5})$$

$$= \int_0^1 \left[\frac{\partial f_s}{\partial \mathbf{x}} \mathbf{x}_\tau \|\mathbf{x}_p\| - \left(\frac{\mathbf{x}_{p\tau}}{\|\mathbf{x}_p\|} \cdot \mathbf{n} \right) \|\mathbf{x}_p\| \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t} + f_s \frac{\mathbf{x}_p \cdot \mathbf{x}_{p\tau}}{\|\mathbf{x}_p\|} \right] dp \quad (\text{A.6})$$

$$= \int_0^1 \left[\frac{\partial f_s}{\partial \mathbf{x}} \mathbf{x}_\tau \|\mathbf{x}_p\| - (\mathbf{x}_{\tau p} \cdot \mathbf{n}) \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t} + f_s \frac{\mathbf{x}_p \cdot \mathbf{x}_{\tau p}}{\|\mathbf{x}_p\|} \right] dp \quad (\text{A.7})$$

$$= \int_0^1 \left[\frac{\partial f_s}{\partial \mathbf{x}} \mathbf{x}_\tau - \frac{\mathbf{x}_{\tau p} \cdot \mathbf{n}}{\|\mathbf{x}_p\|} \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t} + f_s \frac{\mathbf{x}_p \cdot \mathbf{x}_{\tau p}}{\|\mathbf{x}_p\|^2} \right] \|\mathbf{x}_p\| dp \quad (\text{A.8})$$

$$= \int_{s_0}^{s_1} \left[\frac{\partial f_s}{\partial \mathbf{x}} \mathbf{x}_\tau - (\mathbf{x}_{\tau s} \cdot \mathbf{n}) \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t} + f_s (\mathbf{x}_s \cdot \mathbf{x}_{\tau s}) \right] ds \quad (\text{A.9})$$

Applying integration by parts, we have:

$$= \int_{s_0}^{s_1} \frac{\partial f_s}{\partial \mathbf{x}} \mathbf{x}_\tau ds - \int_{s_0}^{s_1} \left[(\mathbf{x}_{\tau s} \cdot \mathbf{n}) \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t} - (\mathbf{x}_{\tau s} \cdot \mathbf{x}_s) f_s \right] ds \quad (\text{A.10})$$

$$= \int_{s_0}^{s_1} \frac{\partial f_s}{\partial \mathbf{x}} \mathbf{x}_\tau ds - \int_{s_0}^{s_1} \left(\mathbf{x}_{\tau s} \cdot \left[\mathbf{n} \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t} - \mathbf{x}_s f_s \right] \right) ds \quad (\text{A.11})$$

$$= \int_{s_0}^{s_1} \frac{\partial f_s}{\partial \mathbf{x}} \mathbf{x}_\tau ds - \int_{s_0}^{s_1} \left[\mathbf{n} \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t} - \mathbf{x}_s f_s \right] \cdot \mathbf{x}_{\tau s} ds \quad (\text{A.12})$$

$$= \int_{s_0}^{s_1} \frac{\partial f_s}{\partial \mathbf{x}} \mathbf{x}_\tau ds + \int_{s_0}^{s_1} \frac{\partial}{\partial s} \left[\mathbf{n} \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t} - \mathbf{x}_s f_s \right] \cdot \mathbf{x}_\tau ds \quad (\text{A.13})$$

$$= \int_{s_0}^{s_1} \frac{\partial f_s}{\partial \mathbf{x}} \mathbf{x}_\tau ds + \int_{s_0}^{s_1} \left[\mathbf{n}_s \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t} + \mathbf{n} \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n} \partial s} + \mathbf{n} \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t}_s - \mathbf{x}_{ss} f_s - \mathbf{x}_s \frac{\partial f_s}{\partial s} \right] \cdot \mathbf{x}_\tau ds \quad (\text{A.14})$$

$$= \int_{s_0}^{s_1} \frac{\partial f_s}{\partial \mathbf{x}} \mathbf{x}_\tau ds + \int_{s_0}^{s_1} \left[\mathbf{n}_s \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t} + \mathbf{n} \mathbf{t}^\top \left(\frac{\partial^2 f_s}{\partial \mathbf{n} \partial \mathbf{x}} \mathbf{x}_s + \frac{\partial^2 f_s}{\partial \mathbf{n}^2} \mathbf{n}_s \right) + \mathbf{n} \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t}_s \right. \quad (\text{A.15})$$

$$\left. - \mathbf{x}_{ss} f_s - \mathbf{x}_s \left(\frac{\partial f_s}{\partial \mathbf{x}} \mathbf{x}_s + \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{n}_s \right) \right] \cdot \mathbf{x}_\tau ds \quad (\text{A.16})$$

$$= \int_{s_0}^{s_1} \frac{\partial f_s}{\partial \mathbf{x}} \mathbf{x}_\tau ds + \int_{s_0}^{s_1} \left[\mathbf{n}_s \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t} + \mathbf{n} \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n} \partial \mathbf{x}} \mathbf{x}_s + \mathbf{n} \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n}^2} \mathbf{n}_s + \mathbf{n} \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t}_s \right. \quad (\text{A.17})$$

$$\left. - \mathbf{x}_{ss} f_s - \mathbf{x}_s \frac{\partial f_s}{\partial \mathbf{x}} \mathbf{x}_s - \mathbf{x}_s \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{n}_s \right] \cdot \mathbf{x}_\tau ds \quad (\text{A.18})$$

$$= \int_{s_0}^{s_1} \left[\frac{\partial f_s}{\partial \mathbf{x}} + \kappa \mathbf{t} \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t} + \mathbf{n} \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n} \partial \mathbf{x}} \mathbf{t} + \kappa \mathbf{n} \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n}^2} \mathbf{t} - \kappa \mathbf{n} \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{n} + \kappa \mathbf{n} f_s \right. \quad (\text{A.19})$$

$$\left. - \mathbf{t} \frac{\partial f_s}{\partial \mathbf{x}} \mathbf{t} - \kappa \mathbf{t} \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t} \right] \cdot \mathbf{x}_\tau ds \quad (\text{A.20})$$

$$= \int_{s_0}^{s_1} \left[\underbrace{\frac{\partial f_s}{\partial \mathbf{x}} - \mathbf{t} \frac{\partial f_s}{\partial \mathbf{x}} \mathbf{t}}_{\mathbf{n} \frac{\partial f_s}{\partial \mathbf{x}} \mathbf{n}} + \underbrace{\kappa \mathbf{t} \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t} - \kappa \mathbf{t} \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{t}}_0 - \kappa \mathbf{n} \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{n} + \mathbf{n} \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n} \partial \mathbf{x}} \mathbf{t} \right. \quad (\text{A.21})$$

$$\left. + \kappa \mathbf{n} \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n}^2} \mathbf{t} + \kappa \mathbf{n} f_s \right] \cdot \mathbf{x}_\tau ds \quad (\text{A.22})$$

$$= \int_{s_0}^{s_1} \left[\mathbf{n} \frac{\partial f_s}{\partial \mathbf{x}} \mathbf{n} - \kappa \mathbf{n} \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{n} + \mathbf{n} \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n} \partial \mathbf{x}} \mathbf{t} + \kappa \mathbf{n} \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n}^2} \mathbf{t} + \kappa \mathbf{n} f_s \right] \cdot \mathbf{x}_\tau ds \quad (\text{A.23})$$

$$= \int_{s_0}^{s_1} \left[\mathbf{n} \left(\frac{\partial f_s}{\partial \mathbf{x}} \mathbf{n} - \kappa \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{n} + \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n} \partial \mathbf{x}} \mathbf{t} + \kappa \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n}^2} \mathbf{t} + \kappa f_s \right) \right] \cdot \mathbf{x}_\tau ds. \quad (\text{A.24})$$

APPENDIX B

DERIVATION OF GRADIENT FLOW FOR RADAR

Our integrand for radar without the weighting is given as:

$$f_s(\mathbf{x}, \mathbf{n}) = \left(\frac{G}{R}\right)^2 (-\mathbf{u} \cdot \mathbf{n}) \quad (\text{B.1})$$

We will find it useful to derive a couple of expressions for later use.

$$\frac{\partial f_s}{\partial \mathbf{x}} = 2\frac{G}{R} \left[\frac{G_{\mathbf{x}}R - GR_{\mathbf{x}}}{R^2} \right] (-\mathbf{u} \cdot \mathbf{n}) + \left(\frac{G}{R}\right)^2 \frac{\partial(-\mathbf{u} \cdot \mathbf{n})}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \mathbf{x}} \quad (\text{B.2})$$

$$= \frac{2G(-\mathbf{u} \cdot \mathbf{n})}{R^2} G_{\mathbf{x}} - \frac{2G^2(-\mathbf{u} \cdot \mathbf{n})}{R^3} \mathbf{u}^\top - \left(\frac{G}{R}\right)^2 \mathbf{n}^\top (\mathbf{I} - \mathbf{u}\mathbf{u}^\top) \frac{1}{R} \quad (\text{B.3})$$

$$= - \left(\frac{2G(\mathbf{u} \cdot \mathbf{n})}{R^2} G_{\mathbf{x}} - \frac{3G^2(\mathbf{u} \cdot \mathbf{n})}{R^3} \mathbf{u}^\top + \frac{G^2}{R^3} \mathbf{n}^\top \right) \quad (\text{B.4})$$

$$\frac{\partial f_s}{\partial \mathbf{n}} = - \left(\frac{G}{R}\right)^2 \mathbf{u}^\top \quad (\text{B.5})$$

$$\frac{\partial^2 f_s}{\partial \mathbf{n}^2} = 0 \quad (\text{B.6})$$

$$\nabla_{\mathbf{n}} \cdot \frac{\partial f_s}{\partial \mathbf{n}} = 0 \quad (\text{B.7})$$

$$\frac{\partial^2 f_s}{\partial \mathbf{n} \partial \mathbf{x}} = \frac{\partial}{\partial \mathbf{x}} \left(-\frac{G^2}{R^2} \mathbf{u} \right) \quad (\text{B.8})$$

$$= - \left[\frac{G^2}{R^2} \frac{\partial \mathbf{u}}{\partial \mathbf{x}} + \mathbf{u} \left(\frac{2GG_{\mathbf{x}}}{R^2} - \frac{2G^2 R_{\mathbf{x}}}{R^3} \right) \right] \quad (\text{B.9})$$

$$= - \left[\frac{G^2}{R^2} (\mathbf{I} - \mathbf{u} \mathbf{u}^\top) \frac{1}{R} + \frac{2G}{R^2} \mathbf{u} G_{\mathbf{x}} - \frac{2G^2}{R^3} \mathbf{u} R_{\mathbf{x}} \right] \quad (\text{B.10})$$

$$= - \left[\frac{G^2}{R^3} \mathbf{I} - \frac{3G^2}{R^3} \mathbf{u} \mathbf{u}^\top + \frac{2G}{R^2} \mathbf{u} G_{\mathbf{x}} \right] \quad (\text{B.11})$$

$$\nabla_{\mathbf{x}} \cdot \frac{\partial f_s}{\partial \mathbf{n}} = \text{Tr} \left(\frac{\partial^2 f_s}{\partial \mathbf{n} \partial \mathbf{x}} \right) = - \left[\frac{2G^2}{R^3} - \frac{3G^2}{R^3} + \frac{2G}{R^2} (\mathbf{u} \cdot G_{\mathbf{x}}) \right] \quad (\text{B.12})$$

$$= \frac{G^2}{R^3} - \frac{2G}{R^2} (\mathbf{u} \cdot G_{\mathbf{x}}) \quad (\text{B.13})$$

Using the previously derived gradient expression, without the weighting, we get:

$$\nabla_C E = \left(\frac{\partial f_s}{\partial \mathbf{x}} \mathbf{n} - \kappa \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{n} + \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n} \partial \mathbf{x}} \mathbf{t} + \kappa \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n}^2} \mathbf{t} + \kappa f_s \right) \mathbf{n} \quad (\text{B.14})$$

$$= \left(\frac{\partial f_s}{\partial \mathbf{x}} \mathbf{n} + \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n} \partial \mathbf{x}} \mathbf{t} - \kappa \frac{\partial f_s}{\partial \mathbf{n}} \mathbf{n} + \kappa \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n}^2} \mathbf{t} + \kappa f_s \right) \mathbf{n} \quad (\text{B.15})$$

$$= \left[\frac{\partial f_s}{\partial \mathbf{x}} \mathbf{n} + \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n} \partial \mathbf{x}} \mathbf{t} - \kappa \left(\frac{\partial f_s}{\partial \mathbf{n}} \mathbf{n} - \mathbf{t}^\top \frac{\partial^2 f_s}{\partial \mathbf{n}^2} \mathbf{t} - f_s \right) \right] \mathbf{n} \quad (\text{B.16})$$

$$= \left[\frac{\partial f_s}{\partial \mathbf{x}} \mathbf{n} + \nabla_{\mathbf{x}} \cdot \frac{\partial f_s}{\partial \mathbf{n}} - \mathbf{n}^\top \frac{\partial^2 f_s}{\partial \mathbf{n} \partial \mathbf{x}} \mathbf{n} - \kappa \left(\frac{\partial f_s}{\partial \mathbf{n}} \mathbf{n} - \nabla_{\mathbf{n}} \cdot \frac{\partial f_s}{\partial \mathbf{n}} + \mathbf{n}^\top \frac{\partial^2 f_s}{\partial \mathbf{n}^2} \mathbf{n} \right. \right. \quad (\text{B.17})$$

$$\left. - f_s \right) \mathbf{n} \quad (\text{B.18})$$

$$= - \left[\left(\frac{2G(\mathbf{u} \cdot \mathbf{n})}{R^2} G_{\mathbf{x}} - \frac{3G^2(\mathbf{u} \cdot \mathbf{n})}{R^3} \mathbf{u}^\top + \frac{G^2}{R^3} \mathbf{n}^\top \right) \cdot \mathbf{n} \right. \\ \left. + \left(\frac{G^2}{R^3} - \frac{2G}{R^2} (\mathbf{u} \cdot G_{\mathbf{x}}) \right) + \left(\frac{G^2}{R^3} - \frac{3G^2(\mathbf{u} \cdot \mathbf{n})^2}{R^3} + \frac{2G(\mathbf{u} \cdot \mathbf{n})(G_{\mathbf{x}} \cdot \mathbf{n})}{R^2} \right) \right. \\ \left. - \kappa \left(\frac{G^2}{R^2} (-\mathbf{u} \cdot \mathbf{n}) - f_s \right) \right] \cdot \mathbf{n} \quad (\text{B.19})$$

$$= \left[-\frac{2G(\mathbf{u} \cdot \mathbf{n})(G_{\mathbf{x}} \cdot \mathbf{n})}{R^2} + \frac{3G^2(\mathbf{u} \cdot \mathbf{n})^2}{R^3} - \frac{G^2}{R^3} - \frac{2G}{R^2} (\mathbf{u} \cdot G_{\mathbf{x}}) + \frac{G^2}{R^3} \right. \quad (\text{B.20})$$

$$\left. + \frac{G^2}{R^3} - \frac{3G^2(\mathbf{u} \cdot \mathbf{n})^2}{R^3} + \frac{2G(\mathbf{u} \cdot \mathbf{n})(G_{\mathbf{x}} \cdot \mathbf{n})}{R^2} \right] \cdot \mathbf{n} \quad (\text{B.21})$$

$$= \frac{G}{R^2} \left[\frac{G}{R} - 2(\mathbf{u} \cdot G_{\mathbf{x}}) \right] \quad (\text{B.22})$$

REFERENCES

- [1] A. J. Yezzi and S. Soatto, “Structure from motion for scenes without features,” in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, IEEE, vol. 1, 2003, pp. I–I.
- [2] A. Yezzi and S. Soatto, “Stereoscopic segmentation,” *International Journal of Computer Vision*, vol. 53, no. 1, pp. 31–43, 2003.
- [3] M. A. Richards, J. Scheer, W. A. Holm, and W. L. Melvin, *Principles of modern radar*. Citeseer, 2010.
- [4] C. F. Barnes, “Synthetic aperture radar,” *Wave Theory Foundations, Analysis and Algorithms*, Barnes, Powder Springs, Georgia, 2015.
- [5] W. M. Brown, “Synthetic aperture radar,” *IEEE Transactions on Aerospace and Electronic Systems*, no. 2, pp. 217–229, 1967.
- [6] W. G. Carrara, R. Goodman, and R. Majewski, “Spotlight synthetic radar: Signal processing algorithms,” *Artech House*, 1995.
- [7] M. Cheney, “A mathematical tutorial on synthetic aperture radar,” *SIAM review*, vol. 43, no. 2, pp. 301–312, 2001.
- [8] C. A. Wiley, *Pulsed doppler radar methods and apparatus*, US Patent 3,196,436, 1965.
- [9] C. Willey, “Synthetic aperture radars—a paradigm for technology evolution,” *IEEE Trans. Aerospace Elec. Sys*, vol. 21, pp. 440–443, 1985.
- [10] A. Ferretti, C. Prati, and F. Rocca, “Permanent scatterers in sar interferometry,” *IEEE Transactions on geoscience and remote sensing*, vol. 39, no. 1, pp. 8–20, 2001.
- [11] D. Massonnet and J.-C. Souyris, *Imaging with synthetic aperture radar*. CRC Press, 2008.
- [12] D. Massonnet, M. Rossi, C. Carmona, F. Adragna, G. Peltzer, K. Feigl, and T. Rabaute, “The displacement field of the landers earthquake mapped by radar interferometry,” *Nature*, vol. 364, no. 6433, p. 138, 1993.

- [13] C. Jakowatz Jr and P. Thompson, “The tomographic formulation of spotlight mode synthetic aperture radar extended to three dimensional targets,” Sandia National Labs., Albuquerque, NM (United States), Tech. Rep., 1992.
- [14] C. V. Jakowatz and P. Thompson, “A new look at spotlight mode synthetic aperture radar as tomography: Imaging 3-d targets,” *IEEE transactions on image processing*, vol. 4, no. 5, pp. 699–703, 1995.
- [15] J. M. Lopez-Sanchez and J. Fortuny-Guasch, “3-d radar imaging using range migration techniques,” *IEEE Transactions on antennas and propagation*, vol. 48, no. 5, pp. 728–737, 2000.
- [16] A. W. Doerry, “Basics of polar-format algorithm for processing synthetic aperture radar images,” *Sandia National Laboratories report SAND2012-3369, Unlimited Release*, 2012.
- [17] C. Nguyen and J. Park, “Stepped-frequency radar sensor analysis,” in *Stepped-Frequency Radar Sensors: Theory, Analysis and Design*. Cham: Springer International Publishing, 2016, pp. 39–64, ISBN: 978-3-319-12271-7.
- [18] M. C. Budge and S. R. German, *Basic RADAR analysis*. Artech House, 2015.
- [19] W. J. Caputi, “Stretch: A time-transformation technique,” *IEEE Transactions on Aerospace and Electronic Systems*, no. 2, pp. 269–278, 1971.
- [20] J. D. Foley, F. D. Van, A. Van Dam, S. K. Feiner, J. F. Hughes, J. Hughes, and E. Angel, *Computer graphics: principles and practice*. Addison-Wesley Professional, 1996, vol. 12110.
- [21] A. J. Stewart and T. Karkanis, “Computing the approximate visibility map, with applications to form factors and discontinuity meshing,” in *Rendering Techniques’ 98*, Springer, 1998, pp. 57–68.
- [22] S. Coorg and S. Teller, “Real-time occlusion culling for models with large occluders,” in *Proceedings of the 1997 symposium on Interactive 3D graphics*, ACM, 1997, 83–ff.
- [23] S. Osher and J. A. Sethian, “Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations,” *Journal of computational physics*, vol. 79, no. 1, pp. 12–49, 1988.
- [24] D. Adalsteinsson and J. A. Sethian, “A fast level set method for propagating interfaces,” *Journal of computational physics*, vol. 118, no. 2, pp. 269–277, 1995.

- [25] S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, and A. Yezzi, “Gradient flows and geometric active contour models,” in *Proceedings of IEEE International Conference on Computer Vision*, IEEE, 1995, pp. 810–815.
- [26] V. Caselles, R. Kimmel, and G. Sapiro, “Geodesic active contours,” in *Proceedings of IEEE international conference on computer vision*, IEEE, 1995, pp. 694–699.
- [27] M. Benyamin, J. Calder, G. Sundaramoorthi, and A. Yezzi, “Accelerated variational pdes for efficient solution of regularized inversion problems,” *Journal of Mathematical Imaging and Vision*, pp. 1–27, 2019.