

ABSTRACT

Title of dissertation: STATISTICAL MODELS AND OPTIMIZATION
ALGORITHMS FOR HIGH-DIMENSIONAL
COMPUTER VISION PROBLEMS

Kaushik Mitra, Doctor of Philosophy, 2011

Dissertation directed by: Professor Rama Chellappa
Department of Electrical and Computer Engineering

Data-driven and computational approaches are showing significant promise in solving several challenging problems in various fields such as bioinformatics, finance and many branches of engineering. In this dissertation, we explore the potential of these approaches, specifically statistical data models and optimization algorithms, for solving several challenging problems in computer vision. In doing so, we contribute to the literatures of both statistical data models and computer vision. In the context of statistical data models, we propose principled approaches for solving *robust regression* problems, both *linear* and *kernel*, and *missing data matrix factorization* problem. In computer vision, we propose statistically optimal and efficient algorithms for solving the *remote face recognition* and *structure from motion* (SfM) problems.

The goal of robust regression is to estimate the functional relation between two variables from a given data set which might be contaminated with outliers. Under the reasonable assumption that there are fewer outliers than inliers in a dataset,

we formulate the robust linear regression problem as a *sparse learning* problem, which can be solved using efficient polynomial-time algorithms. We also provide sufficient conditions under which the proposed algorithms correctly solve the robust regression problem. We then extend our robust formulation to the case of kernel regression, specifically to propose a robust version for relevance vector machine (RVM) regression.

Matrix factorization is used for finding a low-dimensional representation for data embedded in a high-dimensional space. Singular value decomposition is the standard algorithm for solving this problem. However, when the matrix has many missing elements this is a hard problem to solve. We formulate the missing data matrix factorization problem as a *low-rank semidefinite programming* problem (essentially a rank constrained SDP), which allows us to find accurate and efficient solutions for large-scale factorization problems.

Face recognition from remotely acquired images is a challenging problem because of variations due to blur and illumination. Using the convolution model for blur, we show that the set of all images obtained by blurring a given image forms a convex set. We then use convex optimization techniques to find the distances between a given blurred (probe) image and the gallery images to find the best match. Further, using a low-dimensional linear subspace model for illumination variations, we extend our theory in a similar fashion to recognize blurred and poorly illuminated faces.

Bundle adjustment is the final optimization step of the SfM problem where the goal is to obtain the 3-D structure of the observed scene and the camera parameters

from multiple images of the scene. The traditional bundle adjustment algorithm, based on minimizing the l_2 norm of the image re-projection error, has cubic complexity in the number of unknowns. We propose an algorithm, based on minimizing the l_∞ norm of the re-projection error, that has quadratic complexity in the number of unknowns. This is achieved by reducing the large-scale optimization problem into many small scale sub-problems each of which can be solved using second-order cone programming.

STATISTICAL MODELS AND OPTIMIZATION ALGORITHMS
FOR HIGH-DIMENSIONAL COMPUTER VISION PROBLEMS

by

Kaushik Mitra

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2011

Advisory Committee:

Professor Rama Chellappa, Chair/Advisor

Professor Andre L. Tits

Professor Ramani Duraiswami

Professor David Jacobs, Dean's Representative

Professor Ashok Veeraraghavan

© Copyright by
Kaushik Mitra
2011

Dedication

To One and All

Acknowledgments

I owe my gratitude to all the people who have made this dissertation possible and because of whom my graduate experience has been one that I will cherish forever.

First and foremost I'd like to thank my advisor Professor Rama Chellappa for giving me the wonderful opportunity to work with him. Under his able guidance, I had the opportunity to work on many challenging and interesting problems. He has also given me the freedom to explore many new topics and ideas on my own, which has given me the confidence to work as an individual researcher. He has always made himself available for help and advice and I am very grateful to him for that. It has been a great pleasure to work with and learn from such an extraordinary individual.

I would also like to thank Professor Andre L. Tits, Professor Ramani Duraiswami, Professor David Jacobs and Professor Ashok Veeraraghavan for agreeing to serve on my thesis committee and for sparing their invaluable time reviewing the manuscript. Professor Ashok Veeraraghavan has also been my mentor and collaborator and I would like to thank him for his guidance. During my dissertation, I had the opportunity of working closely with some wonderful people - Dr. Sameer Sheorey, who is also a mentor to me, Priyanka Vageeswaran and Jun-Cheng Chen, who were my juniors and their energy and enthusiasm invigorated me.

I would like to thank many professors in the University of Maryland, from whom I have taken courses - (Electrical and Computer Engineering:) Professor

Anthony Ephremides, Professor Haralabos (Babis) Papadopoulos, Professor Andre L. Tits, Professor Rama Chellappa, Professor Min Wu, Professor. Steve Marcus, Professor Sennur Ulukus, (Mathematics and Statistics:) Professor Wojciech Czaja, Professor Mike Fitzpatrick, Professor Grace Yang, Professor Serguei Novikov, Professor Karin Melnick, Professor Richardo Nochetto and Professor Dionisios Margetis. The courses were very enjoyable and formed the foundations for my research work. I would also take this opportunity to thank all my teachers from Indian Institute of Science (Bangalore, India), Institute of Radio Physics and Electronics (Kolkata, India), Presidency College (Kolkata, India) and Kendriya Vidyalaya (Itarsi, India). I am specially grateful to my undergraduate (IRPE) advisor Professor Subal Kar and my graduate (IISc) advisor Professor Anurag Kumar for their able guidance, support and encouragement.

My colleagues at the computer vision laboratory have enriched my graduate life in many ways and deserve a special mention - Ashish Srivastava, Aravind Sundaresan, Ashok Veeraraghavan, Narayanan Ramanathan, Gaurav Aggarwal, Seong-Wook Joo, Naresh Cuntoor, Feng Guo, Arun Mohanchettiar, Pavan Turaga, Soma Biswas, James Sherman, Aswin Sankaranarayanan, Mahesh Ramachandran, Wu Hao, Volkan Cevher, Vishal Patel, Ruonan Li, Dikpal Reddy, Nitesh Shroff, Raghuraman Gopalan, Sima Taheri, Nazre Batool, Priyanka Vageeswaran, Jun-Cheng Chen, Sumit Shekhar, Jaishankar Pillai, Jie Ni, Hien Nguyen, Huy Tho Ho, Garrett Warnell, Qiang Qiu, Dave Shaw, Mohammed Abdelkader, Ming Du, Ming Liu, Wu Tao. Apart from my laboratory colleagues, I would also like to thank my other University colleagues and friends - Anne Jorstad, who was my wonderful office mate;

Ajay Mishra, with whom I had many philosophical discussions on a wide variety of topics; Punyaslok Purkayastha, whose knowledge of maths, physics and literature inspires me; and many other friends and colleagues - Indroda (Indrajit Bhattacharya), Sandeep Manocha and Sameer Kibey.

I would like to acknowledge the help and support of several staff members who made my graduate life smooth - Janice Perrone (CfAR), Maria Hoo (ECE), Arlene Schenk (UMIACS) and the UMIACS computing staff.

I owe my deepest thanks to my family - my mother, father and sister, who have always stood by me and guided me through my career. Words cannot express the gratitude I owe them. I am also grateful to my grandparents, uncles, aunties and cousins, who made this experience enjoyable.

I am thankful to my house mates and friends for making my stay in Maryland a wonderful experience - Bhuwan Thapa, Purushottam Shetty, Bhaskar Dutta, Basudev Roy, Sandeep Haldar, Mainak Sen, Senthilkumar Palaniyandi, Sanjeev Howlader, Muhammad S. Noon, Mili Duggal, Kanthi Sarpatwar, Jagadeesh Jagarlamudi, Nitin Madnani, Ashish Markanday, Padmaja, Punarbasu Purkayastha, Arya Mazumdar, Barna Saha, Biswadip Dey, Arijit Biswas, Saptashati Tania Biswas, and Rajibul Kaji.

It is impossible to remember all, and I apologize to those I've inadvertently left out. Lastly, thank you all and thank God!

Table of Contents

List of Figures	viii
1 Introduction	1
2 Robust Linear Regression Using Sparse Learning for High-Dimensional Applications	14
2.1 Robust Regression Based on Basis Pursuit (BPRR)	18
2.2 Proof of the Main Theorem 2.1.1	24
2.3 A Bayesian Approach: Bayesian Robust Regression (BRR)	29
2.4 Theoretical and Empirical studies of the Parameter space of Robust Regression	32
2.5 Age Estimation From Face Images	36
3 Robust RVM Regression Using Sparse Outlier Model	44
3.1 Robust RVM Regression	46
3.1.1 Model Specification	47
3.1.2 Robust Bayesian RVM (RB-RVM)	48
3.1.2.1 Inference	49
3.1.2.2 Prediction	51
3.1.2.3 Advantage over other Robust RVM Algorithms	51
3.1.3 Basis Pursuit RVM (BP-RVM)	52
3.2 Empirical Evaluation	53
3.3 Robust Image Denoising	57
3.4 Age Estimation from Facial Images	59
4 Large-Scale Matrix Factorization with Missing Data under Additional Constraints	64
4.1 Background: Low-rank semidefinite programming (LRSDP)	67
4.2 Matrix factorization using LRSDP (MF-LRSDP)	68
4.2.1 Noiseless Case	69
4.2.2 Noisy case	70
4.2.3 Enforcing Additional Constraints	72
4.3 Matrix Completion, Uniqueness and Convergence of MF-LRSDP	73
4.3.1 Matrix Completion Theory	74
4.3.2 Relation with Matrix Factorization and its Implications	74
4.4 Experimental Evaluation	75
4.4.1 Evaluation with Synthetic Data	75
4.4.2 Evaluation with Real Data	78
5 Direct Recognition of Faces across Blur and Illumination Variations	82
5.1 Face Recognition Across Blur (FRB)	84
5.2 Incorporating the Illumination Model	89
5.3 Experimental Evaluations	92

5.3.1	Recognition across Blur	93
5.3.2	Incorporating Illumination Model	99
6	A Scalable Projective Bundle Adjustment Algorithm using the l_∞ Norm	107
6.1	Background: geometric reconstruction problems using L_∞ norm . . .	111
6.1.1	Triangulation/Intersection	111
6.1.2	Resection	113
6.2	The l_∞ projective bundle adjustment algorithm	113
6.2.1	Decoupling	114
6.2.2	Cheirality and quasi-affine initialization	115
6.3	Computational complexity and memory requirement	118
6.4	Experiments	120
6.4.1	Convergence	121
6.4.2	Computational scalability	123
6.4.3	Behavior with noise	124
7	Conclusion and Future Directions	127
7.1	Robust Linear Regression Using Sparse Learning for High-Dimensional Applications	127
7.2	Robust RVM Regression Using Sparse Outlier Model	128
7.3	Sparse Regularization for Regression and Classification on Manifolds .	129
7.4	Large-Scale Matrix Factorization with Missing Data under Additional Constraints	130
7.5	Direct Recognition of Faces across Blur and Illumination Variations .	131
7.6	Hierarchical Dictionary for Face and Activity Recognition:	132
7.7	A Scalable Projective Bundle Adjustment Algorithm using the L_∞ Norm	132
	Bibliography	133

List of Figures

1.1	Many computer vision problems are solved using the following framework: first extract relevant features from images/video and then use statistical data models such as regression, classification, matrix factorization, etc. to find pattern/structure in the data.	2
1.2	Generally computer vision data have the following characteristics: they are high-dimensional, outliers are present in the data set and some elements of the data are missing.	3
1.3	<i>Outliers</i> occur frequently in computer vision data set. For example , in finding lines in an image, points belonging to one line are outliers for the other lines. (Image courtesy OpenCV 2.0 C Reference)	4
1.4	<i>Missing data</i> problem arises frequently in Structure from Motion (SfM) problem. In SfM, feature points are tracked through all the images, but since not all the features are visible in all the images, this gives rise to the missing data problem. We will see later that completing the missing tracks solves the SfM problem.	5
1.5	Robust linear regression: The popular linear regression technique “least squares” is very sensitive to outliers. Random Sample Consensus (RANSAC), a robust algorithm, is mostly used for solving low-dimensional vision problems. However, it is a combinatorial algorithm and hence can not be used for solving high-dimensional problems. We propose robust polynomial time algorithms and analyze their performances.	6
1.6	Robust RVM regression: RVM regression is a kernel regression technique, which has been used for solving many problems such as age and pose estimation. However, it is very susceptible to outliers as can be seen here. We propose two robust versions of RVM.	7
1.7	Missing data matrix factorization: We encounter missing data (missing tracks) in the SfM problem. We can solve the SfM problem (complete the missing tracks) by solving a missing data matrix factorization problem. We propose a large-scale factorization algorithm that can handle large amounts of missing data.	8
1.8	Remote Face Recognition: Face recognition from remotely acquired images is a challenging problem because of variations due to blur, illumination, pose and occlusions. We address the problem of recognizing blurred and poorly illuminated faces by using the generative models for blur and illumination variations.	11
1.9	Scalable Bundle Adjustment: Bundle adjustment is the final optimization step of the SfM problem, where the structure and camera parameters are refined starting from an initial reconstruction. We propose an efficient bundle adjustment algorithm based on minimizing the l_∞ -norm of reprojection error. (Image courtesy Dr. Noah Snavely)	12

2.1	Mean estimation error vs. outlier fraction for dimension 2, 6 and 25 respectively. Only BPRR, BRR and M-estimator are shown for dimension 25 as the other algorithms very slow. BRR performs very well for all the dimensions; the other algorithms are comparable with each other.	38
2.2	Recovery rate, i.e. the fraction of successful recovery, vs. outlier fraction for dimensions 2 and 50 for algorithms BPRR, BRR and M-estimators; we do have plots for LMedS and RANSAC as these algorithms are very slow. From the figure we can conclude that each of the algorithms exhibit a sharp transtion from success to failure at a certain outlier fraction.	39
2.3	Phase transition curves of the algorithms BPRR, BRR and M-estimator. BRR gives the best performance followed by BPRR and M-estimator.	40
2.4	Mean angle error vs. inlier noise standard deviation for dimension 6 and 0.4 outlier fraction. All algorithms, except LMedS, perform well.	41
2.5	Some outlier and inliers found by BRR. Most of the outliers were images of older subjects. This could be because a linear (regression) model may not be sufficient to capture the relation between age and facial geometry for all age groups. Since, the majority of the images in the dataset are of young subjects, the older subjects become outliers with respect to them.	42
2.6	Mean absolute error (MAE) of age estimation Vs outlier fraction. BRR has almost constant MAE until outlier fraction increases beyond 0.5.	43
3.1	Prediction by the three algorithms: RVM, RB-RVM and BP-RVM in the presence of symmetric outliers for $N = 100$, $f = 0.2$ and $\sigma = 0.1$. Data which are enclosed by a box are the outliers found by the robust algorithms. Prediction error are also shown in the figures. RB-RVM gives the lowest prediction error.	53
3.2	Prediction by the three algorithms: RVM, RB-RVM and BP-RVM in the presence of asymmetric outliers for $N = 100$, $f = 0.2$ and $\sigma = 0.1$. Data which are enclosed by a box are the outliers found by the robust algorithms. Prediction error are also shown in the figures. Clearly, RB-RVM gives the best result.	54
3.3	Prediction error vs. outlier fraction for the symmetric and asymmetric outlier cases. RB-RVM gives the best result for both the cases. For the symmetric case, BP-RVM gives lower prediction error than RVM but for the asymmetric case they give similar result.	56
3.4	Prediction error vs. inlier noise standard deviation for the symmetric and asymmetric outlier cases. RB-RVM gives the lowest prediction error until about $\sigma = 0.2$, after which RVM gives better result. This is because for our experimental setup, at approximately $\sigma = 0.3$, the distinction between the inliers and outliers cease to exist.	56
3.5	Prediction error vs. number of data points for the symmetric and asymmetric outlier cases. For all the three algorithms, performance improves with increasing N	57

3.6	Results on Salt and pepper noise removal: first column: RVM, second column: RB-RVM, third column: Median filter, fourth column: Gaussian filter. The RMSE values are also shown in the figure; RB-RVM gives the best result.	59
3.7	Mean RMSE value over seven images vs. percentage of salt and pepper noise. RB-RVM gives better performance than the median filter.	60
3.8	Mixture of Gaussian and salt and pepper noise removal experiment: denoised images by RVM and RB-RVM with their corresponding RMSE values. This experiment again shows that the RB-RVM based denoising algorithm gives much better result than the RVM based one.	61
3.9	Some inliers and outliers found by RB-RVM. Most of the outliers are images of older subjects like Outlier A and B. This is because there are less number of samples of older subjects in the FG-Net database. Outlier C has an extreme pose variation from the usual frontal faces of the database; hence, it is an outlier. The facial geometry of Outlier D is very similar to that of younger subjects, such as big forehead and small chin, so it is classified as an outlier.	62
3.10	Mean absolute error (MAE) of age prediction vs. fraction of controlled outliers added to the training dataset. RB-RVM gives much lower prediction error as compared to the RVM. Also, note that the prediction error is reasonable even with outlier fraction as high as 0.7.	63
4.1	(a) Reconstruction rate vs. fraction of revealed entries per column $ \Omega /n$ for 500×500 matrices of rank 5 by MF-LRSDP, alternation and OptSpace. The proposed algorithm MF-LRSDP gives the best reconstruction results since it can reconstruct matrices with fewer observed entries. (b) Time taken for reconstruction by different algorithms. MF-LRSDP takes the least time.	77
4.2	(a) Reconstruction rate vs. fraction of revealed entries per column $ \Omega /n$ for rank 5 square matrices of different sizes n by MF-LRSDP and OptSpace. MF-LRSDP reconstructs matrices from fewer observed entries than OptSpace. (b) Reconstruction rate vs. $ \Omega /n$ for 500×500 matrices of different ranks by MF-LRSDP and OptSpace. Again MF-LRSDP needs fewer observations than OptSpace. (c) RMSE vs. noise standard deviation for rank 5, 200×200 matrices by MF-LRSDP, OptSpace, alternation and damped Newton. All algorithms perform equally well.	78
4.3	Cumulative histogram (of 25 trials) for the Dinosaur, Giraffe and the Face sequence. For all of them, MF-LRSDP consistently gives good results.	80
4.4	(a) Input (incomplete) point tracks of the Dinosaur turntable sequence, (b) reconstructed tracks without orthonormality constraints and (c) reconstructed tracks with orthonormality constraints. Without the constraints many tracks fail to be circular, whereas with the constraints all of them are circular (the dinosaur sequence is a turntable sequence and the tracks are supposed to be circular).	81
5.1	Sample probe images from FERET dataset. The probe images are synthetically blurred with Gaussian filters of $\sigma = 0, 2, 4, 6, 8$ respectively. $\sigma = 0$ stands for 'no blur'.	93

5.2	<i>a)</i> Recognition by our proposed algorithm FRB and <i>b)</i> by FADEIN, LPQ and FADEIN+LPQ (figure courtesy [70]) on the FERET dataset. FRB is better than FADEIN and LPQ. FRB is comparable with FADEIN+LPQ for small values of σ , but outperms it for large values.	95
5.3	The effect of kernel size on the performance of our algorithm FRB. The probe images are blurred by a Gaussian kernel of $\sigma = 4$. From these curves we conclude the following:1) FRB is not very sensitive to the choice of kernel-size and 2) the imposition of symmetry constraints further relaxes the need for accurate choice of kernel-size.	103
5.4	For the experiment on REMOTE dataset, we have divided the probe images into four categories: <i>a)</i> sharp and well-illuminated images, <i>b)</i> sharp and poorly-illuminated images, <i>c)</i> blurred and well-illuminated images and <i>d)</i> blurred and poorly-illuminated images. These images were acquired at distances between 5 – 250 meters.	104
5.5	The nine illumination basis images of an individual in the PIE dataset. These basis images are used in the FRBI algorithm to model illumination variations.	105
5.6	The nine illumination basis images of an individual in the REMOTE dataset. These basis images are used in the FRBI algorithm to model illumination variations. The nine illumination positions from which the basis images are created has been optimized for this dataset. . . .	106
6.1	l_∞ BA Algorithm	118
6.2	l_∞ reprojection error versus iteration for the four algorithms on the data sets: sphere, corridor, hotel and dinosaur. l_∞ error decreases monotonically for l_∞ BA but not so for the other algorithms.	122
6.3	RMS reprojection error versus iteration : RMS error decreases monotonically for l_∞ BA and l_2 BA but not so for WIE and IE. IE fails to converge for the dinosaur data set.	123
6.4	3-D reconstruction result for the datasets, Sphere and Corridor. The Red '*' represents the camera center and Blue 'o' represents the structure point. The first column shows the initialization, second column shows the final reconstruction and the third column shows the groundtruth.	124
6.5	Total convergence time of l_2 BA and l_∞ BA as the number of cameras is varied with number of points fixed at 500.	125
6.6	Behavior of l_∞ BA and l_2 BA with image feature noise for the sphere data set.	126

Chapter 1

Introduction

In recent times data-driven approaches are being used to solve many challenging problems in areas such as bioinformatics, finance and many other engineering sciences. The main reason behind this trend is that some problems are very difficult to model. Modeling difficulty arises because it is not clear what the factors involved in the problem are or how they interact with each other. For example, in bioinformatics one would like to know which genes are responsible for which diseases. Modeling here would mean knowing the functionalities of each gene and how they interact each other. This is, no doubt, a very challenging problem given the fact that there are 20,000 – 25,000 genes in a human cell. Similar situations arise in many other areas, such as predicting financial markets, weather patterns and so on. Statistical data models are very useful in these situations; it is convenient to collect several data or examples and use it to learn the parameters of an appropriate statistical model. For example, in the gene-disease problem, one can collect data of the type ‘active genes in a patients suffering from a certain disease’, and one can then use statistical tools such as missing-data matrix factorization to find the underlying relation. Along with the popularity of statistical data models, the need for efficient computational algorithms is also increasing. As the statistical models become more sophisticated and datasets become larger, there is definitely a need for more efficient

optimization algorithms.

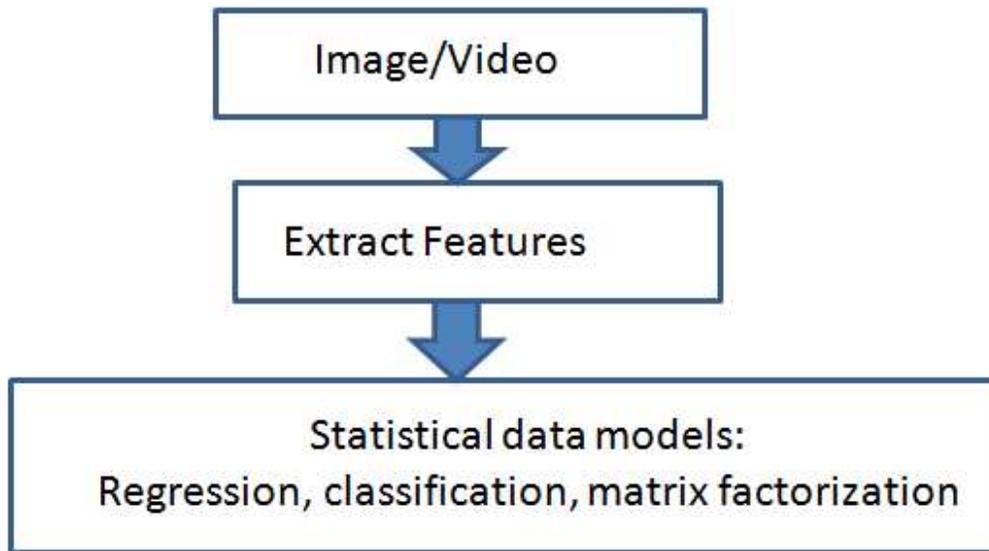


Figure 1.1: Many computer vision problems are solved using the following framework: first extract relevant features from images/video and then use statistical data models such as regression, classification, matrix factorization, etc. to find pattern/structure in the data.

The success of statistical data models and optimization algorithms in other areas motivates us to look for appropriate statistical data models and algorithms in the context of computer vision. There are many problems in computer vision which are difficult to model, such as visual representation of objects and scenes, facial age progression, etc.. If we take the example of facial age progression, there are many factors that play a role, such as bone growth, loss in elasticity of facial muscles, facial fat atrophy, ethnicity, gender, dietary habits, climatic conditions, etc. and it is not easy to model them. Hence, the prevalent approach for solving this problem is to extract relevant features from the face images and use statistical model such

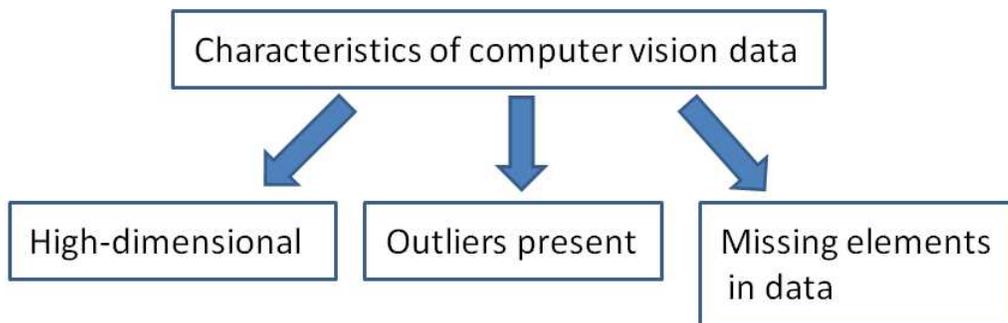


Figure 1.2: Generally computer vision data have the following characteristics: they are high-dimensional, outliers are present in the data set and some elements of the data are missing.

as regression to learn the relation between the extracted features and age [78]. This approach, in general, is used for solving many other vision problems, where the first step involves extracting relevant features from images/video, followed by using statistical data models such as regression, classification, matrix factorization, etc. for finding pattern/structure in the data, see figure 1.1. The need for computationally efficient algorithms has always been felt in vision, the main reason being images, when treated as a vector, are points in very high-dimensional spaces.

Our goal in this dissertation is to design statistical data models and optimization algorithms which can be used for solving many vision problems. Towards this goal, we first list the common characteristics of many computer vision data (see figure 1.2):

- Most of the computer vision data are *high-dimensional*. This becomes clear from the fact that even a small (black and white) image of size 100×100 is a point in $\mathbb{R}^{10,000}$. Also, the recent trend towards concatenating many different

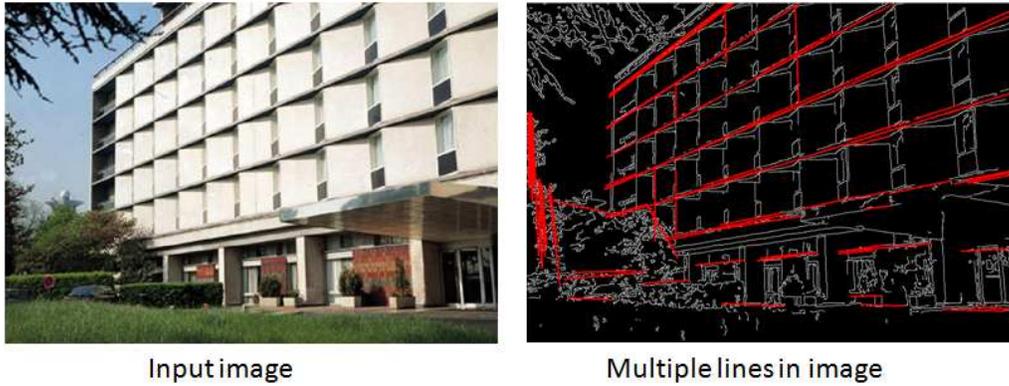


Figure 1.3: *Outliers* occur frequently in computer vision data set. For example , in finding lines in an image, points belonging to one line are outliers for the other lines. (Image courtesy OpenCV 2.0 C Reference)

features such as “histogram of oriented gradients” (HOG) [29], “scale-invariant feature transform” (SIFT) [59], “histogram of Gabor phase patterns” (HGPP) [109], etc. as a big feature vector results in very high-dimensional data. Hence, statistical models and algorithms that we design should be able to handle high-dimensional data.

- *Outliers* (data that deviates from a model by a large extent) occur very frequently in computer vision data sets, see figure 1.3. The main reasons for this are: the presence of multiple models in images/videos and variations in visual data. Multiple models are frequently encountered in the problem of surface reconstruction from range (depth) images, where it is very likely that a scene will have more than one surface (model) and data drawn from one model become outliers for the other models [90]. Multiple models are also encountered while estimating the motion of moving objects in a video and

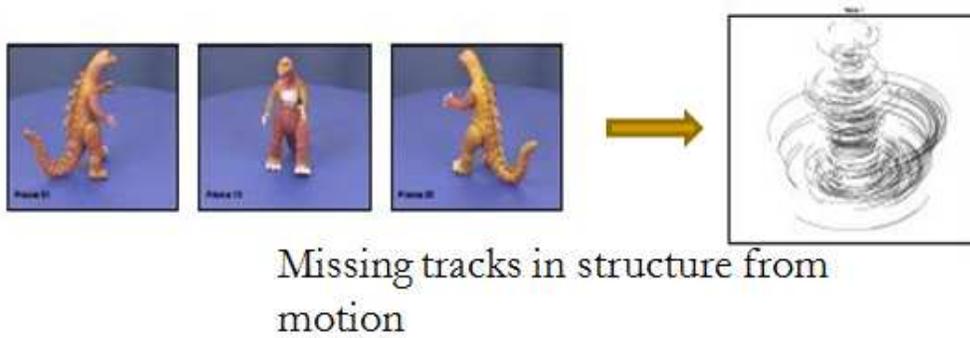


Figure 1.4: *Missing data* problem arises frequently in Structure from Motion (SfM) problem. In SfM, feature points are tracked through all the images, but since not all the features are visible in all the images, this gives rise to the missing data problem. We will see later that completing the missing tracks solves the SfM problem.

finding lines/curves in images. There are many sources of variations in visual data such as that due to illumination, geometry and noise, and if a variation is not accounted for in a data model, then data suffering from that variation are likely to become outliers. In the presence of outliers, it is important to design robust statistical models.

- We also frequently encounter *missing elements* in visual data. For example in the SfM problem [41], where the goal is to reconstruct the 3D scene from multiple images or video, we track 2D features through the images or frames of the video and then estimate the geometry of the scene using the features. However many features are not visible in all the images/frames and this gives rise to the missing data problem (see figure 1.4). The missing data problem also arises when solving the photometric stereo problem [107], where the goal is to reconstruct the surface of an imaged object under different illumination

conditions. Both these problems can be solved by filling in these missing data [14]. Hence, it is important that statistical models, designed for solving computer vision problems, should be able to handle missing elements in the data.

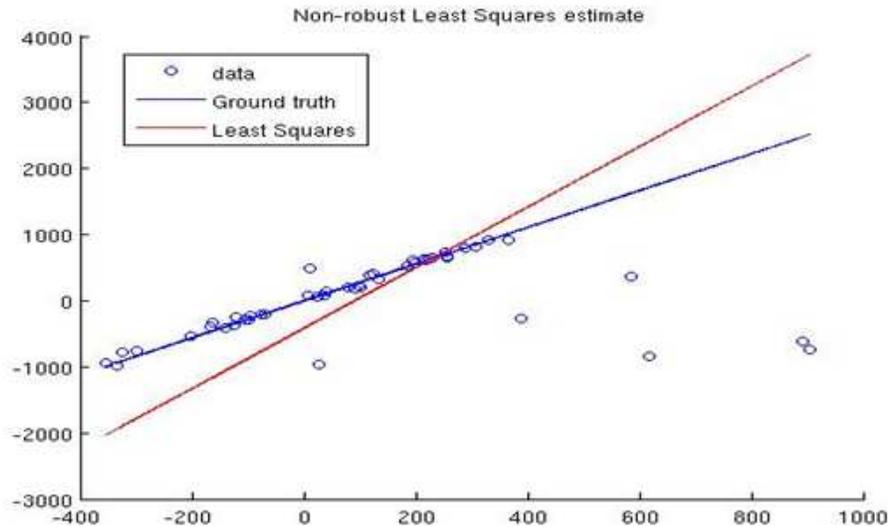


Figure 1.5: **Robust linear regression:** The popular linear regression technique “least squares” is very sensitive to outliers. Random Sample Consensus (RANSAC), a robust algorithm, is mostly used for solving low-dimensional vision problems. However, it is a combinatorial algorithm and hence can not be used for solving high-dimensional problems. We propose robust polynomial time algorithms and analyze their performances.

Keeping the above characteristics of the computer vision datasets in mind, we propose the following statistical data models:

- **Robust Linear Regression For High-Dimensional Data:** The goal of regression is to learn the functional relation between two variables from many

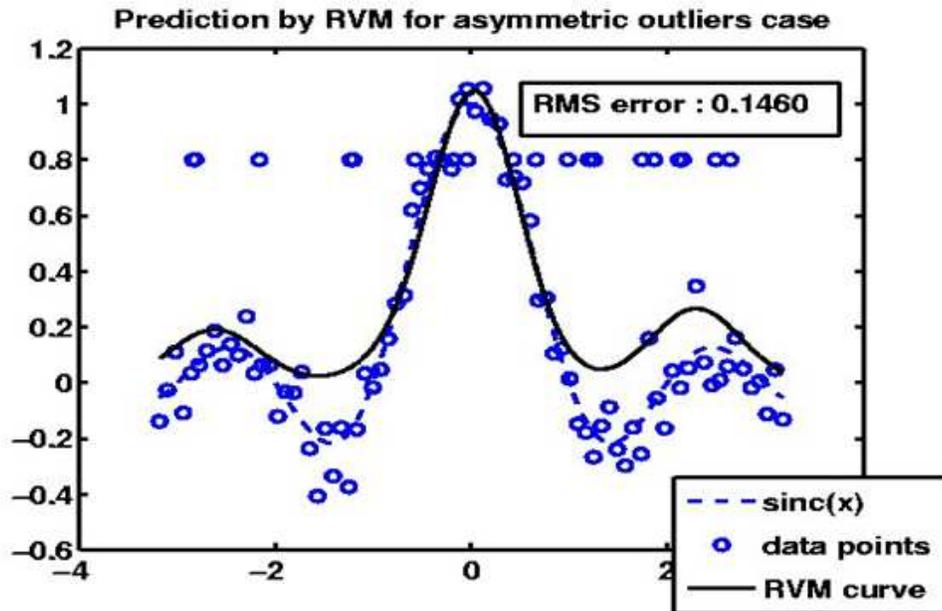


Figure 1.6: **Robust RVM regression:** RVM regression is a kernel regression technique, which has been used for solving many problems such as age and pose estimation. However, it is very susceptible to outliers as can be seen here. We propose two robust versions of RVM.

examples/data. If we know the functional form (linear, quadratic, etc.) of the relation, then the goal of regression becomes estimating the parameters of the function. Many problems in computer vision can be posed as a regression problem. Some examples are: finding primitive structures (lines and curves) in images, epipolar geometry estimation [41], age estimation from facial images [78], human head and body pose estimation [3] and surface estimation from gradient fields [5]. Many of these problems are high-dimensional such as the age, pose and surface estimation problems. And all of these problems usually suffer from outliers and hence we need robust regression algorithms for solving

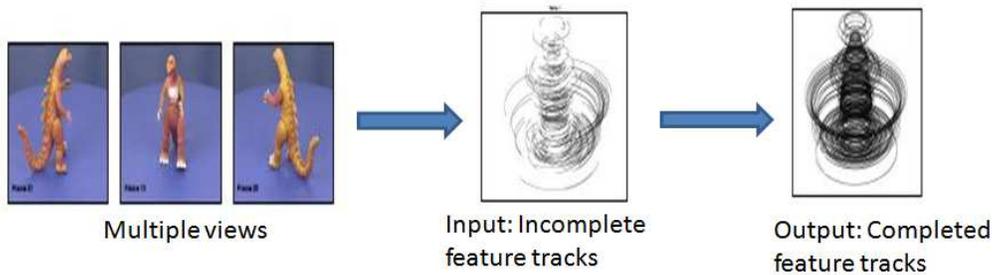


Figure 1.7: **Missing data matrix factorization:** We encounter missing data (missing tracks) in the SfM problem. We can solve the SfM problem (complete the missing tracks) by solving a missing data matrix factorization problem. We propose a large-scale factorization algorithm that can handle large amounts of missing data.

them, see figure 1.5. Low-dimensional problems, such as line/curve estimation and epipolar geometry estimation, are usually solved using the popular (in vision literature) robust algorithm RANSAC [36]. However, this algorithm is combinatorial in the dimension of the problem and hence can not be used for solving high-dimensional problems. We propose polynomial time robust linear regression algorithms, which can be used for solving high-dimensional problems. Using the assumption that outliers in a dataset are usually sparse, we formulate the robust regression problem based on two techniques from sparse representation/learning theory: Basis pursuit [26] and Bayesian sparse learning [95]. We analyze the precise conditions under which the basis pursuit based algorithm can correctly solve the robust regression problem. These conditions are based on the angle difference between the *regressor subspace* and the *outlier subspaces*. We also empirically study the performance of various robust algorithms and use them to solve the age estimating problem. Chapter

2 presents this work in more details.

- **Robust Kernel Regression Using Sparse Outliers Model:** We generalize our robust framework for the linear regression to kernel regression. Linear regression is an example of parametric regression, where we assume the regression model to be of a certain parametric form. However, if we are not certain about the appropriate parametric model to use for a particular problem, the alternative is to use a non-parametric model such as kernel regression. Kernel regression approximates the dependent variable by kernel functions located at each data point. In this dissertation, we consider the *Relevance Vector Machine* (RVM) regression, which is a particular type of kernel regression. In RVM, a Gaussian distribution is assumed for the noise term in the model, which makes it susceptible to the presence of outliers in the data set, see figure 1.6. We propose robust versions of the RVM regression. We decompose the noise term in the RVM formulation into a (sparse) outlier noise term and a Gaussian noise term. We then estimate the outlier noise along with the model parameters. We present two approaches for solving this estimation problem: 1) a Bayesian approach, which essentially follows the RVM framework and 2) a regularization approach based on basis pursuit. In the Bayesian approach, the robust RVM problem essentially becomes a bigger RVM problem with the advantage that it can be solved efficiently by a fast algorithm. Empirical evaluations, and real experiments on image denoising and age estimation demonstrate the better performance of the robust RVM algorithms over that

of the RVM regression. Chapter 3 presents this work in more details.

- **Large-Scale Matrix Factorization in the Presence of Missing Data:**

Low-rank factorization of the “data matrix” (data collected as columns of a matrix) reveals the low-dimensional structure of the data. Many problems in computer vision, such as SfM and photometric stereo, are solved using the low-rank matrix factorization technique. If the data matrix is complete, the low-rank factors can be obtained by singular value decomposition (SVD) of the matrix. However, if there are many missing elements in the matrix, which happens frequently in SfM (see figure 1.7) and photometric stereo problems, it is hard problem to solve. The popular algorithm in vision literature for solving this problem is based on damped Newton’s method [14], which is a very slow and memory intensive algorithm. We formulate the matrix factorization with missing data problem as a *low-rank semidefinite program* (LRSDP) with the advantage that: 1) an efficient quasi-Newton implementation of the LRSDP enables us to solve large-scale factorization problems, and 2) additional constraints such as ortho-normality, required in orthographic SfM, can be directly incorporated in the new formulation. Our empirical evaluations suggest that, under the conditions of matrix completion theory [21], the proposed algorithm finds the optimal solution, and also requires fewer observations compared to the current state of the art algorithms. We further demonstrate the effectiveness of the proposed algorithm in solving the affine SfM problem, non-rigid SfM and photometric stereo problems. Chapter 4 presents this work in more

details.



Figure 1.8: **Remote Face Recognition:** Face recognition from remotely acquired images is a challenging problem because of variations due to blur, illumination, pose and occlusions. We address the problem of recognizing blurred and poorly illuminated faces by using the generative models for blur and illumination variations.

Apart from designing statistical data models and optimization algorithms that can be used for solving many computer vision problems, we also address two specific vision problems and propose statistically optimal and efficient algorithms for solving them.

- **Direct Face Recognition Across Blur and Illumination Variations:**

We are interested in recognizing faces acquired from distant cameras. The main factors that make this a challenging problem are image degradations due to blur and noise, and variations in appearance due to illumination and pose, see figure 1.8. In this dissertation, we address the problem of recognizing faces across blur and illumination variations. The current state of the art approach for recognizing blurred faces first deblurs the face image and then recognize it using classical face recognition algorithms [70]. However, deblurring (blind deconvolution) is an ill-posed problem and, more importantly, is

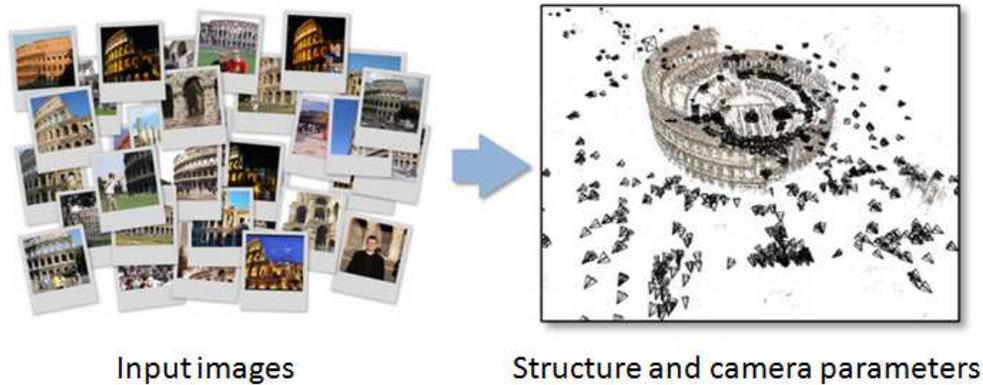


Figure 1.9: **Scalable Bundle Adjustment:** Bundle adjustment is the final optimization step of the SfM problem, where the structure and camera parameters are refined starting from an initial reconstruction. We propose an efficient bundle adjustment algorithm based on minimizing the l_∞ -norm of reprojection error. (Image courtesy Dr. Noah Snavely)

not an essential step for recognizing faces. We take a direct approach for face recognition. Using the convolution model for blur, we show that the set of all images obtained by blurring a given image forms a convex set. We then use the set theoretic notion of distance between a given blurred (probe) image and the gallery sets to find the best match. Further, to handle illumination variations we use the low-dimensional linear subspace model [8], and define a set for each gallery image that represents all possible variations of that gallery image due to blur and illumination. The probe image is then assigned the identity of the closest gallery image. The proposed recognition algorithm is also statistically optimal; it is the maximum likelihood estimate of the blur filter kernel, illumination coefficients and identity. Further, using the set the-

oretic notion of distance between sets, we can characterize the amount of blur our algorithm can handle for a given dataset. Chapter 5 presents this work in more details.

- **A Scalable Bundle Adjustment Algorithm Using the l_∞ Norm:** SfM is the problem of reconstructing the 3-D structure of an observed scene and the camera parameters (orientations and locations) from multiple images or video of the scene. Bundle adjustment is the final optimization step of the SfM problem, where the structure and camera parameters are refined starting from an initial reconstruction, see figure 1.9. Traditionally this is done by minimizing the l_2 -norm of the image reprojection error [7]. LevenbergMarquardt algorithm is used for solving this problem, which has a computational complexity of $O((m+n)^3)$ per iteration and memory requirement of $O(mn(m+n))$, where m is the number of cameras and n is the number of structure points. We propose an algorithm that has a computational complexity of $O(mn(\sqrt{m} + \sqrt{n}))$ per iteration and memory requirement of $O(\max(m, n))$. The proposed algorithm is based on minimizing the l_∞ norm of reprojection error. It alternately estimates the camera and structure parameters, thus reducing the potentially large scale optimization problem to many small scale subproblems each of which is a quasi-convex optimization problem and hence can be solved globally. Experiments using synthetic and real data show that the proposed algorithm gives good performance in terms of minimizing the reprojection error and also has a good convergence rate. Chapter 6 presents this work in more details.

Chapter 2

Robust Linear Regression Using Sparse Learning for High-Dimensional Applications

The goal of regression is to infer a functional relationship between two sets of variables from a given data set. Many a times the functional form is already known and the parameters of the model (function) are estimated from the data set. In most of the data sets, there are some data which differ markedly from the rest of the data; these are known as outliers. The goal of robust regression techniques is to properly account for the outliers while estimating the model parameters. Since, any subset of the data could be outliers, robust regression is, in general, a combinatorial problem and (robust) algorithms such as “least median squares” (LMedS) [81] and RANSAC [36] inherit this combinatorial nature. We propose polynomial-time algorithms and state the conditions under which we can correctly solve the robust regression problem.

We express the regression error as a sum of two error terms: an outlier (gross) error term and an inlier (small) error term. Under the reasonable assumption that the number of outliers is fewer than the number of inliers, the robust regression problem can be formulated as a l_0 -norm regularization problem, where we minimize the number of outliers subject to satisfying the regression model. We provide conditions under which the above optimization problem will find the correct model

parameters (and outliers). These conditions are in terms of the smallest *principal angle* between the *regression subspace* and the *outlier subspaces*, which we show is related to the *restricted isometry constant* of the compressive sensing theory [22]. However, the l_0 -norm regularization problem is a combinatorial problem and hence we relax it to a l_1 -norm regularized problem, which is related to the basis pursuit algorithm [26]. We then show that under stricter conditions on the angular distance between the regression subspace and the outlier subspaces, the proposed algorithm will correctly solve the robust regression problem. We also propose a Bayesian formulation for solving the robust regression problem. We use the sparse Bayesian learning technique [95] to impose a sparse prior on the outliers and then obtain the outliers using maximum a-posterior (MAP) criterion. Finally, we study the theoretical computational complexity of various robust regression algorithms to identify algorithms that are efficient for solving high-dimensional problems.

Related works: LMedS technique [81] minimizes the median of the squared residuals. A random sampling algorithm is used for solving this problem. This sampling algorithm is combinatorial in the dimension (number of the parameters) of the problem which makes LMedS impractical for solving high-dimensional regression problems. The RANSAC algorithm [36] and its improvements such as MSAC, MLESAC [99] are the most widely used robust algorithms in computer vision [90]. RANSAC estimates the model parameters by minimizing the number of outliers, which are defined as data points that have residual greater than a pre-defined threshold. The same random sampling algorithm as used in LMedS is used for solving this problem, which makes RANSAC, MSAC and MLESAC impracti-

cal for high-dimension problems. Another famous class of robust algorithms is the M-estimates [44]. M-estimates are a generalization of the maximum likelihood estimates (MLEs), where the negative log likelihood function of the data is replaced by a robust cost function. Amongst the many possible choices of cost functions, redescending cost functions are the most robust ones. However, these cost functions are non-convex and the resulting non-convex optimization problem has many local minima. Generally, a polynomial time algorithm “iteratively reweighted least squares” (IRLS) is used for solving the optimization problem, which often converges to local minima. There are many other robust algorithms, proposed as improvements over M-estimates, such as S-estimates, L-estimates and MM-estimates, but all of them are solved using the (combinatorial) random sampling algorithm [63], and hence, can not be used for solving high-dimensional problems. Apart from robust cost function-based approaches, there are methods that first identify the outliers using “outlier diagnostics techniques”, remove them, and then use a (non-robust) regression algorithm such as Least Squares (LS) to estimate the model parameters [82]. However, these methods are not known to be very successful when there are many outliers.

A similar mathematical formulation (as robust regression) arises in the context of error-correcting codes over the reals [22], [24]. Error-correcting codes are used for encoding messages in such a way so that it can be reliably transmitted over a channel and correctly decoded at the receiver. The decoding schemes, in particular, are very similar to robust regression algorithms. The decoding scheme used in [22] is the l_1 – *regression* (least absolute deviations). It was shown that if a certain

orthogonal matrix, related to the encoding matrix, satisfies the *restricted isometry property* (RIP) and the gross error vector is sufficiently sparse, then the message can be successfully recovered. In [24], this error-correcting scheme was further extended to the case where the channel could introduce (dense) small errors along with sparse gross errors. Two decoding schemes were proposed and it was shown that if a properly scaled version of the encoding matrix satisfies the RIP property and the gross error vector is sufficiently sparse, the message can be correctly recovered. The robust regression problem is different from the error-correcting codes in the following manner: In error-correcting codes, one is free to design the encoding matrix, whereas, in robust regression we are provided a data set and hence there is no question of designing the regression matrix, which plays a similar mathematically role as the encoding matrix. Also, the sufficient conditions that we provide for correctly estimating the model parameters are more appropriate in the context of robust regression and also tighter than that provided in [24]. Concurrently with us [67], a Bayesian approach based on sparse learning was proposed for solving the robust regression problem in [47]. This approach is similar in principal to our Bayesian approach and the paper reports similar results.

The organization of the rest of this chapter is as follows: in section 2.1, we formulate the robust regression problem as a l_0 -norm regularization problem and relaxed convex versions of it (l_1 regression and modified form of basis pursuit) and provide conditions under which the proposed optimization problems correctly solves the robust regression problem. We prove our main result in section 2.2. In section 2.3, we propose a Bayesian approach for robust regression. In section 2.4, we

perform many empirical experiments to compare various robust algorithms and, finally, in section 2.5, we present a real application of age estimation using the robust algorithms.

2.1 Robust Regression Based on Basis Pursuit (BPRR)

Regression is the problem of estimating the functional relation f between two sets of variables: independent variable or regressor $x \in \mathbb{R}^D$ and dependent variable or regressand $y \in \mathbb{R}$, from many examples pairs (x, y) . In linear regression, the function f is a linear function of the model parameter $w \in \mathbb{R}^D$:

$$y = x^T w + e, \tag{2.1}$$

where e is the observation noise. We want to estimate w from a given training dataset of N observations $(y_i, x_i)_{i=1, 2, \dots, N}$, i.e. $y_i = x_i^T w + e_i$. We can write all the observation equations collectively as:

$$y = Xw + e, \tag{2.2}$$

where $y = (y_1, \dots, y_N)^T$, $X = [x_1^T, \dots, x_N^T] \in \mathbb{R}^{N \times D}$ and $e = (e_1, \dots, e_N)^T$. The most popular estimator of w is the least squares (LS), which is statistically optimal (in the maximum likelihood sense) for independent and identically distributed Gaussian noise case. However, in the presence of outliers or gross error, the noise distribution is far from Gaussian and, hence, LS gives poor estimates of w .

To handle outliers, we express the noise variable e as sum of two independent components, $e = s + n$, where s represents the outliers and n represents the small

noise, which can be modeled, for example, by Gaussian distribution. With this the linear regression model is given by

$$y = Xw + s + n. \quad (2.3)$$

Note that this is an ill-posed problem as there are more unknowns, w and s , than equations and, hence, there are many solutions. Clearly, we need to restrict the solution space in order to make it a well posed problem. A reasonable assumption is that outliers are sparse in a dataset, i.e., the number of outliers are much less than the number of inliers. RANSAC also makes this assumption: it finds that parameter w which results in the least number of data being labeled as outliers. Under this sparse outlier assumption, we should solve the following optimization problem:

$$\min_{s,w} \|s\|_0 \text{ such that } \|\mathbf{y} - \mathbf{X}\mathbf{w} - \mathbf{s}\|_2 \leq \epsilon, \quad (2.4)$$

where $\|s\|_0$ is the number of non-zero elements in s and ϵ is a measure of the magnitude of the small noise n . If we assume n to be a Gaussian random variable, then ϵ may be chosen as a small multiple of the variance. However, before looking at the case where both outliers and small noise is present, we first treat the case where only outliers are present, i.e., $n = 0$.

In the absence of small noise ($n = 0$), we should solve

$$\min_{s,w} \|s\|_0 \text{ such that } y = Xw + s. \quad (2.5)$$

We are interested in the question: Under what conditions, by solving the above equation, can we recover the original w from the observation y ? It is quite obvious that X should be full column rank (as $N \geq D$), otherwise, even when there are

no outliers, we will not be able to recover the original w . To discover the other conditions, we re-write the constraint in (2.5) as

$$y = [X \ I]w_s, \quad (2.6)$$

where I is a $N \times N$ identity matrix and $w_s = [w; s]^1$ is the augmented vector of unknowns. Now, consider a particular dataset y, X where amongst the N data, characterized by the index set $J = [1, 2, \dots, N]$, k of them are affected by outliers. Let these k outlier affected data be specified by the subset $T \subset J$. Then, equation (2.6) can be written as

$$y = [X \ I_T]w_{s^k}, \quad (2.7)$$

where I_T is a matrix consisting of column vectors from I indexed by T , $w_{s^k} = [w; s^k]$ and $s^k \in \mathbb{R}^k$ represents the k non-zero outliers. Given the information about the index subset T , i.e. given which data (indices) are affected by outliers, we can recover w and the non-zero outliers s^k by solving (2.5) if and only if $[X \ I_T]$ is full column rank. The condition $[X \ I_T]$ being full rank can also be expressed in terms of the smallest *principal angle* between the subspace spanned by the regressor, $\text{span}(X)$, and the subspace spanned by outliers, $\text{span}(I_T)$. The smallest principle angle θ between two subspaces \mathcal{U} and \mathcal{W} of \mathbb{R}^N is defined as the smallest angle between a vector in \mathcal{U} and a vector in \mathcal{W} [38]:

$$\cos(\theta) = \max_{u \in \mathcal{U}} \max_{w \in \mathcal{W}} \frac{u^T w}{\|u\| \|w\|}. \quad (2.8)$$

Equivalently for any vectors $u \in \text{span}(X)$ and $w \in \text{span}(I_T)$

$$|u^T w| \leq \delta \|u\| \|w\| \quad (2.9)$$

¹Throughout this chapter, we will use the MATLAB notation $[w; s]$ to mean $[w^T \ s^T]^T$

where $\delta = \cos(\theta)$ is smallest such number. To generalize this inequality for all subset T with cardinality at most k , we introduce the following definition.

Definition 2.1.1. *For every integer $1 \leq k \leq N$ we define a constant δ_k to be the smallest quantity such that for all $u \in \text{span}(X)$ and $w \in \text{span}(I_T)$ with $|T| \leq k$, the following holds*

$$|\langle u, w \rangle| \leq \delta_k \|u\| \|w\| \quad (2.10)$$

The quantity $\delta_k \in [0, 1]$ is a measure of how well separated the regressor subspace $\text{span}(X)$ is from the all the outlier subspaces $\text{span}(I_T)$ with dimension at most k . When $\delta_k = 1$, the regressor subspace and one of the outlier subspaces of dimension at most k , share at least a common vector, whereas, when $\delta_k = 0$, the regressor subspace is orthogonal to all the outlier spaces of dimension at most k . With the definition of δ_k , we are now in a position to state the sufficient conditions for recovering w by solving (2.5).

Proposition 2.1.1. *Assume that $\delta_{2k} < 1$ and X is a full column rank matrix. Then, by solving (2.5), we can recover w exactly if there are at most k outliers in the y variable.*

Proof. The conditions $\delta_{2k} < 1$ and X a full rank matrix together implies that all matrices of the form $[X \ I_T]$ with $|T| \leq 2k$ are full rank. This fact can be proved by a simple contradiction argument.

Now, suppose w_0 and s_0 with $\|s_0\|_0 \leq k$ satisfy the equation

$$y = Xw + s. \quad (2.11)$$

Then to show that we can recover w_0 and s_0 by solving (2.5), it is sufficient to show that there exists no other w and s , with $\|s\|_0 \leq k$, which also satisfy (2.11). We show this by contradiction: Suppose there is another such pair, say w_1 and s_1 with $\|s_1\|_0 \leq k$, which also satisfies (2.11). Then $Xw_0 + s_0 = Xw_1 + s_1$. Re-arranging, we have:

$$[X \ I]\Delta w_s = 0 \tag{2.12}$$

where $\Delta w_s = [\Delta w; \Delta s]$, $\Delta w = (w_0 - w_1)$ and $\Delta s = (s_0 - s_1)$. Since $\|s_0\|_0 \leq S$ and $\|s_1\|_0 \leq S$, $\|\Delta s\|_0 \leq 2k$. If T_Δ denotes the corresponding non-zero index set, then T_Δ has a cardinality of at most $2k$ and, thus, $[X \ I_{T_\Delta}]$ is a full rank matrix. This in turn implies that $\Delta w_s = 0$, i.e. $w_0 = w_1$ and $s_0 = s_1$. Hence, the solution of (2.5) is unique and correct under the assumed conditions. \square

From the above theorem, we can find a lower bound on the maximum number of outliers (in the y variable) that the l_0 norm regression (2.5) can handle in a dataset of regressor matrix X . This is given by the largest integer k such that $\delta_{2k} < 1$. Note that the l_0 norm regression (2.5) is a hard combinatorial problem to solve. So, as in compressive sensing theory, we would like to approximate it by the following convex problem:

$$\min_{s,w} \|s\|_1 \text{ such that } y = Xw + s \tag{2.13}$$

where the $\|s\|_0$ term is replaced by the l_1 norm of s . Note that the above problem can be re-written as $\min_w \|y - Xw\|_1$, and hence this is the l_1 regression problem. Again, we are interested in the question: Under what conditions, by solving the above problem, can we recover the original w ? Not surprisingly, the answer is that

we need a bigger angular separation between the regressor subspace and the outlier subspaces.

Proposition 2.1.1. *Assume that $\delta_{2k} < \frac{2}{3}$ and X is a full column rank matrix. Then, by solving (2.13), we can recover w exactly if there are at most k outliers in the y variable. Furthermore, if there are more than k outliers, then the estimation error of w (Δw) is given in terms of the best k -sparse approximation of the outliers s_k , the vector s with all but the k -largest entries set to zero, by*

$$\|\Delta w\|_2 \leq \tau^{-1} C_0 k^{-\frac{1}{2}} \|s - s_k\|_1, \quad (2.14)$$

where τ is the smallest singular value of X and C_0 is a constant which depends only on δ_{2k} .

Note that if there are at most k outliers, then $s_k = s$, and equation (2.14) implies that $\|\Delta w\|_2 \leq 0$, i.e., w can be exactly recovered. Similar to the l_0 regression case, we can obtain a lower bound on the maximum number of outliers that the l_1 regression can handle in the y variable; it is given by the largest integer k for which $\delta_{2k} < \frac{2}{3}$. Proposition 2.1.1 is a special case of the next theorem which considers the small noise case ($n > 0$). In the presence of small bounded noise with $\|n\|_2 \leq \epsilon$, we propose to solve the following convex approximation of the combinatorial problem (2.4)

$$\min_{\mathbf{s}, \mathbf{w}} \|\mathbf{s}\|_1 \text{ such that } \|\mathbf{y} - \mathbf{X}\mathbf{w} - \mathbf{s}\|_2 \leq \epsilon. \quad (2.15)$$

Note that the above problem is a modified form of the basis pursuit denoising problem [26]. Under the same conditions on the angular separation between the regressor subspace and the outliers subspaces, we have the following result.

Theorem 2.1.1. *Assume that $\delta_{2k} < \frac{2}{3}$, X is a full column rank matrix and (2.15) is feasible. Then the error in estimation of w (Δw) by the solution of (2.15) is given in terms of the best k -sparse approximation of the outliers (s_k) and ϵ as*

$$\|\Delta w\|_2 \leq \tau^{-1}(C_0 k^{-\frac{1}{2}} \|s - s_k\|_1 + C_1 \epsilon), \quad (2.16)$$

where τ is the smallest singular value of X , and C_0, C_1 are constants which depend only on δ_{2k} .

Note that we get fact 2.1.1 by setting $\epsilon = 0$. Also note that if there are at most k outliers, $s_k = s$ and the estimation error $\|\Delta w\|_2$ is bounded by a constant times ϵ . We prove the above theorem in the next section.

2.2 Proof of the Main Theorem 2.1.1

The proof parallels that in [20]. The main assumption of the theorem is in terms of the smallest principal angle between the regressor subspace, $\text{span}(X)$, and the outlier subspaces, $\text{span}(I_T)$. This angle is best expressed in terms of orthonormal bases of the subspaces. I_T is already an orthonormal basis, but we can not say the same for X . Hence we first orthonormalize X by the reduced QR decomposition, i.e. $X = QR$ where Q is an $N \times D$ matrix which forms an orthonormal basis for X and R is an $D \times D$ upper triangular matrix. Since X is assumed to be full rank, R is a full rank matrix. Using this decomposition of X , we can solve (2.15) in an alternative way. First, we substitute $z = R w$ and then solve the problem:

$$\min_{\mathbf{s}, \mathbf{z}} \|\mathbf{s}\|_0 \text{ such that } \|\mathbf{y} - \mathbf{Q}\mathbf{z} - \mathbf{s}\|_2 \leq \epsilon. \quad (2.17)$$

w can be then be obtained by $w = R^{-1}z$. This way of solving for w is exactly equivalent to that of (2.15), and hence for solving practical problems any of the two approaches can be used. However, the proof of the theorem is based on the alternative approach. We first obtain an estimation error bound on z and then use $w = R^{-1}z$ to obtain a bound on w .

For the main proof we will need some more results. One of the results is on the relation between δ_k and a quantity μ_k , defined below, which is very similar to the concept of restricted isometry constant [22].

Definition 2.2.1. *For each integer $k = 1, 2, \dots, N$ we define a constant μ_k as the smallest number such that*

$$(1 - \mu_k)\|x\|^2 \leq \|[Q \ I_T]x\|^2 \leq (1 + \mu_k)\|x\|^2 \quad (2.18)$$

for all T with cardinality at most k .

Lemma 2.2.1. $\delta_k = \mu_k$ for all $k = 1, 2, \dots, N$.

Proof. From definition of δ_k , for any I_T with $|T| \leq k$, z and s :

$$|\langle Qz, I_T s \rangle| \leq \delta_k \|z\| \|s\| \quad (2.19)$$

where we have used $\|Qz\| = \|z\|$ and $\|I_T s\| = \|s\|$ since Q and I_T are orthonormal matrices. Writing $x = [z; s]$, $\|[Q \ I_T]x\|^2$ is given by

$$\begin{aligned} \|[Q \ I_T]x\|^2 &= \|z\|^2 + \|s\|^2 + 2\langle Qz, I_T s \rangle \\ &\leq \|z\|^2 + \|s\|^2 + 2\delta_k \|z\| \|s\| \\ &\leq \|z\|^2 + \|s\|^2 + \delta_k (\|z\|^2 + \|s\|^2), \end{aligned}$$

where we use the fact $2\|z\|\|s\| \leq \|z\|^2 + \|s\|^2$ for the last inequality. Further, using the fact $\|x\|^2 = \|z\|^2 + \|s\|^2$, we get $\|[Q I_T]x\|^2 \leq (1 + \delta_k)\|x\|^2$. Using the inequality $\langle Qz, I_T s \rangle \geq -\delta_k\|z\|\|s\|$, it is easy to show that $\|[Q I_T]x\|^2 \geq (1 - \delta_k)\|x\|^2$. Thus, we have

$$(1 - \delta_k)\|x\|^2 \leq \|[Q I_T]x\|^2 \leq (1 + \delta_k)\|x\|^2. \quad (2.20)$$

This implies $\delta_k \geq \mu_k$. However, since all the inequalities involved can be satisfied with equality, $\delta_k = \mu_k$. \square

Suppose $y = Qz + s + n$ and let z^* and s^* be the solution of (2.17) for this y .

Then

$$\|Q(z - z^*) + (s - s^*)\| \leq \|Qz + s - y\| + \|y - Qz^* - s^*\| \leq 2\epsilon \quad (2.21)$$

This follows from triangular inequality and that both z, s and z^*, s^* are feasible for problem (2.17). Let $\Delta z = z^* - z$ and $h = s^* - s$. For the rest of the proof, we are going to use the following notation: vector x_T is equal to x on the index set T and zero elsewhere. Note that this notation is different from that used for matrices, where I_T denotes the matrix consisting of column vectors from I indexed by T . Now, let's decompose h into a sum of vectors $h_{T_0}, h_{T_1}, h_{T_2}, \dots$, where each of the index set $T_i, i = 0, 1, 2, \dots$, is of cardinality k except for the last index set which can be of lesser cardinality. The index T_0 corresponds to the locations of k largest coefficients of s , T_1 to the locations of k largest coefficients of $h_{T_0^c}$, T_2 to that of the next largest k coefficients of $h_{T_0^c}$ and so on. In the main proof, we will need a bound on the quantity $\sum_{j \geq 2} \|h_{T_j}\|_2$, which we obtain first. We use the following results

from [20]:

$$\sum_{j \geq 2} \|h_{T_j}\|_2 \leq k^{-\frac{1}{2}} \|h_{T_0^c}\|_1 \quad (2.22)$$

and

$$\|h_{T_0^c}\|_1 \leq \|h_{T_0}\|_1 + 2\|s_{T_0^c}\|_1. \quad (2.23)$$

These results correspond to equation (10) and (12) in [20], with some change in notations. The first result holds because of the way h has been decomposed into $h_{T_0}, h_{T_1}, h_{T_2}, \dots$, and the second result is based on $\|s + h\|_1 \leq \|s\|_1$, which holds because $s + h = s^*$ is the minimum l_1 -norm solution of (2.17). Based on the above two equations, we have

$$\begin{aligned} \sum_{j \geq 2} \|h_{T_j}\|_2 &\leq k^{-\frac{1}{2}} \|h_{T_0}\|_1 + 2k^{-\frac{1}{2}} \|s_{T_0^c}\|_1 \\ &\leq \|h_{T_0}\|_2 + 2e_0, \end{aligned} \quad (2.24)$$

where we have used the inequality $k^{-\frac{1}{2}} \|h_{T_0}\|_1 \leq \|h_{T_0}\|_2$ and e_0 is defined as $e_0 = k^{-\frac{1}{2}} \|s_{T_0^c}\|_1$. Since by definition $s_{T_0} = s_k$, the k -sparse approximation of s , $s_{T_0^c} = s - s_k$ and hence $e_0 = k^{-\frac{1}{2}} \|s - s_k\|_1$. With these results, we are in a position to prove theorem 2.1.1.

Proof. Our goal is to find a bound on Δz , from which we can find a bound on Δw .

We do this by first finding a bound for $[\Delta z; h_{T_0 \cup T_1}]$ through bounds on the quantity

$\|Q\Delta z + h_{T_0 \cup T_1}\|^2$. Using $h_{T_0 \cup T_1} = h - \sum_{j \geq 2} h_{T_j}$, we get

$$\|Q\Delta z + h_{T_0 \cup T_1}\|^2 = \langle Q\Delta z + h_{T_0 \cup T_1}, Q\Delta z + h \rangle - \langle Q\Delta z + h_{T_0 \cup T_1}, \sum_{j \geq 2} h_{T_j} \rangle. \quad (2.25)$$

Using triangular inequality, the first term in the right hand side can be bounded as

$$\langle Q\Delta z + h_{T_0 \cup T_1}, Q\Delta z + h \rangle \leq \|Q\Delta z + h_{T_0 \cup T_1}\|_2 \|Q\Delta z + h\|_2. \quad (2.26)$$

Since $h_{T_0 \cup T_1}$ is $2k$ sparse, using (2.20), we get

$$\|Q\Delta z + h_{T_0 \cup T_1}\|_2 \leq \sqrt{1 + \delta_{2k}} \|\Delta z; h_{T_0 \cup T_1}\|_2.$$

Further, using the bound $\|Q\Delta z + h\|_2 \leq 2\epsilon$, see equation (2.21), we get

$$\langle Q\Delta z + h_{T_0 \cup T_1}, Q\Delta z + h \rangle \leq 2\epsilon \sqrt{1 + \delta_{2k}} \|\Delta z; h_{T_0 \cup T_1}\|_2. \quad (2.27)$$

Now, we look at the second term in the right hand side of equation (2.25). Since the support of $h_{T_0 \cup T_1}$ and h_{T_j} , $j \geq 2$ are different, $\langle h_{T_0 \cup T_1}, h_{T_j} \rangle = 0$ for all $j \geq 2$, and we get

$$-\langle Q\Delta z + h_{T_0 \cup T_1}, \sum_{j \geq 2} h_{T_j} \rangle = \sum_{j \geq 2} \langle Q\Delta z, -h_{T_j} \rangle \leq \delta_{2k} \|\Delta z\|_2 \sum_{j \geq 2} \|h_{T_j}\|_2, \quad (2.28)$$

where we used the definition of δ_{2k} and the fact that h_{T_j} is k -sparse, and hence also $2k$ sparse. Further, using (2.24), $\|h_{T_0}\|_2 \leq \|h_{T_0 \cup T_1}\|_2$ and $\|\Delta z\|_2 \leq \|\Delta z; h_{T_0 \cup T_1}\|_2$

$$\delta_{2k} \|\Delta z\|_2 \sum_{j \geq 2} \|h_{T_j}\|_2 \leq \delta_{2k} \|\Delta z\|_2 \|h_{T_0 \cup T_1}\|_2 + 2e_0 \delta_{2k} \|\Delta z; h_{T_0 \cup T_1}\|_2 \quad (2.29)$$

$\|\Delta z\|_2 \|h_{T_0 \cup T_1}\|_2$ can be further bounded by $\frac{1}{2} \|\Delta z; h_{T_0 \cup T_1}\|_2^2$ (by applying the inequality $2ab \leq a^2 + b^2$). Therefore,

$$\delta_{2k} \|\Delta z\|_2 \sum_{j \geq 2} \|h_{T_j}\|_2 \leq \frac{\delta_{2k}}{2} \|\Delta z; h_{T_0 \cup T_1}\|_2^2 + 2e_0 \delta_{2k} \|\Delta z; h_{T_0 \cup T_1}\|_2. \quad (2.30)$$

Finally, we obtain the following bound for $\|Q\Delta z + h_{T_0 \cup T_1}\|^2$

$$\|Q\Delta z + h_{T_0 \cup T_1}\|^2 \leq (2\epsilon \sqrt{1 + \delta_{2k}} + 2e_0 \delta_{2k}) \|\Delta z; h_{T_0 \cup T_1}\|_2 + \frac{\delta_{2k}}{2} \|\Delta z; h_{T_0 \cup T_1}\|_2^2. \quad (2.31)$$

Since $h_{T_0 \cup T_1}$ is $2k$ sparse, from equation (2.20), we get

$$(1 - \delta_{2k}) \|\Delta z; h_{T_0 \cup T_1}\|_2^2 \leq \|Q\Delta z + h_{T_0 \cup T_1}\|_2^2. \quad (2.32)$$

From the above two equations, it follows that

$$(1 - \frac{3}{2}\delta_{2k})\|\Delta z; h_{T_0 \cup T_1}\|_2 \leq 2e_0\delta_{2k} + 2\epsilon\sqrt{1 + \delta_{2k}}. \quad (2.33)$$

Since $\delta_{2k} < \frac{2}{3}$ is an assumption of the theorem, $1 - \frac{3}{2}\delta_{2k} > 0$, and hence

$$\|\Delta z; h_{T_0 \cup T_1}\|_2 \leq \frac{2e_0\delta_{2k}}{1 - \frac{3}{2}\delta_{2k}} + \frac{2\epsilon\sqrt{1 + \delta_{2k}}}{1 - \frac{3}{2}\delta_{2k}}. \quad (2.34)$$

Since $\|\Delta z\|_2 \leq \|\Delta z; h_{T_0 \cup T_1}\|_2$, we obtain

$$\|z\|_2 \leq C_0 k^{-\frac{1}{2}}\|s - s_k\|_1 + C_1\epsilon \text{ where } C_0 = \frac{2\delta_{2k}}{1 - \frac{3}{2}\delta_{2k}}, C_1 = \frac{2\sqrt{1 + \delta_{2k}}}{1 - \frac{3}{2}\delta_{2k}}. \quad (2.35)$$

Using the definition $w = R^{-1}z$, we get $\Delta w \leq \|R^{-1}\|_2\|\Delta z\|_2$, where $\|R^{-1}\|_2$ is the spectral norm of R^{-1} . Note that the spectral norm of R^{-1} is given by its largest singular value, which is the reciprocal of the smallest singular value of R . Further, since $X = QR$ and R share the same singular values, $\|R^{-1}\|_2 = \tau^{-1}$, where τ is the smallest singular value of X . Hence, we have the final result

$$\Delta w \leq \tau^{-1}(C_0 k^{-\frac{1}{2}}\|s - s_k\|_1 + C_1\epsilon). \quad (2.36)$$

□

2.3 A Bayesian Approach: Bayesian Robust Regression (BRR)

We also take a Bayesian approach towards solving (2.4). In the Bayesian approach, a (joint) prior distribution is proposed for the unknown variables of the problem and the (joint) posterior distribution of the variables is computed using the proposed prior and the likelihood distribution. Generally the mean or the mode of this posterior distribution is taken to be the solution. Since we have assumed

outliers are sparse in a dataset, an appropriate prior for s would be the *sparse prior* as introduced in [95]. However, to avoid a choice of prior on w , since we want an unbiased estimate for w , we propose to solve the problem (2.4) in two steps: First, we reduce the joint estimation problem (estimating w and s) to a simpler problem of estimating s . This is done by projecting y onto the left null space of the regressor X , which has contribution only from the outliers s . Recall that X is a full rank $N \times D$ matrix. Let C^T be an orthonormal basis for the left null space of X , i.e. C^T is a $N \times (N - D)$ ortho-normal matrix which satisfies $C \times X = 0$. Pre-multiplying (2.3) by C , we get

$$\begin{aligned} Cy &= CXw + Cs + Cn \\ z &= Cs + g, \end{aligned} \tag{2.37}$$

where $z = Cy$ and $g = Cn$, again, a small noise. We would like to solve the following problem using sparse Bayesian prior on s :

$$\min_s \|s\|_0 \text{ such that } \|z - Cs\|_2 \leq \mu \tag{2.38}$$

Note that μ is related to ϵ of the original problem (2.4). If we assume an isotropic Gaussian distribution for n , then $\mu = \sqrt{(N - D)/N} \times \epsilon$. Once we find a solution for s , we can subtract s from y and estimate w using least squares.

The Bayesian approach towards solving problems of the form (2.38) goes by the name of sparse Bayesian learning [95, 106]. The sparse prior on s is defined in the following manner: Each element of $s = [s_1 s_2 \dots s_N]^T$ is assumed to be a

zero-mean Gaussian random variable with hyper-parameter $\alpha = [\alpha_1 \alpha_2 \dots \alpha_N]^T$:

$$p(s/\alpha) = \prod_{i=1}^N \mathcal{N}(s_i|0, \alpha_i^{-1})$$

where α_i represents the inverse variance of the Gaussian distribution of s_i . Gamma distribution (hyper-prior) is specified for each of the hyper-parameters α_i . This is a hierarchical description of the prior, to get a direct description of the prior we need to marginalize out (integrate over) the hyper-parameters. For example, if a uniform distribution (obtained as a particular parameter setting of the Gamma distribution) is assumed for α_i , then by marginalizing out α_i , the improper prior $p(s_i) = 1/|s_i|$ is obtained, which is a sparsity promoting prior.

The likelihood term is given by

$$p(z/s, \sigma^2) = \mathcal{N}(z|Cs, \sigma^2 I),$$

where Gaussian distribution is assumed for the small noise g and σ is a gamma distributed random variable. With the above prior and likelihood, the maximum a posteriori estimate (MAP) of s is obtained as follows: The unknowns α_i, σ are first solved using evidence maximization technique, which maximizes the marginal distribution $p(y/\alpha_i, \sigma)$ over α_i and σ . These values are then used for obtaining the MAP estimate of s . This Bayesian algorithm has a complexity of $O(N^3)$.

2.4 Theoretical and Empirical studies of the Parameter space of Robust Regression

Three important parameters of the robust regression problem are: fraction of outliers in the dataset f , dimension of the problem D and inlier noise variance σ^2 . We study the performances of the proposed algorithms, BPRR and BRR, and compare them to that of M-estimators, LMedS and RANSAC. The performance criteria are estimation accuracy and computational complexity. We first discuss the theoretical computational complexity of the algorithms and then empirically study them for estimation accuracy.

BPRR (equation (2.15)) is a second order cone programming problem with $N + D$ variables and one cone constraint of dimension N , hence, it has a computational complexity of $O((N + D)^{2.5}N)$ [56]. BRR involves solving the sparse Bayesian learning problem, which has a complexity of $O(N^3)$ [95], and a least squares problem of complexity $O((N + D/3)D^2)$ [38]. M-estimators are usually solved using the IRLS algorithm, which has a complexity of $O((N + D/3)D^2)$. Note that none of these algorithms have any direct dependence on the outlier fraction f or inlier noise variance σ^2 . As discussed in the introduction, LMedS and RANSAC are solved using a random sampling algorithm, where D data are randomly sampled from the data set of N data and the LMedS/RANSAC cost is evaluated based on the these D data. The number of such samplings that we need to perform so as to get a successful sampling (where all the data are inliers) with a high probability p is given

by [82, 36]

$$k = \min\left(\frac{\log(1-p)}{\log(1-(1-f)^D)}, \binom{N}{D}\right). \quad (2.39)$$

Therefore, these algorithms are combinatorial in D . From the above discussion, we can conclude that BPRR, BRR and M-estimates are the feasible algorithms for high-dimensional robust regression problems, whereas LMedS and RANSAC are not.

We perform a series of experiments using synthetically generated data. For each trial in the experiments, we generate the dataset $(x_i, y_i), i = 1, 2, \dots, N$, $x_i \in \mathbb{R}^D$, $y \in \mathbb{R}$, and the model parameters $w \in \mathbb{R}^D$ in the following manner: x_i s are obtained by uniformly sampling a D -dimensional hypercube centered around the origin and w is a randomly sampled from a standard Gaussian random variable. Depending on the outlier fraction f , we randomly categorize the N indices into either inlier or outlier indices. The y_i s corresponding to the inlier indices are obtained from $y_i = \langle x_i, w \rangle + n$, where n is the inlier Gaussian noise $\mathbb{N}(0, \sigma^2)$. The y_i corresponding to the outlier indices are obtained by uniformly sampling the interval $[-r, r]$, where r is the range (maximum absolute value) of the inlier y s. Regression accuracy is measured by the l_2 norm of estimation error of w . BPRR, BRR and RANSAC need estimates of the inlier noise standard deviation, which we provide as the median absolute residual of the l_1 regression. In our experiments, we have used the MATLAB implementation of bisquare (Tukey's biweight) M-estimates, other M-estimates give similar results.

In the first experiment, we study the performances of the algorithms as a function of outlier fraction and dimension. We generate $N = 500$ synthetic data

with inlier noise variance $\sigma = 0.001$. Fig. 2.1 shows the mean estimation error over 20 trials vs. outlier fraction for dimension 2, 6 and 25. For dimension 25, we only show BPRR, BRR and M-estimates as the other algorithms LMedS and RANSAC, which are combinatorial in nature, are very slow. BRR performs very well for all the dimensions. The other algorithms are comparable with each other.

We further study the performances of the algorithms with respect to outlier fraction and dimension using the *phase transition curves*. In compressive sensing theory, where the goal is to find the sparsest solution for an under-determined system of equations, a sharp transition between success and failure of the basis pursuit algorithm has been observed: For a given level of under-determinacy, basis pursuit successfully recovers the correct solution (with high probability) if the sparsity is below a certain level and fails to do so (with high probability) if the sparsity is above that level [30], [31]. This phenomenon is termed phase transition in the compressive sensing literature and it has been used to characterize and compare the performances of several compressive sensing algorithms [62]. We also use this measure to compare the various robust regression algorithms. In the context of robust regression, the notion of under-determinacy depends on N and D . Since, there are N observations and $N + D$ unknowns in robust regression, by varying D for a fixed N we can vary the level of under-determinacy. The notion of sparsity is associated with the outlier fraction. Hence, to obtain the phase transition curves, we vary the dimension D of the problem for a fixed N and for each D find the outlier fraction where the transition from success to failure (in parameter estimation) occurs.

As before, we choose $N = 500$ and $\sigma = 0.001$. We vary D over a range of values

from 1 to 450. At each D , we vary the outlier fractions over a range of values and measure the fraction of trials in which the algorithms successfully found the *correct* solution. We consider a solution to be correct if $\frac{\|w-\hat{w}\|_2}{\|w\|_2} \leq 0.01$. Figure 2.2 shows the fraction of successful recovery vs. outlier fraction for dimensions 2 and 50 for algorithms BPRR, BRR and M-estimators; we do not show LMedS and RANSAC as these algorithms are very slow. From the figure, we can conclude that each of the algorithms exhibit a sharp transition from success to failure at a certain outlier fraction, which confirms that phase transition do occur in robust regression also. For each regression algorithm and dimension, we find that outlier fraction where the probability of success is 0.5. Similar to [62], we use logistic regression to find this outlier fraction. Figure 2.3 shows the phase transition curves of the algorithms; it is easy to conclude that BRR gives the best performance followed by BPRR and M-estimators.

We also study the effect of inlier noise variance on the performance of the algorithms. For this we fixed the dimension at 6, the outlier fraction at 0.4 and the number of data points at 500. Fig. 2.4 shows that all algorithms, except LMedS, perform well. From the above experiments, it is easy to conclude that BRR should be the preferred robust regression algorithm for low as well as high-dimensional problems.

2.5 Age Estimation From Face Images

In this section, we use the BRR algorithm for robust age estimation from face images. We use the publicly available FG-Net dataset ², which contains 1002 facial images of 82 subjects along with their ages. The dependent variable for this problem is the age and the independent variable is a geometric feature obtained by computing the flow field at 68 fiducial features on each image with respect to a reference face image.

We categorize the whole dataset into inliers and outliers using the BRR algorithm. The algorithm found 177 outliers out of the total database of 1002 images. Some of the inliers and outliers are shown in figure 2.5. Most of the outliers were images of older subjects. This could be because a linear model may not be sufficient to capture the relation between age and facial geometry for all age groups. Since, the majority of the images in the dataset are of young subjects, the older subjects become outliers with respect to them. Next, we perform a leave-one-out testing in which the regression algorithm is trained on the entire dataset except for one sample on which testing is done. We measure the mean absolute error (MAE) of age estimation for inliers and outliers separately. The results are shown in Table 2.1. The low inlier MAE and the high outlier MAE indicates that the inlier vs outlier categorization was good.

To further test BRR, we remove the outliers detected in the previous experiment and then introduce controlled outliers. We use 90% of the whole dataset as

²The fg-net aging database, <http://www.fgnet.rsunit.com>

	Inlier MAE	Outlier MAE	All MAE
BRR	3.73	19.14	6.45

Table 2.1: Mean absolute error (MAE) of age estimation for inliers and outliers using BRR. The low inlier MAE and the high outlier MAE indicates that the inlier vs outlier categorization was good.

training set and the remaining 10% as the test set. Controlled outliers are introduced only in the training set and age estimation is done on the test set by both BRR and LS. We vary the percentage of outlier on the training set and measure the MAE of age estimation on the test set. Fig. 2.6 shows that BRR gives much lower MAE as compared to LS. Table 2.2 shows the percentage of correctly detected outliers and inliers wrongly classified as outliers by BRR. BRR detects most of the outliers though it removes some the inliers.

Outlier fraction	0.1	0.2	0.3	0.4	0.5	0.6
Correctly detected outliers	97.3230	96.0524	96.1059	96.0970	95.3474	95.5894
Inlier wrongly classified as outliers	16.2609	16.2415	16.4443	16.2629	16.9942	19.6336

Table 2.2: Outlier detection rate and False alarm rate of BRR for the FG-Net dataset. BRR detects almost all of the outliers though it removes some the inliers.

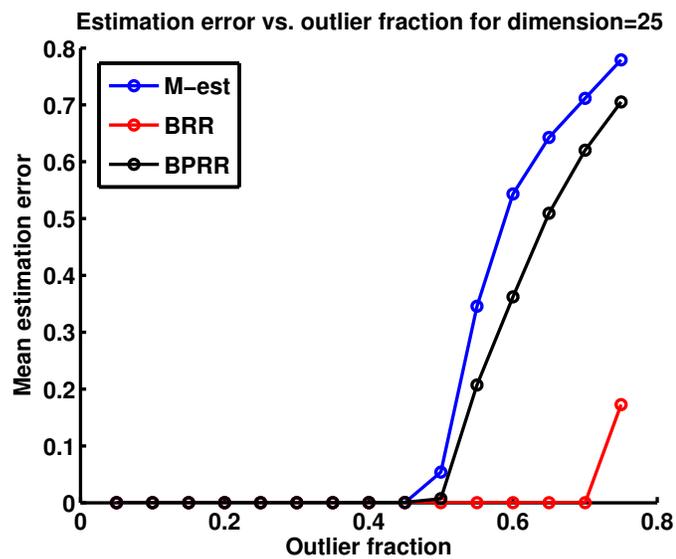
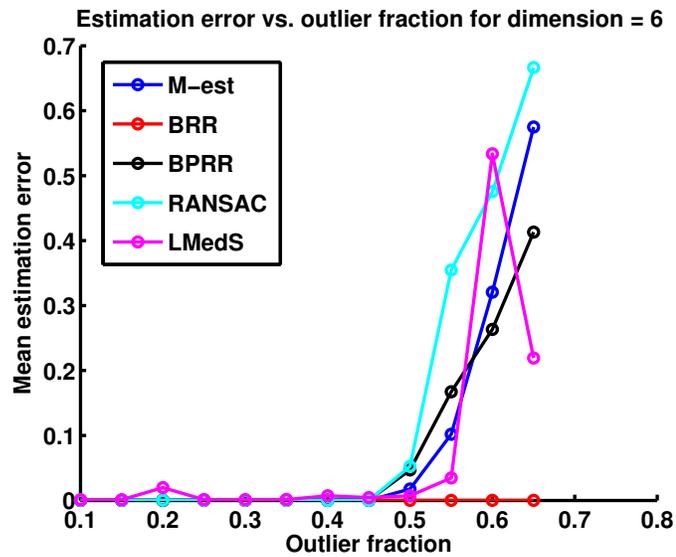
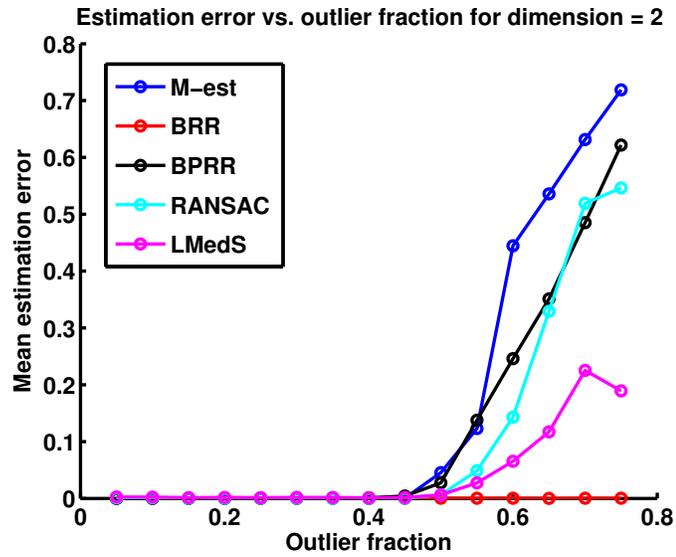


Figure 2.1: Mean estimation error vs. outlier fraction for dimension 2, 6 and 25 respectively. Only BPRR, BRR and M-estimator are shown for dimension 25 as

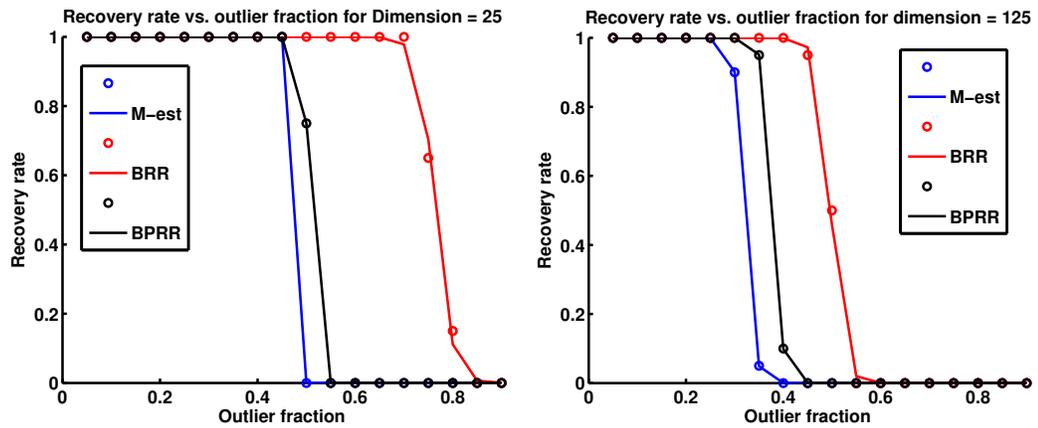


Figure 2.2: Recovery rate, i.e. the fraction of successful recovery, vs. outlier fraction for dimensions 2 and 50 for algorithms BPRR, BRR and M-estimators; we do have plots for LMedS and RANSAC as these algorithms are very slow. From the figure we can conclude that each of the algorithms exhibit a sharp transtion from success to failure at a certain outlier fraction.

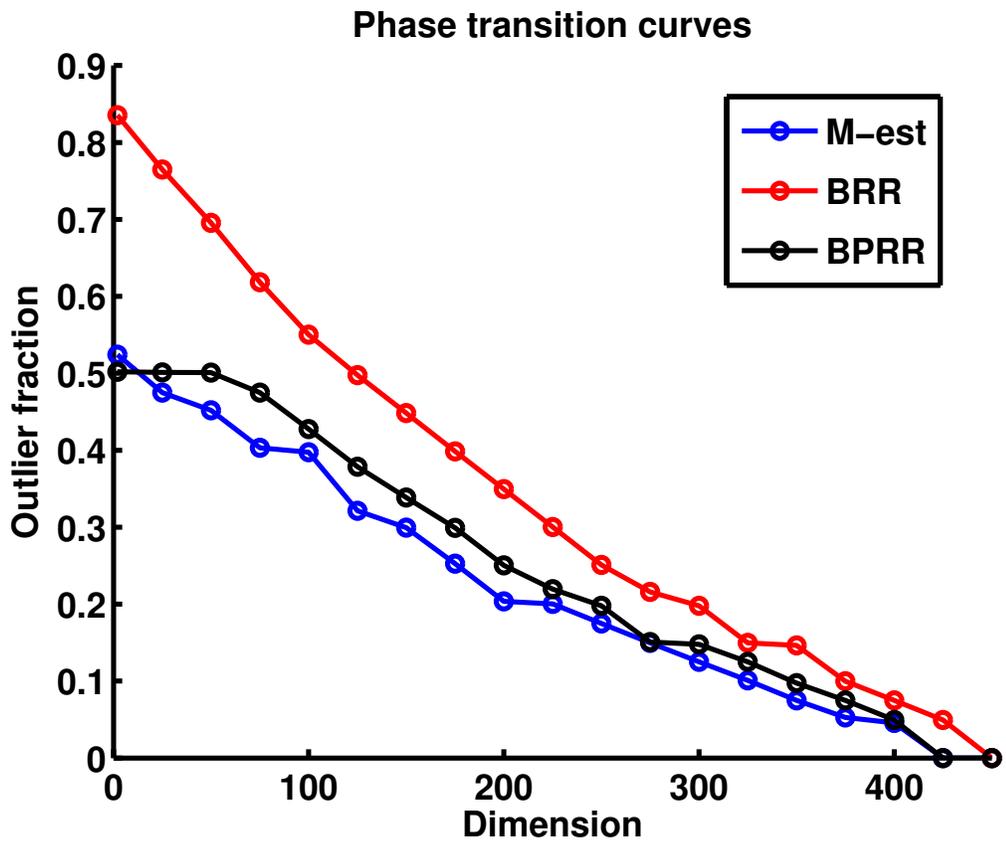


Figure 2.3: Phase transition curves of the algorithms BPRR, BRR and M-estimator.

BRR gives the best performance followed by BPRR and M-estimator.

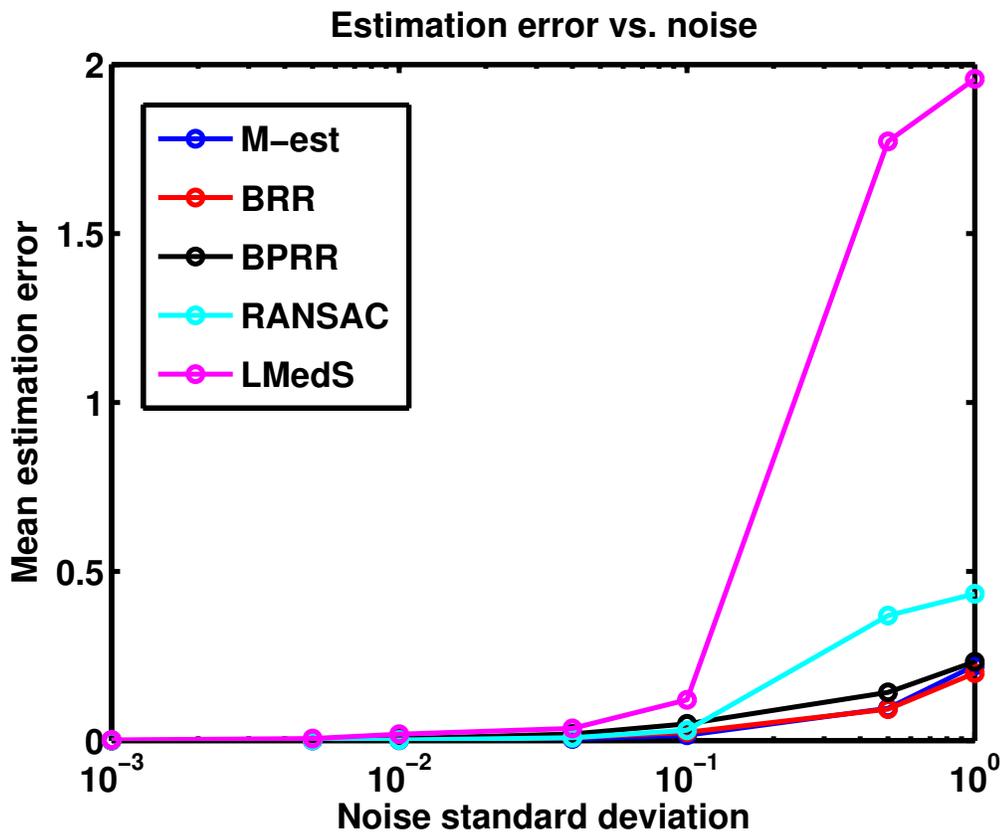


Figure 2.4: Mean angle error vs. inlier noise standard deviation for dimension 6 and 0.4 outlier fraction. All algorithms, except LMedS, perform well.

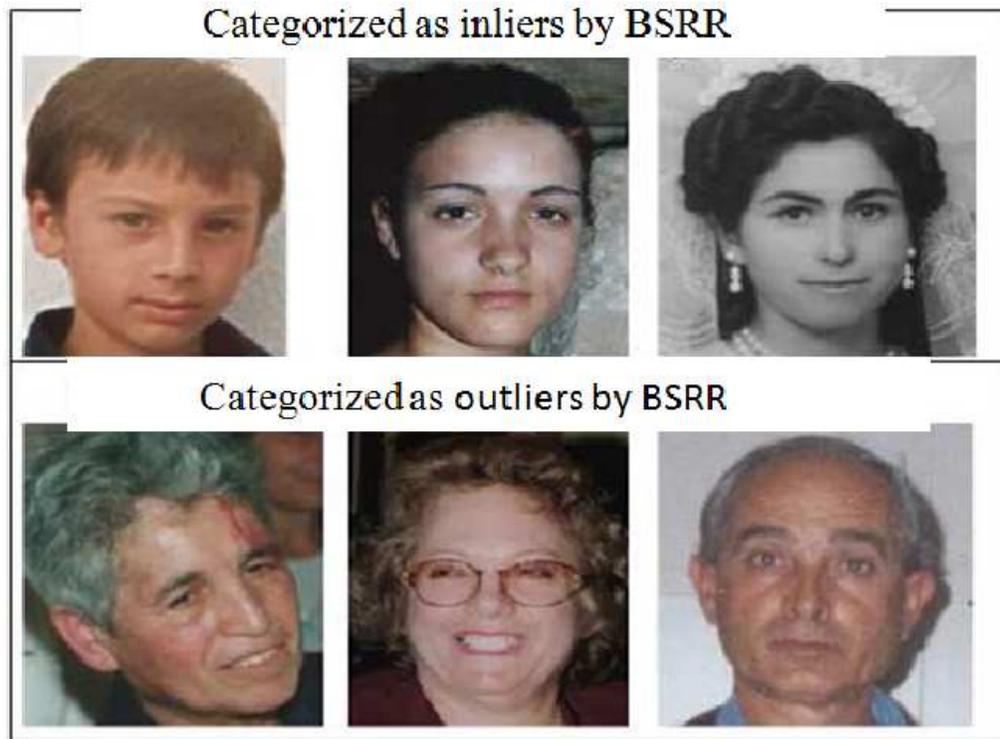


Figure 2.5: Some outlier and inliers found by BRR. Most of the outliers were images of older subjects. This could be because a linear (regression) model may not be sufficient to capture the relation between age and facial geometry for all age groups. Since, the majority of the images in the dataset are of young subjects, the older subjects become outliers with respect to them.

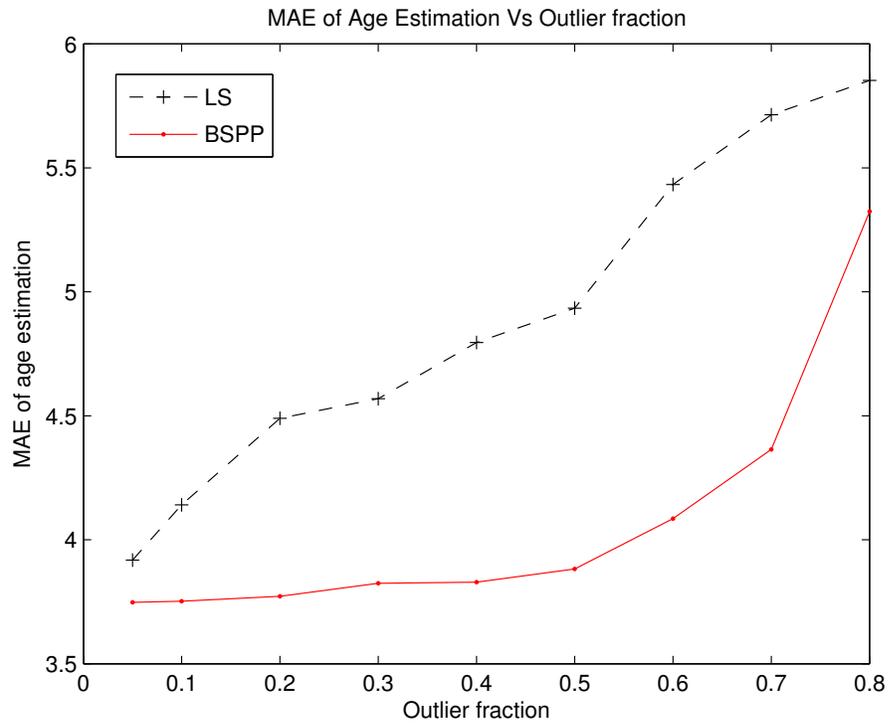


Figure 2.6: Mean absolute error (MAE) of age estimation Vs outlier fraction. BRR has almost constant MAE until outlier fraction increases beyond 0.5.

Chapter 3

Robust RVM Regression Using Sparse Outlier Model

Kernel regression techniques such as Support Vector Regression (SVR) [103], RVM regression [95] and Gaussian processes [79] are widely used for solving many vision problems. Some examples are age estimation from facial images [54, 53, 37, 40], head pose estimation [68], 3D human pose estimation [4] and lighting estimation [86]. Recently, kernel regression has also been used for solving image processing problems such as image de-noising and image reconstruction with a great deal of success [92, 93]. However, many of these kernel regression methods, especially the RVM, are not robust to outliers in the training dataset, and hence, will produce unreliable estimates in the presence of outliers.

To make the RVM model robust to outliers, we decompose the noise term in the RVM model into an outlier noise term, which we assume to be sparse, and a Gaussian noise term. The assumption of outliers being sparse is justified as we generally expect the majority of the data to be inliers. During inference, we estimate the outlier noise along with the model parameters. We present two approaches for solving this estimation problem: 1) a Bayesian approach and 2) an regularization-based approach. In the Bayesian approach, we assume a joint sparse prior for the model parameters and the outliers, and then solve the Bayesian inference problem. The mean of the posterior distribution of the model parameters is used for prediction.

The joint sparse assumption for the model parameters and the outliers, effectively, makes the robust RVM problem a bigger RVM problem with the advantage that we can use a fast algorithm, developed for the RVM [96], to solve this problem. In the regularization-based approach, we propose to minimize the l_0 norm of the model parameters and the outliers, subject to a certain amount of observation error (which depends on the inlier noise variance). However, this is a combinatorial optimization problem and hence can not be used for solving large-scale regression problems. So, we propose to relax the problem to an l_1 regularized problem, which is of the same form as the basis pursuit denoising problem [26]. We then empirically evaluate the robust algorithms by varying the following parameters of the robust regression problem: the outlier fraction, the inlier noise variance and the number of data points in the training dataset. We further demonstrate the effectiveness of the robust approaches in solving the image denoising and age estimation problems.

Related works: Robust versions of the RVM regression have been proposed in [35], [97] and [108]. In [35], the noise term is modeled as a mixture of Gaussian (for the inlier noise), and uniform or Gaussian with large variance for the outlier noise. But the mixture density model makes inference difficult; a variational method is used for solving this problem making it computationally much more expensive than the RVM. In [97], a Student’s t-distribution is assumed for the noise, and the parameters of the distribution are estimated along with the model parameters. Though, this is a very elegant approach, a variational method is used for inference, which makes it computationally expensive. In [108], a trimmed likelihood function is minimized over a ‘trimmed’ subset that does not include the outliers. The robust trimmed

subset and the model parameters are found by an iterative re-weighting strategy, which at each iteration solves the RVM regression problem over the current trimmed subset. However, the method needs an initial robust estimate of the trimmed subset, which determines the accuracy of the final solution. It also needs many iterations, where in each iteration a RVM regression problem is solved, and this makes it slow.

The organization of the rest of this chapter is as follows: in section 3.1, we introduce the RVM regression model and its proposed robust versions: robust Bayesian RVM (RB-RVM) and basis pursuit RVM (BP-RVM). In section 3.2, we evaluate the proposed robust algorithms on synthetically generated data. In section 3.3, we use the RB-RVM algorithm for robust image denoising and in section 3.4, for solving the age estimation problem.

3.1 Robust RVM Regression

For both the robust Bayesian approach and the robust regularization approach, we replace the Gaussian noise assumption in the RVM formulation by an implicit heavy-tailed distribution. This is achieved by decomposing the noise term into a sparse outlier noise term and a Gaussian noise term. The outliers are then treated as unknowns and are estimated together with the model parameters. In the following sub-sections, we first describe the RVM regression model, followed by the robust Bayesian and regularization approaches.

3.1.1 Model Specification

Let $(\mathbf{x}_i, y_i), i = 1, 2, \dots, N$ be the given training dataset with dependent variables $y_i, i = 1, 2, \dots, N$ and independent variables $\mathbf{x}_i, i = 1, 2, \dots, N$. In the RVM formulation, y_i is related to \mathbf{x}_i by the model

$$y_i = \sum_{j=1}^N w_j K(\mathbf{x}_i, \mathbf{x}_j) + w_0 + e_i \quad (3.1)$$

where with each \mathbf{x}_j , there is an associated kernel function $K(\cdot, \mathbf{x}_j)$, and e_i is the Gaussian noise. The objective is to estimate the weight vector $\mathbf{w} = [w_0, w_1, \dots, w_N]^T$ using the training dataset. Once this is done, we can predict the dependent variable y for any new \mathbf{x} by

$$y = \sum_{i=j}^N w_j K(\mathbf{x}, \mathbf{x}_j) + w_0 \quad (3.2)$$

In the presence of outliers, Gaussian noise is not an appropriate assumption for e_i . We propose to split the noise e_i into two components: a Gaussian component n_i and a component due to outliers s_i , which we assume to be sparse. With this, we have

$$y_i = \sum_{j=1}^N w_j K(\mathbf{x}_i, \mathbf{x}_j) + w_0 + n_i + s_i \quad (3.3)$$

In matrix-vector form, this is given by

$$\mathbf{y} = \mathbf{\Phi} \mathbf{w} + \mathbf{n} + \mathbf{s} \quad (3.4)$$

where $\mathbf{y} = [y_1, \dots, y_N]^T$, $\mathbf{n} = [n_1, \dots, n_N]^T$, $\mathbf{s} = [s_1, \dots, s_N]^T$ and $\mathbf{\Phi}$ is a $N \times (N+1)$ matrix with

$$\mathbf{\Phi} = [\phi(\mathbf{x}_1), \phi(\mathbf{x}_2), \dots, \phi(\mathbf{x}_N)]^T, \quad (3.5)$$

where $\phi(\mathbf{x}_i) = [1, K(\mathbf{x}_i, \mathbf{x}_1), K(\mathbf{x}_i, \mathbf{x}_2), \dots, K(\mathbf{x}_i, \mathbf{x}_N)]^T$. The two unknowns \mathbf{w} and \mathbf{s} can be augmented into a single unknown vector $\mathbf{w}_s = [\mathbf{w}^T \mathbf{s}^T]^T$ and the above equation can be written as

$$\mathbf{y} = \Psi \mathbf{w}_s + \mathbf{n} \quad (3.6)$$

where $\Psi = [\Phi | \mathbf{I}]$ is a $N \times (2N + 1)$ matrix with \mathbf{I} , a $N \times N$ identity matrix.

3.1.2 Robust Bayesian RVM (RB-RVM)

In the Bayesian approach, we estimate the joint posterior distribution of \mathbf{w} and \mathbf{s} , given the observations \mathbf{y} and the prior distributions on \mathbf{w} and \mathbf{s} . We then use the mean of the posterior distribution of \mathbf{w} for prediction (3.2). The posterior variance also provides us with a measure of uncertainty in the prediction.

The joint posterior distribution of \mathbf{w} and \mathbf{s} is given by

$$p(\mathbf{w}, \mathbf{s} | \mathbf{y}) = \frac{p(\mathbf{w}, \mathbf{s}) p(\mathbf{y} | \mathbf{w}, \mathbf{s})}{p(\mathbf{y})} \quad (3.7)$$

From (3.6), the likelihood term $p(\mathbf{y} | \mathbf{w}, \mathbf{s})$ is given by

$$p(\mathbf{y} | \mathbf{w}, \mathbf{s}) = \mathcal{N}(\Psi \mathbf{w}_s, \sigma^2 \mathbf{I}) \quad (3.8)$$

where σ^2 is the inlier Gaussian noise variance. To proceed further, we need to specify the prior distribution $p(\mathbf{w}, \mathbf{s})$. We assume that \mathbf{w} and \mathbf{s} are independent: $p(\mathbf{w}, \mathbf{s}) = p(\mathbf{w}) p(\mathbf{s})$. Next, we keep the same ‘sparsity promoting’ prior for \mathbf{w} as in RVM [95], that is,

$$p(\mathbf{w} | \boldsymbol{\alpha}) = \prod_{i=0}^N \mathcal{N}(w_i | 0, \alpha_i^{-1}) \quad (3.9)$$

where $\boldsymbol{\alpha} = [\alpha_0, \alpha_1, \dots, \alpha_N]^T$ is a vector of $(N + 1)$ hyper-parameters. A uniform distribution (hyper-prior) is assumed for each of the α_i (For more details, please see [95]).

For \mathbf{s} , we specify a similar sparsity promoting prior given by

$$p(\mathbf{s}|\boldsymbol{\beta}) = \prod_{i=1}^N \mathcal{N}(s_i|0, \beta_i^{-1}) \quad (3.10)$$

where $\boldsymbol{\beta} = [\beta_1, \beta_2, \dots, \beta_N]^T$ is a vector of N hyper-parameters, and each of the β_i follows a uniform distribution. This completes the description of the likelihood $p(\mathbf{y}|\mathbf{w}, \mathbf{s})$ and the prior $p(\mathbf{w}, \mathbf{s})$. Next, we proceed to the inference stage.

3.1.2.1 Inference

Our inference method follows the RVM inference steps. We first find point-estimates for the hyper-parameters $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ and the inlier noise variance σ^2 by maximizing $p(\mathbf{y}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2)$ with respect to these parameters, where $p(\mathbf{y}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2)$ is given by

$$p(\mathbf{y}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2) = \int p(\mathbf{y}|\mathbf{w}, \mathbf{s}, \sigma^2) p(\mathbf{w}|\boldsymbol{\alpha}) p(\mathbf{s}|\boldsymbol{\beta}) d\mathbf{w} d\mathbf{s} \quad (3.11)$$

Since all the distributions in the right hand side are Gaussian with zero mean, it can be shown that $p(\mathbf{y}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2)$ is a zero-mean Gaussian distribution with covariance matrix $\sigma^2 \mathbf{I} + \boldsymbol{\Psi} \mathbf{A}^{-1} \boldsymbol{\Psi}^T$, where $\mathbf{A} = \text{diag}(\alpha_0, \dots, \alpha_N, \beta_1, \dots, \beta_N)$. The maximization of $p(\mathbf{y}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2)$ with respect to the hyper-parameters $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ and the noise variance σ^2 is known as evidence maximization and can be solved by an EM algorithm [95] or a faster implementation proposed in [96]. We will refer to these estimated parameters as $\boldsymbol{\alpha}_{MP}$, $\boldsymbol{\beta}_{MP}$ and σ_{MP}^2 .

With this point estimation of the hyper-parameters and the noise variance, the (conditional) posterior distribution $p(\mathbf{w}, \mathbf{s} | \mathbf{y}, \boldsymbol{\alpha}_{MP}, \boldsymbol{\beta}_{MP}, \sigma_{MP}^2)$ is given by

$$\frac{p(\mathbf{y} | \mathbf{w}, \mathbf{s}, \sigma_{MP}^2) p(\mathbf{w} | \boldsymbol{\alpha}_{MP}) p(\mathbf{s} | \boldsymbol{\beta}_{MP})}{p(\mathbf{y} | \boldsymbol{\alpha}_{MP}, \boldsymbol{\beta}_{MP}, \sigma_{MP}^2)} \quad (3.12)$$

Since all the terms in the numerator are Gaussian, it can be shown that this is again a Gaussian distribution with covariance and mean given by

$$\boldsymbol{\Sigma} = (\sigma_{MP}^{-2} \boldsymbol{\Psi}^T \boldsymbol{\Psi} + \mathbf{A}_{MP})^{-1} \text{ and } \boldsymbol{\mu} = \sigma_{MP}^{-2} \boldsymbol{\Sigma} \boldsymbol{\Psi}^T \mathbf{y} \quad (3.13)$$

where $\mathbf{A}_{MP} = \text{diag}(\alpha_{MP0}, \dots, \alpha_{MPN}, \beta_{MP1}, \dots, \beta_{MPN})$.

To obtain the posterior distribution $p(\mathbf{w}, \mathbf{s} | \mathbf{y})$, we need to integrate out $\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2$ from $p(\mathbf{w}, \mathbf{s} | \mathbf{y}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2)$, that is,

$$p(\mathbf{w}, \mathbf{s} | \mathbf{y}) = \int p(\mathbf{w}, \mathbf{s} | \mathbf{y}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2) p(\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2 | \mathbf{y}) d\boldsymbol{\alpha} d\boldsymbol{\beta} d\sigma^2 \quad (3.14)$$

However, this is analytically intractable; it has been empirically observed in [95], that for predictive purposes, $p(\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma^2 | \mathbf{y})$ is very well approximated by $\delta(\boldsymbol{\alpha}_{MP}, \boldsymbol{\beta}_{MP}, \sigma_{MP}^2)$.

With this approximation, we have

$$p(\mathbf{w}, \mathbf{s} | \mathbf{y}) = p(\mathbf{w}, \mathbf{s} | \mathbf{y}, \boldsymbol{\alpha}_{MP}, \boldsymbol{\beta}_{MP}, \sigma_{MP}^2) \quad (3.15)$$

Thus, the desired joint posterior distribution of \mathbf{w} and \mathbf{s} is Gaussian with the posterior covariance and mean given by (3.13). For prediction, we use the mean as an estimate of w in the prediction model (3.2).

3.1.2.2 Prediction

We use the prediction model (3.2) to predict \hat{y} for any new data $\hat{\mathbf{x}}$. The predictive distribution of \hat{y} is given by

$$p(\hat{y}|\mathbf{y}, \boldsymbol{\alpha}_{MP}, \sigma_{MP}^2) = \int p(\hat{y}|\mathbf{w}, \sigma_{MP}^2)p(\mathbf{w}|\mathbf{y}, \boldsymbol{\alpha}_{MP}) d\mathbf{w} \quad (3.16)$$

where the posterior distribution of \mathbf{w} , $p(\mathbf{w}|\mathbf{y}, \boldsymbol{\alpha}_{MP})$, can be easily obtained from the joint posterior distribution $p(\mathbf{w}, \mathbf{s}|\mathbf{y}, \boldsymbol{\alpha}_{MP}, \boldsymbol{\beta}_{MP}, \sigma_{MP}^2)$. $p(\mathbf{w}|\mathbf{y}, \boldsymbol{\alpha}_{MP})$ is a Gaussian distribution with mean and covariance given by the mean and covariance of the parameter part (\mathbf{w}) of the \mathbf{w}_s vector, that is,

$$\boldsymbol{\Sigma}_w = \boldsymbol{\Sigma}(1 : N + 1, 1 : N + 1) \text{ and } \boldsymbol{\mu}_w = \boldsymbol{\mu}(1 : N + 1) \quad (3.17)$$

With this, it can be shown that the predictive distribution of \hat{y} is Gaussian with mean $\hat{\mu}$ and variance $\hat{\sigma}^2$ given by

$$\hat{\mu} = \boldsymbol{\mu}_w^T \boldsymbol{\phi}(\hat{\mathbf{x}}) \text{ and } \hat{\sigma}^2 = \sigma_{MP}^2 + \boldsymbol{\phi}(\hat{\mathbf{x}})^T \boldsymbol{\Sigma}_w \boldsymbol{\phi}(\hat{\mathbf{x}}) \quad (3.18)$$

3.1.2.3 Advantage over other Robust RVM Algorithms

The proposed robust Bayesian formulation (RB-RVM) is very similar to the original RVM formulation. All we have to do is, instead of inferring just the parameter vector \mathbf{w} , infer the joint parameter-outlier vector \mathbf{w}_s , by replacing the $\boldsymbol{\Phi}$ matrix with the corresponding $\boldsymbol{\Psi} = [\boldsymbol{\Phi}|\mathbf{I}]$ matrix, and use only the parameter part of the estimated \mathbf{w}_s for prediction. It is this simple modification of the original RVM that gives RB-RVM the computational advantage over [35, 97, 108] because we can use an existing fast implementation of RVM [96] to solve the robust RVM problem.

3.1.3 Basis Pursuit RVM (BP-RVM)

A very similar objective, as in the Bayesian approach, can be achieved by solving the following optimization problem:

$$\min_{\mathbf{w}_s} \|\mathbf{w}_s\|_0 \text{ subject to } \|\mathbf{y} - \Psi \mathbf{w}_s\|_2 \leq \epsilon \quad (3.19)$$

where $\|\mathbf{w}_s\|_0$ is the l_0 norm, which counts the number of non-zero elements in \mathbf{w}_s . The cost function promotes a sparse solution for \mathbf{w}_s and the constraint term is essentially the likelihood term of the Bayesian approach, with ϵ related to the inlier noise variance σ^2 . \mathbf{w} obtained after solving this problem can be used for prediction. However, this is a combinatorial problem; hence, it cannot be solved directly. This problem has been studied extensively in the sparse representation literature [26, 32], where a convex relaxation of the problem is solved:

$$\min_{\mathbf{w}_s} \|\mathbf{w}_s\|_1 \text{ subject to } \|\mathbf{y} - \Psi \mathbf{w}_s\|_2 \leq \epsilon \quad (3.20)$$

where the l_0 norm in the cost function is replaced by the l_1 norm, which makes it a convex problem; hence, it can be solved in polynomial time. This approach is known as Basis Pursuit Denoising (BPD) [26, 32], and we will refer to the robust algorithm based on BPD as the Basis Pursuit RVM (BP-RVM). Initially, the justification for using the l_1 norm approximation was based on empirical observations [26]. However, recently in [23, 32], it has been shown that if \mathbf{w}_s is sparse to begin with, then under certain condition (‘Restricted Isometry Property’ or ‘incoherence’) on the matrix Ψ , (3.19) and (3.20) will have the same solution up to a bounded uncertainty due to ϵ . However, in our case the matrix Ψ depends on the training dataset and the

associated kernel function, and it might not satisfy the desired conditions mentioned above.

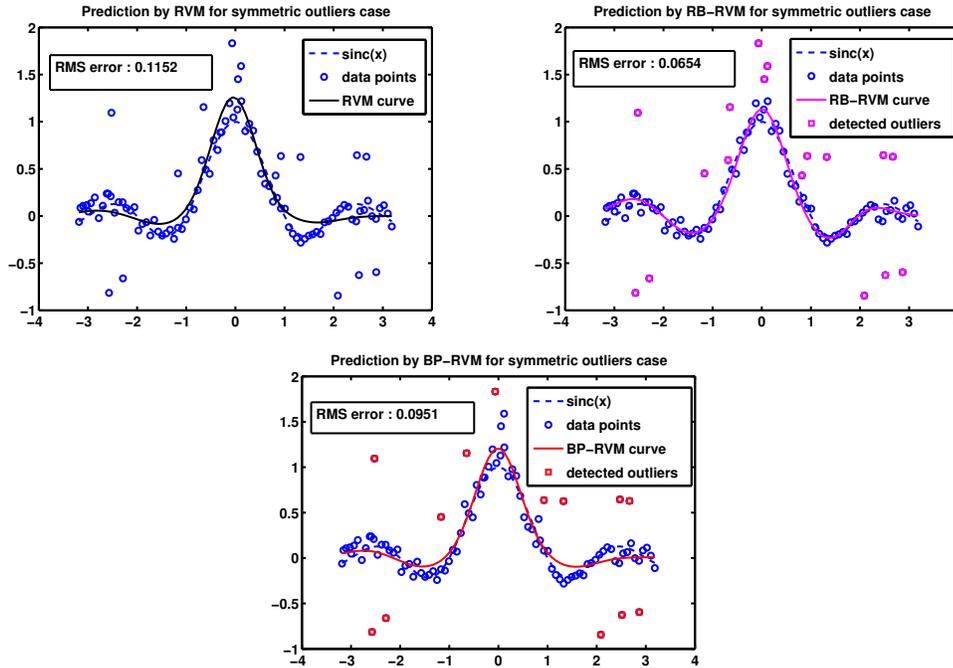


Figure 3.1: Prediction by the three algorithms: RVM, RB-RVM and BP-RVM in the presence of symmetric outliers for $N = 100$, $f = 0.2$ and $\sigma = 0.1$. Data which are enclosed by a box are the outliers found by the robust algorithms. Prediction error are also shown in the figures. RB-RVM gives the lowest prediction error.

3.2 Empirical Evaluation

In this section, we empirically evaluate the proposed robust versions of the RVM, RB-RVM and BP-RVM, with respect to the baseline RVM. We consider three important intrinsic parameters of the robust regression problem: the outlier fraction (f), the inlier noise variance (σ^2) and the number of training data points (N), and study the performance of the three algorithms for different settings of these

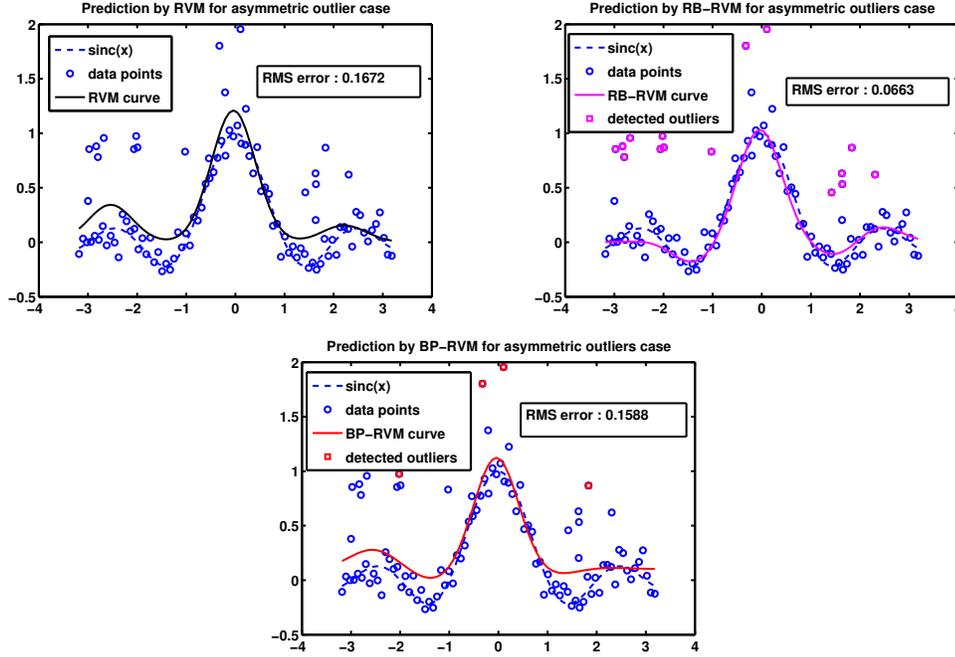


Figure 3.2: Prediction by the three algorithms: RVM, RB-RVM and BP-RVM in the presence of asymmetric outliers for $N = 100$, $f = 0.2$ and $\sigma = 0.1$. Data which are enclosed by a box are the outliers found by the robust algorithms. Prediction error are also shown in the figures. Clearly, RB-RVM gives the best result.

parameters.¹ Next, we describe the experimental setup, which is quite similar to that of [35].

We generate our training data using the normalized sinc function $sinc(x) = \sin(\pi x)/(\pi x)$. y_i of the inlier data are obtained by adding a Gaussian noise $\mathcal{N}(0, \sigma^2)$ to $sinc(x_i)$. For the outliers, we consider two generative models: 1) symmetric and 2) asymmetric. In the symmetric model, y_i is obtained by adding a uniform noise of range $[-1, +1]$ to $sinc(x_i)$, and in the asymmetric model, y_i is obtained by

¹For solving RVM and RB-RVM, we have used the publicly available code in <http://www.vectoranomaly.com/downloads/downloads.htm>. For solving BP-RVM, we have used l1-magic: <http://www.acm.caltech.edu/l1magic/>

adding a uniform noise of range $[0, +1]$ to $\text{sinc}(x_i)$. With each training data x_j , we associate a Gaussian kernel: $K(x, x_j) = \exp(-(x - x_j)^2/r^2)$, with $r = 2$. Figures 3.1 and 3.2 show the performance of the three algorithms for the symmetric and asymmetric outlier cases for $N = 100$, $f = 0.2$ and $\sigma = 0.1$. The performance criterion used for comparison is the root mean square (RMS) prediction error. Note that, after inference, robust methods can also classify the training data as inliers or outliers. We classify a data as an outlier if the prediction error (absolute difference between the predicted and the observed value) is greater than three times the inlier noise standard deviation, which is also estimated during inference. From figures 3.1 and 3.2, we conclude that RB-RVM gives the lowest prediction error, followed by BP-RVM and RVM. In the following sections, we study the performance of the algorithms by varying the intrinsic parameters: f , σ and N .

Varying the Outlier fraction: We vary the outlier fraction f , with the other parameters fixed at $N = 100$ and $\sigma = 0.1$. Figure 3.3 shows the prediction error vs. outlier fraction for the symmetric and asymmetric outliers cases. For both the cases, RB-RVM gives the best result. For the symmetric case, BP-RVM gives lower prediction error than RVM but for the asymmetric case they give similar result.

Varying the Inlier Noise Std: We vary the inlier noise standard deviation σ , with the other parameters fixed at $N = 100$ and $f = 0.2$. Figure 3.4 shows that RB-RVM gives the lowest prediction error until about $\sigma = 0.2$, after which RVM gives better result. This is because in our experimental setup, at approximately $\sigma = 0.3$, the distinction between the inliers and outliers cease to exist. For Gaussian distribution, most of the probability density mass lies within 3σ of the mean, and any

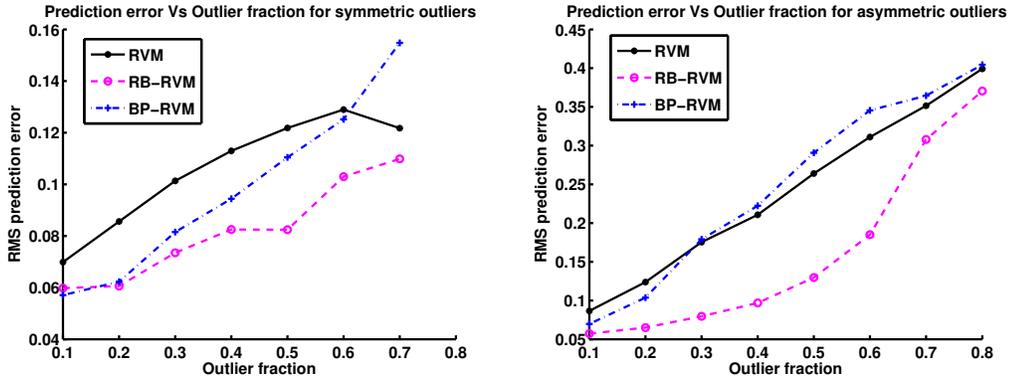


Figure 3.3: Prediction error vs. outlier fraction for the symmetric and asymmetric outlier cases. RB-RVM gives the best result for both the cases. For the symmetric case, BP-RVM gives lower prediction error than RVM but for the asymmetric case they give similar result.

data within this region can be considered as inliers and those outside as outliers. Thus, for $\sigma = 0.3$, $3\sigma = 0.9$; most of the outliers will be within this range and effectively become inliers.

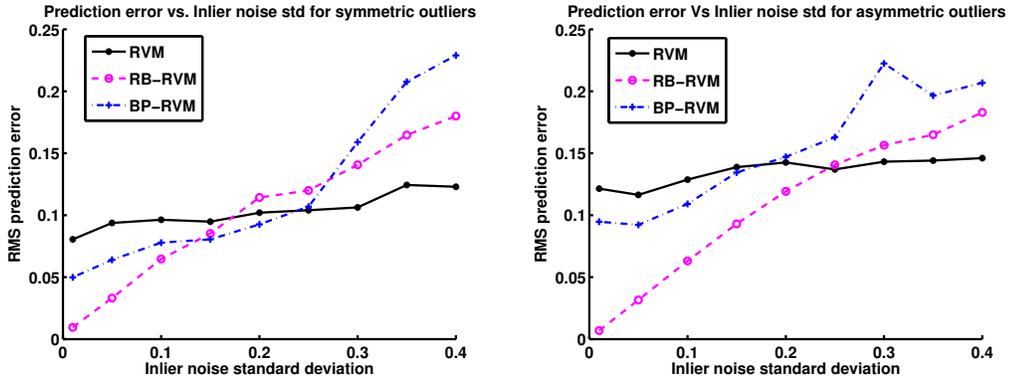


Figure 3.4: Prediction error vs. inlier noise standard deviation for the symmetric and asymmetric outlier cases. RB-RVM gives the lowest prediction error until about $\sigma = 0.2$, after which RVM gives better result. This is because for our experimental setup, at approximately $\sigma = 0.3$, the distinction between the inliers and outliers cease to exist.

Varying the Number of Data Points: We vary the number of data points N , with $f = 0.2$ and $\sigma = 0.1$. Figure 3.5 shows that the performance of all the three

algorithms improve with increasing N .

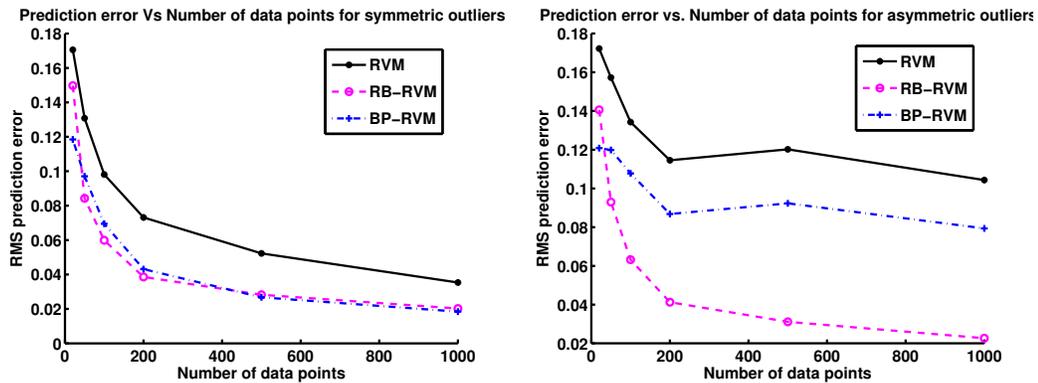


Figure 3.5: Prediction error vs. number of data points for the symmetric and asymmetric outlier cases. For all the three algorithms, performance improves with increasing N .

Discussion: We conclude that in presence of outliers RB-RVM and BP-RVM perform better than RVM. The performance of BP-RVM is poor as compared to RB-RVM; this indicates that the l_1 norm relaxation (3.20) is not a good approximation of the l_0 norm problem (3.19), when Ψ does not satisfy the desired Restricted Isometry Property [23]. Henceforth, we will only consider RB-RVM for solving the image denoising and age regression problems.

3.3 Robust Image Denoising

Recently, kernel regression has been used for solving a number of traditional image processing tasks such as image denoising, image interpolation and super-resolution with a great deal of success [92, 93]. The success of these kernel regression methods prompted us to test RB-RVM for solving the problem of image denoising in the presence of salt and pepper noise. Salt and pepper noise are randomly occurring white and black pixels in an image and can be considered as outliers.

Any image $I(x, y)$ can be considered as a surface over a 2D grid. Given a noisy image, we can use regression to learn the relation between the intensity and the 2D grid of the image. If some kind of a local smoothness is imposed by the regression machine, we can use it for denoising the image. Here, we consider RVM and RB-RVM for achieving this purpose. We divide the image into many (overlapping) patches, and for each patch we infer the parameters of RVM and RB-RVM. We then use the inferred parameters for predicting the intensity of the central pixel of the patch, which is the denoised intensity at that pixel. This is done for all the pixels of the image to obtain the denoised image. Motivated by [92], we consider a composition of Gaussian and polynomial kernel as the choice of kernel in our regression machines. The Gaussian kernel is defined as $K_g(\mathbf{x}, \mathbf{x}_j) = \exp(-\|\mathbf{x} - \mathbf{x}_j\|^2/r^2)$, where r is the scale of the Gaussian kernel, and the polynomial kernel is defined as $K_p(x, x_j) = (\mathbf{x}^T \mathbf{x}_j + 1)^p$, where p is the order of the polynomial kernel. We consider kernels of the form: $K(\mathbf{x}, \mathbf{x}_j) = K_g(\mathbf{x}, \mathbf{x}_j)K_p(\mathbf{x}, \mathbf{x}_j)$.

To test the proposed kernel denoising algorithms, we add 20% salt and pepper noise to the original images. For RVM and RB-RVM, we choose patch size of 6×6 , $r = 2.1$ and $p = 1$. Figure 3.6 shows the image denoising result by RVM, RB-RVM and 3×3 median filter. The denoised images and the corresponding RMSE values show that RB-RVM gives the best denoising result. Next, we vary the amount of salt and pepper noise, and obtain the mean RMSE value over the commonly used images of Barbara, House, Boat, Baboon, Pepper and Elaine. Figure 3.7 shows that RB-RVM gives better result than the median filter, which is the most commonly used filter for denoising images with salt and pepper noise. Further, we test RB-

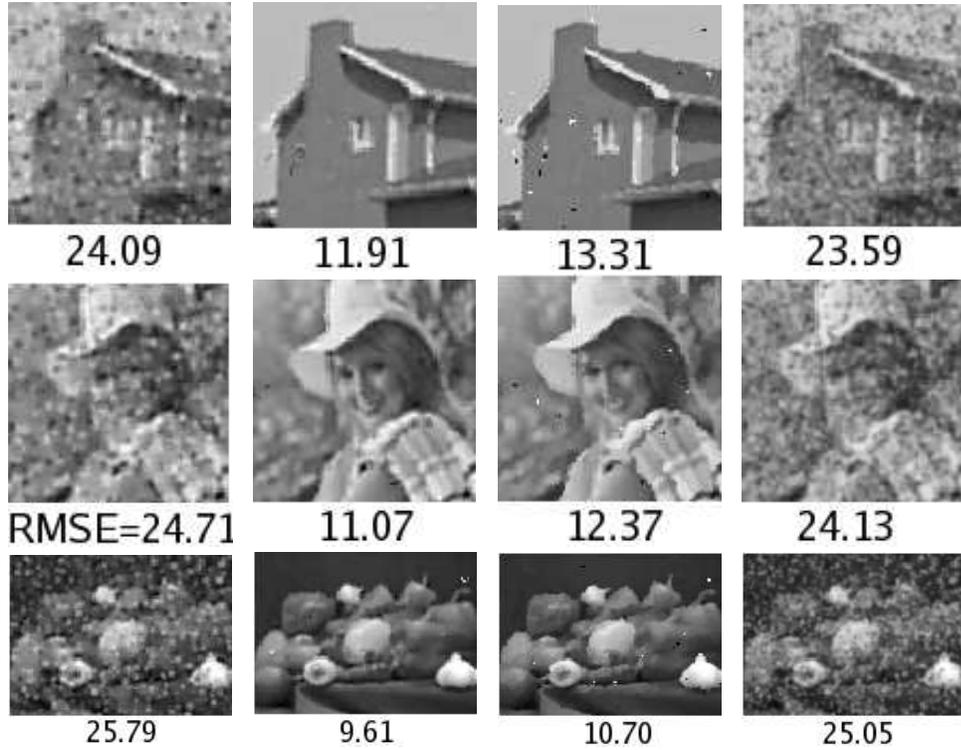


Figure 3.6: Results on Salt and pepper noise removal: first column: RVM, second column: RB-RVM, third column: Median filter, fourth column: Gaussian filter. The RMSE values are also shown in the figure; RB-RVM gives the best result.

RVM for denoising an image corrupted by a mixture of Gaussian noise of $\sigma = 5$ and 5% salt and pepper noise. From figure 3.8, we conclude again that RB-RVM gives much better denoising result as compared to RVM.

3.4 Age Estimation from Facial Images

The goal of facial age estimation is to estimate the age of a person from his/her image. The most common approach for solving this problem is to extract some relevant features from the image, and then learn the functional relationship between these features and the age of the person using regression techniques [54, 53, 37, 40].

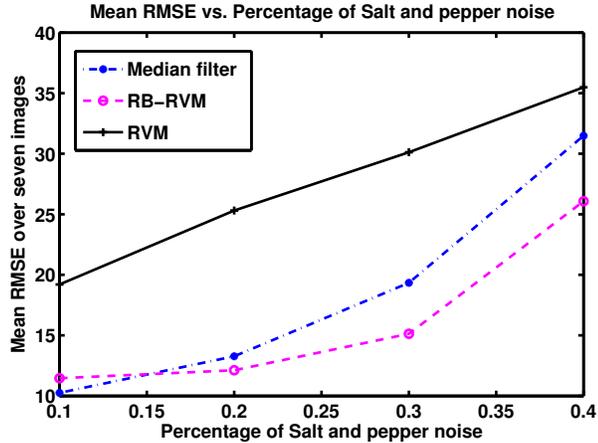


Figure 3.7: Mean RMSE value over seven images vs. percentage of salt and pepper noise. RB-RVM gives better performance than the median filter.

Here, we intend to test the RB-RVM regression for the age estimation problem. For our experiments, we use the publicly available FG-Net dataset [1], which contains 1002 images of 82 subjects at different ages. As a choice of features, we use geometric features proposed in [100], which are obtained by computing the 'flow field' at 68 fiducial points with respect to a reference face image.

To decide on a particular kernel for regression, we perform leave-one-person-out testing, by RB-RVM, for different choices of kernel. Table 3.1 shows the mean absolute error (MAE) of age prediction for different values of the scale parameter r of the Gaussian kernel. $r = 0.2$ gives the best result, and we use this value of r for all the subsequent experiments. Next, we use RB-RVM to categorize the whole dataset into inliers and outliers. The algorithm found 90 outliers; some of the inliers and outliers are shown in figure 3.9. With this knowledge of the inliers and the outliers, we perform the leave-one-person-out test again. Table 3.2 shows the mean absolute error (MAE) of age prediction for the inliers and the outliers separately. The small

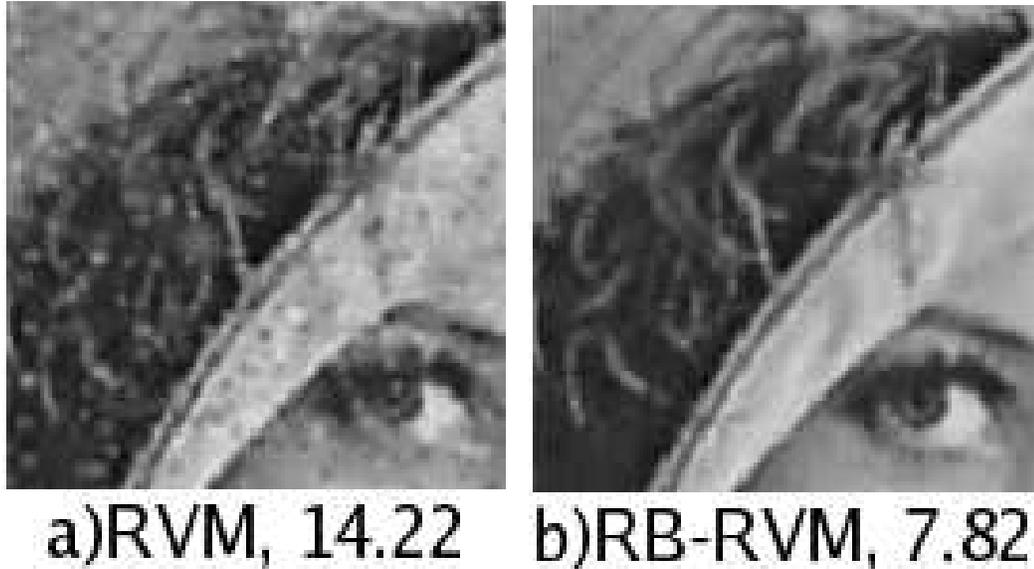


Figure 3.8: Mixture of Gaussian and salt and pepper noise removal experiment: denoised images by RVM and RB-RVM with their corresponding RMSE values. This experiment again shows that the RB-RVM based denoising algorithm gives much better result than the RVM based one. prediction error for the inliers and the large prediction error for the outliers indicate that the inlier vs. outlier categorization by RB-RVM was good. Table 3.2 also shows that the prediction error of the RB-RVM for the whole dataset is lower than that of the RVM. To put the numbers in the table in context, the state-of-the-art algorithm [40] gives a prediction error of 5.07 as compared to the prediction error of 4.61 obtained for the inliers by the RB-RVM.

r	0.1	0.2	0.3	0.4
MAE	7.10	6.52	6.54	6.62

Table 3.1: Mean absolute error (MAE) of age prediction for different values of the scale parameter r of the Gaussian kernel. The prediction errors are for the leave-one-person-out testing by RB-RVM. $r = 0.2$ gives the best result, and we use this r for all the subsequent experiments.

To further test RB-RVM, we add various amount of controlled outliers. Before



Figure 3.9: Some inliers and outliers found by RB-RVM. Most of the outliers are images of older subjects like Outlier A and B. This is because there are less number of samples of older subjects in the FG-Net database. Outlier C has an extreme pose variation from the usual frontal faces of the database; hence, it is an outlier. The facial geometry of Outlier D is very similar to that of younger subjects, such as big forehead and small chin, so it is classified as an outlier.

doing this, we remove the outliers found in the previous experiment. We use 90% of this new dataset as the training set and the remaining 10% as the test set. We introduce controlled outliers only in the training set, and perform age prediction on the test set by both RVM and RB-RVM. We vary the fraction of the outliers on the training set and measure the age prediction error on the test set. Figure 3.10 shows that RB-RVM gives much lower prediction error as compared to RVM. This experiment again suggests that RB-RVM should be preferred over RVM for the age estimation problem.

	Inlier MAE	Outlier MAE	All MAE
RB-RVM	4.61	25.87	6.52
RVM	N.A.	N.A.	6.80

Table 3.2: Mean absolute error (MAE) of age prediction for the inliers, outliers and the whole dataset using RB-RVM. Since RVM does not differentiate between inliers and outliers, we only show the prediction error for the whole dataset. The small MAE for the inliers and the large MAE for the outliers indicates that the inlier vs. outlier categorization by RB-RVM was good. Also, note that the prediction error of the RB-RVM for the whole dataset is lower than that of the RVM.

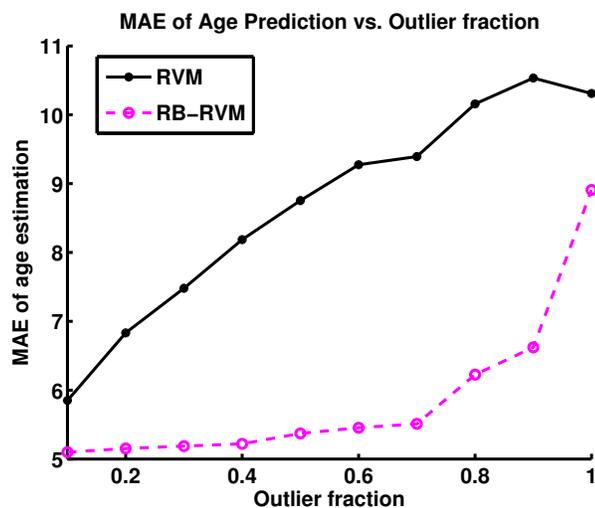


Figure 3.10: Mean absolute error (MAE) of age prediction vs. fraction of controlled outliers added to the training dataset. RB-RVM gives much lower prediction error as compared to the RVM. Also, note that the prediction error is reasonable even with outlier fraction as high as 0.7.

Chapter 4

Large-Scale Matrix Factorization with Missing Data under Additional Constraints

Many computer vision problems such as SfM [98], non-rigid SfM [12] and photometric stereo [42] can be formulated as a matrix factorization problem. In all these problems, the measured data can be arranged as a matrix of a known rank and the low-ranks factors of this matrix (obtained by matrix factorization) provide the solution of the problems. Let M be the measurement matrix of dimension $m \times n$ and rank r . The objective is to factorize this measurement matrix M into factors A and B of dimensions $m \times r$ and $n \times r$, respectively such that the error $\|M - AB^T\|$ is minimized. When all the elements of M are known, and assuming that the elements are corrupted by Gaussian noise, the solution to this problem is given by the singular value decomposition (SVD) of M . However, in most real applications, many of the elements of M will be missing and we need to solve a modified problem given by:

$$\min_{A,B} \|W \odot (M - AB^T)\|_F^2 + \lambda_1 \|A\|_F^2 + \lambda_2 \|B\|_F^2 \quad (4.1)$$

where \odot is the Hadamard element-wise product, W is a weight matrix with zeroes at indices corresponding to the missing elements of M , and $\|A\|_F^2$, $\|B\|_F^2$ are regularization terms which prevent data over-fitting. Matrix factorization with missing data is a difficult non-convex problem with no known globally convergent algorithm. The damped Newton algorithm [14], a variant of Newton's method, is one of the

most popular algorithms for solving this problem. However, this algorithm has high computational complexity and memory requirements and so cannot be used for solving large scale problems.

We formulate the matrix factorization with missing data problem as a *low-rank semidefinite program* LRSDP [16], which is essentially a rank constrained semidefinite programming problem (SDP) and was proposed to solve large SDP in an efficient way. The advantages of formulating the matrix factorization problem as a LRSDP problem are the following: 1) It inherits the efficiency of the LRSDP algorithm. The LRSDP algorithm is based on a quasi-Newton method, which has lower computational complexity and memory requirements than that of Newton’s method, and so is ideally suited for solving large scale problems. 2) Many additional constraints, such as the ortho-normality constraints for the orthographic SfM problem, can be easily incorporated into the LRSDP-based factorization formulation; this is possible because of the flexible framework of the LRSDP (see section 4.1).

Related works: Algorithms for matrix factorization in the presence of missing data can be broadly divided into two main categories: initialization algorithms and iterative algorithms. Initialization algorithms [98, 46, 39, 64, 94] generally minimize an algebraic or approximate cost of (4.1) and are used for providing a good starting point for the iterative algorithms. Iterative algorithms are those algorithms that directly minimize the cost function (4.1). Alternation algorithms [84, 105, 45, 2, 11, 50], damped Newton algorithm [14] and our approach fall under this category. Alternation algorithms are based on the fact that if one of the factors A or B is known, then there are closed form or numerical solutions for the other

factor. Though the alternation-based algorithms minimize the cost in each iteration, they suffer from flatlining, requiring an excessive number of iterations before convergence [14]. To solve this problem, damped Newton and hybrid algorithms between damped Newton and alternation were proposed in [14]. Although these algorithms give very good results, they cannot be used for solving large-scale problems because of their high computational complexity and memory requirements. Other algorithms, based on Newton’s method, have been proposed in [17, 72], which also cannot be used for solving large-scale problems.

The matrix factorization with missing data problem is closely related to the matrix completion problem [21]. The goal of matrix completion is to find a low-rank matrix which agrees with the observed entries of the matrix M . Recently, many efficient algorithms have been proposed for solving this problem [19, 60, 65, 55, 52, 66]. Some of them [55, 52, 66] are formulated as matrix factorization problems. However, these algorithms can not handle additional constraints. Matrix factorization also arises while solving the collaborative filtering problem. Collaborative filtering is the task of predicting the interests of a user by collecting the taste information from many users, for example in a movie recommendation system. In [88], collaborative filtering is formulated as a matrix completion problem and solved using a semidefinite program. Later a fast version, using conjugate gradient, was proposed in [80], but this also cannot handle additional constraints.

The organization of the rest of the chapter is as follows: in section 4.1, we set up the necessary background for LRSDP. In section 4.2, we formulate the factorization problem as a LRSDP problem and in section 4.3, discuss its relation with the matrix

completion problem. In section 4.4, we experimentally evaluate the LRSDP-based factorization algorithm on synthetic data and for some computer vision problems drawn from SfM and photometric stereo.

4.1 Background: Low-rank semidefinite programming (LRSDP)

LRSDP was proposed in [16] to efficiently solve a large scale SDP [101]. In the following paragraphs, we briefly define the SDP and LRSDP problems, and discuss the efficient algorithm used for solving the LRSDP problem.

SDP is a subfield of convex optimization concerned with the optimization of a linear objective function over the intersection of the cone of positive semidefinite matrices with an affine space. The standard-form SDP is given by:

$$\min C \bullet X \text{ subject to } A_i \bullet X = b_i, \quad i = 1, \dots, k \quad X \succeq 0 \quad (4.2)$$

where C and A_i are $n \times n$ real symmetric matrices, b is k -dimensional vector, and X is an $n \times n$ matrix variable, which is required to be symmetric and positive semidefinite, as indicated by the constraint $X \succeq 0$. The operator \bullet denotes the inner product in the space of $n \times n$ symmetric matrices defined as $A \bullet B = \text{trace}(A^T B) = \sum_{i=1}^n \sum_{j=1}^n A_{ij} B_{ij}$. The most common algorithms for solving (4.2) are the interior point methods [101]. However, these are second-order methods, which need to store and factorize a large (and often dense) matrix and hence are not suitable for solving large scale problems.

In LRSDP a change of variables is introduced as $X = RR^T$, where R is a real, $n \times r$ matrix with $r \leq n$. This has the advantage that it removes the non-linear

constraint $X \succeq 0$, which is the most challenging aspect of solving (4.2). However, this comes with the cost that the problem may no longer be a convex problem. The LRSDP formulation is given by:

$$(N_r) \quad \min C \bullet RR^T \text{ subject to } A_i \bullet RR^T = b_i, \quad i = 1, \dots, k \quad (4.3)$$

Note that the LRSDP formulation depends on r ; when $r = n$, (4.3) is equivalent to (4.2). But the intention is to choose r as small as possible so as to reduce the number of variables, while the problem remains equivalent to the original problem (4.2).

A non-linear optimization technique called the augmented Lagrangian method is used for solving (4.3). The majority of the iterations in this algorithm involve the minimization of an augmented Lagrangian function with respect to the variable R which is done by a limited memory BroydenFletcherGoldfarbShanno (BFGS) method. BFGS, a quasi-Newton method, is much more efficient than Newton's method both in terms of computations and memory requirement. The LRSDP algorithm further optimizes the computations and storage requirements for sparse C and A_i matrices, which is true for problems of our interest. For further details on the algorithm, see [16, 15].

4.2 Matrix factorization using LRSDP (MF-LRSDP)

In this section, we formulate the matrix factorization with missing data as an LRSDP problem. We do this in the following stages: in section 4.2.1, we look at the noiseless case, that is, where the measurement matrix M is not corrupted with

noise, followed by the noisy measurement case in section 4.2.2, and finally in section 4.2.3, we look at how additional constraints can be incorporated in the LRSDP formulation.

4.2.1 Noiseless Case

When the observed elements of the $m \times n$ dimensional measurement matrix M are not corrupted with noise, a meaningful cost to minimize would be:

$$\min_{A,B} \|A\|_F^2 + \|B\|_F^2 \text{ subject to } (AB^T)_{i,j} = M_{i,j} \text{ for } (i,j) \in \Omega, \quad (4.4)$$

where Ω is the index set of the observed entries of M , and A, B are the desired factor matrices of dimensions $m \times r$ and $n \times r$ respectively. We assume r is known, for example, in affine SfM $r = 4$ and in photometric stereo $r = 3$. To formulate this as a LRSDP problem, we introduce a $(m+n) \times r$ dimensional matrix $R = \begin{pmatrix} A \\ B \end{pmatrix}$.

Then

$$RR^T = \begin{pmatrix} AA^T & AB^T \\ BA^T & BB^T \end{pmatrix} \quad (4.5)$$

We observe that the cost function $\|A\|_F^2 + \|B\|_F^2$ can be expressed as $\text{trace}(RR^T)$ and the constraints as $(RR^T)_{i,j+m} = M_{i,j}$. Thus, (4.4) is equivalent to:

$$\min_R \text{trace}(RR^T) \text{ subject to } (RR^T)_{i,j+m} = M_{i,j} \text{ for } (i,j) \in \Omega \quad (4.6)$$

This is already in the LRSDP form, since we can express the above equation as

$$\min_R C \bullet RR^T \text{ subject to } A_l \bullet RR^T = b_l, \quad l = 1, \dots, |\Omega| \quad (4.7)$$

where C is an $(m+n) \times (m+n)$ identity matrix, and to simplify the notations we have introduced the index l with $\Omega(l) = (i, j) \quad l = 1, \dots, |\Omega|$. A_l are sparse matrices with the non-zero entries at indices $(i, j+m)$ and $(j+m, i)$ equal to $1/2$ and $b_l = M_{i,j}$. This completes the formulation of the matrix factorization problem as an LRSDP problem for the noiseless case. Next we look at the noisy case.

4.2.2 Noisy case

When the observed entries of M are corrupted with noise, an appropriate cost function to minimize would be:

$$\min_{A,B} \|W \odot (M - AB^T)\|_F^2 + \lambda \|A\|_F^2 + \lambda \|B\|_F^2 \quad (4.8)$$

where \odot is the Hadamard element-wise product and W is a weight matrix with zeros corresponding to the missing entries and 1 to the observed entries in M . To formulate this as an LRSDP problem, we introduce noise variables $e_l, l = 1, 2, \dots, |\Omega|$ which are defined as $e_l = (M - (AB^T))_l$. Now, (4.8) can be expressed as

$$\min_{A,B,e} \|e\|_2^2 + \lambda \|A\|_F^2 + \lambda \|B\|_F^2 \text{ subject to } (M - AB^T)_l = e_l \text{ for } l = 1, 2, \dots, |\Omega| \quad (4.9)$$

Next, we aim to formulate this as a LRSDP problem. For this, we construct an augmented noise vector $E = [e^T \quad 1]^T$ and define R to be

$$R = \begin{pmatrix} \begin{pmatrix} A \\ B \end{pmatrix} & 0 \\ 0 & E \end{pmatrix} \quad (4.10)$$

R is a ‘block-diagonal’ matrix, where the blocks are of sizes $(m+n) \times r$ and $(|\Omega|+1) \times 1$ respectively. With this definition, RR^T is a block-diagonal matrix given by

$$RR^T = \begin{pmatrix} \begin{pmatrix} AA^T & AB^T \\ BA^T & BB^T \end{pmatrix} & 0 \\ 0 & EE^T \end{pmatrix} \quad (4.11)$$

We can now express (4.8) in the following LRSDP form:

$$\min_R C \bullet RR^T \text{ subject to } A_l \bullet RR^T = b_l, \quad l = 1, \dots, |\Omega| + 1 \quad (4.12)$$

with

$$C = \begin{pmatrix} \lambda I_{(m+n) \times (m+n)} & 0 \\ 0 & I_{(|\Omega|+1) \times (|\Omega|+1)} \end{pmatrix} \quad (4.13)$$

Note that the number of constraints $|\Omega| + 1$ in (4.12) is one more than the number of observations $|\Omega|$. This is because the last constraint is used to set $E_{|\Omega|+1} = 1$, which is done by choosing $A_{|\Omega|+1}$ to be a sparse matrix with the non-zero entry at index $(|\Omega| + l + m + n, |\Omega| + 1 + m + n)$ equal to 1 and $b_{|\Omega|+1} = 1$. For the remaining values of l , the A_l are sparse matrices with the non-zero entries at indices $(i, j + m)$, $(j + m, i)$, $(|\Omega| + 1 + m + n, l + m + n)$ and $(l + m + n, |\Omega| + 1 + m + n)$ equal to $1/2$ and $b_l = M_l$. Note that (4.12) is a block-LRSDP problem (R has a block-diagonal structure), which is a simple extension of the original LRSDP problem [15]. This completes the LRSDP formulation for the noisy case. Next, we look at incorporating additional constraints in this framework.

4.2.3 Enforcing Additional Constraints

Many additional constraints can be easily incorporated in the LRSDP formulation. We illustrate this using the specific example of orthographic SfM [98]. SfM is the problem of reconstructing the scene structure (3-D point positions and camera parameters) from 2-D projections of the points in the cameras. Suppose that $m/2$ cameras are looking at n 3-D points, then under the *affine camera model*, the 2-D imaged points can be arranged as an $m \times n$ measurement matrix M with columns corresponding to the n 3-D points and rows corresponding to the $m/2$ cameras (2 consecutive rows per camera). M is a rank 4 matrix and can be factorized as $M = AB^T$, where A is a $m \times 4$ camera matrix and B is a $n \times 4$ structure matrix with the last column of B an all-one vector, i.e. $B = [X \ \mathbf{1}]$. Under the *orthographic camera model*, A has more structure (constraints). To state these constraints precisely, we express the A matrix as $A = [P \ t]$, where P is a $m \times 3$ sub-matrix consisting of the first three columns and t is the last column vector. A satisfies the following constraints: rows of P that corresponds to the same camera are ortho-normal. This implies that the diagonal elements of the matrix PP^T are equal to 1 (normality constraint) and appropriate off-diagonal elements are 0 (orthogonality constraint). Now, $AB^T = PX^T + t\mathbf{1}^T$ and the observation error can be expressed as $e_{i,j} = (M - PX)_{i,j} - t_i$ for $(i,j) \in \Omega$. A meaningful optimization problem to solve here would be to minimize the observation error subject to the ortho-normality

constraints:

$$\begin{aligned}
\min_{e,P,X,t} \|e\|_2^2 \quad & \text{subject to } e_{i,j} = (M - PX)_{i,j} - t_i, \quad (i,j) \in \Omega \\
& (PP^T)_{k,k} = 1, \quad k = 1, 2, \dots, m \\
& (PP^T)_{k,l} = 0, \text{ if } k \text{ and } l \text{ are rows from same camera} \quad (4.14)
\end{aligned}$$

To formulate this as an LRSDP problem, we introduce the augmented translation variable $T = [t^T \quad 1]^T$, and propose the following block-diagonal matrix R :

$$R = \begin{pmatrix} \begin{pmatrix} P \\ X \end{pmatrix} & 0 & 0 \\ 0 & T & 0 \\ 0 & 0 & E \end{pmatrix} \quad (4.15)$$

With this definition of R , we can express (4.14) as a LRSDP problem, following steps similar to the previous sections. This completes our illustration on the incorporation of the ortho-normality constraints for the orthographic SfM case. This example should convince the reader that many other application-specific constraints can be directly incorporated into the LRSDP formulation; this is because of the underlying SDP structure of the LRSDP.

4.3 Matrix Completion, Uniqueness and Convergence of MF-LRSDP

In this section, we state the main result of the matrix completion theory and discuss its implications for the matrix factorization problem.

4.3.1 Matrix Completion Theory

Matrix completion theory considers the problem of recovering a low-rank matrix from a few samples of its entries:

$$\min_X \text{rank}(X) \text{ subject to } X_{i,j} = M_{i,j} \text{ for } (i,j) \in \Omega \quad (4.16)$$

More specifically, it considers the following questions: 1) when does a partially observed matrix have a unique low-rank solution? 2) How can this matrix be recovered?

The answers to these questions were provided in theorem 1.3 of [21] which states that if 1) the matrix M , that we want to recover, has row and column spaces incoherent with the standard basis and 2) we are given enough entries ($\geq O(rd^{6/5} \log d)$, where $d = \max(m, n)$), then there exists a unique low-rank solution to (4.16). Further, the solution can be obtained by solving a convex relaxation of (4.16) given by:

$$\min_X \|X\|_* \text{ subject to } X_{i,j} = M_{i,j} \text{ for } (i,j) \in \Omega \quad (4.17)$$

where $\|X\|_*$ is the nuclear norm of X , given by the sum of its singular values.

4.3.2 Relation with Matrix Factorization and its Implications

In matrix completion the objective is to find a minimum rank matrix which agrees with the partial observations (4.16), whereas in matrix factorization we assume the rank r to be known, as in the problems of SFM and photometric stereo, and we use the rank as a constraint. For example, in our LRSDP formulation, we have imposed this rank constraint by fixing the number of columns of the factors A and B to r . However, though the matrix completion and factorization problems

are defined differently, they are closely related as revealed by their very similar Lagrangian formulations. This fact has been used in solving the matrix completion problem via matrix factorization with an appropriate rank [55, 52, 66]. We should also note that matrix completion theory helps us answer the question raised in [14]: when is missing data matrix factorization unique (up to a gauge)? And from the discussion in the previous section, it should be clear that the conditions of the matrix completion theory are sufficient for guaranteeing us the required uniqueness. Further, in our experimental evaluations (see next section), we have found that the LRSDP formulation, though a non-convex problem in general, typically converges to the global minimum solution under these conditions.

4.4 Experimental Evaluation

We evaluate the performance of the proposed LRSDP-based factorization algorithm (MF-LRSDP) on both synthetic and real data and compare it against other algorithms such as alternation [14], damped Newton [14] and OptSpace [52], which is one of state-of-the-art algorithms for matrix completion.

4.4.1 Evaluation with Synthetic Data

The important parameters in the matrix factorization with missing data problem are: the size of the matrix M characterized by m and n , rank r , fraction of missing data and the variance σ^2 of the observation noise. We evaluate the factorization algorithms by varying these parameters. We consider two cases: data

without noise and data with noise. For synthetic data without noise, we generate $n \times n$ matrices M of rank r by $M = AB^T$, where A and B are $n \times r$ random matrices with each entry being sampled independently from a standard Gaussian distribution $\mathcal{N}(0, 1)$. Each entry is then revealed randomly according to the missing data fraction. For synthetic data with noise, we add independent Gaussian noise $\mathcal{N}(0, \sigma^2)$ to the observed entries generated as above.

Exact Factorization: a first comparison. We study the reconstruction rate of different algorithms by varying the fraction of revealed entries per column ($|\Omega|/n$) for noiseless 500×500 matrices of rank 5. We declare a matrix to be reconstructed if $\|M - \hat{M}\|_F / \|M\|_F \leq 10^{-4}$, where $\hat{M} = \hat{A}\hat{B}$ is the reconstructed matrix and $\|\cdot\|_F$ denotes the Frobenius norm. Reconstruction rate is defined as the fraction of trials for which the matrix was successfully reconstructed. In all the synthetic data experiments, we performed 10 trials. Figure 4.1(a) shows the reconstruction rate by MF-LRSDP, alternation and OptSpace. MF-LRSDP gives the best reconstruction results as it needs fewer observations for matrix reconstruction than the other algorithms. It is followed by OptSpace and alternation, respectively. MF-LRSDP also takes the least time, followed by OptSpace and alternation. For similar comparison to other matrix completion algorithms such as ADMiRA [55], SVT [19] and FPCA [60], the interested reader can look at [52], where OptSpace was shown to be consistently better than these algorithms. For the remaining experiments on synthetic data, we compare MF-LRSDP against OptSpace. Note that we have not included the damped Newton algorithm in this comparison because it is very slow for matrices of this size.

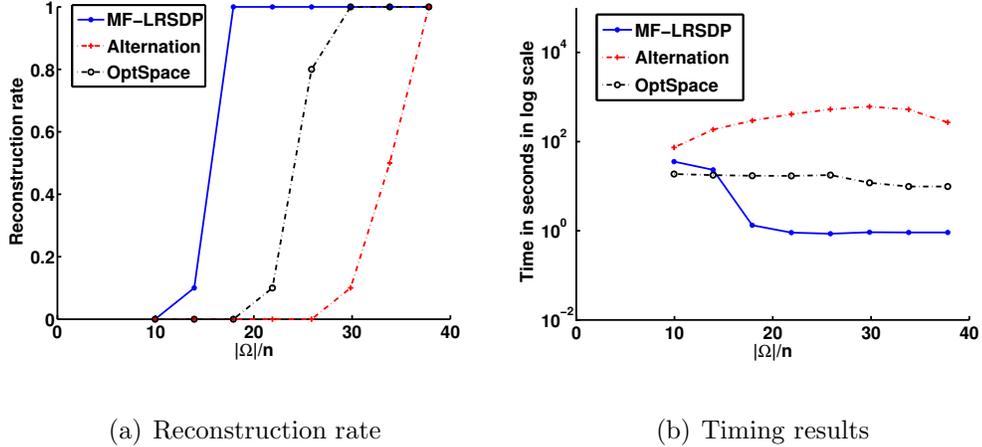


Figure 4.1: (a) Reconstruction rate vs. fraction of revealed entries per column $|\Omega|/n$ for 500×500 matrices of rank 5 by MF-LRSDP, alternation and OptSpace. The proposed algorithm MF-LRSDP gives the best reconstruction results since it can reconstruct matrices with fewer observed entries. (b) Time taken for reconstruction by different algorithms. MF-LRSDP takes the least time.

Exact Factorization: vary size. We study the reconstruction rate vs. fraction of revealed entries per column $|\Omega|/n$ for different sizes n of rank 5 square matrices by MF-LRSDP and OptSpace. Figure 4.2(a) shows that MF-LRSDP reconstructs matrices from fewer observed entries than OptSpace.

Exact Factorization: vary rank. We study the reconstruction rate vs. $|\Omega|/n$ as we vary the rank r of 500×500 matrices. Figure 4.2(b) again shows that MF-LRSDP gives better results than OptSpace.

Noisy Factorization: vary noise standard deviation. For noisy data, we use the root mean square error $\text{RMSE} = 1/\sqrt{mn} \|M - \hat{M}\|_F$ as a performance measure. We vary the standard deviation σ of the additive noise for rank 5, 200×200 matrices and study the performance by MF-LRSDP, OptSpace, alternation and damped Newton. Figure 4.2(c) shows that all the algorithms perform equally well.

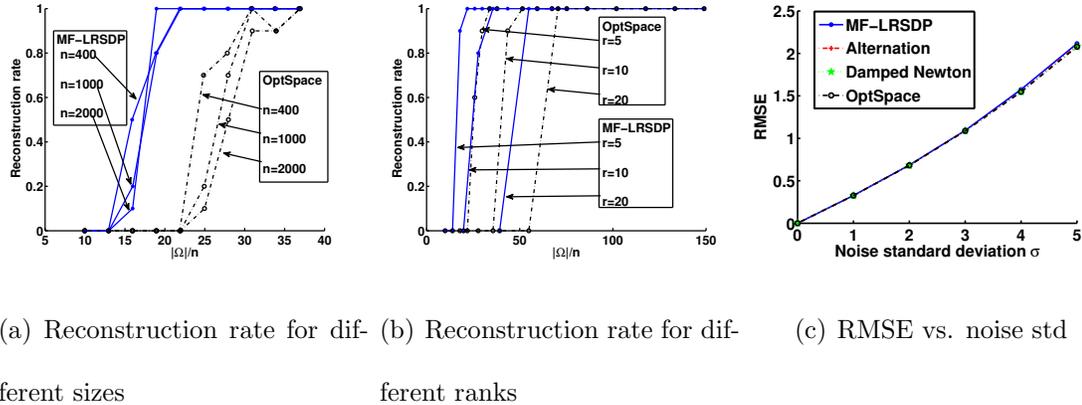


Figure 4.2: (a) Reconstruction rate vs. fraction of revealed entries per column $|\Omega|/n$ for rank 5 square matrices of different sizes n by MF-LRSDP and OptSpace. MF-LRSDP reconstructs matrices from fewer observed entries than OptSpace. (b) Reconstruction rate vs. $|\Omega|/n$ for 500×500 matrices of different ranks by MF-LRSDP and OptSpace. Again MF-LRSDP needs fewer observations than OptSpace. (c) RMSE vs. noise standard deviation for rank 5, 200×200 matrices by MF-LRSDP, OptSpace, alternation and damped Newton. All algorithms perform equally well.

For timing comparisons, please refer to the supplementary material.

4.4.2 Evaluation with Real Data

We consider three problems: 1) affine SfM 2) non-rigid SfM and 3) photometric stereo.

Affine SfM. As discussed in section 4.2.3, for affine SfM, the $m \times n$ measurement matrix M is a rank 4 matrix with the last column of matrix B an all-one vector. M is generally an incomplete matrix because not all the points are visible in all the cameras. We evaluate the performance of MF-LRSDP on the ‘Dinosaur’ sequence used in [14, 17], for which M is a 72×319 matrix with 72% missing entries. We perform 25 trials and at each trial we provide the same random initializations

to MF-LRSDP, alternation and damped Newton (OptSpace has its only initialization technique). We use the root mean square error over the observed entries, $\|W \odot (M - \hat{M})\|_F / \sqrt{|\Omega|}$, as our performance measure. Figure 4.3 shows the cumulative histogram over the RMS pixel error. MF-LRSDP gives the best performance followed by damped Newton, alternation and OptSpace. We further tested the algorithms on a 'longer Dinosaur', the result of which is provided in the supplementary material.

Non-rigid SfM. In non-rigid SfM, non-rigid objects are expressed as a linear combination of b basis shapes. In this case, the $m \times n$ measurement matrix M can be expressed as $M = AB^T$, where A is an $m \times 3b$ matrix and B is an $n \times 3b$ matrix [12]. This makes M a rank $3b$ matrix. We test the performance of the algorithms on the 'Giraffe' sequence [14, 17] for which M is a 240×167 matrix with 30% missing entries. We choose the rank as 6. Figure 4.3 shows the cumulative histogram of 25 trials from which we conclude that MF-LRSDP, alternation and damped Newton give good results.

Photometric Stereo. Photometric stereo is the problem of estimating the surface normals of an object by imaging that object under different lighting conditions. Suppose we have n images of the object under different lighting conditions with each image consisting of m pixels (m surface normals) and we arrange them as an $m \times n$ measurement matrix M . Then under Lambertian assumptions, we can express M as $M = AB^T$, where A is an $m \times 3$ matrix representing the surface normals and reflectance and B is an $n \times 3$ matrix representing the light-source directions and intensities [42]. Thus, M is a rank 3 matrix. Some of the image pixels are likely to

be affected by shadows and specularities and those pixels should not be included in the M matrix as they do not obey the Lambertian assumption. This makes M , an incomplete matrix. We test the algorithms on the ‘Face’ sequence [14, 17] for which M is a 2944×20 matrix with 42% missing entries. The cumulative histogram in figure 4.3 shows that MF-LRSDP and damped Newton gives the best results followed by alternation and OptSpace.

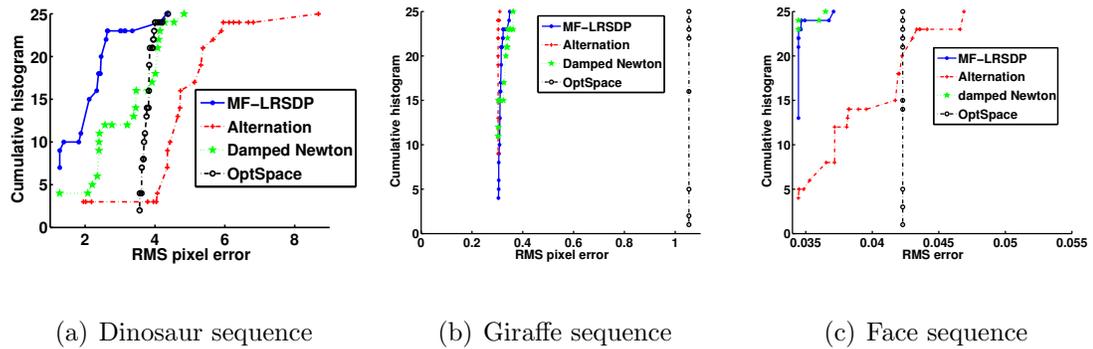
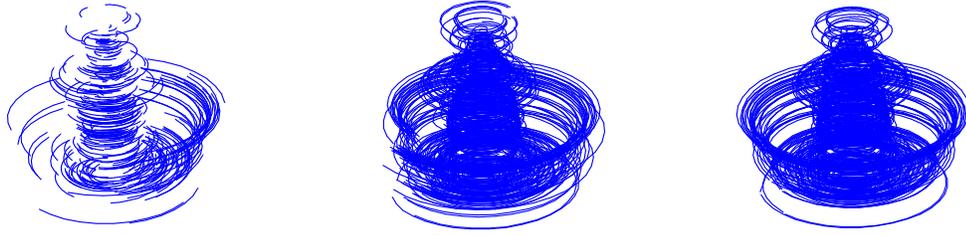


Figure 4.3: Cumulative histogram (of 25 trials) for the Dinosaur, Giraffe and the Face sequence. For all of them, MF-LRSDP consistently gives good results.

Additional constraints: Orthographic SfM. Orthographic SfM is a special case of affine SfM, where the camera matrix A satisfies the additional constraint of ortho-normality, see section 4.2.3. We show here that incorporating these constraints leads to a better solution. Figure 4.4 shows the input point tracks, reconstructed point tracks without the constraints and reconstructed point tracks with the constraints for the Dinosaur turntable sequence. Without the constraints many tracks fail to be circular, whereas with the constraints all of them are circular (the dinosaur sequence is a turntable sequence and the tracks are supposed to be circular). Thus, incorporating all the constraints of a problem leads to better solution

and MR-LRSDP provides a very flexible framework for doing so.



(a) Input point tracks (b) Reconstructed tracks with- (c) Reconstructed tracks with
out constraints constraints

Figure 4.4: (a) Input (incomplete) point tracks of the Dinosaur turntable sequence, (b) reconstructed tracks without orthonormality constraints and (c) reconstructed tracks with orthonormality constraints. Without the constraints many tracks fail to be circular, whereas with the constraints all of them are circular (the dinosaur sequence is a turntable sequence and the tracks are supposed to be circular).

Chapter 5

Direct Recognition of Faces across Blur and Illumination Variations

Face recognition is one of the most important problems of computer vision and significant progress has been made towards a solution for this problem [110]. Under controlled environments, with well regulated illumination, expression and pose conditions, the current state of the art face recognition algorithms perform quite well. However, for images acquired in uncontrolled environments, it is still a very challenging problem. We are interested in recognizing faces acquired from distant cameras. The main factors that make this a challenging problem are image degradations due to blur and noise, and variations in appearance due to illumination and pose [69]. In this dissertation, we address the problem of recognizing faces across blur and illumination variations.

The current state of the art approach for recognizing blurred faces first deblurs the face image and then uses this deblurred image for recognition [70]. However, this involves solving the challenging problem of blind image deconvolution, which is not a necessary step for solving the recognition problem. We take a direct approach for recognizing blurred faces. Using the convolution model for blur, that is a blurred image can be modeled by convolution of a sharp image with a blur filter kernel, we show that the set of all images obtained by blurring a given image forms a convex set. Hence, we can associate such a set with each of the gallery images. Given a

probe image, we find its distance from each of the convex sets (associated with the gallery images) and assign it the identity of the gallery image with the minimum distance. We show that this algorithm is also statistically optimal; it is the maximum likelihood estimator of the blur kernel and the identity. Further, based on the set theoretic notions of blur, we present an algorithm to characterize the amount of blur our algorithm can handle for a given data set.

In uncontrolled environments, for good recognition performance, it is imperative to model the appearance of a face under different illumination conditions. Towards this end, we use the low-dimensional linear subspace model theory proposed in [8, 77]. For each face in the gallery, we create nine basis images that spans its corresponding low-dimensional subspace. Using these basis images and the convolution model for blur, we associate a (non-convex) set, which represents all variations due to blur and illumination, with each gallery image. Given a probe image, we find the closest such set and assign its identity to the probe image. The main optimization step involves solving a quadratically constrained quadratic programming problem (QCQP), which we solve by alternately optimizing over the blur kernels and the illumination coefficients. The proposed algorithm is also statistically optimal; it is the maximum likelihood estimator of the blur kernel, illumination coefficients and identity.

Related works: Face recognition from blurred images can be classified into three major approaches. In one approach, blur invariant features are extracted from the blurred image and then used for recognition. [6] follows this approach, where local phase quantization (LPQ) [71] method is used to extract blur invariant features.

Though this approach works very well for small blurs, it is not so effective for large blurs [70]. In another approach, the blurred image is first deblurred and then used for recognition. This is the approach taken in [43] and [70]. As discussed earlier, the drawback of this approach is that it first solves the challenging problem of blind image deconvolution, which is not a necessary step for solving the face recognition problem. Also, in [70], statistical models are learned for each blur kernel type and amount; this step might become infeasible when we try to capture the complete space of blur kernels. Finally, the last approach is the direct recognition approach. This is the approach taken in [89] and by us. In [89], artificially blurred versions of the gallery images are created and the blurred probe image is matched to them. Again, it is not possible to capture the whole space of blur kernels using this method. We avoid this problem by optimizing over the space of blur kernels. Our approach also has the additional advantage of incorporating the low-dimensional linear subspace model for capturing illumination variations.

The organization of the rest of this chapter is as follows: In section 5.1 we propose our approach for recognizing blurred faces, in section 5.2 we incorporate the illumination model in our approach and in section 5.3 we perform experiments to evaluate the efficacy of our approach on many synthetic and real datasets.

5.1 Face Recognition Across Blur (FRB)

We first present the convolution model for blur. Next, we describe the set of all images obtained by blurring a given image. We then describe our recognition

algorithm based on distances from such set. We also present an algorithm that characterizes the amount of blur our algorithm can handle in a given dataset.

Convolution model for blur. A pixel in a blurred image is the weighted average of that pixel’s neighborhood in the original image. Thus, blur is modeled by a convolution operation between the original image and a blur kernel (filter) which represents the weights. Let I be the original image and H be the blur kernel, then the blurred image I_b is given by

$$I_b = I * H \tag{5.1}$$

where $*$ represents the convolution operation. For a blur kernel size of $(2k + 1) \times (2k + 1)$, which is generally much smaller than the image size, the blurred image I_b is given by

$$I_b(n_1, n_2) = \sum_{i=-k}^k \sum_{j=-k}^k H(i, j) I(n_1 - i, n_2 - j) \tag{5.2}$$

Blur kernels satisfy the following properties: their coefficients are non-negative $H \geq 0$ and they sum up to 1 ($\sum_{i=-k}^k \sum_{j=-k}^k H(i, j) = 1$). These properties basically represents the fact that the weights in the weighted averaging operation are non-negative and sum up to unity. The blur kernel H may possess additional structures depending on the type of blur, such as circular-symmetry for out-of-focus blur, and these structures could be exploited during recognition.

The set of all blurred images. We want to characterize the set of all images obtained by blurring a given image I . To do that we re-write (5.1) in a matrix-vector form. Let $h \in \mathbb{R}^{(2k+1)^2}$ be the vector obtained by concatenating the columns of H , i.e. $h = H(:)$, and similarly $i_b \in \mathbb{R}^N$ be the representation of I_b in the vector form.

Then (5.2) suggests that we can write (5.1), along with the blur kernel constraints, as

$$i_b = \{Ah \mid h \geq 0, \|h_i\|_1 = 1\} \quad (5.3)$$

where A is a $N \times (2k + 1)$ matrix, obtained from I , with each row of A representing the neighborhood pixel intensities about the pixel indexed by the row. From the above equation, it is clear that the set of all blurred images obtained from I is given by

$$B = \{Ah \mid h \geq 0, \|h_i\|_1 = 1\} \quad (5.4)$$

We have the following result about the set B .

Proposition 5.1.1. *The set of all images B obtained by blurring an image I forms a convex set. Moreover, this convex set is given by the convex hull of the columns of matrix A , which represents the neighborhood structure of the blur operation on I .*

Proof. Let i_1 and i_2 be elements from the set B . Then there exists h_1 and h_2 , with both satisfying the conditions $h \geq 0$ and $\|h\|_1 = 1$, such that $i_1 = Ah_1$ and $i_2 = Ah_2$. To show the set B is convex we need to show that for any λ satisfying $0 \leq \lambda \leq 1$, $i_3 = \lambda i_1 + (1 - \lambda)i_2$ is an element of B . Now

$$\begin{aligned} i_3 &= \lambda i_1 + (1 - \lambda)i_2 \\ &= A(\lambda h_1 + (1 - \lambda)h_2) \\ &= Ah_3. \end{aligned} \quad (5.5)$$

Note that h_3 satisfies both the non-negativity and sum conditions and hence i_3 is

an element of B . Thus, B is a convex set. B is defined as

$$B = \left\{ \sum_i A_i h_i \mid h_i \geq 0, \sum_i h_i = 1 \right\}, \quad (5.6)$$

which by definition is the convex hull of the columns A_i of A . \square

A geometric face recognition algorithm. Let $I_j, j = 1, 2, \dots, M$ be the set of M gallery images. From analysis given above, every gallery image I_j has an associated convex set of blurred images B_j . Given the probe image I_b , we find the minimum distance between the image and a point in the set B_j by solving:

$$r_j = \min_h \|I_b - A_j * h\|^2 \text{ subject to } h \geq 0, \|h\|_1 = 1 \quad (5.7)$$

This is a convex quadratic program which can be solved efficiently. We compute r_j for each $j = 1, 2, \dots, M$ and assign I_b the identity of the gallery image with minimum r_j . If there are multiple images per class (person), we can use the k-nearest neighbor rule, i.e. we arrange the r_j in an ascending order and find the class with repeats the most in the first k instances. In this algorithm, we can also incorporate additional information about the type of blur. The most commonly occurring blurs are out-of-focus, motion and atmospheric blur [10]. The out-of-focus blur and the atmospheric blur are circular-symmetric, i.e. the coefficients of H at the same radius are equal, whereas the motion blur is symmetric about the origin, i.e. $H(i, j) = H(-i, -j)$. So if we know the type of blur, we can impose its corresponding symmetry constraint while solving for (5.7). Imposing these constraints reduces the number of parameters in the optimization problem, giving better recognition accuracy and faster solutions.

Statistical interpretation of the algorithm. Though our algorithm is geometrically motivated it also has a statistical interpretation. For that we need to

introduce the concept of noise in the blur model (5.1):

$$I_b = I * H + n, \quad (5.8)$$

where n is the image noise. If we assume a Gaussian distribution $\mathcal{N}(0, \sigma^2 I)$ for noise, then the likelihood of I_b given I and H is $\mathcal{N}(I_b - I * H, \sigma^2 I)$. Given the gallery images $I_j, j = 1, 2, \dots, M$ and the probe image I_b , the maximum likelihood estimate for the gallery image and the blur kernel H is obtained by solving

$$[\hat{j}, \hat{H}] = \arg \max_{j, H} \mathcal{N}(I_b - I_j * H) \text{ subject to } H \geq 0, \sum H = 1 \quad (5.9)$$

Using the fact that maximizing the likelihood is same as minimizing the negative log likelihood and the matrix-vector notations, we get

$$[\hat{j}, \hat{h}] = \arg \min_{j, h} \|I_b - A_j h\|^2 \text{ subject to } h \geq 0, \|h\|_1 = 1, \quad (5.10)$$

The joint minimization over the index j and blur kernel H can also be solved by first minimizing over H , i.e. by solving (5.7), and then minimizing r_j over the index j . This is exactly our proposed algorithm.

Effect of the amount of blur on recognition. Given a dataset of gallery images, we can make a prediction for the amount of blur that our algorithm can handle. Here, by amount of blur we mean the length of the blur vector h . For any given length of blur kernel, we can find the separation between the convex sets B_j associated with each gallery image I_j . We define the separation between two sets B_i and B_j as the minimum distance $s(i, j)$ between two points from each set:

$$s(i, j) = \min_{h_i, h_j} \|A_i h_i - A_j h_j\|^2 \text{ subject to } h_i \geq 0, h_j \geq 0, \|h_i\|_1 = 1, \|h_j\|_1 = 1 \quad (5.11)$$

Note that this is again a convex quadratic program and hence can be efficiently solved. With increasing amount of blur, all the pairwise distances will decrease. At a certain blur amount, the separation between a pair of sets might become zero which essentially means that the corresponding sets have non-zero intersection. Any probe image that falls in such an intersection can be classified as belonging to either of the classes, leading to poor recognition results. Thus, the amount of blur which reduces the minimal separation between sets to zero is a good measure of the amount of blur a particular dataset can handle.

5.2 Incorporating the Illumination Model

The facial images of a person under different illuminations can look very different, and hence for any recognition algorithm to work in practice, it must account for these variations. First, we discuss about the low-dimensional subspace model for handling appearance variations due to illumination. Next, we use this model along with the convolution model to define the set of images of face under all possible lighting conditions and blur. We then propose a recognition algorithm based on minimizing the distance of the probe image from such sets.

Low-dimensional linear model for illumination variations. It has been shown in [8, 77] that when the object is convex and Lambertian, the set of images form a low-dimensional linear subspace of approximate dimension 9. Though a human face is not exactly convex or Lambertian, it is very close to being one and hence the nine-dimensional subspace model captures its variations due to illumina-

tion quite well [34]. The nine-dimensional linear subspace corresponding to a face image I can be characterized by 9 basis images. In terms of these nine basis images $I_m, m = 1, 2, \dots, 9$, an image I of a person under any illumination condition can be written as

$$I = \sum_{m=1}^9 \alpha_m I_m \quad (5.12)$$

where $\alpha_m, m = 1, 2, \dots, 9$ are the corresponding linear coefficients. To obtain these basis images, we use the “universal configuration” of lighting positions proposed in [27]. These are a configuration of 9 lighting positions $s_m, m = 1, 2, \dots, 9$ such that images taken under these lighting positions can serve as basis images for the subspace. These basis images are generated using the Lambertian reflectance model:

$$I_m(i, j) = \rho(i, j) \max(\langle s_m, n(i, j) \rangle, 0) \quad (5.13)$$

where $\rho(i, j)$ and $n(i, j)$ are the albedo and surface-normal corresponding to pixel (i, j) . We use the average 3-D face normals for n and we approximate the albedo ρ by a well-illuminated gallery image under diffuse lighting. One can, potentially, also use the algorithm in [9] to estimate the albedo and the surface-normals from the single gallery image.

The set of all images under different lighting and blur. For a given face characterized by the nine basis images $I_m, m = 1, 2, \dots, 9$, the set of images under all possible lighting conditions and blur is given by

$$BI = \left\{ I = \sum_{m=1}^9 \alpha_m A_m h \mid h \geq 0, \|h\|_1 = 1 \right\}, \quad (5.14)$$

where we have used the matrix-vector notations for I, I_m and H introduced in

section 5.1. This set is not a convex set though if we fix either the filter kernel h or the illumination condition α_m the set becomes convex.

Face recognition across blur and illumination (FRBI). Corresponding to each gallery image $I_j, j = 1, 2, \dots, M$, we obtain the nine basis images $I_{j,m}, m = 1, 2, \dots, 9$ corresponding to each of the gallery images. Given a probe image I_b , we find the minimum distance between the image and a point in the set BI_j by solving:

$$r_j = \min_{h, \alpha_m} \|I_b - (\sum_{m=1}^9 \alpha_m A_{j,m} h)\|^2 \text{ subject to } h \geq 0, \|h\|_1 = 1 \quad (5.15)$$

We then assign to the probe image the identity of the gallery image that has the minimum residual value r_j . If there are multiple gallery images per person, we can use the k-nearest neighbor rule for assigning identity. The major computational step of the algorithm is the optimization problem of (5.15). This is a non-convex problem. To solve this problem we use an alternation algorithm in which we alternately minimize over h and α s, i.e., in one step we minimize over h keeping α_m s fixed and in the other step we minimize over α_m s keeping h fixed and we iterate till convergence. Each step is now a convex problem: the optimization over h given α_m reduces to the same problem as (5.7) and the optimization of α s given h is just a linear least squares problem. One can also formulate the optimization problem (5.15) as a non-convex quadratically constrained quadratic program (QCQP) and obtain a lower bound by solving a relaxed convex problem via semidefinite programming (SDP) [102]. Because of large number of variables involved, we could not pursue this approach. Similar to the previous section, we can show that the proposed algorithm can also be derived as a maximum likelihood estimator of the blur kernel,

illumination coefficients and the identities.

Like section 5.1, we can also find the amount of blur and illumination variations our algorithm can handle for any given dataset. For each gallery image I_j , we have an associated set BI_j of images which captures all the variations due to blur and illumination of I_j . For a given amount of blur we can find the separations between any pairs of sets BI_i and BI_j by solving

$$s(i, j) = \min_{h_i, \alpha_m, h_j, \alpha_n} \left\| \sum_{m=1}^9 \alpha_m A_{i,m} h_i - \sum_{n=1}^9 \alpha_n A_{j,n} h_j \right\|^2$$

subject to $h_i \geq 0, h_j \geq 0, \|h_i\|_1 = 1, \|h_j\|_1 = 1.$ (5.16)

The amount of blur for which the minimum separation between any pair of sets becomes zero is a good measure of the maximum blur that the proposed algorithm can handle for a given dataset under all possible illumination conditions.

5.3 Experimental Evaluations

We evaluate our algorithm on some synthetically blurred datasets like FERET [75] and PIE [85], and a real dataset of remotely acquired face images with lots of blur and illumination variations [69], which we will refer to as the REMOTE dataset. In section 5.3.1 we evaluate our ‘blur only’ recognition algorithm (FRB) of section 5.1 on well-illuminated but blurred images and in section 5.2 we evaluate our ‘joint blur and illumination’ algorithm (FRBI) of section 5.2 on blurred as well as poorly illuminated images.

5.3.1 Recognition across Blur

We evaluate our algorithm for handling blur on the artificially blurred images of the FERET dataset and the naturally blurred images of the REMOTE dataset.

Experiments on the FERET dataset. On this dataset we compare our algorithm with FADEIN, the algorithm proposed in [70]. For a fair comparison, we use the same experimental set-up as used for FADEIN, i.e., a subset of 1001 individuals from the ‘fa’ and ‘fb’ sub-directories of FERET are used as gallery and probe respectively. There is one image per person in the gallery and the probe images are obtained by artificially blurring the fb images by Gaussian kernels of σ values 0, 2, 4, 6, 8, see figure 5.1.

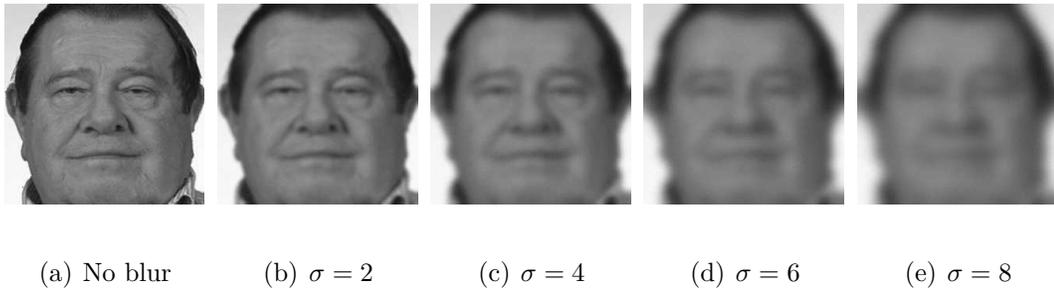


Figure 5.1: Sample probe images from FERET dataset. The probe images are synthetically blurred with Gaussian filters of $\sigma = 0, 2, 4, 6, 8$ respectively. $\sigma = 0$ stands for ‘no blur’.

To handle small variations in illumination we histogram-equalize all the images in the gallery and probe datasets. While optimizing over the space of blur kernels in (5.7), we use kernel size of $2\sigma + 1$ along with circular symmetry constraint for the blur kernel. Figure 5.2(a) shows the recognition result for different σ values obtained by our algorithm. For comparison we show the corresponding recognition results by

FADEIN, LPQ [71] and FADEIN+LPQ [70] as well. LPQ (local phase quantization) are blur insensitive image descriptors which has been used in [6] for recognizing blurred faces. FADEIN+LPQ has been proposed in [70] as an improvement over FADEIN, where a LPQ descriptor is constructed from the deblurred image produced by FADEIN. Referring back to figure 5.2(b), we can conclude that FRB is better than FADEIN and LPQ. FRB is comparable with FADEIN+LPQ for small values of σ , but outperforms it for large values.

Next we test the effectiveness of our algorithm in handling motion blur. For this experiment, we use a smaller dataset of 100 images each from the fa and fb sub-directories of FERET. As before images from fa form our gallery, and the images from fb are blurred with different motion kernels to form the probe. While optimizing over the space of blur kernels in (5.7), we impose symmetry about the origin. The recognition results by FRB are tabulated in 5.1. It is easy to conclude that FRB performs well for motion-blurred images too. To generalize our experiments further we construct datasets where no implicit assumption can be made on the type of blur. We blur each image of the (smaller) fb dataset by 3 different motion blur kernels and 3 different Gaussian blur kernels. While solving for the blur kernels, we do not impose any symmetry constraints. The results are tabulated in 5.2. Clearly, our algorithm fares well in this case too.

In all the previous experiments, we made an optimal choice for the kernel-size. For example, for a Gaussian blur of specific σ , we chose a kernel size of $2\sigma + 1$. Here we study the effect of kernel size on the performance of our algorithm. We also examine the implications of imposing the symmetry constraints on the performance.

We create a probe dataset by blurring images from the smaller fb sub-directory by Gaussian blur of $\sigma = 4$ and solve the optimization problem (5.7) with choices of kernel size ranging from 1 to $16\sigma + 1$. We perform two experiments: One where we impose the circular symmetry constraint and the other where we do not impose any symmetry constraint. From figure 5.3 we can conclude the following: 1) Our algorithm is not very sensitive to the choice of kernel-size and 2) the imposition of symmetry constraints further relaxes the need for accurate choice of kernel-size.

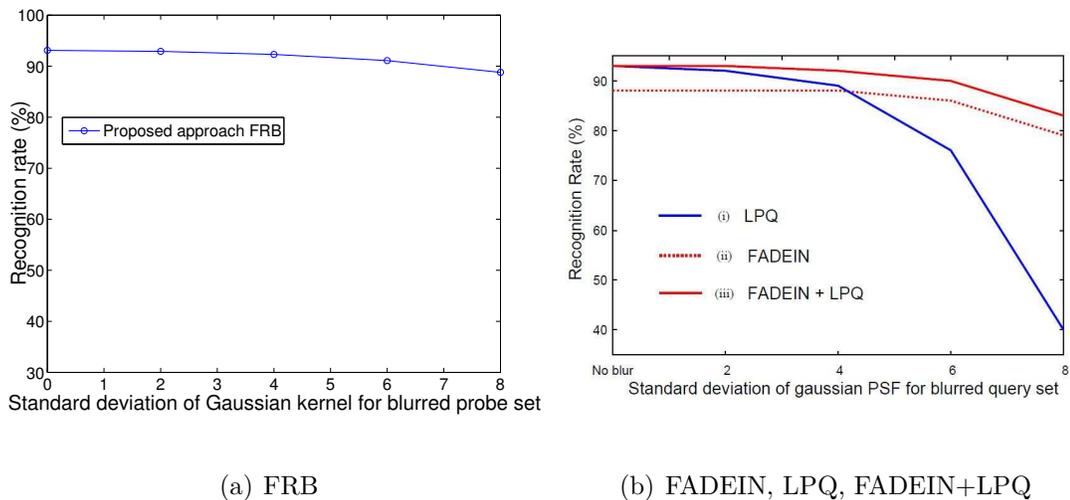


Figure 5.2: *a)* Recognition by our proposed algorithm FRB and *b)* by FADEIN, LPQ and FADEIN+LPQ (figure courtesy [70]) on the FERET dataset. FRB is better than FADEIN and LPQ. FRB is comparable with FADEIN+LPQ for small values of σ , but outperms it for large values.

Experiments on the REMOTE dataset. Images in the REMOTE dataset were captured from distances ranging between 5 to 250 meters under uncontrolled outdoor conditions. Hence, the images suffer from varying amount of blurs, mostly, low-resolution and out of focus blurs. It also has large variations in illumination

Motion blur kernel (L,θ)	(1,0)	(5,0)	(5,45)	(10,45)	(10,90)	(20,0)	(20,45)
Recognition rate (%)	100	100	100	98	98	100	98

Table 5.1: Recognition results for images from the FERET dataset blurred by different motion kernels. While solving for blur kernel in (5.7) we also impose the ‘symmetry about the origin’ constraint on the kernel. These results show that FRB generalizes well for motion blur.

Blur kernels	No Blur	M(10,45)	M(10,90)	M(20,0)	G(2)	G(4)
Recognition rate (%)	100	100	98	96	100	100

Table 5.2: Recognition results for images from FERET that have been degraded by different blur kernels. $M(L,\theta)$ represents motion blur and $G(\sigma)$ represents Gaussian blur. No symmetry constraints have been imposed while solving (5.7). This shows that FRB generalizes well for all types of blur.

Probe image categories	FRB with no illumination pre-processing	FRB with histogram equalization
Sharp and well illuminated	41.0	55.7
Sharp and poorly illuminated	29.0	38.2
Blurred and well illuminated	36.7	50.4
Blurred and poorly illuminated	32.1	42.4
Overall recognition result	35.2	47.5

Table 5.3: Recognition result by FRB on the REMOTE dataset with and without illumination pre-processing. This shows that even a simple illumination pre-processing step such as histogram-equalization improves the recognition result by about 12% and hence there is clearly a need for better illumination modeling.

and pose, and suffer from occlusions. The dataset has been manually labeled with qualitative labels specifying the amount of blur (mild or severe), (good or poor) illumination, (frontal or profile) pose and (no or partial or full) occlusion. Since we are handling blur and illumination, we use a subset of images that are mostly frontal and has no occlusions but has large variations due to blur and illumination. The dataset contains images of 17 individuals. We designed our gallery to have a single frontal, sharp and well-illuminated image per person. The rest of the images form the probe set. We further sub-divided the probe set into four categories: 1) sharp and well-illuminated images (290 images), 2) sharp and poorly-illuminated images (217 images), 3) blurred and well-illuminated images (371 images) and 4) blurred and poorly-illuminated images (271 images). Figure 5.4 shows some images from each category. We use our FRB algorithm to perform face recognition on this dataset. Since most of the images have low-resolution and out-of-focus blur, we have used the circular-symmetry constraint while solving for the blur kernel. We have used a blur kernel size of 17×17 . Table 5.3 shows the recognition results by the algorithm. We have done two experiments, one on the raw images and another on histogram-equalized images. From our experiment, we can conclude the following:

- 1) this dataset is a much more challenging dataset than the FERET dataset and
- 2) even a simple illumination pre-processing step such as histogram-equalization improves the recognition result by about 12% and hence there is clearly a need for better illumination modeling.

5.3.2 Incorporating Illumination Model

We evaluate FRBI, our proposed algorithm to handle blur and illumination variations simultaneously, on PIE and REMOTE datasets both of which have significant variations in illumination. On the PIE dataset, first we test the efficacy of the FRBI algorithm in modeling illumination only without taking blur into consideration. The PIE dataset contains images of 68 individuals under different illumination conditions. We follow the same experimental set up as in the face recognition experiments of [9]: Recognition is performed across illumination with images from one illumination condition forming the gallery and the images from a different illumination set forming the probe. We use the well-illuminated images under the illumination condition f_{21} as our gallery. Since, in this experiment, we are testing only the illumination model of the FRBI algorithm, we only optimize over the space of illumination coefficients α_m , while H is set to the impulse function in (5.15). Table 5.4 shows the result from our algorithm and that from [9]. The recognition algorithm proposed in [9] estimates the albedo maps of the gallery and probe images and use them for recognition. Our better results indicate that we have modeled the illumination well in FRBI.

In our next experiment, we artificially blur the PIE dataset and then use FRBI to perform recognition on it. We use the images under the illumination condition f_{21} as our gallery, and thus our gallery has a single image per individual. We divide the rest of the illumination conditions into two categories: 1) sharp and well-illuminated images ($f_{09}, f_{11}, f_{12}, f_{20}$) and 2) sharp and poorly-illuminated im-

	f_{09}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}	f_{20}	f_{22}	Ave
Proposed approach	98.6	100	100	100	100	100	100	100	100	100	99.9
Albedo-based approach [9]	99	100	100	100	100	93	94	85	100	97	97

Table 5.4: We test the illumination modeling capability of the proposed algorithm FRBI on the PIE dataset. We use the well-illuminated images under the illumination condition f_{21} as our gallery and the rest of the images in $f_{09}, f_{11}, f_{12}, f_{13}, f_{14}, f_{15}, f_{16}, f_{17}, f_{20}, f_{22}$ as probe. We also compare our results with the albedo-based recognition algorithm of [9]. From our better results we can conclude that we have modeled the illumination well in FRBI.

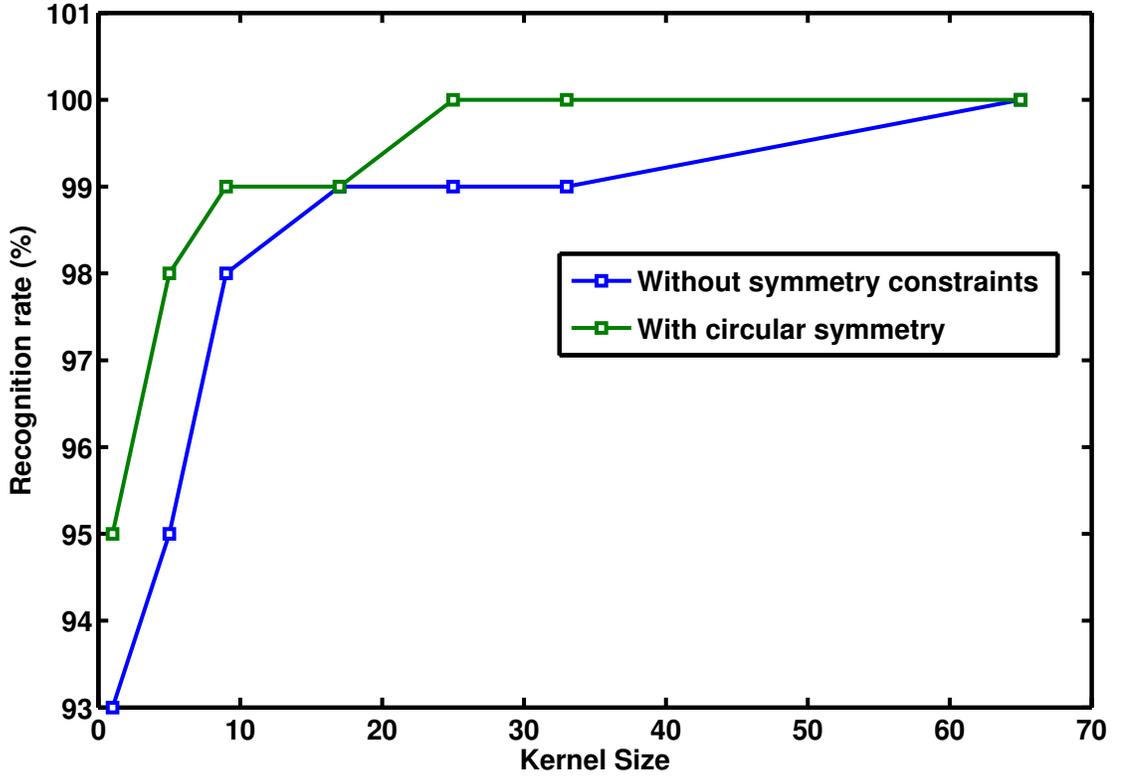
Probe image categories	FRB with histogram equalization	FRBI
Sharp and well illuminated	94.5	99.7
Sharp and poorly illuminated	72.5	100
Blurred and well illuminated	75.7	99.7
Blurred and poorly illuminated	59.5	99.1
Overall recognition result	71.1	99.5

Table 5.5: Recognition results by FRB with histogram pre-processing and FRBI on the PIE dataset. FRBI has an average accuracy rate of almost 100%, whereas that of FRB is 71%.

Probe image categories	FRB with no illumination pre-processing	FRB with histogram equalization	FRBI
Sharp and well illuminated	41.0	55.7	53.8
Sharp and poorly illuminated	29.0	38.2	44.7
Blurred and well illuminated	36.7	50.4	45.3
Blurred and poorly illuminated	32.1	42.4	47.6
Overall recognition result	35.2	47.5	47.9

Table 5.6: Recognition results by FRB with and without illumination pre-processing and FRBI on the REMOTE dataset. On the poor-illumination categories, FRBI gives better results than FRB with histogram pre-processing. However, on the well-illuminated image categories FRB with illumination pre-processing gives better result than FRBI. This result can be attributed to the fact that we have used an average 3-D face model for obtaining the nine basis images which, effectively, means that we are not using the discriminative nature of the shape information for recognition. However, potentially, we can improve the performance of FRBI by using individual 3-D shape models, which can be estimated using the method presented in [9].

ages ($f_{13}, f_{14}, f_{15}, f_{16}, f_{17}, f_{22}$). We further blur these images by Gaussian blur of $\sigma = 1, 2$ to obtain two more categories: 3) well-illuminated and blurred images and 4) poorly-illuminated and blurred. Next for each gallery image we obtain the corresponding nine illumination basis images. Figure 5.5 shows the basis images of a gallery image. After obtaining these basis images, we perform recognition using FRBI. Table 5.5 shows the results obtained by FRBI and by FRB with histogram pre-processing. FRBI has an average accuracy rate of almost 100% whereas that of FRB is 71%. This experiment clearly demonstrates the importance of proper illumination modeling along with the blur. Next we use FRBI to perform recognition on the challenging REMOTE dataset. Figure 5.6 shows the nine illumination basis images of an individual in the REMOTE dataset. These basis images are used in the FRBI algorithm to model illumination variations. Table 5.6 shows the result by FRBI for the 4 categories of probe images. It also shows the result obtained by FRB, with and without the pre-processing step of histogram-equalization. On the poor-illumination categories FRBI gives better results than FBI with histogram pre-processing. However, on the well-illuminated image categories FRB with illumination pre-processing gives better result than FRBI. This result can be attributed to the fact that we have used an average 3-D face model for obtaining the nine basis images which effectively means that we are not using the discriminative nature of the shape information for recognition. However, potentially, we can improve the performance of FRBI by using individual 3-D shape models which can be estimated in the framework of [9].



(a)

Figure 5.3: The effect of kernel size on the performance of our algorithm FRB. The probe images are blurred by a Gaussian kernel of $\sigma = 4$. From these curves we conclude the following: 1) FRB is not very sensitive to the choice of kernel-size and 2) the imposition of symmetry constraints further relaxes the need for accurate choice of kernel-size.



(a) Sharp and well-illuminated

(b) Sharp and poorly-illuminated



(c) Blurred and well-illuminated

(d) Blurred and poorly-illuminated

Figure 5.4: For the experiment on REMOTE dataset, we have divided the probe images into four categories: *a*) sharp and well-illuminated images, *b*) sharp and poorly-illuminated images, *c*) blurred and well-illuminated images and *d*) blurred and poorly-illuminated images. These images were acquired at distances between 5 – 250 meters.

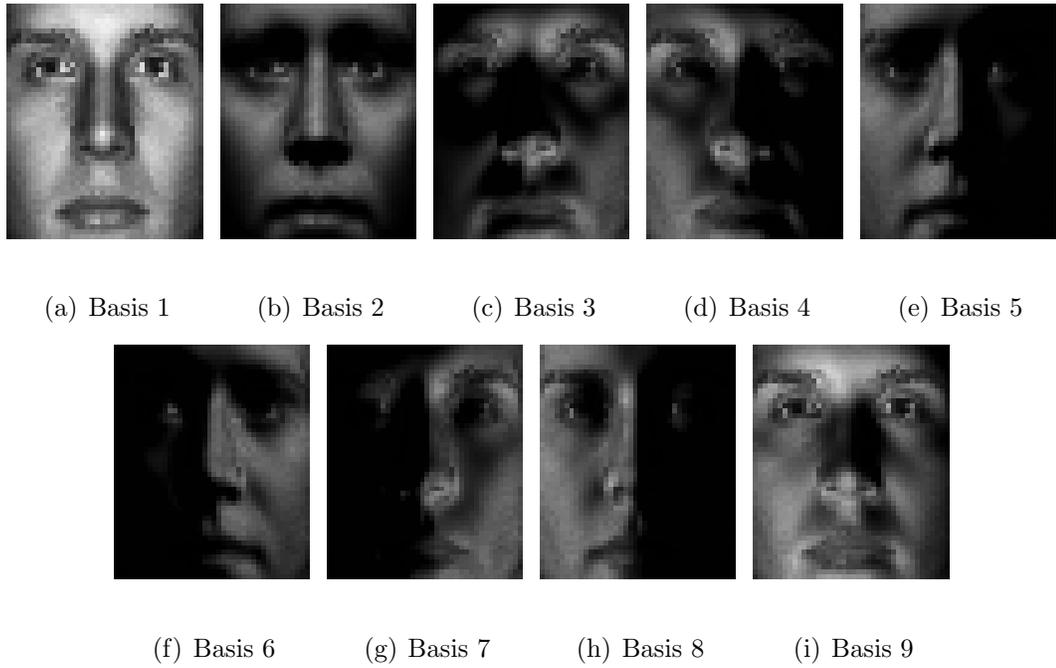


Figure 5.5: The nine illumination basis images of an individual in the PIE dataset. These basis images are used in the FRBI algorithm to model illumination variations.

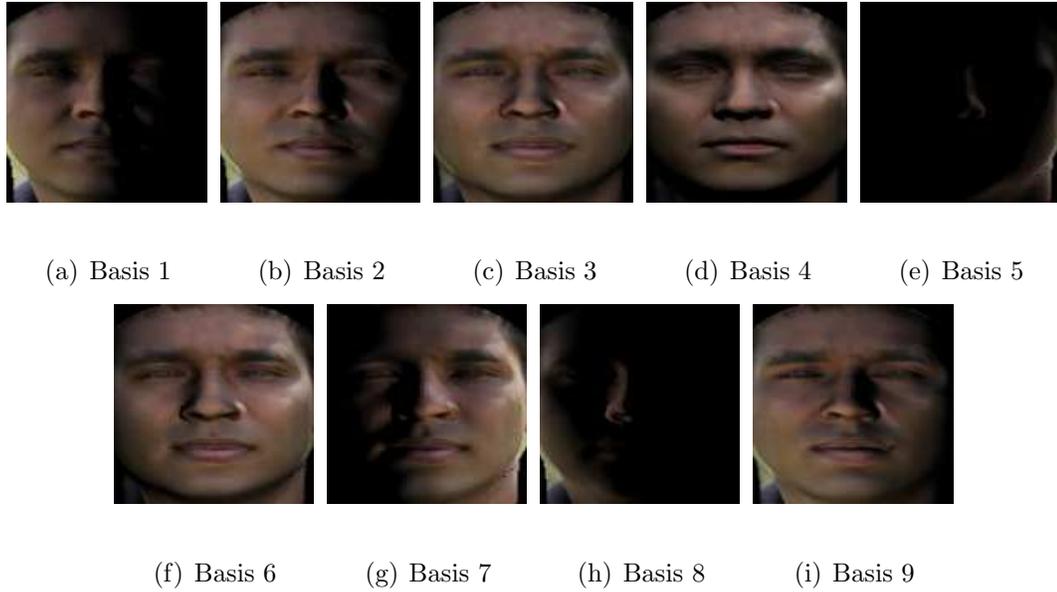


Figure 5.6: The nine illumination basis images of an individual in the REMOTE dataset. These basis images are used in the FRBI algorithm to model illumination variations. The nine illumination positions from which the basis images are created has been optimized for this dataset.

Chapter 6

A Scalable Projective Bundle Adjustment Algorithm using the l_∞ Norm

Structure from Motion is the problem of reconstructing the 3-D structure of an observed scene and the camera parameters (orientations and locations) from multiple images or video of the scene. Bundle adjustment is the final optimization step of the SfM problem, where the structure and camera parameters are refined starting from an initial reconstruction. This is done by minimizing a norm of the image reprojection error, which is defined as the error between the reprojected image points, obtained from the current estimates of camera and structure parameters, and the observed image points.

The l_2 norm of reprojection error is the most commonly used cost function [7]. The main reason for this choice of the norm is that the cost function becomes a differentiable function of parameters and this allows the use of gradient and Hessian-based optimization methods. However, there are two main problems in minimizing this cost function: One, it is a non-linear and non-convex function of the camera and structure parameters. Even the simpler problem of estimating the structure parameters given the camera parameters, known as the triangulation or intersection problem, is a nonlinear and non-convex optimization problem. The corresponding cost function might have multiple minima and finding the global minimum is a

difficult problem. The same is true for the problem of estimating the camera parameters given the structure parameters; this problem is known as the resection problem. Another problem with minimizing the l_2 norm-based cost function is its high computational requirements. A second order algorithm, Levenberg-Marquardt(LM), is used for solving the problem, which has a computational complexity of $O((m+n)^3)$ per iteration and memory requirement of $O(mn(m+n))$, where m is the number of cameras and n is the number of structure points. There exists a more efficient algorithm, sparse bundle adjustment [58], that takes advantage of the sparse structure of the Jacobian matrix of the problem. The computational complexity of this algorithm is $O(m^3 + mn)$ per iteration and the memory requirement is $O(mn)$.

Another norm that is geometrically and statistically meaningful is the l_∞ norm. Minimizing the l_∞ norm is the same as minimax estimation in statistics. Apart from this significance, the l_∞ norm of reprojection error has a nice analytical form for the triangulation and resection problems: It is a *quasi-convex* function of the unknown parameters. A quasi-convex function has the property that any local minimum is also a global minimum. The global minimum can be obtained using a bisection algorithm ([48],[50]). Further, each step of the bisection algorithm can be solved by checking for the feasibility of a second order cone programming problem(SOCP) [48],[50], for which efficient software packages such as SeDuMi [91] are readily available.

The availability of efficient means of finding the global solution for the triangulation and resection problems in the l_∞ norm prompts us to look for bundle adjustment algorithms using the same norm. Joint structure and camera parameter

estimation in the l_∞ norm is not a quasi-convex optimization problem because of the non-linear coupling of structure and camera parameters. However, if we fix one of the unknowns, say structure, and optimize over the camera parameters, we are back to the original problem of l_∞ resection. The next step would be to fix the camera parameters and optimize over the structure (l_∞ triangulation problem). These two steps could be iterated till convergence to a local minimum. This algorithm is an instance of alternation algorithms, and, more specifically, of resection-intersection algorithms in bundle adjustment [7]¹. The proposed algorithm, using the l_∞ norm, has two advantages. One, fixing one of the unknown set of parameters, say the structure parameters, the camera parameters estimation problem can be solved for each camera separately, which effectively reduces the high-dimensional problem to many low-dimensional subproblems. The same is true when the camera parameters are fixed and the structure parameters are estimated. Hence, a high-dimensional parameter estimation problem gets transformed into many low-dimensional subproblems. The second advantage is that the subproblems that we have to solve are all quasi-convex optimization problems, whose global minimum can be efficiently found.

Our goal is to design a *projective bundle adjustment* algorithm and so the triangulation and resection subproblems have to be solved in the projective space [41]. In [48] and [50], the l_∞ triangulation problem is solved in the Euclidean/affine space where the optimization is done over the convex region *in front of* all the cameras from which the point is visible. This region is well defined in Euclidean/affine space,

¹In this discussion, we use “intersection” and “triangulation” interchangeably.

but not in the projective space. The search for this region can increase the computational cost of the projective triangulation problem [49]; the same is true for the projective resection problem. We have avoided these computations by initializing our algorithm using a *quasi-affine reconstruction*, which can be easily obtained from an initial projective reconstruction by solving a linear programming problem [41].

Other related works: There are resection-intersection algorithms based on the l_2 norm. The algorithms proposed by Chen et. al. [25] and Mahamud et. al. [83] are some examples of this. These algorithms have almost the same computational complexity as our algorithm but they are based on minimizing algebraic errors, which are approximations of the l_2 reprojection error. These approximations make them susceptible to wrong solutions [83].

The organization of the rest of this chapter is as follows: In section 6.1, we provide some necessary background on solving the triangulation and resection problems in the l_∞ framework. In section 6.2, we discuss the proposed l_∞ bundle adjustment algorithm. In section 6.3, we compare the computational complexity and memory requirements of our algorithm with the l_2 bundle adjustment algorithm and the l_2 based resection-intersection algorithms. In section 6.4, we evaluate our algorithm for convergence, computational complexity, and robustness to noise with appropriate comparisons to other algorithms.

6.1 Background: geometric reconstruction problems using L_∞ norm

We give a very brief overview of the problem of minimizing the L_∞ norm for (Euclidean/affine) triangulation/intersection and resection problems. For more details see Kahl [48] and Ke et. al [50]. We begin with the definition of a quasi-convex function since the triangulation and resection cost functions reduce to this form.

Definition 1. *A function $f(X)$ is a quasi-convex function if its sublevel sets are convex.*

6.1.1 Triangulation/Intersection

Let $P^j = (a^{jT}; b^{jT}; c^{jT}), i = 1, 2, \dots, M$ be the 3×4 projection matrices for M cameras and $(u^j, v^j), j = 1, 2, \dots, M$ be the images of the unknown 3D point X in these M cameras. The problem is to estimate X given P^j and (u^j, v^j) . Let \tilde{X} be the homogeneous coordinate of X i.e, $\tilde{X} = (X; 1)$. Then the reprojected 2-D image point in camera j (in Euclidean coordinates) is given by $\left(\frac{a^{jT}\tilde{X}}{c^{jT}\tilde{X}}, \frac{b^{jT}\tilde{X}}{c^{jT}\tilde{X}}\right)$ and the L_2 norm of reprojection error function is given by:

$$\begin{aligned} f^j(X) &= \left\| \left(\frac{a^{jT}\tilde{X}}{c^{jT}\tilde{X}} - u^j, \frac{b^{jT}\tilde{X}}{c^{jT}\tilde{X}} - v^j \right) \right\|_2 \\ &= \left\| \left(\frac{a^{jT}\tilde{X} - u^j c^{jT}\tilde{X}}{c^{jT}\tilde{X}}, \frac{b^{jT}\tilde{X} - v^j c^{jT}\tilde{X}}{c^{jT}\tilde{X}} \right) \right\|_2 \end{aligned} \quad (6.1)$$

In Euclidean triangulation, i.e., when the projection matrices are of the form $P^j = [R^j t^j]$, where $R^j \in SO(3)$ and $t^j \in \mathbb{R}^3$, the fact that X is in front of camera i is

expressed by: $c^{jT} \tilde{X} > 0$ (also known as cheirality constraint). With this constraint, we have:

$$f^j(X) = \frac{\left\| \begin{matrix} a^{jT} \tilde{X} - w^j c^{jT} \tilde{X}, b^{jT} \tilde{X} - v^j c^{jT} \tilde{X} \end{matrix} \right\|_2}{c^{jT} \tilde{X}} \quad (6.2)$$

$$= \frac{p^j(X)}{q^j(X)}$$

$p^j(X)$ is a convex function because it is a composition of a convex function (norm) and an affine function. $q^j(X)$ is an affine function. Functions of the form of $f^j(X)$ are quasi-convex [102]. The L_∞ norm of reprojection error is

$$F_\infty(X) = \max_j f^j(X), \quad (6.3)$$

which is again a quasi-convex function, as point-wise maximum of quasi-convex functions is also quasi-convex [102].

Minimization of the quasi-convex function F_∞ can be done using a bisection algorithm in the range of F_∞ ([48], [50]). One step in the bisection algorithm involves solving the following feasibility problem:

$$\text{find } X \quad \text{s.t.} \quad X \in S_\alpha \quad (6.4)$$

where S_α is the alpha sub-level set of $F_\infty(X)$ with the cheirality constraint $q^j(X) > 0$.

For triangulation,

$$S_\alpha = \{X | f^j(X) \leq \alpha, q^j(X) > 0, \forall j\}$$

$$= \{X | p^j(X) - \alpha q^j(X) \leq 0, q^j(X) > 0, \forall j\} \quad (6.5)$$

S_α is a convex set and hence we have to solve a convex feasibility problem. Moreover, since $p^j(X)$ is a L_2 norm, this problem is a second order cone programming problem which can be efficiently solved using software packages like Sedumi [91].

6.1.2 Resection

Here we are given N 3D points $X_i, i = 1, 2, \dots, N$ and their corresponding image points $(u_i, v_i), i = 1, 2, \dots, N$. The problem is to estimate the 3×4 camera projection matrix $P = [a^T; b^T; c^T]$. The reprojected 2-D image point corresponding to the i^{th} 3-D point is given by $\left(\frac{a^T \tilde{X}_i}{c^T \tilde{X}_i}, \frac{b^T \tilde{X}_i}{c^T \tilde{X}_i}\right)$ and the l_2 norm of reprojection error function is given by:

$$f_i(P) = \left\| \left(\frac{a^T \tilde{X}_i - u_i c^T \tilde{X}_i}{c^T \tilde{X}_i}, \frac{b^T \tilde{X}_i - v_i c^T \tilde{X}_i}{c^T \tilde{X}_i} \right) \right\|_2 \quad (6.6)$$

The l_∞ norm of reprojection error is

$$F_\infty(P) = \max_i f_i(P), \quad (6.7)$$

Again, the l_∞ reprojection error is a quasi-convex function of the unknown camera parameters and the global minimum can be obtained as in the triangulation case [48],[50].

6.2 The l_∞ projective bundle adjustment algorithm

For l_∞ projective bundle adjustment, we propose an iterative algorithm based on the principle of resection-intersection. We partition the unknown structure and camera parameters into two separate sets and minimize the l_∞ norm of reprojection

error over one set of parameters while keeping the other set fixed. In the resection step, the minimization is done over the camera parameters while keeping the structure parameters fixed and in the intersection step, the optimization is done over the structure parameters while keeping the camera parameters fixed. These resection-intersection steps are iterated many times till the algorithm converges to a stationary point.

The resection and intersection steps of the proposed algorithm are still a high-dimensional optimization problem. In section 6.2.1, we show that how these two steps can be further simplified by solving a large number of small optimization problems. In section 6.2.2, we discuss the correct way to initialize our algorithm.

6.2.1 Decoupling

Consider the intersection step of the algorithm, where the camera parameters are fixed and minimization of the l_∞ norm of reprojection error is done over the structure parameters. Let $P^j, j = 1, 2, \dots, M$ be the given projection matrices of M cameras and $X_i, i = 1, 2, \dots, N$ be the N 3D points, which are to be estimated. Let f_i^j be the l_2 norm of reprojection error for the i -th 3D point imaged in the j -th camera.

The l_∞ norm of reprojection error is:

$$\begin{aligned}
 F_\infty(X_1, X_2, \dots, X_N) &= \max_{i,j} f_i^j(X_1, X_2, \dots, X_N) \\
 &= \max_i \max_j f_i^j(X_i) \\
 &= \max_i f_{\infty,i}(X_i)
 \end{aligned} \tag{6.8}$$

where,

$$f_{\infty,i}(X_i) = \max_j f_i^j(X_i). \quad (6.9)$$

From equation (6.8), we conclude that $F_{\infty}(X_1, X_2, \dots, X_N)$ can be minimized jointly over all the structure variables by minimizing each of the $f_{\infty,i}(X_i)$ individually over X_i . Hence, the large optimization problem can be solved by solving many (N) small problems. Moreover, we can solve all these problems in parallel. The same is true for the resection step; we can optimize each of $f_{\infty,j}(P^j)$ individually over P^j to obtain the joint optimal solution $F_{\infty}(P^1, P^2, \dots, P^N)$.

6.2.2 Cheirality and quasi-affine initialization

A camera projection matrix is of the form $P = K[R \quad t]$, where K is a 3×3 upper-triangular matrix (which has information about the focal length of the camera and is called the internal parameter matrix) and $R \in SO(3)$, $t \in \mathbb{R}^3$ are rotation and translation of the camera coordinate system with respect to the world coordinate system (the 3-D points are described with respect to this system). In a reconstruction problem, the goal is to find the camera projection matrices and 3-D structure points from their 2-D image points. While solving the reconstruction problem, if we impose the upper-triangular matrix constraint on K and the ortho-normality constraint on R , the reconstruction we obtain is an Euclidean reconstruction, i.e., the reconstructed camera matrices and structure points are related to the original quantities by a Euclidean transformation (global rotation and translation). However, if we do not impose the above constraints and just solve for an unconstrained

P matrix, the reconstruction that we get is a projective reconstruction, i.e., it is related to the original reconstruction by a projective transformation. Generally, we first obtain a projective reconstruction, and later use the various constraints to find a projective transformation that can convert the projective reconstruction to an Euclidean one [41]. Our goal is to design a projective bundle adjustment algorithm.

Since bundle adjustment algorithm is an iterative algorithm, we need to initialize the algorithm with an initial reconstruction. The usual way to initialize a projective bundle adjustment algorithm is a projective reconstruction obtained from the given images. Any of the methods mentioned in [41] can be used for projective reconstruction. However doing this for the proposed algorithm will increase its computational complexity. To understand and get around this problem, we need to understand a property known as cheirality.

Let $\mathbf{X} = (X, Y, Z, T)$ be a homogeneous representation of a point and $P = [a^T; b^T; c^T] = [M \ p_4]$ be the projection matrix of a camera, with M a 3×3 sub-matrix and p_4 a column vector. The imaged point x is given by $P\mathbf{X} = \omega\hat{x}$, where \hat{x} denotes the homogeneous representation of x in which the last coordinate is 1. The depth of the point \mathbf{X} with respect to the camera is given by:

$$\text{depth}(\mathbf{X}; P) = \frac{\text{sign}(\det M)\omega}{T||m_3||} \quad (6.10)$$

where m_3 is the third row of M [41]. A point \mathbf{X} is said to be in front of the camera if and only if $\text{depth}(\mathbf{X}; P) > 0$.

Definition 2. *The quantity $\text{sign}(\text{depth}(\mathbf{X}; P))$ is known as the cheirality of the point \mathbf{X} with respect to the camera [41].*

If $\det M > 0$ and $T > 0$, then $c^T \mathbf{X} > 0$ implies that the point is in front of the camera (since $\omega = c^T \mathbf{X}$). Cheirality is an invariant quantity under Euclidean, affine transformations and quasi-affine transformations, but it is not so under projective transformation [41]. In section 6.1, since we were solving Euclidean triangulation and resection problems, we were justified in using the cheirality constraint. However, when solving the projective triangulation problem, we can't just restrict our search for X in the convex region of $\{X : c^{jT} X > 0, \forall j\}$. If there are M cameras, then the M principal planes divide the projective space P^3 into $(M^3 + 3M^2 + 8M)/6$ regions [49]. The l_∞ cost, with respect to X , has to be minimized over each of these regions and the minimum among them is the desired solution of the projective triangulation problem [49].

From the discussion above, to avoid additional computations, we should initialize the projective bundle adjustment algorithm with either Euclidean, affine or quasi-affine reconstruction. Quasi-affine reconstruction is the best choice as it is very easy to convert any (initial) projective reconstruction into a quasi-affine one. The only information required for this conversion is the fact that if a point is imaged by a camera, then it must be in front of the camera. The transformation that takes a projective reconstruction to a quasi-affine reconstruction can be found by solving a linear programming problem [41].

To summarize, we first obtain an initial projective reconstruction from the images and then convert this to a quasi-affine reconstruction by solving a linear programming problem. This reconstruction is then used as an initialization for our algorithm. After this initialization, we can use the triangulation/resection method

Algorithm *Bundle Adjustment using l_∞ norm minimization*

Input: Set of images

Output: Projective reconstruction

1. Do initial projective reconstruction from set of images.
 2. Convert to quasi-affine reconstruction.
 3. **while** Reprojection error $> \epsilon$
 4. **do** Get camera parameters for each camera by l_∞ resection.
 5. Get 3D structure parameters for each point by l_∞ triangulation.
-

Figure 6.1: l_∞ BA Algorithm

of section 6.1. The summary of the algorithm is given in Figure 6.1.

6.3 Computational complexity and memory requirement

This section first describes the computational complexity and memory requirement of the proposed algorithm and then compares it with that of l_2 based bundle adjustment and l_2 based resection-intersection algorithms.

As discussed in section 6.2.1, at any time we are either solving the triangulation problem for one structure point or the resection problem for one camera. We first analyze the computational complexity and memory requirements for solving one triangulation problem. The triangulation problem is solved by a bisection algorithm

[48],[50]. At each step of the bisection, we solve a convex feasibility problem, given in (6.5) of section 6.1. If the point that we are triangulating is visible in m cameras then we have to solve m second order cone feasibility problem. This problem has a computational complexity of $O(m^{1.5})$ and a memory requirement of $O(m)$ [56]. By analogy, the resection problem for one camera has an computational complexity $O(n^{1.5})$ and memory requirement $O(n)$ where n points are visible in that camera. Now consider one iteration of our algorithm. For the case where there are m cameras and n points and all the points are visible in all the cameras, per iteration empirical computational complexity is $O(mn(\sqrt{m} + \sqrt{n}))$ and the memory requirement is $O(\max(m, n))$. Furthermore a parallel implementation of the algorithm is possible because during each resection/intersection step all the cameras/points can be estimated at the same time. This is because of the decoupling discussed in section 6.2.1. Such an implementation will result in a reduction of computational complexity.

The l_2 norm bundle adjustment algorithm (l_2 BA) [7] is based on the Levenberg-Marquardt (LM) method. The central step involves solving an equation with all the camera and structure parameters as unknowns. Hence its computational complexity is $O((m + n)^3)$ per iteration. For memory requirement, we can consider the Jacobian which is $O(mn(m + n))$. There exists a sparse LM method which uses the fact that the Jacobian matrix for the bundle adjustment problem has a sparse structure [41]. For this method, computation complexity is $O(m^3 + mn)$ per iteration. The memory requirement is $O(mn)$ [41]. The l_2 based resection-intersection algorithms [25],[83] have computational complexity of $O(mn)$ per iteration and same memory requirement as our algorithm, i.e, $O(\max(m, n))$ [83]. But since these algorithms

minimize approximate algebraic errors, they are not so reliable as we found in our experiments 6.4.1.

6.4 Experiments

We have done experimental evaluation of the proposed algorithm (l_∞ BA) for convergence, computational scalability and robustness to noise. Comparisons with l_2 bundle adjustment based on LM algorithm (l_2 BA) and l_2 resection-intersection algorithms are also given. For l_2 BA we have used a publicly available implementation of projective bundle adjustment [58] based on the sparse LM method. The l_2 based resection-intersection algorithms that we have compared with are the Weighted Iterative Eigen algorithm (WIE) proposed by Chen et al [25] and a variation of the same algorithm where we avoid the reweighting step, henceforth called the IE algorithm. In section 6.4.1, while studying the convergence of the four algorithms we found that the performance of WIE and IE are unreliable and hence in the rest of the sections we have compared l_∞ BA with only l_2 BA.

For initial reconstruction, we have used the projective factorization method of Triggs et al [13] with proper handling of missing data. This reconstruction was then converted to a quasi-affine reconstruction. Any other projective reconstruction followed by a conversion to a quasi-affine reconstruction will also work fine. When comparing different algorithms, all of them have been provided with the same initial reconstruction.

6.4.1 Convergence

Here we study the convergence with iteration of l_∞ BA, l_2 BA, WIE and IE algorithms. Experiments are done on one synthetic data set and three real data sets. The synthetic data set consists of 100 points distributed uniformly within a unit sphere and 50 cameras in a circle around the sphere looking straight at the sphere. Gaussian noise of standard deviation 1 pixel is added to the feature points. In all the experiments, reprojection error for this data set is the mean reprojection error over ten trials. The real data sets used are the corridor and dinosaur data set² and the hotel data set³. We have used a subset of 464 structure points from the dinosaur data set and a subset of 60 views from the hotel data set. For the corridor and the dinosaur data sets, feature points are already available and we have used them as they are. For the hotel data set we have used the KLT tracker to track feature points and then used Torr’s Matlab SFM Toolbox [99] to remove the outliers.

We compare the convergence of the algorithms in the l_∞ norm of reprojection error and the Root Mean Squares reprojection error (RMS) which is a measure of the l_2 norm. Figure 6.2 shows that the l_∞ error decreases monotonically for l_∞ BA, but not so for the other algorithms. Figure 6.3 shows RMS error decreases (almost) monotonically for l_∞ BA and l_2 BA but not so for WIE and IE. From Figure 6.3, we can further conclude the following. All the algorithms converge well for the sphere data set. For the corridor data set, WIE converges at a higher value than others. For the hotel data set, both WIE and IE first diverge and then later converge. For

²<http://www.robots.ox.ac.uk/~vgg/data/data-mview.html>

³<http://vasc.ri.cmu.edu/idb/>

the dinosaur data set, IE fails to converge. To further study the nature of WIE and IE, we added Gaussian noise of standard deviation 1 pixel to the real data sets and found that the algorithms fail to converge at many trials. However, each time these algorithms have converged in the algebraic cost that they minimize. This study tells us that algebraic cost based algorithms may not be very reliable. For all of the above data sets, our algorithm converges within ten iterations with similar RMS reprojection error as l_2 BA. Figure 6.4 shows the final 3-D reconstruction by our algorithm for the datasets, sphere and corridor.

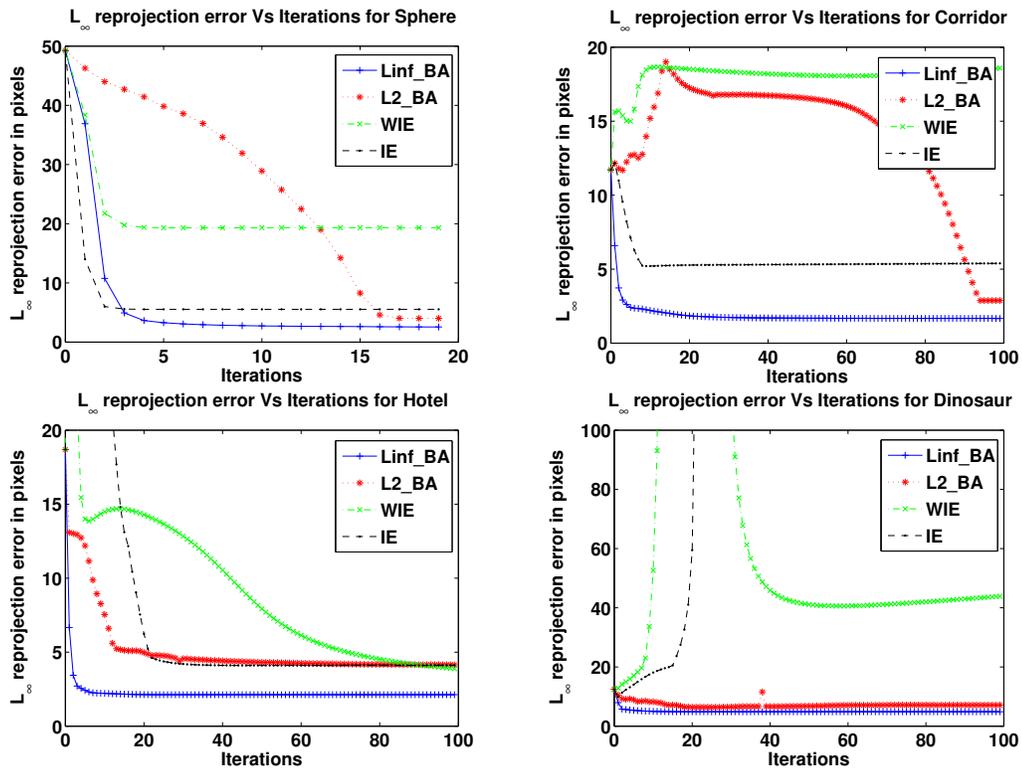


Figure 6.2: l_∞ reprojection error versus iteration for the four algorithms on the data sets: sphere, corridor, hotel and dinosaur. l_∞ error decreases monotonically for l_∞ BA but not so for the other algorithms.

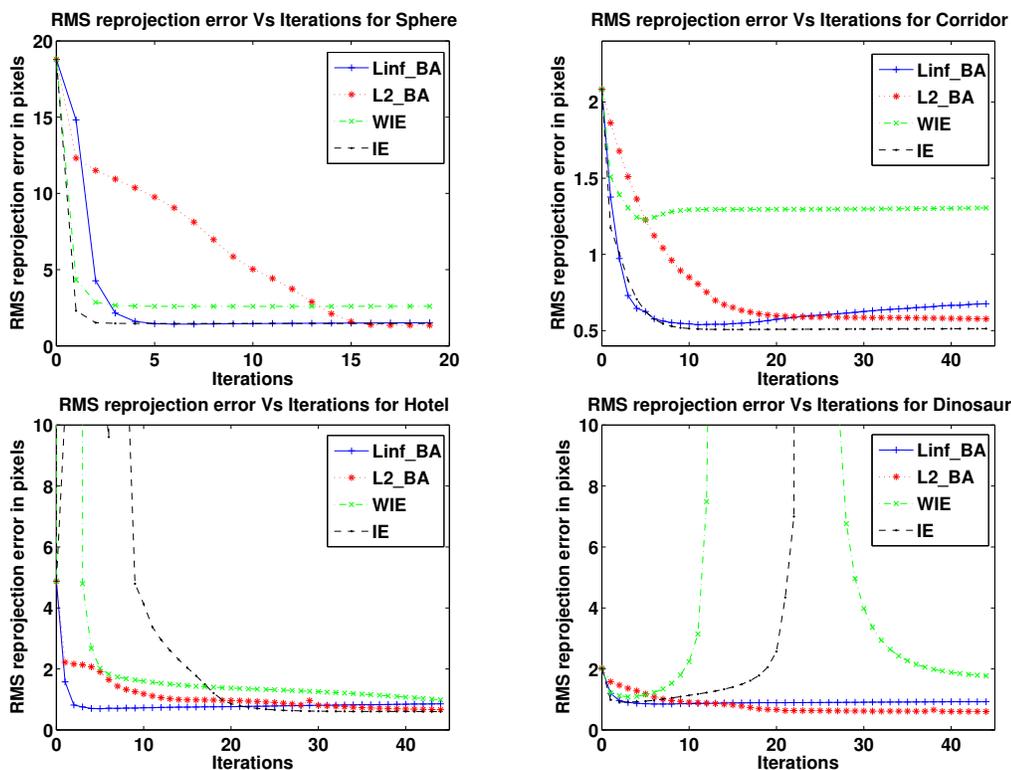


Figure 6.3: RMS reprojection error versus iteration : RMS error decreases monotonically for l_∞ BA and l_2 BA but not so for WIE and IE. IE fails to converge for the dinosaur data set.

6.4.2 Computational scalability

We did experiment on the synthetic sphere data set to compare the total convergence time for l_2 BA and l_∞ BA as the number of cameras is varied with the number of points fixed at 500, Figure 6.5. To ensure a fair comparison, both the algorithms were implemented in Matlab with the computationally intensive routines as mex files. l_2 BA converges at about 10 iterations and l_∞ BA at about 2 iterations. Figure 6.5 clearly shows that our algorithm has the advantage in terms of time from 250 cameras onwards. For a video with 30 frames per second this is approximately

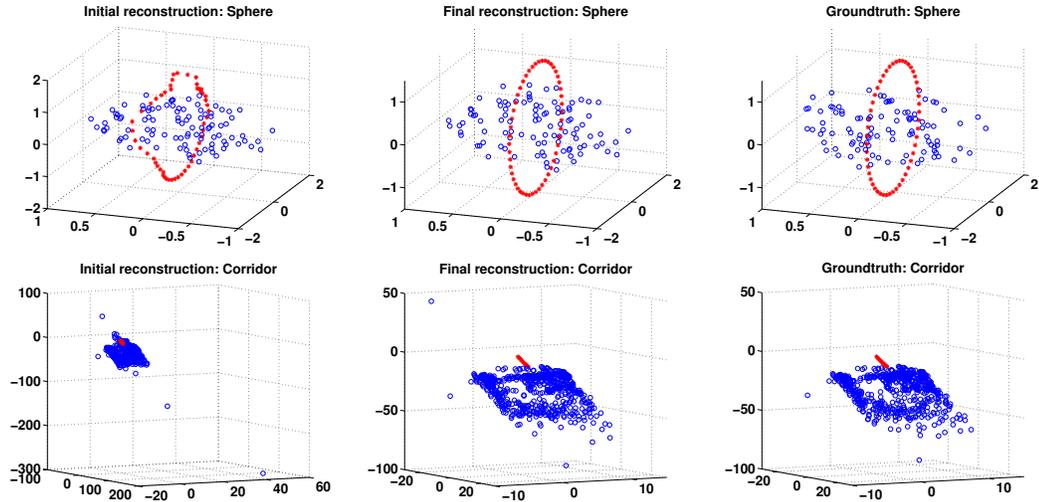


Figure 6.4: 3-D reconstruction result for the datasets, Sphere and Corridor. The Red '*' represents the camera center and Blue 'o' represents the structure point. The first column shows the initialization, second column shows the final reconstruction and the third column shows the groundtruth.

8 sec of data. Note that we have to estimate the camera parameters corresponding to each frame of the video. Thus our algorithm is suitable for solving reconstruction problems for video data where the number of frames can be large.

Recently, there has been some work on faster computations of l_∞ triangulation and resection problems [18] and incorporating this will reduce the convergence time of our algorithm. Further reduction in convergence time is possible by a parallel implementation, which we have not done here.

6.4.3 Behavior with noise

Gaussian noise of different standard deviations are added to the feature points. Figure 6.6 shows the RMS reprojection error in pixels with noise for the synthetic

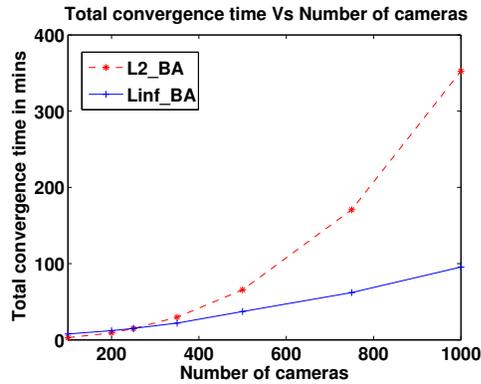


Figure 6.5: Total convergence time of l_2 BA and l_∞ BA as the number of cameras is varied with number of points fixed at 500.

data set, sphere. Generally the l_∞ norm has the reputation of being very sensitive to noise, but here we see a graceful degradation with noise. Further to handle noise with strong directional dependence, we can incorporate the directional uncertainty model of Ke et. al. [51] into the resection and triangulation steps of our algorithm, though we have not done it here. We have not considered outliers here, as bundle adjustment is considered to be the last step in the reconstruction process and outlier detection is generally done in the earlier stages of the reconstruction. In fact as mentioned earlier in section 6.4.1, we have removed the outliers from the hotel data set before the initial reconstruction step.

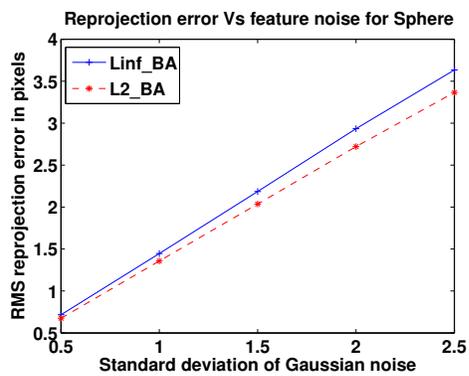


Figure 6.6: Behavior of l_∞ BA and l_2 BA with image feature noise for the sphere data set.

Chapter 7

Conclusion and Future Directions

We summarize and suggest future directions for each of the topics covered in the dissertation. We also propose interesting directions for some related topics.

7.1 Robust Linear Regression Using Sparse Learning for High-Dimensional Applications

Successful robust regression algorithms, such as LMedS and RANSAC are combinatorial in the dimension of the problem, and hence are not useful for solving high-dimensional problems. We proposed robust polynomial time algorithms based on techniques from sparse learning theory. We decomposed the error term in regression as the sum of two terms: an outlier or gross error term, which is assumed to be sparse, and an inlier or small error term. We then formulated the robust regression problem as an l_0 -norm optimization problem and stated the conditions under which it can correctly recover the model parameters in presence of k outliers: The smallest principal angle between the regression subspace and all the $2k$ -dimensional outlier subspaces should be greater than zero and X should be full column rank. Since the above optimization is a combinatorial problem, we proposed a relaxed convex problem BPRR, which is a modified version of the basis pursuit algorithm. We then showed that if the smallest principal angle between the regression and all

the $2k$ -dimensional outlier subspaces is more than $\cos^{-1}(\frac{2}{3})$ and X is full column rank, then BPRR finds the correct model parameters provided there are at most k outliers. We also proposed a Bayesian approach, BRR, for solving the robust regression problem, which is based on the sparse Bayesian learning technique. We then empirically studied the parameter space of the robust regression algorithm, which showed that BRR gives the best performance.

Finding the Maximum Number of Outliers that a Dataset can Handle. The sufficient conditions that we provided for BPRR, Theorem 2.1.1, are in terms of the quantity δ_k (cosine of the smallest principal angle between the regression subspace and all k dimensional outlier subspace). The largest integer k for which $\delta_{2k} < \frac{2}{3}$ provides us a lower bound on the maximum number of outliers (in y variable) that a given dataset can handle. However, the computation of this quantity is itself a combinatorial problem. An interesting direction of research would be to find greedy algorithms that can provide lower and/or upper bound on the maximum number of outliers that a given dataset can handle.

7.2 Robust RVM Regression Using Sparse Outlier Model

We extended our robust linear regression formulation to a particular kernel (non-linear) regression technique, the RVM regression. We explored two natural approaches for incorporating robustness in the RVM model: a Bayesian approach and a regularization approach. In the Bayesian approach (RB-RVM), the robust RVM problem is formulated as a bigger RVM problem with the advantage that it can

be solved efficiently by a fast algorithm. The regularization approach (BP-RVM) is based on the Basis Pursuit Denoising algorithm, which is a popular algorithm in the sparse representation literature. Empirical evaluations of the two robust algorithms show that RB-RVM performs better than BP-RVM. Further, we used RB-RVM to solve the robust image denoising and age estimation problem, which clearly demonstrated the superiority of RB-RVM over the original RVM. As a future direction of research, it would be interesting to look at a similar robust version for RVM classification. Also, the RB-RVM can be applied for solving the image interpolation problem and the 3d human pose estimation problem, where RVM regression gives one of the best performances [4].

7.3 Sparse Regularization for Regression and Classification on Manifolds

There are many applications in vision such as dynamic textures [87], human activity modeling and recognition [104] and shape analysis [73], where the data lies on a non-Euclidean manifold. We are interested in developing regression and classification techniques which would be suitable for such problems. Recent papers by Pelletier et. al. [74, 57] have proposed kernel techniques for regression and classification on closed Riemannian manifolds. However, these techniques lack proper regularization and hence may not *generalize* well, i.e., they may not predict well for unseen data. It would be interesting to look at sparse regularization for these problems. Another direction would be to make them robust to outliers.

7.4 Large-Scale Matrix Factorization with Missing Data under Additional Constraints

Many problems in computer vision, such as SfM and photometric stereo, can be formulated as a missing-data matrix factorization problem, which is a hard problem to solve. We have formulated this problem as a low-rank semidefinite programming problem (MF-LRSDP). MF-LRSDP is an efficient algorithm that can be used for solving large-scale factorization problems. It is also flexible for handling many additional constraints such as the ortho-normality constraints of the orthographic SfM. Our empirical evaluations on synthetic data show that it needs fewer observations for matrix factorization as compared to other algorithms and it gives very good results on the real problems of SfM, non-rigid SfM and photometric stereo. We note that though MF-LRSDP is a non-convex problem, it finds the global minimum under the conditions of the matrix completion theory. As a future work, it would be interesting to find a theoretical justification for this.

Subspace Clustering in the presence of Missing Data. As seen in Chapter 4, the motion of a single object can be well formulated by missing data matrix factorization. If there are multiple objects undergoing different motions, then it can be shown that this problem can be formulated as a subspace clustering problem, where each cluster represents a single motion. For solving this problem, Vidal et. al. [33] have proposed a sparse subspace clustering technique. It would be interesting to extend this technique to the missing data scenario.

7.5 Direct Recognition of Faces across Blur and Illumination Variations

Motivated by the problem of remote face recognition, we have addressed the problem of recognizing blurred and poorly-illuminated faces. We have used the convolution model for blur and a low-dimensional linear subspace model for illumination to propose a direct recognition method. For each gallery image, we have an associated set which represents all the variations due to blur and illumination. Given a probe image, we find its distance from each such (gallery) set and assign it the identity of the closest (gallery) set. We have shown that this algorithm, though based on set theoretic concept, is also statistically optimal; it gives the maximum likelihood estimates for the blur kernel, illumination coefficients and identity. We also provided a way to theoretically characterize the amount of blur our algorithm can handle in a given dataset. Finally, we have demonstrated very good recognition results on many synthetic and real datasets. As an extension, it would be interesting to address the problem of pose variations under the same framework. Also, instead of maximizing the likelihood of the probe image over the joint space of identities, blur kernels and illumination coefficients, one can maximize the marginal likelihood of the probe image over the space of identities. This can be done by integrating the joint likelihood function over the space of blur kernels and illumination coefficients. This approach is likely to improve the recognition accuracy but it will also increase the computational complexity of the algorithm.

Beyond Nearest Neighbor Classification for Face Recognition Across

Blur and Illumination. The proposed algorithm is a nearest neighbor algorithm for the recognizing faces across blur and illumination variations. It is a well known fact that nearest neighbor classifiers are computationally intensive and do not generalize well. It would be interesting to explore other classifiers, such as support vector machines (SVM) [28], for solving this problem.

7.6 Hierarchical Dictionary for Face and Activity Recognition:

Dictionary based face and activity recognition is a promising new direction [61]. We are interested in learning hierarchical (multi-resolution) dictionaries, which will reveal the proper structure of the data and will also lead to scalable algorithms for dictionary-based recognition.

7.7 A Scalable Projective Bundle Adjustment Algorithm using the

L_∞ Norm

The traditional bundle adjustment algorithm, based on minimizing the L_2 norm of the image re-projection error, has cubic complexity in the number of unknowns, and hence, is slow. We have proposed an efficient projective bundle adjustment algorithm using the L_∞ norm. It is a resection-intersection (coordinate descent/alternation) based algorithm which converts the large scale optimization problem to many small scaled ones. It is possible to make the present algorithm faster using a parallel implementation and by a more efficient implementation of L_∞ resection and triangulation.

Bibliography

- [1] The fg-net aging database, <http://www.fgnet.rsunit.com>.
- [2] H. Aanæs, R. Fisker, K. Åström, and J. M. Carstensen. Robust factorization. *IEEE TPAMI*, 2002.
- [3] A. Agarwal and B. Triggs. 3d human pose from silhouettes by relevance vector regression. In *CVPR*, 2004.
- [4] Ankur Agarwal and Bill Triggs. Recovering 3d human pose from monocular images. *IEEE TPAMI*, 2006.
- [5] A. Agrawal, R. Raskar, and R. Chellappa. An algebraic approach to surface reconstruction from gradient fields. In *Intl Conf. Computer Vision*, 2005.
- [6] T. Ahonen, E. Rahtu, V. Ojansivu, and J. Heikkila. Recognition of blurred faces using local phase quantization. In *International Conference on Pattern Recognition*, 2008.
- [7] R. Hartley B. Triggs, P. McLauchlan and A. Fitzgibbon. Bundle adjustment - a modern synthesis. In *Vision Algorithms: Theory and Practice, Springer-Verlag*, 2000.
- [8] Ronen Basri and David W. Jacobs. Lambertian reflectance and linear subspaces. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2003.
- [9] Soma Biswas, Gaurav Aggarwal, and Rama Chellappa. Robust estimation of albedo for illumination-invariant matching and shape recovery. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2009.
- [10] Alan C. Bovik. The essential guide to image processing. *Elsevier*, 2009.
- [11] S. Brandt. Closed-form solutions for affine reconstruction under missing data. In *Stat. Methods for Video Proc. (ECCV 02 Workshop)*, 2002.
- [12] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3d shape from image streams. In *CVPR*, 2000.
- [13] B. Triggs. Factorization methods for projective structure and motion. In *Proc. IEEE Conf. on CVPR*, 1996.
- [14] A. M. Buchanan and A. W. Fitzgibbon. Damped newton algorithms for matrix factorization with missing data. In *CVPR*, 2005.
- [15] S. Burer and C. Choi. Computational enhancements in low-rank semidefinite programming. *Optimization Methods and Software*, 2006.

- [16] S. Burer and R.D.C. Monteiro. A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Mathematical Programming (series B)*, 2001.
- [17] Pei C. Optimization algorithms on subspaces: Revisiting missing data problem in low-rank matrix. *IJCV*, 2008.
- [18] A. P. Eriksson C. Olsson and F. Kahl. Efficient optimization for l_∞ -problems using pseudoconvexity. In *IEEE Int. Conf. Computer Vision*, 2007.
- [19] J. Cai, E. J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 2010.
- [20] E. J. Candes. The restricted isometry property and its implications for compressed sensing. *Comptes Rendus Mathématique*, 2008.
- [21] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations on Computational Mathematics*, 2009.
- [22] E. J. Candes and T. Tao. Decoding by linear programming. *IEEE Transactions on Information Theory*, 2005.
- [23] E. J. Candes and M. Wakin. An introduction to compressive sampling. *IEEE Signal Processing Magazine*, 2008.
- [24] E.J. Candes and P.A. Randall. Highly robust error correction byconvex programming. *Information Theory, IEEE Transactions on*, 2008.
- [25] Q. Chen and G. Medioni. Efficient iterative solution to m-view projective reconstruction problem. In *IEEE Conf. on CVPR*, 1999.
- [26] Scott Shaobing Chen, David L. Donoho, and Michael A. Saunders. Atomic decomposition by basis pursuit. *SIAM Jour. Scient. Comp.*, 1998.
- [27] Kuang chih Lee, Jeffrey Ho, and David Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005.
- [28] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 1995.
- [29] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.
- [30] D. L. Donoho. High-dimensional centrally symmetric polytopes with neighborliness proportional to dimension. *Discrete Computational Geometry*, 2006.
- [31] D. L. Donoho and J. Tanner. Counting faces of randomly projected polytopes when the projection radically lowers dimension. *J. Amer. Math. Soc.*, 2009.

- [32] David L. Donoho, Michael Elad, and Vladimir N. Temlyakov. Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Transactions on Information Theory*, 52(1), 2006.
- [33] E. Elhamifar and R. Vidal. Sparse subspace clustering. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2009.
- [34] R. Epstein, P.W. Hallinan, and A.L. Yuille. 5 plusmn;2 eigenimages suffice: an empirical investigation of low-dimensional lighting models. In *Physics-Based Modeling in Computer Vision, 1995., Proceedings of the Workshop on*, 1995.
- [35] Anita C. Faul and Michael E. Tipping. A variational approach to robust regression. In *ICANN*, 2001.
- [36] M.A. Fischler and R.C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Comm. Assoc. Mach.*, 1981.
- [37] Yun Fu, Ye Xu, and Thomas S. Huang. Estimating human age by manifold analysis of face pictures and regression on aging features. In *ICME*, 2007.
- [38] G. H. Golub and C. F. Van Loan. Matrix computations. *Johns Hopkins University Press, Baltimore, Md*, 1996.
- [39] N. Guilbert, A.E. Bartoli, and A. Heyden. Affine approximation for direct batch recovery of euclidian structure and motion from sparse data. *IJCV*, 2006.
- [40] Guodong Guo, Yun Fu, Charles R. Dyer, and Thomas S. Huang. Image-based human age estimation by manifold learning and locally adjusted robust regression. *IEEE Transactions on Image Processing*, 2008.
- [41] R. Hartley and A. Zisserman. Multiple view geometry in computer vision. *Cambridge University press, 2nd edition*, 2004.
- [42] H. Hayakawa. Photometric stereo under a light source with arbitrary motion. *JOSA*, 1994.
- [43] Hao Hu and Gerard de Haan. Low cost robust blur estimator. In *ICIP*, 2006.
- [44] P.J. Huber. Robust statistics. *Wiley Series in Probability and Statistics*, 1981.
- [45] D. Q. Huynh, R. Hartley, and A. Heyden. Outlier correction in image sequences for the affine camera. In *ICCV*, 2003.
- [46] D. W. Jacobs. Linear fitting with missing data for structure-from-motion. *CVIU*, 2001.
- [47] Y. Jin and B. D. Rao. Algorithms for robust linear regression by exploiting the connection to sparse signal recovery. In *ICASSP*, 2010.

- [48] F. Kahl. Multiple view geometry and the l_∞ -norm. In *IEEE Int. Conf. On Computer Vision*, 2005.
- [49] F. Kahl and R. Hartley. Multiple view geometry under the l_∞ -norm. *IEEE Tran. Pattern Analysis and Machine Intelligence*, 2008.
- [50] Q. Ke and T. Kanade. Quasiconvex optimization for robust geometric reconstruction. In *IEEE Int. Conf. On Computer Vision*, 2005.
- [51] Q. Ke and T. Kanade. Uncertainty models in quasiconvex optimization for geometric reconstruction. In *IEEE CVPR*, 2006.
- [52] R. H. Keshavan and S. Oh. A gradient descent algorithm on the grassman manifold for matrix completion. *CoRR*, *abs/0910.5260*, 2009.
- [53] A Lanitis, C Draganova, and C. Christodoulou. Comparing different classifiers for automatic age estimation. *IEEE TSMC*, 2004.
- [54] A. Lanitis, C. J. Taylor, and T. F. Cootes. Toward automatic simulation of aging effects on face images. *IEEE TPAMI*, 2002.
- [55] K. Lee and Y. Bresler. Admira: Atomic decomposition for minimum rank approximation. *CoRR*, *abs/0905.0044*, 2009.
- [56] M. S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebert. Applications of second-order cone programming. *Linear Algebra and its Applications*, 1998.
- [57] J. Loubes and B. Pelletier. A kernel-based classifier on a riemannian manifold. *Statistics and Decisions*, 2008.
- [58] M. I. A. Lourakis and A. A. Argyros. The design and implementation of a generic sparse bundle adjustment software package based on the levenberg-marquardt algorithm. In *ICS/FORTH Technical Report No. 340*, 2004.
- [59] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 2004.
- [60] S. Ma, D. Goldfarb, and L. Chen. Fixed point and bregman iterative methods for matrix rank minimization. *Mathematical Programming*, 2009.
- [61] J. Mairal, F. Bach, J. Ponce, G. Shapiro, and A. Zisserman. Supervised dictionary learning. In *Advances in Neural Information Processing Systems (NIPS)*, 2008.
- [62] A. Maleki and D. L. Donoho. Optimally tuned iterative reconstruction algorithms for compressed sensing. *IEEE Journal of Selected Topics in Signal Processing*, 2010.
- [63] R. A. Maronna, R. D. Martin, and V. J. Yohai. Robust statistics, theory and methods. *Wiley Series in Probability and Statistics*, 2006.

- [64] D. Martinec and T. Pajdla. 3d reconstruction by fitting low-rank matrices with missing data. In *CVPR*, 2005.
- [65] R. Mazumder, T. Hastie, and R. Tibshirani. Spectral regularization algorithms for learning large incomplete matrices. http://www-stat.stanford.edu/hastie/Papers/SVD_JMLR.pdf, 2009.
- [66] R. Meka, P. Jain, and I. S. Dhillon. Guaranteed rank minimization via singular value projection. *CoRR*, abs/0909.5457, 2009.
- [67] K. Mitra, A. Veeraraghavan, and R. Chellappa. Robust regression using sparse learning for high dimensional parameter estimation problems. In *ICASSP*, 2010.
- [68] Erik Murphy-Chutorian and Mohan Manubhai Trivedi. Head pose estimation in computer vision: A survey. *IEEE TPAMI*, 31, 2009.
- [69] J. Ni and R. Chellappa. Evaluation of state-of-the-art algorithms for remote face recognition. In *ICIP*, 2010.
- [70] Masashi Nishiyama, Abdenour Hadid, Hidenori Takeshima, Jamie Shotton, Tatsuo Kozakaya, and Osamu Yamaguchi. Facial deblur inference using subspace analysis for recognition of blurred faces. *Accepted in IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010.
- [71] Ville Ojansivu and Janne Heikkil. Blur insensitive texture classification using local phase quantization. In *Image and Signal Processing*. Springer Berlin / Heidelberg, 2008.
- [72] T. Okatani and K. Deguchi. On the wiberg algorithm for matrix factorization in the presence of missing components. *IJCV*, 2007.
- [73] V. Patrangenaru and K. V. Mardia. Affine shape analysis and image analysis. In *22nd Leeds Annual Statistics Research Workshop*, 2003.
- [74] B. Pelletier. Non-parametric regression estimation on closed riemannian manifolds. *Journal of Nonparametric Statistics*, 2006.
- [75] P. Jonathon Phillips, Hyeonjoon Moon, Syed A. Rizvi, and Patrick J. Rauss. The feret evaluation methodology for face-recognition algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000.
- [76] J. Portilla, V. Strela, M. Wainwright, and E. P. Simoncelli. Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Transactions on Image Processing*, 2003.
- [77] Ravi Ramamoorthi and Pat Hanrahan. A signal-processing framework for inverse rendering. In *SIGGRAPH*, 2001.

- [78] N. Ramanathan, R. Chellappa, and S. Biswas. Computational methods for modeling facial aging: A survey. *J. Vis. Lang. Comput.*, 2009.
- [79] Carl Edward Rasmussen and Christopher K. I. Williams. Gaussian processes for machine learning. *The MIT press*, 2006.
- [80] J. D. M. Rennie and N. Srebro. Fast maximum margin matrix factorization for collaborative prediction. In *ICML*, 2005.
- [81] P. J. Rousseeuw. Least median of squares regression. *J. of Amer. Sta. Assoc.*, 1984.
- [82] P. J. Rousseeuw and A. M. Leroy. Robust regression and outlier detection. *Wiley Series in Prob. and Math. Stat.*, 1986.
- [83] Y. Omori S. Mahamud, M. Herbert and J. Ponce. Provably-convergent iterative methods for projective structure from motion. In *IEEE Conf. on CVPR*, 2001.
- [84] H. Shum, K. Ikeuchi, and R. Reddy. Principal component analysis with missing data and its application to polyhedral object modeling. *IEEE TPAMI*, 1995.
- [85] Terence Sim, Simon Baker, and Maan Bsat. The cmu pose, illumination, and expression (pie) database, 2002.
- [86] Terence Sim and Takeo Kanade. Combining models and exemplars for face recognition: An illuminating example. In *CVPR 2001 Workshop on Models versus Exemplars in Computer Vision*, 2001.
- [87] Stefano Soatto, Gianfranco Doretto, and Ying Nian Wu. Dynamic textures. In *ICCV*, 2001.
- [88] N. Srebro, J. D. M. Rennie, and T. Jaakkola. Maximum-margin matrix factorization. In *NIPS*, 2004.
- [89] Inna Stainvas and Nathan Intrator. Blurred face recognition via a hybrid network architecture. *Pattern Recognition, International Conference on*, 2000.
- [90] Charles V. Stewart. Robust parameter estimation in computer vision. *SIAM Reviews*, 1999.
- [91] J. Sturm. Using sedumi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization methods and software*, 1999.
- [92] Hiroyuki Takeda, Sina Farsiu, and Peyman Milanfar. Robust kernel regression for restoration and reconstruction of images from sparse noisy data. In *ICIP*, 2006.
- [93] Hiroyuki Takeda, Sina Farsiu, and Peyman Milanfar. Kernel regression for image processing and reconstruction. *IEEE TIP*, 2007.

- [94] J. P. Tardif, A. Bartoli, M. Trudeau, N. Guilbert, and S. Roy. Algorithms for batch matrix factorization with application to structure-from-motion. In *CVPR*, 2007.
- [95] Michael E. Tipping. Sparse bayesian learning and the relevance vector machine. *J. Mach. Learn. Res.*, 2001.
- [96] Michael E. Tipping and Anita Faul. Fast marginal likelihood maximisation for sparse bayesian models. In *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, 2003.
- [97] Michael E. Tipping and Neil D. Lawrence. Variational inference for student-models: Robust bayesian interpolation and generalised component analysis. *Neurocomputing*, 69(1-3), 2005.
- [98] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *IJCV*, 1992.
- [99] P. H. S. Torr. A structure and motion toolkit in matlab “interactive adventures in s and m”. In *Technical report, MSR-TR-2002-56*, 2002.
- [100] P. Turaga, S. Biswas, and R. Chellappa. Role of geometry of age estimation. In *ICASSP*, 2010.
- [101] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Rev.*, 1996.
- [102] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review*, 1996.
- [103] V. N. Vapnik. The nature of statistical learning theory. 1995.
- [104] Ashok Veeraraghavan, Amit K. Roy-Chowdhury, and Rama Chellappa. Matching shape sequences in video with applications in human movement analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005.
- [105] R. Vidal and R. Hartley. Motion segmentation with missing data using powerfactorization and gpca. In *In CVPR*, 2004.
- [106] D. P. Wipf and B. D. Rao. Sparse bayesian learning for basis selection. *IEEE Trans. Signal Process*, 52, 2004.
- [107] R. J. Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineerings*, 1980.
- [108] B. Yang, Z. Zhang, and Z. Sun. Robust relevance vector regression with trimmed likelihood function. In *IEEE Sig. Proc. Letters*, 2007.

- [109] B. Zhang, S. Shan, X. Chen, and W. Gao. Histogram of gabor phase patterns (hgpp): A novel object representation approach for face recognition. *IEEE Transactions on Image Processing*, 2007.
- [110] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Comput. Surv.*, 2003.