Visual data processing over structured dictionaries with applications in light field imaging

Chen, Jie

2016

<https://hdl.handle.net/10356/68594>

<https://doi.org/10.32657/10356/68594>

Visual Data Processing
over Structured Dictionaries
with Applications in Light Field
Imaging

CHEN JIE

School of Electrical and Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirement for the degree of
Doctor of Philosophy

Acknowledgement

First and foremost I want to thank my advisor Lap-Pui Chau. It has been an honor to be his Ph.D student. I appreciate all his contributions of time, ideas, and funding to make my Ph.D experience productive and enjoyable.

My gratitude extends to the Nanyang Technological University, the Infinitus research lab, for the excellent hardware and professional technical support provided.

I would like to thank the members of prof Chau's research group: Boonseng Chew, Cheenhau Tan, Zhenpeng Bian, Junhui Hou, Alexander Matyasko and Hui Liu. The group has been a source of friendships as well as good discussion and collaboration.

I thank individuals that have been involved in my personal and professional time during the four years of my Ph.D study: Zhen Zuo, Yan Wang, Yi Yu, Xiang Li, Peng Hu, Yusong Hu, John Yang, Vinh Luong, and Zaifeng Yang. The four years have been a struggling journey of academic endeavor, but with all those individuals around, it has never been lonely.

Finally, I would like to thank my parents Guiying, and Lyuyue Chen, who raised me with love and supported me in all my pursuits. I have been doing all my best to make you proud.

Chen Jie

Singapore, August 2015

Abstract

Structure exists in all forms of visual data in the nature. In this thesis, we focus on modeling these data structures using sparse representation techniques over a redundant dictionary, which has been proven as an efficient tool in numerous visual signal processing applications. We propose several types of redundant dictionaries with specially designed structures that adapts to the unique application scenarios. Different sparse coding and dictionary training strategies are investigated. Considerations are given to challenging issues such as multi-scale dictionary cross-scale interactions, dictionary disparity segment de-correlation, perspective-shifted dictionary sparse coding acceleration etc. All efforts aim to provide a more powerful frame for representing complicated visual data structures.

For sparse signal representation, the sparsity across the scales is a promising yet under investigated direction. In this thesis, we design a multi-scale sparse representation scheme to explore such potential. A multi-scale dictionary (MD) structure is designed. A Cross-scale Matching Pursuit (CMP) algorithm is proposed for multi-scale sparse coding. Two dictionary learning methods: Cross-scale Cooperative Learning (MD/CCL), and Cross-scale Atom Clustering (MD/CAC) are proposed with each focusing on one of the two important attributes of an efficient multi-scale dictionary: the similarity, and uniqueness of corresponding atoms in different scales. We analyze

and compare their different advantages in the application of image denoising under different noise levels, where both methods produce state-of-the-art denoising results.

The light field (LF) is a function that describes the intensities of light rays in all possible propagation directions. The LF contains large volumes of visual information that can provide a comprehensive understanding of the 3D environment of the scene. In this thesis, a light field dictionary (LFD) based on perspective-shifting is proposed for sparse representation of the highly correlated light field. A two-stage coding algorithm is proposed which uses the Winner-Take-All (WTA) hashing strategy to narrow down the search range for light field sparse coding. The algorithm proves to be able to increase the coding efficiency by almost three times and keep the reconstruction quality almost the same with the original OMP coding.

A compressed sensing framework is proposed for the sampling and reconstruction of a high resolution light field based on a coded aperture camera. Two separate methods, i.e., Sub-Aperture Scan (SAS) and Normalized Fluctuation (NF) are proposed to acquire/calculate the scene disparity, which will be used during the light field reconstruction with the proposed disparity-aware dictionary. A hardware implementation of the proposed light field acquisition/reconstruction scheme is carried out. Both quantitative and qualitative evaluation shows the proposed methods produce state-of-the-art performance in both reconstruction quality and computation efficiency.

Then, a light field compression framework based on the LFD is proposed for efficient storage and transmission of the bulky LF data. A highly efficient adaptive guided filtering algorithm is also proposed for the LF disparity/depth map post-processing. Both Quantitative and qualitative simulations validate the efficiency of the proposed methods.

Contents

Acknowledgement	i
Abstract	ii
Table of Contents	viii
List of Figures	xv
List of Tables	xvii
List of Abbreviations	xviii
List of Notations	xix
1 Introduction	1
1.1 Motivation and Objective	1
1.1.1 Multi-scale Dictionary for Image Denoising	2
1.1.2 Compressed Light Field Sensing and Reconstruction Based on Perspective-Shifted Dictionaries	3
1.1.3 Light Field Data Compression and Post-Processing	4
1.2 Contributions of the Thesis	5

CONTENTS

1.3	Organizations of the Thesis	7
2	Background and Review	9
2.1	Introduction	9
2.2	Sparse Representation and Compressed Sensing	10
2.2.1	The Wavelet Transform	10
2.2.2	The promotion of solution sparsity	11
2.2.3	Solution Uniqueness Analysis	13
2.2.4	Sparsity Pursuit Algorithms	15
2.3	Light Field Acquisition and Data Processing	16
2.3.1	The Light Field Representation	16
2.3.2	Structure of A Microlens-Array Based Light Field Camera	19
2.3.3	The Light Field Data and Epi-polar Plane Image	20
2.4	Conclusion	21
3	Multi-scale Dictionary For Visual Signal Processing	22
3.1	Introduction	22
3.1.1	Related Work	24
3.1.2	Our Contributions	26
3.2	Multiscale Sparse Representation Model	27
3.2.1	The Cross-Scale Matching Pursuit	28
3.3	Multi-scale Dictionary Learning	29
3.3.1	Multi-scale Dictionary Cross-scale Cooperative Learning (MD/CCL)	30
3.3.2	Multi-scale Dictionary Cross-scale Atom Clustering (MD/CAC)	32
3.4	Experiments and Results	37
3.4.1	Dictionary Sparse Coding RMSE	37

CONTENTS

3.4.2	Visual and Quantitative Evaluation of the Multi-scale Dictionary Learning Outcomes	39
3.4.3	MD Parameter Analysis	41
3.4.4	Image Denoising Method and Result	44
3.5	Conclusion: Similarity or Uniqueness	52
4	Perspective Shifted Dictionary for Light Field Representation and Its Fast Coding Technique	55
4.1	Introduction	55
4.2	Perspective Shifted Light Field Dictionary	57
4.3	Two Stage Coding for light field reconstruction based on LFD	62
4.4	Model Evaluation	66
4.5	Conclusion	69
5	Light Field Compressed Sensing	70
5.1	Introduction	70
5.1.1	Light Field Acquisition	70
5.1.2	Light Field Reconstruction	71
5.1.3	Our Contribution	73
5.2	Vignetting and Sensing Matrix Optimization for the Light Field Dictionary	74
5.2.1	The Vignetting Pattern Matrix	74
5.2.2	Sensing Matrix Optimization	77
5.3	Light Field Acquisition and Reconstruction	78
5.3.1	Perspective Flow calculation	78
5.3.2	Light Field Reconstruction	81

CONTENTS

5.4	A Hardware Implementation	82
5.5	Evaluation and Analysis	85
5.5.1	Evaluation of the Sub-aperture Scan Process	85
5.5.2	Measurement Mask Pattern Evaluation	86
5.5.3	Quantitative Evaluation of the Light Field Reconstruction	87
5.5.4	Quantitative Evaluation of Disparity Restoration	90
5.5.5	Qualitative Evaluation of the Reconstructed Light Field	92
5.6	Conclusion	92
6	Light Field Data Post-Processing: Compression and Depth Estima- tion	98
6.1	Introduction	98
6.1.1	Light Field Compression	98
6.1.2	Scene Depth Estimation Based on Light Field Data	99
6.2	<i>LFD Codec</i> Based on Sparse Coding Over A Light Field Dictionary . .	101
6.2.1	Disparity Map Voting	101
6.2.2	Light Field Sparse Coding	103
6.3	Adaptive Guided Filtering Algorithm for Light Field Depth Estimation	104
6.3.1	Definition of Filter Kernel	105
6.3.2	The Patch Window Size Variation Mechanism	107
6.4	Model Evaluation	109
6.4.1	Evaluation of the <i>LFD Codec</i>	109
6.4.2	Evaluation of the AGF Model for Estimation of Scene Depth . .	112
6.5	Conclusion	114
7	Conclusion and Future Work	115

CONTENTS

7.1	Conclusion	115
7.2	Future Work	117
7.2.1	Toward A More Flexible Multi-scale Structure	117
7.2.2	Light Field Capturing and Data Processing	118
	Publication List	120
	Bibliography	132

List of Figures

2.1	demonstration of different norms on 2D vectors.	12
2.2	demonstration of expanding L_p -norm ball to intersect with the feasible set plane $Ax = y$. (a) and (b) are for L_1 -norm and L_2 -norm, respectively. The cross sign shows the point of intersection.	13
2.3	demonstration of light field representation as a 4D data structure. . . .	17
2.4	Demonstration of the image synthesis process for a (a) conventional camera, and (b) micro-lens array based light field camera.	18
2.5	A sensor image directly taken from a micro-lens array based light field camera. (b) shows the raw sensor image, (a) shows a reconstructed image based on the raw data from (b). Photo courtesy of Lytro.	19
2.6	(a) Illustration of the Epi-polar image extraction in a micro-lens array based camera.	20
2.7	Epi-polar Image from the synthesis light field image “Buddha” [1]. . . .	20
3.1	Four natural images with “multi-scale” patterns.	23
3.2	Demonstration of the scale transform matrix “ R_p^l ”. Bilinear interpolation is used for the transformation.	31

LIST OF FIGURES

3.3	Structure of matrix M_k^l . “Bright” columns are fidelity terms, the “dark” column is the cross-scale similarity term.	36
3.4	Coding performance comparison in terms of RMSE between four different dictionaries. Data are the average of five images: <i>Barbara</i> , <i>Peppers</i> , <i>Lena</i> , <i>Boat</i> and <i>House</i>	38
3.5	2-scale MD/CCL dictionary learned for the image <i>Barbara</i> : (a) dictionary in scale 64×256 , (c) dictionary in scale 144×256 ; (b) are the zoomed in part taken from (a) and (b) in red rectangles.	40
3.6	2-scale MD/CAC dictionary for the image <i>Barbara</i> : (a) dictionary in scale 64×256 , (b) dictionary in scale 144×256 ; (c) and (d) are the zoomed in part taken from (a) and (b) in red dotted and solid rectangles respectively.	40
3.7	Comparison on number of matched atoms across different scales after each dictionary update iteration on the image <i>Barbara</i>	42
3.8	Multi-scale dictionary scale selection frequency ratio during CMP for different images. (a) uses a 2-scale dictionary (patch size 8×8 , 12×12 for each scale), and (b) uses a 3-scale dictionary (patch size 8×8 , 12×12 and 16×16 for each scale)	43
3.9	MD/CAC CMP coding performance analysis (in terms of RMSE) under different dictionary atom numbers and coding cardinalities.	44
3.10	A geometrical demonstration of the multi-scale coding process for application of image denoising.	46

LIST OF FIGURES

3.11	Difference of denoising results between 2-scale ($8 \times 8 / 12 \times 12$) MD/CCL, and MD/CAC with MD (no cooperative learning or cross-scale clustering applied) as reference. Results are average of six images: <i>Barbara</i> , <i>Peppers</i> , <i>Lena</i> , <i>Boat</i> , <i>House</i> and <i>Straw</i>	48
3.12	Difference of denoising results between 2-scale ($8 \times 8 / 12 \times 12$) MD/CCL, MD/CAC and MD, with single scale KSVD (8×8) as reference. Results are average of six images: <i>Barbara</i> , <i>Peppers</i> , <i>Lena</i> , <i>Boat</i> , <i>House</i> and <i>Straw</i>	49
3.13	Denoising result on image <i>Straw</i> with Gaussian noise $\sigma = 15$. (a) is the noisy image, the part in red rectangle is zoomed in from (b) to (f), which are the denoising result using (b) K-SVD (64×256 , $26.09dB$), (c) K-SVD (144×256 , $23.80dB$), (d) 2-scale dictionary proposed in [2] ($24.01dB$), (e) 2-scale MD/CCL ($26.50dB$) and (f) 2-scale MD/CAC ($26.20dB$).	52
3.14	Denoising result on image <i>Barbara</i> with Gaussian noise $\sigma = 15$. (a) noisy image, the part in red rectangle is zoomed in from (b) to (f), which are the denoising result using (b) K-SVD (64×256 , $32.37dB$), (c) K-SVD (144×256 , $32.28dB$), (d) 2-scale dictionary proposed in [2] ($32.47dB$), (e) 2-scale MD/CCL ($32.61dB$) and (f) 2-scale MD/CAC ($32.50dB$).	53
4.1	A 5×5 LF view configuration: red and blue arrows represent relative perspective shifting distances in the horizontal and vertical directions for each view.	59

LIST OF FIGURES

4.2	Formation of the proposed LF dictionary by perspective shifting. (a) shows the extraction process of the LF atom ($\sqrt{N^v} \times \sqrt{N^v}$ in grey colour) from a globally trained image dictionary ($\sqrt{N^c} \times \sqrt{N^c}$) with adjacent disparity value dp' . The dotted line surrounded area are symmetric replication of the pattern at the border of the solid rectangle inside. (b) shows the situation when adjacent disparity is at another increased value dp'' . (c) shows the structure of the LF dictionary D_{LF} . Each column is concatenated with vectorized view patches. The different brightness for each LF atom represents different adjacent view disparities.	60
4.3	Four of the LF atoms in D_{LF} . Each row is a different atom pattern; each column has a different adjacent view disparity.	61
4.4	Three sample atoms that moves patterns from outside of the central patch, which are suitable for describing irregular patterns under occlusions and specularities.	61
4.5	Angular compressive light field imager architecture	62
4.6	Diagram of the two-stage coding algorithm. Note in the perspective-shifted LF dictionary, atoms with same colored have identical central view, but with different perspective-shifting amounts.	67
4.7	Performance evaluation of the two-stage coding method. In the legend, <i>Recon. Err #</i> stands for reconstruction error when using proposed two-stage coding method; <i>Recon. Err</i> stands for reconstruction error using single-stage OMP coding. And similar with the notation <i>CPU Time #</i> , <i>CPU Time</i>	68

LIST OF FIGURES

5.1	Juxtaposed sub-matrices $V_G(v)$ ($v = 1, 2, 3, 4$), with the vignetting pattern for each sub aperture on their respective diagonals.	76
5.2	Sub-aperture images with vignetting. First row are the aperture shapes, second row are the sub-aperture images with vignetting, third row are the calibrated vignetting pattern for each sub-aperture view. The brighter pixels indicates higher transmission rate.	76
5.3	Proposed camera system for light field acquisition.	82
5.4	The hardware implementation for concept proof for the proposed system. Right angle prisms are currently not used in our current stage of implementation, however the concept of sub-aperture scanning and LF reconstruction is validated using this set-up.	84
5.5	Evaluation of aperture scan number and measurement number's effect on final reconstruction quality for SAS. The data is collected on there LF images <i>Buddha</i> , <i>Dragons</i> , and <i>Maria</i> with 5×5 views.	86
5.6	Evaluation of the modulation mask quality with respect to the mask elements' total neighborhood correlation defined in Eqn. (5.5) for the LF images <i>Buddha</i> , <i>Dragons</i> , and <i>Maria</i> repsectively in (a), (b), and (c).	87
5.7	LF reconstruction performance comparison for competing methods. Both reconstruction PSNR and computation time are evaluated. The curves and bars are average of three synthetic LF data <i>Buddha</i> , <i>Maria</i> and <i>Dragons</i>	90
5.8	9 of the reconstructed 5×5 angular views with measurements taken using the proposed LF camera. The NF method is used, 2 measurements are taken. Red dotted lines are added to demonstrate the disparity between different reconstructed views.	93

LIST OF FIGURES

5.9	LF reconstruction details for <i>Buddha</i> . Areas in red rectangles are zoomed-in for better observation. Areas in blue rectangles show perspective differences from different LF views. The NF method is used, 2 measurements are taken with reconstruction PSNR 33.35.	94
5.10	LF reconstruction details for <i>Dragons</i> . Areas in red rectangles are zoomed-in for better observation. Areas in blue rectangles show perspective differences from different LF views. The NF method is used, 2 measurements are taken with reconstruction PSNR 31.84.	94
5.11	Refocusing demonstration on the reconstructed light field based on synthetic data <i>Buddha</i> . 5 sub-aperture scans and only 1 extra measurement is used for the reconstruction. The reconstruction PSNR is 33.64 dB. For better illustration of the focus change, the details in blue and red rectangles in (a), (b) are zoomed in bellow in (c) and (d) respectively.	95
5.12	Refocusing demonstration on the reconstructed light field based on measurements from our proposed LF camera. The NF method is used, and 2 measurements are taken. For better illustration of the focus change, the details in blue and red rectangles in (a), (b) are zoomed in bellow in (c) and (d) respectively.	96
6.1	System illustration of the proposed <i>LFD codec</i>	102
6.2	AGF processing result for the light field <i>Maria</i> . (b) is the zoomed in disparity map calculated based on EPI, low confidence values are removed. (c) is the AGF output disparity map. (d) is the ground truth disparity map imported from [1].	108

LIST OF FIGURES

6.3	Comparison of LF reconstruction PSNR for different transform models with different percentages of non-zero coefficients. The horizontal axis represents the percentage of non-zero coefficients, and vertical axis represents LF reconstruction PSNR. The three methods in comparison are: 2D, and 3D Wavelet Transform, and the proposed LFD compression model.	110
6.4	Comparison of LF reconstruction PSNR with <i>H.264 codec</i> at different compression rates for different bench mark light field images. The horizontal axis represents bit-rate (in bpp), and vertical axis represents reconstruction PSNR.	110
6.5	Comparison of LF disparity reconstruction PSNR between <i>H.264 codec</i> with proposed <i>LFD codec</i> at different bit-rates. The horizontal axis represents bit-rate (in bpp), and vertical axis represents LF disparity reconstruction PSNR.	111
6.6	AGF processing result for the light field <i>Maria</i> and <i>MonasRoom</i> . (a) the input images, (b) the confidence thresholded disparity maps, (c) the AGF processing results.	112
6.7	AGF processing result for the light field <i>Buddha</i> . (a) is input image, (b) is the disparity map calculated based on EPI. (c) is the processing result using GCL proposed in [3]. (d) is our result.	113

List of Tables

3.1	Denoising result for six test images with added zero mean white gaussian noise of different intensities ranging from 5 to 50. The six data listed for each table entry are results from: top row from left to right, (1) single-scale KSVD dictionary (dimension 64×256); (2) single-scale KSVD dictionary (dimension 144×256); (3) 2-scale dictionary trained by KSVD separately each scale; bottom row from left to right, (4) 2-scale dictionary proposed in [2], (5) 2-scale dictionary trained by MD/CCL; (6) 2-scale dictionary trained by MD/CAC. Best performance in terms of PSNR (in dB) are in bold.	50
5.1	Quantitative evaluation of LF reconstruction for synthesized LF data <i>Maria</i> , <i>Buddha</i> [1], and <i>Dragons</i> [4] with different number of measurements. The four entries for each unit are (from top to bottom): (1) coded aperture method [5]; (2) coded sensor-plane method; [4]; (3) sub-aperture scan method using 5 scan images; (4) normalized fluctuation method. Best performance, i.e. highest PSNR (in dB) and shortest time, are put in bold for each unit. Compression ratio is calculated as measurement no./reconstructed angular sub-view no.	91

LIST OF TABLES

5.2	Quantitative evaluation of LF disparity reconstruction PSNR (in dB) for synthesized LF data <i>Maria</i> , <i>Buddha</i> [1], and <i>Dragons</i> [4] with dif- ferent number of measurements.	92
-----	---	----

List of Abbreviations

AGF	Adaptive Guided Filtering
CMP	Cross-Scale Matching Pursuit
CS	Compressed Sensing
EPI	Epi-Polar Image
GF	Guided Filter
K-SVD	K-means clustering Singular Value Decomposition
LF	Light Field
LFD	Light Field Dictionary
MD	Multi-scale Dictionary
MD/CCL	Multi-scale Dictionary Cross-scale Cooperative Learning
MD/CAC	Multi-scale Dictionary Cross-scale Atom Clustering
MRA	Multi-resolution Analysis
NF	Normalized Fluctuation
OMP	Orthogonal Matching Pursuit
SAS	Sub-Aperture Scan
SVD	Singular Value Decomposition
WTA	Winner-Take-All

List of Notations

Summary of Key notations, in order of their appearances in each Chapter.

Chapter 2

A	The linear system matrix
b	Input signal/measurement
x	System solution
$\phi_{j,k}$	The scaling function for scale j , and shifting location k
$\psi_{j,k}$	The wavelet function for scale j , and shifting location k
V_j	Space spanned by orthogonal basis $\{\phi_{j,k}(2^j x - k), k \in Z\}$
W_j	Space spanned by orthogonal basis $\{\psi_{j,k}(2^j x - k), k \in Z\}$

Chapter 3

D	Redundant dictionary
K	Total number of dictionary atoms
x	Input signal
α	Sparse coding coefficient vector

N_l	Spatial dimension of the atom ($\sqrt{N_l} \times \sqrt{N_l}$) of scale l
L	Total number of scales
D_l	Dictionary for the scale l
d_k^l	k_{th} dictionary atom in scale l
x^l	Input signal of scale l
$\hat{\varepsilon}_i^l$	Average pixel representation error for the pixel i in scale l
R_p^l	linear operator that bi-linearly interpolates a dictionary atom from scale $\mathbb{R}^{N_p \times 1}$ to $\mathbb{R}^{N_l \times 1}$
λ	Regularization parameter
$\mathbb{S}_k^l$	Set of image patches that have used the dictionary atom d_k^l during sparse coding
Σ_k^l	Number of elements in set $\mathbb{S}_k^l$
δ_k^l	Atom clustering center calculated as: $\delta_k^l = R_p^l d_{k_c}^p$
M_k^l	Appended error matrix calculated as $M_k^l = [E_k^l, \lambda * \delta_k^l]$

Chapter 4

D^c	Global dictionary trained with natural images
N^v	Spatial dimension of the light field atoms ($\sqrt{N^v} \times \sqrt{N^v}$)
N^c	Dimension of the global image dictionary atoms ($\sqrt{N^c} \times \sqrt{N^c}$)
dp	Unit disparity value between adjacent views
V	Total number of light field views
K^c	Number of atoms for the global image dictionary
S	Length of disparity vector
x	Input light field measurement

$[x]_{\#}$	Hash coded signal
Φ	Modulation mask for the light field views
D_{LF}	Light Field Dictionary
$[D_{LF}]_{\#}$	Hash coded light field dictionary
$[D_{LF}^m]_{\#}$	Median hash codes for all atoms in $[D_{LF}]_{\#}$ that share the same central atom
$D_{LF}^{\subset}$	Subset of D_{LF} whose central atom hash codes have been picked in Stage 1 of the Two-stage coding
α	Light field coding coefficient vector
ε	Light field coding error target
N_P	Number of permutations performed and the code length for the WTA hash code
N_W	Window size for the max operation during WTA hash coding

Chapter 5

F	The separation between the exit pupil of the lens and the camera sensor
$L_f(x, y, s, t)$	A continuous 4D light field
$L_F(x, y, v)$	A discrete light field
V	Total number of light field views
$M_V(v)$	The modulation mask value for the view v
$V_G(v)$	Vignetting matrix for the view v
$\Phi(v)$	The projection matrix that combines the physical effects of mask modulation and vignetting
$f[A, \vec{b}]$	An operator that translates a matrix A to the direction specified by $\vec{b}$
Rn	Total neighborhood correlation for the modulation mask
N_R	Size of the pool of random generated mask patterns

N_P	Size of the pool of candidate mask patterns
I_A^v	Sub-aperture image for the light field view v
$\tilde{I}_A^v$	Vignetting rectified sub-aperture image for the light field view v
P_X, P_Y	Horizontal and vertical optical flow
P	The perspective flow
$\Theta(\cdot)$	Discretizing and Mapping operator
I_M^k	Image taken with aperture modulation mask M_V^k
$\tilde{I}_M^k$	Vignetting normalized image taken with aperture modulation mask M_V^k
D_{LF}^S	Dictionary segment with atoms in D_{LF} that share the same disparity/ perspective shifting amount $dp(S)$

Chapter 6

D_{LF}^S	Dictionary segment with atoms in D_{LF} that share the same disparity/perspective shifting amount
E_X, E_Y	Epi-Polar image of the light field with fixed spatial axis x and y, respectively
S_X, S_Y	Disparity voting score for the light field with fixed spatial axis x and y, respectively
C_e	Edge confidence for disparity estimation
P	Estimated disparity map from voting
$\Phi(P)$	Quantized disparity map P
a_k, b_k	Affine transform coefficients for the patch centered at pixel k
w_k	Patch window of size for the AGF
I_i	Guidance RGB image
q_i	Output disparity value as a patch
β_i	Thresholded edge confidence for pixel i
ε	Regularizer that prevents a_k from getting too large

wn_i	Number of patches that include the pixel i
w_i	AGF filter patch size

Chapter 1

Introduction

Structure exists in all forms of visual data in the nature. In this thesis, we focus on modeling these data structures using sparse representation techniques over a redundant dictionary, which has been proven as an efficient tool in numerous visual signal processing applications. We propose several types of redundant dictionaries with specially designed structures that adapts to the unique application scenarios.

For a structured dictionary, problems naturally arise regarding how to learn the atoms in a way that each component of the structure (components could be of different scales, segments, or view points etc.) could cooperate to produce a combined optimal output. Other problems also arise such as on how to handle the coding complexity brought about either by the large size or by the complicated dictionary structure. And these problems are certainly not trivial.

1.1 Motivation and Objective

In this thesis, we focus on the problem of designing complicated dictionary structures for sparse representations of different types of visual data. Based on those specially

1.1. MOTIVATION AND OBJECTIVE

designed structures, different dictionary training methodologies, and sparse coding strategies will be investigated in different application scenarios such as multi-scale signal denoising, light field compressed sensing, light field data compression and post-processing etc. A brief introduction to these application scenarios will be given in the following sections. The theoretical backgrounds for sparse signal representation and light field imaging will be given in Chapter 2.

1.1.1 Multi-scale Dictionary for Image Denoising

Images of natural scenes tend to produce repeated patterns. These primary patterns may appear at different locations, with various translation, rotation, and scaling. For sparse visual signal representation, the sparsity across the scales is a promising yet under investigated direction.

Research can be found that focus on learning dictionaries for the wavelet coefficients [6] [7] [8], which is a multi-resolution analysis based framework. The image pyramid model is employed in the works of [9] and [8] for a multi-scale representation. In these methods, the “scaling” is realized by a separate transform/decomposition operator, the dictionaries themselves remain “single scale”; the efficiency of the combination of a separate transform operator with sparse coding is questionable. In another work [2] [10], a simple quadtree based multi-scale dictionary is proposed, however the cross-scale dependencies are not investigated due to its structural limitations.

We aim at designing a more direct and more efficient multi-scale sparse representation framework, separate from the wavelet decomposition, which was involved in most previous available multi-scale schemes. The designed multi-scale dictionary should be multi-resolution in itself, with dictionary atoms learned for different signal scales. Cross-scale sparsity and similarity need to be carefully investigated.

1.1.2 Compressed Light Field Sensing and Reconstruction Based on Perspective-Shifted Dictionaries

The theoretical principle of light field imaging has been introduced over a century ago [11, 12], however it is only with the recent technological development, it has become practical and popular. One technical challenge to record the light field is the extremely large amount of information the light field contains, the other is in the hardware design that can resolve the light ray integration.

Most light field acquisition methods share the camera pixel resources between spatial and angular dimensions, such as the micro-lens array based methods [13] [14], and the kaleidocamera [15]. Spatial modulation techniques are also applied in light field acquisition, such as sensor plane modulation [4], and aperture plane modulation [5, 16–19].

The hope of capturing the light field with both high spatial and angular resolution gives rise to a lot of new researches on applying compressed sensing techniques to light field imaging [4] [18] [5] [20]. However, the dictionary learning as well as the data reconstruction is usually extremely computationally expensive (for [4], it takes more than 10 hours on a 24-core processor to learn the dictionary, and takes 18 hours for the reconstruction of light field with spatial resolution 480×270 and angular resolution 5×5 on a 8-core workstation).

In this thesis, we aim at designing a compressed sensing framework for light field acquisition that can produce a high resolution output, with a computationally efficient algorithm to reconstruct the light field.

1.1.3 Light Field Data Compression and Post-Processing

The high dimensional LF (Light field) data is usually very large, even for ones that is discrete in the angular dimension. Light field compression therefore is an important issue in its storage as well as transmission.

For compression, the LF is usually treated as a 3D structured data, and video compression methods could be applied by treating the angular indices as time axis in a video [21] [22] [23] [24]. The strong correlations within a LF lies not only among the LF views, but also between the compensation elements between adjacent LF views, i.e., there exists a linear relationship between compensation elements among the LF views. This suggests that higher compression ratio and performances could be achieved if the interior structure among all the views could be exploited. This inspired us to use a dictionary synthesized with such linear interior structure. A higher compression ratio is expected, and we will evaluate its efficiency in light field data reconstruction quality as well.

The challenge of depth inference from light field is similar to that of traditional stereo vision, however instead of a pair of input images, the light field extends the disparity space to a continuous or multiple discrete ones. However the “aperture” problem that plagues many other computer vision problems persists, which means the directly inferred disparity map will be noisy with a lot of false estimations especially in texture-less regions. Therefore, filtering and post-processing is always need for an optimized output [13,25,26]. Notably, Wanner et al. [3] used a energy minimization approach combined with state-of-the-art convex optimization algorithms; they produced high quality depth maps, however at the cost of very high computational complexity. A “fine-to-coarse” approach was implemented in Kim’s recent work [27], where disparity estimations from lower resolution EPIs were used to interpolate the

1.2. CONTRIBUTIONS OF THE THESIS

missing data in higher resolution. The algorithm is slightly time consuming but more efficient than [3].

The concept of “Guided Filtering” was first introduced by He et al. in [28], where a guidance image was used to filter another image. Details are preserved where the corresponding area in the guided image is with edges or high variation, and smoothed otherwise. This property of the filter could be extremely useful in filtering noisy LF disparity estimations, since we aim to sharpen the disparity edges and align them to the image edges that is used as guide, and to fill texture-less regions with surrounding disparity values. Issues such as how to place more weight on the high estimation confidence pixels and reject those with suspicion during the filtering, and how to properly choose adaptive filter size need to be carefully investigated to fully realize the potential of the Guided Filter.

1.2 Contributions of the Thesis

Structure exists in all forms of vision data in the nature. In this thesis, we focus on modeling these data structures using sparse representation techniques over a redundant dictionary, which has been proven as an efficient tool in numerous visual signal processing applications.

For sparse signal representation, the sparsity across the scales is a promising yet under investigated direction. In this thesis, we design a multi-scale sparse representation scheme to explore such potential. A multi-scale dictionary (MD) structure is designed. A Cross-scale Matching Pursuit (CMP) algorithm is proposed for multi-scale sparse coding. Two dictionary learning methods: Cross-scale Cooperative Learning MD/CCL), and Cross-scale Atom Clustering (MD/CAC) are proposed with each focusing on one of the two important attributes of an efficient multi-scale dictionary:

1.2. CONTRIBUTIONS OF THE THESIS

the similarity, and uniqueness of corresponding atoms in different scales. We analyze and compare their different advantages in the application of image denoising under different noise levels, where both methods produce state-of-the-art denoising results.

The LF contains large volumes of visual information that can provide a comprehensive understanding of the 3D environment of the scene. In this thesis, a light field dictionary (LFD) based on perspective-shifting is proposed for sparse representation of the highly correlated light field. A two-stage coding algorithm is proposed which uses the Winner-Take-All (WTA) hashing strategy to narrow down the search range for light field sparse coding. The algorithm proves to be able to increase the coding efficiency by almost three times and keeping the reconstruction quality almost the same with the original OMP coding.

A compressed sensing framework is proposed for the sampling and reconstruction of a high resolution light field based on a coded aperture camera. Two separate methods, i.e., Sub-Aperture Scan (SAS) and Normalized Fluctuation (NF) are proposed to acquire/calculate the scene disparity, which will be used during the light field reconstruction with the proposed disparity-aware dictionary. A hardware implementation of the proposed light field acquisition/reconstruction scheme is carried out. Both quantitative and qualitative evaluation shows the proposed methods produce state-of-the-art performance in both reconstruction quality and computation efficiency.

Then, a light field compression framework based on the LFD is proposed for efficient storage and transmission of the bulky LF data. A highly efficient adaptive guided filtering algorithm is also proposed for the LF disparity/depth map post-processing. Both Quantitative and qualitative simulations validate the efficiency of the proposed methods.

1.3 Organizations of the Thesis

The rest of the thesis is organized as follows.

Chapter 2 introduces the background for the related research in sparse representation and light field imaging.

In Chapter 3, a multi-scale sparse representation scheme is introduced. A multi-scale dictionary (MD) structure is designed. A Cross-scale Matching Pursuit (CMP) algorithm is proposed for multi-scale sparse coding. Two dictionary learning methods: Cross-scale Cooperative Learning MD/CCL), and Cross-scale Atom Clustering (MD/CAC) are proposed with each focusing on one of the two important attributes of an efficient multi-scale dictionary: the similarity, and uniqueness of corresponding atoms in different scales.

In Chapter 4, a light field dictionary (LFD) based perspective shifting of central view atom is proposed for sparse representation of the high dimensional light field data. The extreme large dimension of the LFD poses challenge for LF reconstruction. A two-stage coding algorithm is proposed to speed up the coding process.

In Chapter 5, a compressed sensing framework is proposed for the sampling and reconstruction of a high resolution light field based on a coded aperture camera. Based on the LF dictionary proposed in Chapter 4, two separate methods, i.e., Sub-Aperture Scan (SAS) and Normalized Fluctuation (NF) are proposed to acquire/calculate the scene disparity for the light field reconstruction. A hardware implementation of the proposed light field acquisition/reconstruction scheme is carried out.

In Chapter 6, a compression framework is proposed. A LF dictionary via perspective-shifting is employed, and a sparse coding procedure based on LF disparity voting is proposed. As an important LF post-processing technique, an adaptive guided filtering (AGF) algorithm is proposed to calculate an optimized output disparity/depth map.

1.3. ORGANIZATIONS OF THE THESIS

The thesis will be concluded in Chapter 7. Promising research directions are recommended later in this chapter.

Chapter 2

Background and Review

2.1 Introduction

In this Chapter, theoretical background and recent research related to the topics of this thesis will be reviewed.

Section 2.2 introduces the basic theories under sparse representation and compressed sensing. Section 2.2.1 reviews the Wavelet transform, and how it is designed to sparsify signal representation. This naturally leads to the topic in Section 2.2.2 on how to promote sparse solution for a general linear system. The uniqueness of a sparse solution for a given system is discussed in Section 2.2.3. Finally, we proceed to the topic of sparsity pursuit algorithms. Greedy pursuit algorithms and convex relaxation algorithms will be discussed in Section 2.2.4 and 2.2.4, respectively.

Section 2.3 introduces the theories of light field imaging and its post-processing techniques. Section 2.3.1 describes the 4D representation for the light field which will be used throughout the rest of this thesis. Section 2.3.2 introduces the basic structure of a micro-lens array based light field camera. Section 2.3.3 reviews the

theory of Epi-Polar images and related light field data post-processing techniques.

2.2 Sparse Representation and Compressed Sensing

2.2.1 The Wavelet Transform

It is a well observed fact that visual signals (in forms of image, video, etc.) reside in a much lower dimensional ambience in the signal space, and they can be well approximated by a linear combination of few basis, i.e., signal can be sparsely represented in the transform domain. Numerous work has been done in the past few decades trying to find a good transformation that produces sparse representation.

In Mallat's multi-resolution analysis framework (MRA) [29], a scaling function is first chosen as ϕ , such that the family $\{\phi_{j,k}(2^j x - k), k \in Z\}$ forms an orthonormal basis for the space V_j . It's theoretically proven that if the function $f(x) \in V_j$, then its scaled version $f(2x) \in V_{j+1}$ is spanned by $\{\phi_{j+1,k}(x) = \sqrt{2}\phi_{j,k}(2^{j+1}x + k), k \in Z\}$. The spaces V_j and V_{j+1} are "similar", and $V_j \subset V_{j+1}$ [30].

Based on ϕ , a wavelet function ψ is then calculated as:

$$\psi(x) = \sqrt{2} \sum_k (-1)^k h(-k+1) \phi(2x - k), \quad (2.1)$$

where $h(k) = \langle \phi(x), \sqrt{2}\phi(2x - k) \rangle$. It is then proven the space W_j spanned by the wavelet is orthogonal complement to V_j inside V_{j+1} . When the scaled and shifted versions of the wavelets $\{\phi_{j,k}, \psi_{j,k}\}$ integrate with the signal, it transforms the signal into another domain that has sparse coefficients. The design of $\{\phi, \psi\}$ is critical, it determines the parsimony of transformation coefficients. $\{\phi, \psi\}$ should be a globally optimal pattern in the subspace of the signal, in order to sparsify the coefficients, and keep the representation to a desired precision.

Ingrid Daubechies made her efforts into designing a hierarchy of $\{\phi, \psi\}$ that has a compact support and with a prescribed number of continuous derivative [31]. Some scale-adaptive image denoising algorithms based on such properties produce very good performances such as those reported in [32], [33], [34]. Extension methods such as ridgelet [35], curvlet [36], bandlet [37], and oriented wavelet [38] were later proposed to further adapt the wavelet basis $\{\phi, \psi\}$ to different image geometrical attributes.

2.2.2 The promotion of solution sparsity

Consider a matrix $A \in \mathbb{R}^{n \times m}$, and a system equation of $Ax = b$. If $n \leq m$, this system is an under-determined system, which means it has more unknowns than equations given as system conditions. Given that b is in the span of the column space of A , the system $Ax = b$ will have infinite number of solutions.

In order to get one optimum solution for $Ax = b$, regularization is needed. The most well known choice of the regularizer the squared Euclidean norm $\|x\|_2^2$:

$$\min_x \|x\|_2^2 \text{ subject to } Ax = b. \quad (2.2)$$

The use of a L_2 -norm based regularizer has been popular due to its simplicity since a closed-form unique solution is available:

$$\hat{x} = A^T(AA^T)^{-1}b. \quad (2.3)$$

The Wiener filter is such an example of L_2 -norm based method.

In order to promote the sparsity of the solution x , we first demonstrate the behavior of different L_p -norms at unit vector length. For the case when x is a two dimensional vector, the curves are drawn in Fig. 2.1. As shown in the figure, when p

Figure 2.1: demonstration of different norms on 2D vectors.

goes from 2 to 0.5, the unit norm ball becomes pointier along each axis. An inference would be appropriate that when p goes to 0, the unit norm ball becomes the axis itself. This indeed conforms with the definition of a vector zero-norm, which is the number of non-zero elements in the vector.

Then we look at the following problem:

$$\min \|x\|_p^p, \text{ subject to } Ax = b. \quad (2.4)$$

The constraint function $Ax = b$ forms a plane where vectors on the plane form a feasible set of solutions for the problem in Eqn. 2.4. A unique solution will be determined as the vector with smallest L_p norm. This process could be imagined as expanding the L_p -norm ball from the origin, until it first touches the feasible set plane. Figure 2.2 shows such situation when x is a three dimensional vector. As can be seen from Fig. 2.2, the solution produced by minimum L_1 norm in (b) tend

Figure 2.2: demonstration of expanding L_p -norm ball to intersect with the feasible set plane $Ax = y$. (a) and (b) are for L_1 -norm and L_2 -norm, respectively. The cross sign shows the point of intersection.

to happen at one of the axis (which means coefficients for all other axis are zero, therefore producing a sparse solution). However for L_2 -norm in (a), such situation will not occur.

2.2.3 Solution Uniqueness Analysis

Given an under-determined system $Ax = b$ ($A \in \mathbb{R}^{n \times m}$, $n \leq m$), one critical concern is, whether a unique sparse solution exists. To answer this question, we need to use the notion of matrix *Spark*. But first of all, recall the definition of matrix *Rank*:

Definition 2.1. The largest number of linearly-independent columns of a matrix A is called its *Rank*.

As comparison, matrix *Spark* is defined as [39]:

Definition 2.2. The smallest number of linearly-dependent columns of a matrix A is called its *Spark*.

2.2. SPARSE REPRESENTATION AND COMPRESSED SENSING

The definition of *Spark* is crucial in the proof of the uniqueness of the sparse solution. Let's first visit one fundamental theorem for the *Spark* of a matrix:

Theorem 2.3. *If vector x is in the null space of A : $Ax = 0$, then:*

$$\|x\|_0 \geq \text{Spark}(A). \quad (2.5)$$

Proof. For any vectors in the null-space of A , i.e., $Ax = 0$, the coefficient elements that correspond to linearly-independent columns must be zero; for the columns that are linearly-dependent, to make the count of non-zero elements to be as small as possible, and also to make sure that x not being a zero vector, at least a number of $N_0 = \text{Spark}$ non-zero coefficient elements are required to configure those dependent columns to sum to zero. $\square$

Theorem 2.3 will be used in determining whether a sparse solution x is unique:

Theorem 2.4. *If a system defined by the equation: $Ax = y$ has a solution such that $\|\hat{x}\|_0 \leq \text{Spark}(A)/2$, then $\hat{x}$ is the sparsest solution among all its feasible counterparts.*

Proof. Suppose there was another solution z to the system such that $Az = y$. The subtraction of the two solutions will be in the null space of A : $A(x-z) = 0$. According to Theorem 2.3,

$$\|x\|_0 + \|z\|_0 \geq \|x - z\|_0 \geq \text{Spark}(A). \quad (2.6)$$

If a certain solution satisfies $\|\hat{x}\|_0 \leq \text{Spark}(A)/2$, then according to Eqn. 2.6, all other possible solutions must satisfy $\|z\|_0 \geq \text{Spark}(A)/2$. Therefore, $\hat{x}$ is the sparsest solution. $\square$

2.2.4 Sparsity Pursuit Algorithms

The sparse coding problem can be expressed as following:

$$\hat{x} = \arg \min_x \|b - Ax\|_2^2 \text{ subject to } \|x\|_0^0 \leq L, \quad (2.7)$$

where L is the maximum number of dictionary atoms allowed to be used. In another expression, his problem can also be formulated as:

$$\min_x \|x\|_0^0 \text{ subject to } \|b - Ax\|_2^2 \leq \varepsilon, \quad (2.8)$$

where ε is the coding error tolerance.

The L_0 -norm problem in Eqn. (2.7) and Eqn. (2.7) are intrinsically NP hard: all possible combinations of dictionary atoms in D need to be tested in order to find an optimum solution. Two types of solutions are available for this problem: greedy pursuit, and convex relaxation, and they will be reviewed in this section.

Greedy Pursuit: Orthogonal Matching Pursuit

The core idea of greedy pursuit is to compromise the globally optimal solution, which requires exhaustive combinatorial listing, for a series of locally optimal supports. The most famous greedy pursuit algorithm is Orthogonal Matching Pursuit (OMP). Algorithm 2.1 is a detailed explanation of the OMP.

In each iteration, the residual is projected to the subplane spanned by columns of A , and the largest projection is chosen to be the new support and added to the current support of the solution. As after each update, the residual $r^{(k)}$ is orthogonal to the chosen supports, and hence the name Orthogonal Matching Pursuit.

2.3. LIGHT FIELD ACQUISITION AND DATA PROCESSING

Algorithm 2.1 The OMP sparse coding algorithm

Input parameter: target error ε_0 , and maximum support number L_0

Initialization: $k = 1$; $x^{(1)} = 0$; $r^{(1)} = b - Ax^{(1)}$, $Support\{x\} = \phi$

while Residual $r^{(k)} > \varepsilon_0$ and support number $k \leq L_0$ **do**

Sweep through all a_j (a_j is column j of A), find j_0 that produces the smallest error:

$$\varepsilon(j_0) = \min_j \|a_j z_j - r^{(k-1)}\|_2^2, \quad z_j = a_j^T r^{(k-1)} / \|a_j\|_2^2 \quad (2.9)$$

Update support $S^{(k)} = S^{(k-1)} \cup a_{j_0}$

Update provisional solution: compute x^k , the minimizer of $\|Ax - b\|_2^2$ subject to $Support\{x\} = S^{(k)}$

Update residual: $r^{(k)} = b - Ax^{(k)}$

$k = k + 1$

end while

Convex relaxation: Basis-Pursuit

As discussed in Section 2.2.2 L_1 -norm has the potential for promoting sparse solutions [40]. Therefore another way to solve the sparse coding problem is to relax L_0 -norm in Eqn. (2.7) and Eqn. (2.8) to L_1 -norm. Since L_1 -norm is convex and piece-wise continuous. As such, it can be solved using modern interior-point methods [41], simplex methods [42], homotopy methods [43], or other techniques that has been developed to sophistication over the past few decades.

2.3 Light Field Acquisition and Data Processing

2.3.1 The Light Field Representation

The light field (LF) is a function that describes the intensities of light rays in all possible propagation directions. As shown in Fig. 2.4(a), light rays both from the world plane and the main lens plane propagate in all directions, the spatial property of each ray can be described with a 5D plenoptic function $L(x, y, z, \theta, \phi)$ (x , y , and z

Figure 2.3: demonstration of light field representation as a 4D data structure.

for the light origin location, θ and ϕ for the propagation direction), and the magnitude of each ray is given by the radiance. The light field, which is the ensemble of all light rays in the optical system, contains rich information with respect to each scene point's real world spatial location, which could be used for the calculation of the scene depth, and post refocusing manipulations, etc.

Multiple choices are available for the representation of light field data. In this thesis, we use the 4D (two-plane) method, where the light field data are organized as a collection of pinhole views from different aperture (main lens) positions. Illustrated in Fig. 2.3, we relate to the 2D aperture plane as $\Pi(s, t)$, and the 2D image plane within each view as $\Omega(x, y)$. Each light ray can be expressed as $L(x, y, s, t)$.

For conventional cameras, converging lenses are used to create a 2D projection of the high dimensional LF data $L(x, y, s, t)$, as shown in Fig. 2.4(a). Classical radiometry shows that the irradiance from the aperture of the main lens onto a point on the sensor is equal to the following weighted integral of the radiance coming through

Figure 2.4: Demonstration of the image synthesis process for a (a) conventional camera, and (b) micro-lens array based light field camera.

the lens:

$$I(x, y) = \frac{1}{F^2} \iint L_f(x, y, s, t) \cos^4 \theta \, ds \, dt, \quad (2.10)$$

where F is the separation between the exit pupil of the lens and the camera sensor, L_f is a continuous light field and I is camera sensor image. θ is the angle between the light ray (x, y, s, t) and the sensor plane normal. $\cos^4 \theta$ is the falloff factor that represents the reduced effect of rays striking the sensor from oblique directions [14].

Figure 2.5: A sensor image directly taken from a micro-lens array based light field camera. (b) shows the raw sensor image, (a) shows a reconstructed image based on the raw data from (b). Photo courtesy of Lytro.

2.3.2 Structure of A Microlens-Array Based Light Field Camera

A typical light field camera structure is illustrated in Fig. 2.4(b), where a micro-lens array is inserted into a conventional camera [13, 14], and placed at the original camera sensor plane. The camera sensor is slightly moved back a small distance. The micro-lens diverges the focused light shaft onto the camera sensor with respect to their originating point from the main lens plane. The Sensor records each light ray's spatial information on both the main lens and the micro-lens array plane [44].

Fig. 2.5 shows a sensor image directly taken from the Lytro camera, which is a state-of-the-art micro-lens array based light field camera. As can be seen from Fig. 2.5(b), each bright circle is the image under a micro-lens. All pixels from the same position inside each circle will form the image (a), which is one of the angular views of the light field.

Figure 2.6: (a) Illustration of the Epi-polar image extraction in a micro-lens array based camera.

Figure 2.7: Epi-polar Image from the synthesis light field image “Buddha” [1].

2.3.3 The Light Field Data and Epi-polar Plane Image

For a light field data represented in 4D format $L(x, y, s, t)$, When we fix a vertical line in the camera main lens (aperture) plane by setting $t = t'$, and in the micro-lens array plane by setting $y = y'$, the light ray will be restricted to a subspace $\{E(x, s) | E(x, s) = L(x, y', s, t')\}$, the resulting image $E(x, s)$ is called the Epi-polar Plane Image (EPI). See the demonstration in Fig. 2.6.

Fig. 2.7 shows an example of EPI from the synthesis light field image “Buddha” [1]. EPI reveals the linear spatial disparity of a scene point among different viewing directions. By detecting the slope of each tilting lines on EPI, a disparity map, and further, scene depth map can be calculated [3, 27].

2.4 Conclusion

In this Chapter, theoretical background and recent research on topics related to this thesis has been reviewed. The basic theories under sparse representation and compressed sensing have been introduced. Wavelet transform have been reviewed first, and then, the topic on how to promote a sparse solution for a general linear system, as well as the uniqueness of the solution have been discussed. Sparsity pursuit algorithms are also properly reviewed.

Subsequently, the theories of light field imaging and its post-processing techniques are reviewed. The 4D representation and the basic structure of a micro-lens array based light field camera have been introduced. The theory of Epi-Polar images and related light field data post-processing techniques have been introduced.

The indepth discussion for a multi-scale sparse representation scheme for visual signal processing will be covered in the next chapter.

Chapter 3

Multi-scale Dictionary For Visual Signal Processing

3.1 Introduction

Images of natural scenes tend to produce repeated patterns. These primary patterns may appear at different locations, with various translation, rotation, and scaling. Our daily visual perception experience tells us: pattern similarity of a natural scene resides not only within a single scale, but across multiple ones. An obvious example is shown in Fig. 3.1(a), where seven matryoshka dolls are juxtaposed in two rows. Most of the details on the dolls are identical and vary only in scale. Other examples, such as the spiral steps in 3.1(b), and the intricate layered carvings on the Gothic style arch in Fig. 3.1(c), gradually diminish in a perspective view. Less obvious but still discernible is the vein texture on the wings of a monarch butterfly in Fig. 3.1(d). All these images are good examples of primary patterns repeating themselves across scales in natural imagery.

Figure 3.1: Four natural images with “multi-scale” patterns.

It is a well observed fact that visual signals reside in a much lower dimensional ambience in signal space. The distribution tends to cluster around multiple local centroids. This observation gives rise to the thrive of research on example-based dictionary learning for sparse image representation, which has received enormous success in computer vision and pattern recognition applications in recent years. The key characteristic of an efficient dictionary is its atoms’ ability to generalize the distribution centroids of possible signals. Essentially, if all the signal centroids with high frequencies of occurrence are known and used as dictionary atoms, the dictionary can be considered as a good one.

Dictionary learning is most frequently expressed as the following optimization problem [45]:

$$\min_{\alpha} \|\alpha\|_0 \text{ subject to } \|x - D\alpha\|_2 \leq \epsilon, \quad (3.1)$$

3.1. INTRODUCTION

where $x \in \mathbb{R}^{N \times 1}$ is the input signal, which is an image patch reshaped into a vector. $D \in \mathbb{R}^{N \times K}$ is the dictionary with K atoms. α is the coding coefficient vector for x , the non-zero element of which is limited to a small number to ensure that the representation is sparse. Numerous work, such as convex relaxation [46], greedy iterative update methods etc [47], has been done to solve the problem in Eqn.(3.1). With the signal x and the dictionary D stretching across multiple scales, the aim of this Chapter is to investigate the potential of cross-scale atom similarity based on the framework of Eqn. (3.1).

3.1.1 Related Work

The idea of “cross-scale sparsity” is not new. In Mallat’s multi-resolution analysis (MRA) [29] framework, a scaling and wavelet function pair $\{\phi, \psi\}$ is chosen, and then the scaled and shifted versions of ψ are integrated with the signal to transform the signal into domains of different resolutions. The parsimony of the transformation coefficients relies on the design of $\{\phi, \psi\}$, the shape of which should be a global optimal pattern in the subspace of the signal across the scales. The wavelet decomposition is a celebrated achievement in this direction [31]. Some scale-adaptive image denoising algorithms based on the wavelet structure produce very good performances such as reported in [33], [34], [48], [32].

With the recent success of example-based dictionary learning for sparse signal representation [49], and its overwhelming comparative advantage over the pre-constructed wavelet methods in various applications, such as visual signal compression [50] [51], denoising [52, 53], interpolation [54] [55] and visual recognition [56–59] etc, the idea of adapting the concept of “scaling” from the wavelet to the dictionary learning based methods looks promising.

3.1. INTRODUCTION

One line of work focuses on learning dictionaries for the wavelet coefficients directly. Ophir et al. [6] proposed to train sub-dictionaries at different wavelet bands, which managed to squeeze out some of the redundancy left by the wavelet decomposition, specifically the spatial correlation between wavelet coefficients in the same band or between bands, and is able to produce a much sparser representation than wavelet and single scale K-SVD. Yan et al. [7] further proposed to utilize clustering on data samples for the training of many sub-dictionaries at each wavelet decomposition level to improve coding efficiency and reduce artifacts. Another similar work was done by Skretting et al. [8], where dictionaries are trained in both the image and wavelet domains for the application of image compression.

Another line of work took advantage of the image pyramid model. In the recent work of Hughes et al. [9], after decomposing an input image into a pyramid of distinct frequency bands, sparse coding and dictionary learning are performed on individual levels of the resulting pyramid. In their model, a set of dictionary atoms are shared and learned across all scales. A statistical model that allows for efficient inference of parameters is proposed. Another work done by Liu et al. in [60] investigated the idea of dictionary refinement from lower to higher scales, which led to a more globally optimal dictionary. Although the final output dictionary is still single scale, we can see the process of pattern generalization across different scales.

All these methods, either using wavelet transform or pyramid decomposition, share one common trait: the “scaling” is brought about by a separate transform/decomposition operator, but the dictionaries themselves still remain “single scale”; the efficiency of the combination of a separate transform operator with sparse coding is questionable.

Another line of work focuses on designing a dictionary itself that is multi-scale. Mairal et al. [2] [10] used a multi-scale quadtree model, where a big image patch

3.1. INTRODUCTION

is decomposed along the tree to sub-patches of smaller scales, and a dictionary is learnt at each scale. The quadtree structure constrains smaller patches from shifting inside larger ones. Different dictionary scales cooperate and combine to depict a signal of interest, which produces a much sparser representation than its single scale counterparts. This model produces convincing results in image denoising. However, being independent from a cross-scale operator (e.g., wavelet transform operator) implies that the proposed model, although being multi-scale, overlooks the cross-scale dependencies of the signal, which makes it sub-optimal in this sense.

In [61] Aharon et al. gave an interesting example of how dictionary atoms in their different forms (a 2D dictionary in this case) could be manipulated into the existing scheme of sparse coding and dictionary learning. Although it is on a different topic, this work inspired the authors in how to manipulate signals from different scales into one processing framework.

3.1.2 Our Contributions

We aim at designing a more direct and efficient multi-scale sparse representation framework, separate from the wavelet decomposition, which was involved in most previous available multi-scale schemes. The designed multi-scale dictionary is multi-resolution in itself, with dictionary atoms learnt for different signal scales. Cross-scale sparsity and similarity is carefully investigated.

Based on this idea, a Cross-scale Matching Pursuit (CMP) algorithm is proposed for multi-scale sparse coding. Two dictionary learning methods: Cross-scale Cooperative Learning (MD/CCL), and Cross-scale Atom Clustering (MD/CAC) are proposed with each focusing on one of the two important attributes of an efficient multi-scale dictionary, i.e., the pattern similarity and uniqueness of corresponding atoms in dif-

3.2. MULTISCALE SPARSE REPRESENTATION MODEL

ferent scales. We analyze and compare their different advantages in the application of image denoising under different noise levels, where both methods produce state-of-the-art denoising results.

The rest of this Chapter is organized as follows. In Section 3.2 the structure of the proposed multi-scale dictionary is explained. The sparse coding algorithm based on such dictionary structure is also introduced in this section. In Section 3.3 two different dictionary learning strategies: MD/CCL and MD/CAC are explained in details. The parameter analysis of the proposed multi-scale scheme, as well as their image denoising performance data are presented in Section 3.4. Section 3.5 concludes the work.

3.2 Multiscale Sparse Representation Model

In this section, we explain the structure of the proposed multi-scale dictionary, and then we proceed to introduce the Cross-scale Matching Pursuit (CMP) sparse coding algorithm.

First, an image I is decomposed into overlapped patches in different scales, i.e., patches of different sizes $\sqrt{N_l} \times \sqrt{N_l}$. Then, patches are reshaped into vectors $x_i^l \in \mathbb{R}^{N_l \times 1}$, where $l = 1, \dots, L$ is the scale index, and L is the total number of scales used. Patches are placed on corresponding points on grids in different scales, and they are indexed from the top-left most pixel i . We do not use the central pixel to index a patch since we do not assume that cross-scale similarity resides at the same indexed location or anywhere in the vicinity.

Let $D_l \in \mathbb{R}^{N_l \times K}$ be the dictionary in scale l (K is the number of dictionary atoms in each scale). The multi-scale dictionary is the collection of $\{D_l | l = 1, \dots, L\}$. For every larger scale in $\{D_l\}$, the dictionary atom dimension increases by a fixed ratio as

3.2. MULTISCALE SPARSE REPRESENTATION MODEL

compared to the previous scale, and the image patch data size increases accordingly.

3.2.1 The Cross-Scale Matching Pursuit

Assume that we have a multi-scale dictionary $\{D_l | l = 1, \dots, L\}$, the Cross-scale Matching Pursuit (CMP) algorithm aims to find the sparse coding coefficients for a signal of interest.

Let α_i^l be the sparse coding coefficient vector for the signal x_i^l over the dictionary D_l . Orthogonal matching pursuit (OMP) [62] is applied to calculate sparse vectors within each single scale. Let N_l be the dimensionality of x_i^l . The average pixel representation error $\hat{\varepsilon}_i^l$ can be calculated as:

$$\hat{\varepsilon}_i^l = \|x_i^l - D_l \alpha_i^l\|_2^2 / N_l, \quad (3.2)$$

For the group of signal vectors with the same index i but from different scales $\{x_i^l | l = 1, \dots, L\}$, one scale l_0 will be chosen if the dictionary atoms in D_{l_0} better represent the signal. The criterion for the selection is defined as:

$$l_0 = \arg \min_l \{\hat{\varepsilon}_i^l f(\|\alpha_i^l\|_0)\}, \quad (3.3)$$

where $\|\alpha_i^l\|_0$ calculates the number of non-zero coefficients in α_i^l , and $f(t)$ is a kernel function positively proportional to parameter t . In our current implementation, $f(t) = \sqrt{t}$.

The criterion in Eqn. (3.3) takes two factors into account during scale selection. The first factor is the representation error: for a fair comparison across scales, the vector error is divided by its dimensionality, which actually measures the average pixel error. The second factor is the representation sparsity: measured in the l_0 norm

3.3. MULTI-SCALE DICTIONARY LEARNING

(number of non-zero coefficients) of the coefficient. The two factors ensure the fidelity and sparsity of a representation, which are the key concerns of the CMP. The steps of the CMP algorithm are generalized in Algorithm 3.1.

Algorithm 3.1 The cross-scale matching pursuit

input parameter: input image I ; multi-scale dictionary $\{D_l | l = 1, \dots, L\}$; target coding error $\hat{\varepsilon}_0$.

-Decompose I into patches of different scales, and form into vectors: $\{x_i^l | i = 1, \dots; l = 1, \dots, L\}$

for each image patch $i = 1, \dots$ **do**

for each dictionary scale $l = 1, \dots, L$ **do**

 -Set initial support for x_i^l as $S_i^l = \emptyset$; set initial residual: $\hat{\varepsilon}_i^l = \|x_i^l\|_2^2/N_l$

while $\hat{\varepsilon}_i^l > \hat{\varepsilon}_0$ **do**

 -Sweep current residual $\hat{\varepsilon}_i^l$ through all atoms in D_l , find the atom $d_{k_0}^l$ that produces the largest projection: $k_0 = \arg \max_k \{d_k^{lT} \hat{\varepsilon}_i^l\}$

 -Update support $S_i^l = S_i^l \cup d_{k_0}^l$.

 -Calculate coefficient s.t. $\text{Supp}\{\alpha_i^l\} = S_i^l$: $\alpha_i^l = \arg \min_{\alpha} \|D_l \alpha - r_i^l\|_2^2$

 -Update current residual according to: $\hat{\varepsilon}_i^l = \|x_i^l - D_l \alpha_i^l\|_2^2/N_l$

 -Select scale $l_0 = \arg \min_l \{\hat{\varepsilon}_i^l f(\|\alpha_i^l\|_0)\}$, set coefficient(s) value $\alpha_i^l = 0$, for all $l \neq l_0$.

3.3 Multi-scale Dictionary Learning

An efficient dictionary is able to maximize the sparsity level, and minimize the reconstruction error in a representation, which is expressed in Eqn. (3.1). In an effort to further promote sparsity in a multi-scale set-up, we propose two dictionary learning methods: Multi-scale Dictionary Cross-scale Cooperative Learning (MD/CCL), and Multi-scale Dictionary Cross-scale Atom Clustering (MD/CAC). Each of these methods focuses on one of the two important attributes of an efficient multi-scale dictionary, i.e., the pattern similarity and uniqueness of corresponding atoms in different scales. Both methods bear distinctive advantages under different circumstances, the

3.3. MULTI-SCALE DICTIONARY LEARNING

details of which will be explained in this section.

3.3.1 Mult-scale Dictionary Cross-scale Cooperative Learning (MD/CCL)

For the first attempt, we try to implement a dictionary structure in which corresponding atoms in all larger scales equal to one atom from the base scale ($l = 1$), up to a linear interpolation:

$$D_l = \{d_k^l \mid d_k^l = R_1^l d_k^1, k = 1, \dots, K\}. \quad (3.4)$$

Where d_k^l is the k th dictionary atom in $D_l \in \mathbb{R}^{N_l \times K}$. The matrix R_p^l ($p = 1$) is a linear operator matrix that bilinearly interpolates a dictionary atom from $d_k^p \in \mathbb{R}^{N_p \times 1}$ to $d_k^l \in \mathbb{R}^{N_l \times 1}$. R_p^l relates the pixel to be interpolated with the location of the original pixels that will contribute to the interpolation, and it stores the bilinear coefficients. The functional diagram of R_p^l has been illustrated in Fig. 3.2. In such a structure, dictionary atoms with the same index k at different scales are related by R_p^l , which enforces cross-scale similarity for every atom across the scales.

Suppose we already have the sparse coding coefficients α_i^l using CMP on D_l ($l = 1, \dots, L$). We propose to update the dictionary D_l to minimize the following target error function:

$$\hat{d}_k^1 = \arg \min_{d_k^1} \sum_{l=1}^L \|x_i^l - \sum_{k=1}^K R_1^l d_k^1 \alpha_i^l(k)\|_2^2 / N_l. \quad (3.5)$$

Obviously, the only variable in Eqn. (3.5) is d_k^1 , which is from the base scale ($l = 1$); all the larger scale atoms will be calculated according to Eqn. (3.4) once d_k^1 is decided.

The Stochastic Gradient Descent (SGD) method [63] is used to update the multi-scale dictionary. The SGD method have been used quite extensively in adaptive filters (least-mean-squares), computerized tomography, and neural network learning algorithms, etc [64]. According to Eqn. (3.5), the partial derivative of the target

Figure 3.2: Demonstration of the scale transform matrix “ R_p^l ”. Bilinear interpolation is used for the transformation.

function over d_k^1 is:

$$\begin{aligned} & \frac{\partial}{\partial d_k^1} \left\{ \sum_{l=1}^L \|x_i^l - \sum_{k=1}^K R_1^l d_k^1 \alpha_i^l(k)\|_2^2 / N_l \right\} \\ &= 2 \times \sum_{l=1}^L \left(\sum_{k=1}^K R_1^l d_k^1 \alpha_i^l(k) - x_i^l \right) \left(\sum_{k=1}^K \alpha_i^l(k) R_1^l \right) / N_l, \end{aligned} \quad (3.6)$$

the dictionary update equation can therefore be written as:

$$\begin{aligned} (d_k^1)_{new} &= (d_k^1)_{old} - \mu \cdot \sum_{l=1}^L \left(\sum_{k=1}^K R_1^l d_k^1 \alpha_i^l(k) - x_i^l \right) \times \\ & \quad \left(\sum_{k=1}^K \alpha_i^l(k) R_1^l \right) / N_l, \end{aligned} \quad (3.7)$$

where μ is the step size parameter, which decides the dictionary update rate after each incoming image patch x_i^l . The SG algorithm converges well given the number of

3.3. MULTI-SCALE DICTIONARY LEARNING

image patches (and of different scales) is very large. The name “cooperative learning” is given to this method based on the fact that signals across all scales contribute their part during the dictionary update of the base scale atoms. Such a cooperative learning manner is very beneficial for its application in signal/image denoising, since noise is usually scale-independent. Thus, jointly using image samples from different scales will average out the noise more effectively.

3.3.2 Multi-scale Dictionary Cross-scale Atom Clustering (MD/CAC)

Unlike the “brute force” that we used in MD/CCL, where all atoms across all scales are forced to be bilinearly similar, we propose a more adaptive learning method that aims at enforcing cross-scale similarities and preserving scale-unique patterns at the same time.

With the coding coefficient vectors $\{\alpha_i^l\}$ calculated from CMP, we introduce a new dictionary structure in which a certain atom d_k^l has a matched similar atom $d_{k_c}^{l_c}$ in another scale. The matching is done according to the following criterion:

$$\{k_c, l_c\} = \arg \min_{j, p} \left\| d_k^l - R_p^l d_j^p \right\|_2^2. \quad (3.8)$$

Eqn. (3.8) basically looks for the most similar atom in the rescaled dictionary $R_p^l D^p$. R_p^l is the same scale transform matrix as in Eqn. (3.4), except that for MD/CCL, R_p^l is always an interpolator ($p = 1, p < l$); however for MD/CAC, R_p^l can be either an interpolator or a decimator depending on the relation between p and l .

The dictionary learning can be thought of as a cross-scale clustering process. Matched atoms will be drawn similar to each other after it is updated according to

3.3. MULTI-SCALE DICTIONARY LEARNING

Eqn. (3.9):

$$\begin{aligned} \hat{d}_k^l = \arg \min_{d_k^l} \sum_{i \in \mathbb{S}_i^l} & \| [x_i^l - \sum_{m \neq k} d_m^l \alpha_i^l(m)] - d_k^l \alpha_i^l(k) \|_2^2 \\ & + \lambda \| d_k^l - R_p^l d_{k_c}^p \|_2^2, \end{aligned} \quad (3.9)$$

where the set $\mathbb{S}_k^l$ is formed by finding patches in scale l that used atom d_k^l during CMP coding. According to the previous section, only the patches that have been selected by CMP to be the best representative scale for the image patch at position i will have a value for α_i^l ; otherwise, $\alpha_i^l = 0$. Therefore, $\mathbb{S}_k^l = \{i | \forall i, \alpha_i^l(k) \neq 0\}$. The first term of Eqn. (3.9) calculates the representation error when atom d_k^l is not used. The second term is the cross-scale atom clustering term. Similar dictionary atoms across scales are drawn closer after each update because of this term. λ is a tuning parameter.

Our algorithm is inspired by K-SVD [52], and comprises two steps at each iteration: CMP coding and dictionary update. The CMP algorithm outputs the coding coefficients α_i^l based on the current multi-scale dictionary $\{D^l | l = 1, \dots, L\}$, and then the cross-scale atom clustering algorithm will update the dictionary based on the current α_i^l . Each atom in each dictionary scale is updated separately. For the dictionary atom d_k^l of scale l with index k , the algorithm finds the set $\mathbb{S}_k^l$ that consists of image patches that have used the dictionary atom d_k^l during the last CMP. For each $x_i^l \in \mathbb{S}_k^l$, the representation error is calculated as:

$$\epsilon_i^l = x_i^l - \sum_{m \neq k} d_m^l \alpha_i^l(m), \quad x_i^l \in \mathbb{S}_k^l, \quad (3.10)$$

where the contribution from the atom d_k^l is excluded. All ϵ_i^l s calculated from $x_i^l \in \mathbb{S}_k^l$ are then juxtaposed as columns to form the matrix $E_k^l \in \mathbb{R}^{N_l \times \Sigma_k^l}$, where Σ_k^l equals

3.3. MULTI-SCALE DICTIONARY LEARNING

the number of elements in set $\mathbb{S}_k^l$.

The algorithm then tries to find the nearest atom $d_{k_c}^p$ in scale p ($p \neq l$) based on Eqn. (3.8), and uses it as the current clustering centroid:

$$\delta_k^l = R_p^l d_{k_c}^p. \quad (3.11)$$

δ_k^l is then appended to the last column of E_k^l to form the matrix $M_k^l \in \mathbb{R}^{N_l \times (\Sigma_k^l + 1)}$:

$$M_k^l = [E_k^l, \lambda * \delta_k^l]. \quad (3.12)$$

Here, λ is a tuning parameter. The algorithm then applies SVD on $M_k^l = U\Delta V^T$. The column in U that corresponds to the largest singular value in Δ will be used to update the dictionary atom d_k^l ; the atom's coding coefficient vector $\alpha_i^l(k)_{i \in \mathbb{S}_k^l}$ will be updated with the first Σ_k^l elements of the column in V that also corresponds to the largest singular value in Δ .

After going through all the dictionary atoms in scale l , the algorithm goes on to iterate through atoms in all other scales, with difference only in the scaling operator R_p^l , where it could change from being an bilinear interpolator to a decimator.

The final updated multi-scale dictionary will be used as the new input to the CMP in the next iteration, until the termination condition is reached. Algorithm 3.2 gives a detailed description of the cross-scale atom clustering algorithm.

In Algorithm 3.2, we enforce all the atoms at one scale to find their matched atoms at another scale. This naturally raises the concern: what if some patches at one scale do not have cross-scale similarities. Refer to Fig. 3.3 which illustrates the structure of the error matrix $M_k^l \in \mathbb{R}^{N_l \times (\Sigma_k^l + 1)}$. The first Σ columns (bright) are data representation errors from a certain dictionary atom d_k^l . Each column represents an

3.3. MULTI-SCALE DICTIONARY LEARNING

Algorithm 3.2 Dictionary learning with cross-scale atom clustering.

input parameter: input image Patches X_i ; multiscale dictionary $\{D^l | l = 1, \dots, L\}$.
repeat
 cross-scale sparse coding:
 Apply the CMP algorithm and calculate the coefficients α_i^l for each patch based on the current multi-scale dictionary $\{D^l | l = 1, \dots, L\}$.
 dictionary update:
 for each dictionary scale $l = 1, \dots, L$ **do**
 for each dictionary atom $k = 1, \dots, K$ **do**
 -Find the set of patches that use this atom: $\mathbb{S}_k^l = \{i | \alpha_i^l(k) \neq 0\}$
 -For each patch $x_i^l \in \mathbb{S}_k^l$, Compute the representation error for each patch: $\epsilon_i^l = x_i^l - \sum_{m \neq k} d_m^l \alpha_i^l(m)$
 -Find the nearest interpolated/decimated atom in other scale(s), and calculate the displacement: $\delta_k^l = \min_{k_c} \|d_k^l - R_p^l d_{k_c}^p\|_2^2$
 -Set E_k^l as matrix with columns of $\{\epsilon_i^l\}_{i \in \mathbb{S}_k^l}$. Append δ_k^l with tuning coefficient λ to the last column of E_k^l to form matrix $M_k^l = [E_k^l, \lambda * \delta_k^l]$, $M_k^l \in \mathbb{R}^{N_l \times (\Sigma_k^l + 1)}$.
 -Apply SVD on $M_k^l = U \Delta V^T$, use the column in U and first N_l elements of the column in V that correspond to the largest singular value in Δ to update d_k^l and $\alpha_i^l(k)_{i \in \mathbb{S}_k^l}$, respectively.
 until *iterate J times*;

3.3. MULTI-SCALE DICTIONARY LEARNING

Figure 3.3: Structure of matrix M_k^l . “Bright” columns are fidelity terms, the “dark” column is the cross-scale similarity term.

image patch that has selected d_k^l during CMP coding; the last (dark) column is the rescaled atom (via scaling operator R_p^l) from another dictionary scale ($d_{k_c}^p$) that has been matched with d_k^l . For a certain atom pattern that is unique and popular for a single scale, which means this atom is used frequently for that scale, the error matrix M_k^l will have more “bright” columns; so when M_k^l is later used for SVD, the cross-scale matching column (dark) will have less impact during the update, thus preserving the scale uniqueness. In conclusion, although the algorithm enforces cross-scale matching for every atom, the learning process vary with respect to each atom’s scale-dependent properties.

We comment a bit more about the choice of a bilinear interpolator/decimator for the scaling operator R_p^l . Since dictionary update is an iterative process, a high quality interpolation/decimation [65] [66] is unnecessary, and the bilinear operator is sufficient and efficient for MD/CAC.

3.4 Experiments and Results

We carry out experiments on the Cross-scale Matching Pursuit (CMP), the Cross-scale Cooperative Learning (MD/CCL), and the Cross-scale Atom Clustering (MD/CAC) algorithms, to evaluate the proposed multi-scale sparse coding and dictionary learning scheme.

3.4.1 Dictionary Sparse Coding RMSE

With the multi-scale structure and added sparsity across the signal scales, we expect the proposed multi-scale dictionary to represent images with less error than single-scale dictionaries, provided that coefficient cardinalities (number of non-zero elements allowed for each coefficient vector) are identical. Comparisons have been carried out between several competing methods:

- KSVD: a single-scale dictionary of dimension 64×256 (patch size: 8×8 , atom number: 256) trained with the KSVD algorithm [45];
- MD: a 2-scale Dictionary of dimension $64 \times 128 / 144 \times 128$ (patch sizes: $8 \times 8 / 12 \times 12$, atom numbers: 128 / 128, 256 in total), trained with the KSVD algorithm for each scale separately without any cross-scale interactions;
- MD/CCL: a 2-scale Dictionary of dimension $64 \times 128 / 144 \times 128$ (patch sizes: $8 \times 8 / 12 \times 12$, atom numbers: 128 / 128, 256 in total), trained with the proposed Cross-scale Cooperative Learning method;
- MD/CAC: also 2-scale Dictionary of dimension $64 \times 128 / 144 \times 128$ (patch sizes: $8 \times 8 / 12 \times 12$, atom numbers: 128 / 128, 256 in total), trained with the proposed Cross-scale Atom Clustering method.

Figure 3.4: Coding performance comparison in terms of RMSE between four different dictionaries. Data are the average of five images: *Barbara*, *Peppers*, *Lena*, *Boat* and *House*.

For a fair comparison, the atom numbers of different dictionaries have been set to be the same 256. Four dictionaries are used to code images with different limits of cardinalities. For KSVD, Orthogonal Matching Pursuit [62] is used for coding; for MD, MD/CCL and MD/CAC, CMP is used. The coding Root Mean Square Errors (RMSE) are shown in Fig. 3.4, which are averages of five input images: *Barbara*, *Peppers*, *Lena*, *Boat* and *House*; all images are of dimension 512×512 .

Obviously from Fig. 3.4, the three 2-scale dictionaries all show significant decrease of coding RMSE against the single-scale dictionary (in the scale of 2+), provided that dictionary atom numbers as well as coding cardinalities are equivalent. This experiment validates our expectation that a multi-scale dictionary structure provides a more powerful frame for describing visual signals. Among the three multi-scale counterparts, MD/CAC produces the best coding RMSE when cardinality gets larger.

3.4.2 Visual and Quantitative Evaluation of the Multi-scale Dictionary Learning Outcomes

Visual Illustration of MD/CCL and MD/CAC Learning Outcome

Using a 2-scale DCT dictionary (dimension 64×256 and 144×256) as the initial dictionary, and training patches from the image *Barbara* (overlapping patches of size 8×8 and 12×12), we train the multi-scale dictionaries with the two methods herein proposed: MD/CCL and MD/CAC. Both training processes iterate between CMP coding and dictionary update for 10 iterations.

Fig. 3.5 shows the dictionary learning result for MD/CCL. Dictionary atoms from different scales (shown in Fig. 3.5(a) and (c) respectively) have visually similar patterns. Fig. 3.5(b) zooms in some atoms from the two scales that are related by the interpolation operator R_p^l . As can be seen, the larger scale (12×12 , in Fig. 3.5(b) bottom row) can depict the patterns in better detail than the smaller scale (8×8 , in Fig. 3.5(b) top row); while smaller atoms can cooperate with larger atoms to depict similar image patterns in a slightly smaller resolution.

Fig. 3.6 shows the dictionary learning result for MD/CAC. The atom orders in the larger dictionary scale (12×12 , in Fig. 3.6(b)) have been rearranged to their corresponding matched atoms' positions in the smaller scale (8×8 , in Fig. 3.6(a)) for better visual comparison. The effect of cross-scale clustering is obvious: Fig. 3.6(d) zooms in the six atoms in solid rectangles in Fig. 3.6(a) and (b). These atoms have similar patterns, but are of different resolutions; Fig. 3.6(c) highlights the six atoms in dotted rectangles in Fig. 3.6(a) and (b). These atoms have unique and scale-dependent patterns after dictionary training.

Figure 3.5: 2-scale MD/CCL dictionary learned for the image *Barbara*: (a) dictionary in scale 64×256 , (c) dictionary in scale 144×256 ; (b) are the zoomed in part taken from (a) and (b) in red rectangles.

Figure 3.6: 2-scale MD/CAC dictionary for the image *Barbara*: (a) dictionary in scale 64×256 , (b) dictionary in scale 144×256 ; (c) and (d) are the zoomed in part taken from (a) and (b) in red dotted and solid rectangles respectively.

3.4. EXPERIMENTS AND RESULTS

MD/CAC Atom Matching Property Evaluation

The atom matching scheme proposed for MD/CAC is neither injective, nor surjective, i.e., multiple atoms from different scales could choose the same atom as their match, and some atoms might not get chosen at all during the dictionary update. Experiment was carried on the image *Barbara*, in which we calculated how many different (k_c, l_c) values there were for all the atoms across the scales after each dictionary update, and the result is shown in Fig. 3.7. As a comparison, we also calculated the total number of (k_c, l_c) values when no clustering is implemented ($\lambda = 0$ in Eqn. (3.9)).

As can be seen from Fig. 3.7, for both occasions - with or without clustering - the number of matched atoms is high at iteration 1, due to the fact that the initial multi-scale DCT dictionary has a diverse range of frequency variation. After several dictionary updates, atoms get more adapted to image specific features. When this is done without cross-scale interaction, the matching number drops because natural image signals have a limited frequency band, multiple atoms from one scale are easily matched to several popular low frequency atoms in another scale, and therefore the decrease for the blue curve in Fig. 3.7. However, the number remains high when cross-scale interaction is involved (shown as the red curve in Fig. 3.7). Pattern-specific similarities across the scales have been found and encouraged, which highlights the contribution of the proposed cross-scale matching scheme.

3.4.3 MD Parameter Analysis

Atom Numbers in Different Scales

One important parameter of the multi-scale dictionary is the atom numbers in different scales. A natural expectation is that dictionary scales with higher resolution should have more atoms: since they bear more information, and hold a higher re-

3.4. EXPERIMENTS AND RESULTS

Figure 3.7: Comparison on number of matched atoms across different scales after each dictionary update iteration on the image *Barbara*.

dundancy threshold. Such reasoning is correct when different dictionary scales are trained separately without cross-scale cooperation/interaction. However, since we are using the CMP coding method to choose among atoms in different scales for a best fit for each signal, i.e., CMP judges which scale is better at the task. In this regard, we need to carry out an experiment to calculate the atom selection ratio among each scales when CMP is used for sparse coding, and the result should have the final say on how many atoms should be assigned to each scale.

Fig. 3.8 shows the experiment result for five different images. As can be seen from the figure, the smaller scale (8×8) has a much higher frequency to be selected by the CMP than the larger scales (12×12 for Fig. 3.8(a), (b), and 16×16 for Fig. 3.8(b)). The frequency ratio is approximately 6:1 in favour of the smaller scales.

Based on this data, it is safe to conclude: when CMP is used for sparse coding, if the atom number is sufficient for a smaller scale to be redundant, then this atom

3.4. EXPERIMENTS AND RESULTS

Figure 3.8: Multi-scale dictionary scale selection frequency ratio during CMP for different images. (a) uses a 2-scale dictionary (patch size 8×8 , 12×12 for each scale), and (b) uses a 3-scale dictionary (patch size 8×8 , 12×12 and 16×16 for each scale)

number will be enough, if not too many, for the larger dictionary scale(s) to be sufficiently redundant as well. This justifies the proposed multi-scale dictionary structure, and such structure facilitates sparse coding as well as dictionary learning in return.

Total Atom Number Across the Scales

We have already explained the reason for choosing equal atom numbers in different MD scales in the last experiment. Next, we evaluate the choice of total atom number across the scales.

Experiment has been carried out on the image *Barbara*, in which we calculate the CMP coding RMSE using learned MDs with different atom numbers. The RMSE are calculated with different specified coding cardinalities. The MDs are of dimension: $(64 \times K, 144 \times K)$, where K is the variable for atom number in each MD scale. All the MDs are trained using the proposed MD/CCL method. The result is shown in Fig. 3.9.

Figure 3.9: MD/CAC CMP coding performance analysis (in terms of RMSE) under different dictionary atom numbers and coding cardinalities.

As can be seen from the figure, along the “Atom Number” axis, after $K = 256$, the RMSE decrease rate will become much slower for most cardinality values along the “Cardinality” axis. Giving consideration to both representation efficiency and computational simplicity, we choose $K = 256$ for each MD scale in our current implementation.

3.4.4 Image Denoising Method and Result

The most direct application of sparse representation is signal denoising. In this section, we will describe how to implement image denoising based on the proposed multi-scale dictionary structure. Both quantitative and qualitative performance comparison with other state-of-the-art image denoising methods will be given.

3.4. EXPERIMENTS AND RESULTS

The denoising method

Suppose a noise-free image is divided into patches of L different scales, and then vectorized as signals $\{x_i^l | x_i^l \in \mathbb{R}^{N_l \times 1}, l = 1, 2 \dots L\}$. Image patches from different scales with same index “ i ” share the same starting pixel. The image is corrupted with additive zero-mean white Gaussian noise n_l , with standard deviation σ :

$$y_i^l = x_i^l + n_i^l. \quad (3.13)$$

With the noisy observations $\{y_i^l | y_i^l \in \mathbb{R}^{N_l \times 1}, l = 1, 2 \dots L\}$, we aim to find the best estimate of the original signal.

The denoising process is largely similar to the CMP algorithm, however with a slight difference. A geometrical illustration of the multi-scale denoising process has been depicted in Fig. 3.10. For an easier spatial perception, a 2-scale dictionary with each scale’s dimensionality: $N_1 = 2$ and $N_2 = 3$ is used. y_i^1 and y_i^2 are the corrupted signals at the same pixel location.

Fuzzy spheres are created around the vectors y_i^1 and y_i^2 with diameters δ_l :

$$\delta_l = C \times \sqrt{N_l} \times \sigma, \quad (3.14)$$

where σ is the noise intensity, and C is a constant. As compared to noise, image signal usually live in a lower dimensional subspace. For this case, the subspace is either a line $\Omega_{\mathcal{S}_j}^1$ for $N_1 = 2$; or a plane $\Omega_{\mathcal{S}_j}^2$ for $N_2 = 3$ ($j = 1, 2, \dots$). Both subspaces are spanned by a sparse set of atoms $\mathcal{S}$ from D_1 and D_2 respectively in their corresponding dimensions.

The denoising algorithm uses “representation sparsity” and “small sparse coding error” as the two main priors for visual signal estimation. For each scale, the denoising

Figure 3.10: A geometrical demonstration of the multi-scale coding process for application of image denoising.

algorithm chooses the subspace $\tilde{\mathcal{S}}$ with least number of support $\|\mathcal{S}\|_0$ that first enters the fuzzy sphere. The noise-free signal x_i^l is believed to lie within such sphere with high probability. The projection onto the chosen subspace $\Omega_{\tilde{\mathcal{S}}}$ is the current scale's prediction for x_i^l . The algorithm then chooses between scales to represent the current signal $\{y_i^1, y_i^2\}$ by finding the smallest target function value defined in Eqn. (3.3).

After sparse coding, the final image will be reconstructed by calculating the average of all overlapped patches. As the scale at each patch location may vary, each pixel will be involved different times by neighboring patches which can be of different scales. A matrix will be maintained to record this information for each pixel during the sparse coding phase, which will later be used as weighting coefficient during the final averaging.

3.4. EXPERIMENTS AND RESULTS

The performance

We will use the aforementioned method to perform image denoising over different types of single-scale and multi-scale dictionaries to compare their performances.

First, we evaluate the advantage the proposed cross-scale learning methods can provide when a multi-scale dictionary is used for image denoising. For that purpose, a multi-scale dictionary trained separately for each scale without any cross-scale interaction is used as reference. The result is shown in Fig. 3.11. The three different 2-scale dictionaries listed for comparison are of the same dimension (dimension $64 \times 256 / 144 \times 256$), and the PSNRs shown are the average of denoising result on six different images: *Barbara*, *Peppers*, *Lena*, *Boat*, *House* and *Straw*. As can be seen in Fig. 3.11, the denoising performance increases by approximately 0.2+ dB on average for all noise levels when the multi-scale dictionary is trained with cross-scale interaction. To be specific, MD/CAC shows better performance with low noise levels ($\sigma = 5$), MD/CCL performs better when noise level increases. This result can be understood this way: MD/CAC shows advantage for low noise levels as it is more flexible and is better at preserving cross-scale similarity and uniqueness at the same time, but only when noise is relatively low. When noise increases and becomes dominant, scale uniqueness will decrease because added noise is scale irrelevant, and cross-scale similarity becomes more difficult to capture. As MD/CCL fixes atoms' cross scale atom matching relations, and learns the atoms in a cooperative manner, i.e., signal from different scales all contribute to the learning outcome of the same atom, hence it is more resilient to larger noise. In spite of their different advantages for different noise levels, one conclusion can be drawn: multi-scale dictionaries learned via cross-scale cooperation/interaction show significant advantage in the application of image denoising over multi-scale dictionaries trained separately for each scale. The pursuit for

3.4. EXPERIMENTS AND RESULTS

Figure 3.11: Difference of denoising results between 2-scale ($8 \times 8/12 \times 12$) MD/CCL, and MD/CAC with MD (no cooperative learning or cross-scale clustering applied) as reference. Results are average of six images: *Barbara*, *Peppers*, *Lena*, *Boat*, *House* and *Straw*.

cross-scale similarity and cross-scale sparsity is therefore justified.

Second, we evaluate the advantage of a multi-scale dictionary when used for image denoising over a single-scale dictionary. The result is shown in Fig. 3.12, where a single-scale KSVD dictionary (of dimension 64×256) is used as reference. The data in Fig. 3.12 are also the averages of the same six input images used in Fig. 3.11. A complete table of performance data is listed in Table 3.1, where an additional entry of single-scale KSVD dictionary (dimension 144×256) is also involved for better comparison.

As shown in the figure and the table, multi-scale dictionaries MD/CCL and MD/CAC show an average of approximately 0.2 dB advantage in denoising performances over single-scale KSVD trained dictionary. Notably, the 2-scale dictionary

Figure 3.12: Difference of denoising results between 2-scale ($8 \times 8/12 \times 12$) MD/CCL, MD/CAC and MD, with single scale KSVD (8×8) as reference. Results are average of six images: *Barbara*, *Peppers*, *Lena*, *Boat*, *House* and *Straw*.

3.4. EXPERIMENTS AND RESULTS

Table 3.1: Denoising result for six test images with added zero mean white gaussian noise of different intensities ranging from 5 to 50. The six data listed for each table entry are results from: top row from left to right, (1) single-scale KSVD dictionary (dimension 64×256); (2) single-scale KSVD dictionary (dimension 144×256); (3) 2-scale dictionary trained by KSVD separately each scale; bottom row from left to right, (4) 2-scale dictionary proposed in [2], (5) 2-scale dictionary trained by MD/CCL; (6) 2-scale dictionary trained by MD/CAC. Best performance in terms of PSNR (in dB) are in bold.

σ /PSNR	5 /34.15			10 /28.13			15 /24.61			20 /22.11			25 /20.17			50 /14.15		
barbara	38.02	36.80	38.09	34.59	34.14	34.53	32.37	32.28	32.39	30.87	30.90	30.81	29.60	29.68	29.52	25.47	26.39	25.58
	38.19	38.22	38.21	34.62	34.70	34.60	32.47	32.61	32.50	30.95	31.01	30.93	29.94	29.76	29.62	26.53	26.12	25.78
peppers	37.98	36.91	38.00	35.01	34.86	34.96	33.50	33.29	33.47	32.39	32.16	32.34	31.48	31.36	31.47	28.22	28.50	28.25
	38.06	38.02	38.07	35.12	35.03	35.10	33.51	33.60	33.57	32.37	32.46	32.42	31.49	31.59	31.56	28.38	28.47	28.37
lena	38.54	37.68	38.58	35.49	34.91	35.42	33.67	33.40	33.52	32.34	32.24	32.24	31.32	31.29	31.22	27.81	28.12	27.69
	38.60	38.61	38.62	35.58	35.64	35.51	33.71	33.81	33.65	32.46	32.52	32.35	31.48	31.45	31.30	28.24	28.18	28.02
boat	37.04	35.66	37.13	33.68	33.25	33.65	31.78	31.48	31.72	30.36	30.38	30.29	29.28	29.12	29.17	26.01	26.14	25.93
	37.31	37.10	37.17	33.71	33.74	33.69	31.67	31.87	31.77	30.37	30.50	30.38	29.27	29.35	29.23	26.21	26.33	26.07
house	43.27	43.27	42.71	39.11	39.40	38.73	36.90	37.41	36.90	35.70	35.96	35.53	34.51	34.70	34.49	30.75	31.49	30.93
	42.96	43.37	43.40	39.09	39.40	39.44	37.22	37.45	37.52	35.81	36.14	36.16	34.77	34.90	34.93	31.68	31.53	31.39
straw	30.89	24.71	30.09	28.38	24.26	28.03	26.09	23.80	26.18	24.43	23.10	24.49	23.20	22.34	23.29	19.83	19.32	19.81
	31.30	32.11	32.13	28.29	29.06	28.64	24.01	26.50	26.20	23.49	24.71	24.40	22.81	23.34	23.10	19.39	19.80	19.53
average	37.62	35.84	37.43	34.38	33.47	34.22	32.39	31.94	32.36	31.02	30.79	30.95	29.91	29.79	29.86	26.35	26.66	26.36
	37.74	37.91	37.93	34.40	34.60	34.50	32.10	32.64	32.54	30.91	31.22	31.11	29.95	30.06	29.96	26.74	26.74	26.51

3.4. EXPERIMENTS AND RESULTS

trained without cross-scale interaction (MD in Fig. 3.12) actually performs poorer than its single-scale counterpart. This could be explained by the fact that: since MD shows overwhelmingly better efficiency in signal representation (already validated in previous experiment in Fig. 3.4), such capability could extend to noise representation as well. Therefore, the denoising performance of a multi-scale dictionary learned without cross-scale interaction is actually worse than a single-scale dictionary; this in a way further shows the necessity and effectiveness of the cross-scale interaction schemes herein proposed for MD/CCL and MD/CAC.

We also include the result from [2] for comparison. In [2] the authors proposed several additional algorithmic improvements during the process of sparse coding (imposed preference for the DC component, OMP stopping criteria, etc.). In order to focus only on the advantage brought about by the multi-scale scheme, we have carefully re-implemented their algorithm without the additional improvements. The patch sizes for each noise level are: 10×10 for $\sigma = 5$; 12×12 for $\sigma = 10$; 16×16 for $\sigma = 15$, 20 and 25; 20×20 for $\sigma = 50$, respectively. The results are listed in Table 3.1. It can be seen from the table that the proposed multi-scale scheme outperforms the method in [2], except for a few cases at larger noise levels ($\sigma = 50$).

Visual comparison of these competing denoising methods are provided in Fig. 3.13 and Fig. 3.14. Fig. 3.13 shows the result for the image *Straw* contaminated with gaussian noise $\sigma = 15$. The image contains an enormous amount of cross-scale patterns. As can be seen, the proposed multi-scale methods in Fig. 3.13(e) and (f) preserve the straw edges and texture details better than other methods. Fig. 3.14 gives another visual example on the image *Barbara* contaminated with gaussian noise $\sigma = 15$. The cross-scale mechanism introduced herein also helps to remove noise in textureless parts of the image better than the other methods, as can be seen from the

Figure 3.13: Denoising result on image *Straw* with Gaussian noise $\sigma = 15$. (a) is the noisy image, the part in red rectangle is zoomed in from (b) to (f), which are the denoising result using (b) K-SVD (64×256 , $26.09dB$), (c) K-SVD (144×256 , $23.80dB$), (d) 2-scale dictionary proposed in [2] ($24.01dB$), (e) 2-scale MD/CCL ($26.50dB$) and (f) 2-scale MD/CAC ($26.20dB$).

zoomed in part of the wall in the background.

3.5 Conclusion: Similarity or Uniqueness

In this Chapter, we proposed a multi-scale dictionary structure, a multi-scale sparse coding algorithm (CMP), and two different types of cross-scale dictionary learning methods: cross-scale cooperative learning (MD/CCL), and cross-scale atom clustering (MD/CAC). Our aim is to study the cross-scale interactions of a multi-scale dictio-

Figure 3.14: Denoising result on image *Barbara* with Gaussian noise $\sigma = 15$. (a) noisy image, the part in red rectangle is zoomed in from (b) to (f), which are the denoising result using (b) K-SVD (64×256 , $32.37dB$), (c) K-SVD (144×256 , $32.28dB$), (d) 2-scale dictionary proposed in [2] ($32.47dB$), (e) 2-scale MD/CCL ($32.61dB$) and (f) 2-scale MD/CAC ($32.50dB$).

nary, and its applications. The proposed multi-scale dictionary, along with the CMP coding algorithm, show better performance in sparse representation over single-scale dictionaries. Experiments also showed its advantage in image denoising.

The focus of this Chapter is on the cross-scale interaction of the multi-scale dictionary. MD/CCL is better at enforcing cross-scale similarity because of its fixed atom matching relations; MD/CAC is more flexible in representing scale-unique patterns due to its re-calculation of clustering centers during each dictionary update. The two attributes: similarity and uniqueness that characterize the two proposed cross-scale

3.5. CONCLUSION: SIMILARITY OR UNIQUENESS

dictionary learning methods, showed their respective advantages when used in image denoising applications at different noise levels. The compromise between the two attributes in a multi-scale dictionary is vital not just in image denoising, but also in many other multi-scale scenarios.

In conclusion, the multi-scale dictionaries herein proposed are advantageous over single-scale dictionaries in sparse coding and signal denoising, and the cross-scale interaction during dictionary learning is vital in exploiting such advantages.

Chapter 4

Perspective Shifted Dictionary for Light Field Representation and Its Fast Coding Technique

4.1 Introduction

The light field (LF) is a function that describes the intensities of light rays in all possible propagation directions. Conventional cameras use a converging lens to record a 2D projection of the high dimensional LF data. This inevitably loses a lot of information, especially each light ray's directional information. The light field cameras are designed to resolve the integration process and capture the extra directional information, which is useful in many applications such as refocusing [67], depth inference [3] [68], and 3D reconstruction [27] [69] etc.

Recent emerging light field capture technologies such as Lytro [14] (using an inserted micro-lens array), Pelican [70] (multi-camera array), or the software controlled

4.1. INTRODUCTION

“Focus Sweep” technology by Nokia, each provides different but efficient designs to solve the aforementioned difficulties to some extent. However, the compromise between spatial and angular resolution seems inevitable. And this hope of capturing the light field with both high spatial and angular resolution gives rise to a lot of new research, notably on applying compressed sensing techniques to light field imaging.

Ashok et. al. first attempted to exploit the spatio-angular correlations inherent in the light fields, and simulated the performance of the compressed light field imager based on a principal component (PC) measurement basis, which they prove to achieve four times fewer measurements and three times shorter exposure to capture light field with quality compared to non-compressed schemes [5]. Babacan et. al. proposed to reconstruct the light field based on a Bayesian reconstruction algorithm, using a few angular coded light field measurements. Notably, total variation image prior and cross-image prior are used in their Bayesian framework [18].

Recently Marwah et. al proposed a new framework in [71] where they code the light field near the camera sensor plane, and is able to reconstruct the light field based on a single measurement. The reconstruction is based on sparse coding over a trained light field dictionary. This dictionary is of high signal dimension ($\mathbb{R}^{(11 \cdot 11 \cdot 5 \cdot 5) \times 1}$) with a very large number of atom numbers (more than 2300), and therefore this method is inefficient in dictionary training and sparse coding.

In this chapter, we propose and explain a perspective-shifted light field dictionary (LFD), with a structure that is much more easily trained, and provides many advantages for signal reconstruction and compression (will be explained in later chapters). Because of the high dimensionality of the LF data, the light field dictionary contains large number of atoms, which poses a big challenge for efficient computation during sparse reconstruction. A two-stage coding algorithm is designed for the LFD. It in-

corporate an interim Winner-Take-All (WTA) hash coding stage into the Orthogonal Matching Pursuit (OMP) algorithm, which proves to speed up the OMP by almost three times but still maintains the reconstruction quality.

4.2 Perspective Shifted Light Field Dictionary

Fig. 4.1 shows the typical structural layout of a 4D light field [72]. In the figure, each small square represents a separate stereo viewing point, from which one image is hypothetically taken for each LF view. The red and blue arrows show the displacements of each view with respect to the central view in the horizontal and vertical directions respectively, and the displacement values are recorded in matrices M_H and M_V . For a 5×5 light field, M_H and M_V take the following form:

$$M_H = \begin{bmatrix} -2 & -2 & -2 & -2 & -2 \\ -1 & -1 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 & 2 \end{bmatrix},$$

$$M_V = \begin{bmatrix} -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \end{bmatrix}. \quad (4.1)$$

The spatial relations among each view can be considered as linear shifting, and the disparity of the same scene point at different views can be expressed as (with

respect to the central view):

$$S_d(dp, v) = dp \times (M_H(v) + M_V(v)), \quad v = 1, 2, \dots, V, \quad (4.2)$$

where dp is the unit disparity value between adjacent views. dp is a parameter related to the light field camera's aperture dimension, and the scene point's distance to the camera. Once the camera's settings are fixed, dp is only inversely related to the scene depth. For different views, v is the LF view index, and V is the total number of views, which is 5×5 for the case in Fig. 4.1 and 4.2.

Here we make the following assumption:

Assumption 4.1. Pixels in a small patch area around that scene point have the same disparity (depth).

Based on this assumption, we can predict the patch appearance in all other views by simply shifting the central view. According to Eqn. (4.2), the shifting distance is $S_d(dp, v)$ for the view v with unit disparity dp .

Based on this idea, a LF dictionary could be created by shifting the atoms in a normal image dictionary for different LF views (v) at different disparity values (dp) [72]. It is expected that the resultant dictionary atoms are able to sparsely represent the LF data. The dictionary perspective-shifting process will be generalized in detail as follows:

- First, apply the KSVD [45] method, a global dictionary $D^c \in \mathbb{R}^{N^c \times K^c}$ is trained from 10 benchmark natural images which are decomposed into overlapping patches of $\sqrt{N^c} \times \sqrt{N^c}$ pixels. K^c is the atom number.
- Second, for a certain unit disparity value dp , and a certain atom in D^c , an array of smaller patches ($\sqrt{N^v} \times \sqrt{N^v}$ pixels) are extracted from the atom

Figure 4.1: A 5×5 LF view configuration: red and blue arrows represent relative perspective shifting distances in the horizontal and vertical directions for each view.

($\sqrt{N^c} \times \sqrt{N^c}$ pixels, and $N^v \leq \frac{N^c}{4}$), with extraction centre locations decided by S_d defined in Eqn. (4.2). As illustrated in Fig. 4.2(a) the grey squares are the 1st, 5th, and 25th views to be extracted. The extracted patches are then vectorized and concatenated as a vector to be the new atom for the LF dictionary $d_{LF} \in \mathbb{R}^{(N^v \cdot V) \times 1}$. The vectorization and concatenation process is illustrated in Fig. 4.2(c).

- Third, the previous step is repeated for a vector of different disparity values dp which span the LF dictionary's disparity dimension: $dp \in \{dp(1), dp(2) \dots dp(S)\}$. S is the total number of different unit disparity values. Fig. 4.2(b) shows the extraction process for a different dp value (dp'').
- Last, repeat the second and third step for all K^c atoms in D^c to create the final LF dictionary: $D_{LF} \in \mathbb{R}^{(N^v \cdot V) \times (S \times K^c)}$.

Still in Fig. 4.2(a) and (b), notice as dp increases, the array of black dots expands accordingly. They stand for the extraction centers for different views. The view shift-

Figure 4.2: Formation of the proposed LF dictionary by perspective shifting. (a) shows the extraction process of the LF atom ($\sqrt{N^v} \times \sqrt{N^v}$ in grey colour) from a globally trained image dictionary ($\sqrt{N^c} \times \sqrt{N^c}$) with adjacent disparity value dp' . The dotted line surrounded area are symmetric replication of the pattern at the border of the solid rectangle inside. (b) shows the situation when adjacent disparity is at another increased value dp'' . (c) shows the structure of the LF dictionary D_{LF} . Each column is concatenated with vectorized view patches. The different brightness for each LF atom represents different adjacent view disparities.

Figure 4.3: Four of the LF atoms in D_{LF} . Each row is a different atom pattern; each column has a different adjacent view disparity.

Figure 4.4: Three sample atoms that moves patterns from outside of the central patch, which are suitable for describing irregular patterns under occlusions and specularities.

ing/extraction process applies bi-linear interpolation when the patch centers fall on sub-pixel positions. And d_k^c (shown as solid rectangle) will be padded with symmetrically repeated patterns along its borders in case the shifted patch reach out of its area (the padded area is in dotted rectangles). Some sample dictionary atoms are shown in Fig. 4.3.

The shifting process will move some patterns outside of the central patch into the center at certain disparities. This gives the dictionary flexibility to describe irregular patterns such as under occlusions and specularities. Some example atoms are shown in Fig. 4.4.

4.3. TWO STAGE CODING FOR LIGHT FIELD RECONSTRUCTION BASED ON LFD

Figure 4.5: Angular compressive light field imager architecture

4.3 Two Stage Coding for light field reconstruction based on LFD

Due to the large dictionary size (both signal dimension, and atom number), normal sparse coding techniques become quite computationally inefficient. We propose a two-stage coding technique for the proposed LFD.

Before we start describing the algorithm, the basic architecture of a compressed sensing light field camera will be introduced. Fig. 4.5 shows a typical prototype with a coded aperture mask [5]. Light rays from the world scene will pass through the camera lens, and before they are converged toward the camera sensor plane, they are modulated at the camera aperture. Each light field view (at a different location on the aperture) will be modulated with a constant transmission rate, and then they will integrate at the camera sensor plane.

The final captured image can be expressed in matrix multiplication form:

$$x = \Phi l, \quad \Phi = [\Phi_1 \quad \Phi_2 \quad \dots \quad \Phi_V], \quad (4.3)$$

4.3. TWO STAGE CODING FOR LIGHT FIELD RECONSTRUCTION BASED ON LFD

where $x \in \mathbb{R}^{N \times 1}$ is the compressed sensing measurement for the original light field data $l \in \mathbb{R}^{(N \cdot V) \times 1}$. x has been vectorized from its original 2D form $\mathbb{R}^{\sqrt{N} \times \sqrt{N}}$. $\Phi \in \mathbb{R}^{N \times (N \cdot V)}$ is the sensing matrix, which is concatenated by different Φ_v ($\Phi_v \in \mathbb{R}^{N \times N}$, $v = 1, 2, \dots, V$), with each Φ_v being an identity matrix multiplied by the constant that equals to the coded aperture mask value for the view v . The modulation mask is a 5×5 random matrix with each element range: $(0, 1]$.

To recover the light field data l from its measurements x , Eqn. (4.3) poses an inverse problem in which the number of measurements (N) is significantly smaller than the number of unknowns ($N \cdot V$). Since the light field data has a strong spatio-angular correlation, therefore can be sparsely represented by a few signal examples [71]. With the proposed perspective-shifted light field dictionary D_{LF} , the sensing process in Eqn. (4.3) can be re-written as:

$$x = \Phi D_{LF} \alpha, \quad (4.4)$$

and the reconstruction problem for the light field data can be expressed as:

$$\arg \min_{\alpha} \|x - \Phi D_{LF} \alpha\|_2^2 + \lambda \|\alpha\|_0^0. \quad (4.5)$$

$$l = D_{LF} \alpha \quad (4.6)$$

To reconstruct l , Orthogonal Matching Pursuit (OMP) algorithm could be applied in a similar manner as it is used in image denoising [52]. However, the extremely large dimension of D_{LF} makes the greedy algorithm extremely time consuming and inefficient.

Now we will explain the two-stage coding algorithm in detail, which is designed to speed up the coding process.

4.3. TWO STAGE CODING FOR LIGHT FIELD RECONSTRUCTION BASED ON LFD

Stage 1: WTA Hash Code Matching for Disparity Invariant Patterns

The algorithm aims to narrow down the search range of $\Phi D_{LF} \in \mathbb{R}^{N^v \times (K^c \cdot S)}$ for the signal x in the first stage. This is achieved by calculating the hamming distances [73] between the WTA hash codes [74] of the median filtered LFD and the hash codes of the signal.

WTA Hashing non-linearly transforms the signal from its input feature space into sparse codes, and the coding process for a input vector $x \in \mathbb{R}^{N^v \times 1}$ is described as following:

- Perform N_P ($N_P = 12$ in our experiment) times of random permutations on the elements of x .
- For each permutation, pick up the index that belongs to the largest element among the first N_W ($N_W = 8$ in our experiment) elements.
- Concatenate the N_P maximum elements, and output the final hash code vector for x of length N_P .

In order to use the WTA hash coding method to increase the efficiency of light field sparse coding, the following assumption on light field measurements is used.

Assumption 4.2. Light field measurements (in forms of $\Phi d_k^{dp} \in \mathbb{R}^{N^v \times 1}$ for the LF atoms) with identical central view (same index k) has identical WTA Hash code disregarding their different perspective-shifting amount (different dp).

We calculate the hash code $[d_k^{dp}]_{\#} \in \mathbb{R}^{N_P \times 1}$ for each integrated light field atom Φd_k^{dp} , $k = 1, \dots, K^c$, $dp \in \{dp(1), dp(2), \dots, dp(S)\}$. Based on Assumption 4.2, measurement Φd_k^{dp} with identical index k but different dp values should produce the same

4.3. TWO STAGE CODING FOR LIGHT FIELD RECONSTRUCTION BASED ON LFD

hash code. In practice, we use the median value among $\{[\Phi d_k^{dp}]_{\#} \mid dp = dp(1), dp(2), \dots, dp(S)\}$ to represent their “identical” hash code:

$$[\Phi d_k^m]_{\#} = \text{median}([\Phi d_k^{dp}]_{\#}, dp = dp(1), dp(2), \dots, dp(S), \quad (4.7)$$

where $[\Phi d_k^m]_{\#} \in \mathbb{R}^{N_P \times 1}$, and the final hash coded light field dictionary will be $[\Phi D_{LF}^m]_{\#} \in \mathbb{R}^{N_P \times K^c}$.

The hash code for the input data $x \in \mathbb{R}^{N_v \times 1}$ is calculated as $[x]_{\#} \in \mathbb{R}^{N_P \times 1}$. For the hash code matching, hamming distance [73] is calculated between $[x]_{\#}$ and each columns of $[\Phi D_{LF}^m]_{\#}$, and the smallest $K_{\#}^c$ items will be chosen for the next coding stage. Since size of the original light field dictionary is greatly reduced (S times smaller), and hamming distance is calculated instead of euclidean, this matching process is extremely efficient. Moreover, since rank correlation [75] is stable to noise and variations that do not alter implicit ordering, this method is therefore more resilient to the disparity-induced measurement differences, which further strengthens Assumption 4.2.

Stage 2: Exact signal matching

In Stage 1, $K_{\#}^c$ hash codes that stand for $K_{\#}^c$ groups of LFD atoms have been picked out. The $K_{\#}^c$ hash codes will be expanded to form a sub-set of the modulated dictionary $\Phi D_{LF}^{\subset} \in \mathbb{R}^{N_v \times (K_{\#}^c \cdot S)}$ for the second stage OMP matching:

$$\Phi D_{LF}^{\subset} = \{\Phi d_k^{dp} \mid k \in K_{\#}^c, dp = dp(1), dp(2), \dots, dp(S)\}. \quad (4.8)$$

In stage 2, the LF atom in $\Phi D_{LF}^{\subset}$ that produces the maximum projection for the current signal residual will be chosen for coding, and the residual will be updated

4.4. MODEL EVALUATION

accordingly. The residual will be used for the next iteration of the two-stage coding; with a new hash code calculated for it. The iteration stops until a certain residual ε_0 is reached. The complete algorithm for the two-stage coding is described in Algorithm 4.1. Fig. 4.6 gives a flow chart for the algorithm.

Algorithm 4.1 Two-Stage coding for light field reconstruction

input parameter: input LF measurement x ; mask modulated LF dictionary ΦD_{LF} ; target coding error ε_0

- Hash code all the atoms in ΦD_{LF} with code length N_P : $[\Phi D_{LF}]_{\#} \in \mathbb{R}^{N_P \times (K^c \cdot S)}$
- Find the median hash code for all atoms that share the same central atom, and produce the hash code dictionary: $[\Phi D_{LF}^m]_{\#} \in \mathbb{R}^{N_P \times K^c}$
- Initialize the coding error $\varepsilon = x$

while $\varepsilon > \varepsilon_0$ **do**

Stage 1: WTA hash code filtering

- Hash code the current residual ε as $\varepsilon_{\#}$.
- Find the $K_{\#}^c$ hash codes in $[\Phi D_{LF}^m]_{\#}$ that produce smallest hamming distances with $\varepsilon_{\#}$.

Stage 2: Exact residual matching

- Take atoms in ΦD_{LF} whose hash codes have been picked in **Stage 1**, and form the LFD subset: $\Phi D_{LF}^{\subseteq} \in \mathbb{R}^{N^v \times (K_{\#}^c \cdot S)}$
- Sweep current residual ε through all atoms in $\Phi D_{LF}^{\subseteq}$, find the atom Φd_{k_0} that produces the largest projection: $k_0 = \arg \max_k \{(\Phi d_k)^T \varepsilon\}$
- Update support $S = S \cup d_{k_0}$.
- Calculate coefficient s.t. $Supp\{\alpha\} = S$: $\alpha = \arg \min_{\alpha} \|\Phi D_{LF}^{\subseteq} \alpha - x\|_2^2$
- Update current residual according to: $\hat{\varepsilon} = \|x - \Phi D_{LF}^{\subseteq} \alpha\|_2^2$

4.4 Model Evaluation

In order to evaluate the proposed two-stage coding algorithm in context of compressed light field sensing, we create a perspective-shifted light field dictionary $D_{LF} \in \mathbb{R}^{(64 \cdot 25) \times (31 \cdot 400)}$ is trained with 10 different images. The disparity scale vector for the dictionary is defined as: $[-3.0, -2.8, \dots, 2.8, 3.0]$. A uniformly distributed random aperture mask (with range $(0, 1]$) is generated for Φ .

Figure 4.6: Diagram of the two-stage coding algorithm. Note in the perspective-shifted LF dictionary, atoms with same colored have identical central view, but with different perspective-shifting amounts.

Three light field images *Maria*, *Buddha* [1], and *Dragons* [4] [1] are used for the experiment. The dimensions of the LF images have all been decimated to 5×5 . 50K random samples of light field data $l \in \mathbb{R}^{(64 \cdot 25)}$ are used for the evaluation. Matlab function *cputime* is used to calculate the average running time (average over all 50K samples) for each modulated light field data entry $x \in \mathbb{R}^{64 \times 1}$, $x = \Phi l$. We also calculated the average reconstruction error between the recovered light field (based on Eqn. (4.5) and (4.6)) with the ground truth data l , and the results are shown in Fig. 4.7. The horizontal axis indicates the number of atoms $K_{\#}^c$ picked after the first-stage hash code matching. Experiments are also carried out using original OMP

4.4. MODEL EVALUATION

Figure 4.7: Performance evaluation of the two-stage coding method. In the legend, *Recon. Err #* stands for reconstruction error when using proposed two-stage coding method; *Recon. Err* stands for reconstruction error using single-stage OMP coding. And similar with the notation *CPU Time #*, *CPU Time*.

coding directly over D_{LF} for comparison.

As can be seen from the figure, there is a jump in computation time with respect to $K_{\#}$ at around 18. The reconstruction error for the two-stage coding approaches to that of single-stage coding very quickly, and is almost the same near $K_{\#} = 16$, surpassing it after $K_{\#} = 28$. However the CPU Time by two-stage coding at $K_{\#} = 18$ is nearly one third of that by single-stage coding. This experiment validates the efficiency of the proposed two-stage coding scheme, when we choose the optimal $K_{\#}^c = 18$.

The sudden jump of computation time may vary when a different desktop is used, and this is most likely related to the parameter of CPU bit-width. Currently we are using a 64-bit desktop for the experiment. It is safe to infer, when a wider bit-width CPU is used, the “jump” will occur at a higher $K_{\#}^c$ number, and the optimal choice

4.5. CONCLUSION

of $K_{\#}^c$ may increase accordingly.

During the Two Stage coding, in Stage 1, not only is the number of atoms drastically reduced from $K^c \cdot S$ to K^c (S times smaller than the original D_{LF} , $S = 31$, $K^c = 400$), so is the vector length for each dictionary atom reduced from $N_v \cdot V$ to N_P (more than 80 times shorter, $N_v \cdot V = 64 \cdot 25 = 1600$, $N_P = 18$). In Stage 2, the dictionary dimension is reduced to $K_{\#}^c \cdot S = 18 \cdot 31 = 558$. In general, the total number of atoms being searched during the Two Stage coding process is $K_c + K_{\#}^c \cdot S = 400 + 558 = 958$, instead of the original $K^c \cdot S = 400 \cdot 31 = 12400$, with the atoms length in the first stage reduced from 1600 to 18. Both the searching size and complexity are greatly reduced, which explains the reason for the speed up.

4.5 Conclusion

In this Chapter, a light field dictionary (LFD) based perspective shifting of central view atom is proposed for sparse representation of the high dimensional light field data. A simple light field compressed sensing framework based on aperture coding is introduced to evaluate the efficiency of the proposed LFD.

The extreme dimensionality of the LFD poses great challenge for light field reconstruction. A two-stage coding algorithm is proposed which uses the Winner-Take-All (WTA) hashing strategy to narrow down the search range for light field sparse coding. The algorithm proves to be able to increase the coding efficiency by almost three times and keep the reconstruction quality almost the same with original OMP coding.

Chapter 5

Light Field Compressed Sensing

5.1 Introduction

5.1.1 Light Field Acquisition

The theoretical principle of light field imaging have been introduced over a century ago [11, 12], however it is only with the recent technological development, it has become practical and popular. One technical challenge to record the light field is the extremely large amount of information the light field contains, the other is in the hardware design that can resolve the light ray integration.

A micro-lens array is useful in diverging the light rays in a micro scale so that the converging light ray's original directional information could be recorded by the camera sensors closely behind it [13]. The famous Lytro Camera falls into this category [14]. Except from a micro-lens array, a spatial modulation mask can also be inserted closely in front of the sensor for an angular modulated sampling of the light field [4]. The images generated with this method are generally with high quality. However since the sensor resource has to shared between spatial and angular dimensions, it greatly

5.1. INTRODUCTION

compromises the final output image resolution.

Except from the camera sensor plane, spatial modulation can also be performed near the system aperture. Such designs and their performance analysis can be found in [5,16–18,76]. In [19], the author uses a Liquid Crystal on Silicon as spatial modulator, which increased the efficiency of aperture coding in terms of light transmission rate and resolution.

In [15] Manakov et al. introduced a kaleidocamera for light field capturing. The camera uses two parallel mirrors to form a kaleidoscope cavity, which physically copies a 3×3 sub-aperture image collage. The system is designed as an add-on to existing DSLR cameras and lenses. Since it is based on spatial copying, it also suffers from the resolution limit since camera sensor has to be shared between the 9 views.

Other light field acquisition hardware include: the Pelican [70], where a monolithic multi-camera array system is used; and the HTC One M8's Duo Camera, which uses a combination of software and an additional lens that collects rudimentary depth information for light field reconstruction. These technologies generally produces much lower imaging quality, however their low-cost makes them competitive in mobile applications.

5.1.2 Light Field Reconstruction

The post-processing and reconstruction of the light field data is necessary for almost every light field camera, and therefore it is a critical procedure which decides the quality of the final output of the system. Even for cameras that share sensor resource between the dimensions [14], the raw output tends to be dim and distorted which requires intensity and contrast adjustment as well as interpolation [77] to rectify the final output.

5.1. INTRODUCTION

For the kaleidocamera [15], which has an initial 3×3 copies of the sub-aperture views, needs to be interpolated to higher resolution based on optical flow between adjacent views, before they can be used for refocusing or other light field applications.

The hope of capturing the light field with both high spatial and angular resolution gives rise to a lot of new researches, notably on applying compressed sensing techniques to light field imaging. [?]. Ashok et. al. first attempted to exploit the spatio-angular correlations inherent in the light field [5]. Babacan et. al. proposed to reconstruct the light field using a coded aperture camera based on a Bayesian reconstruction algorithm. Kamal et al. [20] used compressed acquisitions from a camera array modulated by random convolution to reconstruct the LF. A set of directional wavelet transforms are used for the reconstruction to exploit the sparsity in the angular domain. Other LF compressed sensing examples are also found in applications such as hyperspectral imaging [78] and compressive focal stack imaging [79]. Attention is also paid to sparsity in the transformed domain, for example, Shi et al. proposed an algorithm that is optimized for sparsity in the continuous Fourier spectrum to reconstruct the light field [80].

Marwah et. al proposed a light field sparse representation framework in [4] where they reconstruct the light field based on sparse coding over a trained LF dictionary, using measurements from a coded-sensor plane camera. The dictionary learning method and its coding algorithm is extremely computation intense (takes more than 10 hours on a 24-core processor to learn the dictionary, and takes 18 hours for the reconstruction of light field with spatial resolution 480×270 and angular resolution 5×5 on a 8-core workstation). The compressed sensing and reconstruction scheme proposed in this work can produce better LF reconstruction quality with much lower computation cost.

5.1.3 Our Contribution

In this work a compressed sensing framework is proposed for the sampling and reconstruction of a high resolution light field based on a coded aperture camera.

Based on the Light Field Dictionary (LFD) proposed in Section 4.2, a sub-aperture vignetting matrix is incorporated to model the light field integration over imaging lenses. The designed shifting process makes the dictionary atoms disparity-aware, which facilitates the light field reconstruction in both reconstruction quality and computational simplicity.

Two different light field sensing methods that fully take the advantage of such LF dictionary are proposed. A Sub-Aperture Scan (SAS) process is introduced to quickly capture the perspective disparity flow of the scene during camera's focus-detection process before final image shot is taken. Another method, i.e., Normalized Fluctuation (NF) is subsequently proposed to calculate the disparity from aperture modulated images. Both SAS and NF have their respective advantages in terms of hardware implementation or reconstruction quality, which will be explained in detail in this chapter. The disparity information for each image patch will be used to extract the corresponding dictionary segment during light field reconstruction. This greatly boosts the computation efficiency, and experiments also show that it increases reconstruction quality as well, especially for cases with low measurement numbers.

Finally, a hardware implementation of the proposed light field acquisition/reconstruction scheme is carried out. A Liquid Crystal on Silicon (LCOS) device is used as aperture modulator. An additional optical scheme design is also proposed to integrate the whole scheme into one opto-mechanical module that can best serve our design ideology. Both quantitative and qualitative evaluation shows the proposed methods produces state-of-the-art performance in both reconstruction quality and computation

efficiency.

5.2 Vignetting and Sensing Matrix Optimization for the Light Field Dictionary

Based on the Light Field Dictionary (LFD) introduced in Section 4.2, the issues regarding to its application in light field acquisition will be addressed in this section. To be specific, the vignetting pattern calibration, and sensing matrix Φ optimization will be discussed in detail.

5.2.1 The Vignetting Pattern Matrix

For most cameras, the light volume that reaches the image sensor tends to decrease from the image center to the corners, which is the phenomenon known as vignetting [81]. Classical radiometry shows that the irradiance from the aperture of a lens onto a point on the film is equal to the following weighted integral of the radiance coming through the lens:

$$I(x, y) = \frac{1}{F^2} \iint L_f(x, y, s, t) \cos^4 \theta \, ds \, dt, \quad (5.1)$$

where F is the separation between the exit pupil of the lens and the camera sensor, L_f is a continuous light field and I is camera sensor image. Four parameters are used to indicate a light ray: (s, t) are coordinates on the lens aperture plane, and (x, y) are coordinates on the camera sensor plane. θ is the angle between ray (x, y, s, t) and the sensor plane normal. $\cos^4 \theta$ is the falloff factor that represents the reduced effect of rays striking the sensor from oblique directions [14].

In this Chapter we aim at the reconstruction of a discrete light field L with limited number of measurements from the camera sensor. We change the continuous light

5.2. VIGNETTING AND SENSING MATRIX OPTIMIZATION FOR THE LIGHT FIELD DICTIONARY

field $L_f(x, y, s, t)$ in Eqn. (5.1) to a discrete one $L_F(x, y, v)$, where $v = 1, 2, \dots, V$ is the LF view index. The resulting light field integration equation becomes:

$$I(x, y) = \frac{1}{F^2} \sum_v L_F(x, y, v) M_V(v) \cos^4 \theta. \quad (5.2)$$

$M_V(v)$ is the modulation mask value for the view v . For conventional camera without a modulation mask, $M_V(v) = 1$ for all views $v = 1, \dots, V$. Eqn. (5.2) can be written in matrix format:

$$I = \Phi L. \quad (5.3)$$

Where L is vectorized form of L_F , different LF views are directly concatenated. $\Phi \in \mathbb{R}^{N^v \times (N^v \cdot V)}$ is the projection matrix that sums all the LF views together. Φ can be expressed as the juxtaposition of V sub-matrices $\Phi(v)$, $v = 1, 2, \dots, V$, and it has two functions:

$$\Phi(v) = V_G(v) \cdot M_V(v). \quad (5.4)$$

1. incorporate the sub-aperture vignetting effect: Fig. 5.1 shows the configuration of the vignetting matrix V_G . Each sub matrix $V_G(v)$, ($v = 1, 2, 3, 4$) stands for its corresponding sub aperture's vignetting pattern.
2. modulate the sub-aperture signal: an aperture modulation mask is installed on the lens aperture plane in this work, a modulation coefficient $M_V(v)$ is multiplied on light rays of that sub aperture view.

The vignetting pattern for each sub-aperture view is related to the camera parameters and the aperture shape. Numerous efforts have been made in modelling the vignetting pattern in a camera, most of which adopt a polynomial parametric model such as in [82] and [17]. In this work however, we choose to use experiment measure-

5.2. VIGNETTING AND SENSING MATRIX OPTIMIZATION FOR THE LIGHT FIELD DICTIONARY

Figure 5.1: Juxtaposed sub-matrices $V_G(v)$ ($v = 1, 2, 3, 4$), with the vignetting pattern for each sub aperture on their respective diagonals.

Figure 5.2: Sub-aperture images with vignetting. First row are the aperture shapes, second row are the sub-aperture images with vignetting, third row are the calibrated vignetting pattern for each sub-aperture view. The brighter pixels indicates higher transmission rate.

ment to calibrate the vignetting pattern as a non-parametric model. By directing a white LED light source into the camera sensor, each sub-aperture view's vignetting pattern can be easily recorded. After normalization, the vignetting pattern can be directly used as diagonals in $V_G(v)$. Shown in Fig. 5.2 are the calibration results for a simple 4×4 light field.

5.2.2 Sensing Matrix Optimization

The selection of a good measurement matrix has been studied in many recent works. Based on a random matrix, which is generally considered to be good at providing a powerful frame in compressed sensing, the authors in [83] proposed to optimize the sensing matrix for a given dictionary to make each atom as orthogonal to each other as possible. In our proposed system, the sensing matrix has a specific structure (as shown in Fig. 5.1), which constrains the matrix elements to be related to each other, therefore the optimization method in [83] is not suitable in our system.

According to our experiments, mask patterns generated from uniform random process with a large neighborhood correlation produces outstanding reconstruction performance. The mask's total neighborhood correlation is defined as:

$$Rn = \sum_{i,j \in \Omega_{i,j}} \frac{M_V \odot f[M_V, (i,j)]}{\|M_V\|_2^2 \|f[M_V, (i,j)]\|_2^2} \quad (5.5)$$

where $\Omega_{i,j}$ is the 3×3 neighbourhood around the mask element (i,j) . $f[A, \vec{b}]$ is an operator that translates a matrix A to the direction specified by $\vec{b}$.

For a single measurement, the mask pattern is selected from N_R (N_R should be large enough, $N_R=10000$ in our experiment) random generated normal distribution, and then among these N_R masks, the one with the maximum Rn is chosen as one candidate mask pattern.

When multiple measurements are taken, the masks used for each one should be as orthogonal to each other as possible. We choose among the pool of N_P ($N_P=1000$ in our experiment) mask candidates, and the next mask pattern will be chosen among

5.3. LIGHT FIELD ACQUISITION AND RECONSTRUCTION

these candidates according to the criterion:

$$l' = \arg \min_l \sum_{k=1}^{K_I} \frac{M_V^k \odot M_V^l}{\|M_V^k\|_2^2 \|M_V^l\|_2^2}, l = 1, \dots, N_P, \quad (5.6)$$

where $M_V^k, k = 1, \dots, K_I$ are the masks already chosen, and l' is the new mask to be added from the mask candidate pool.

5.3 Light Field Acquisition and Reconstruction

5.3.1 Perspective Flow calculation

In order to take advantage of the proposed disparity-aware LF dictionary, a rudimentary disparity/depth information of the scene need to be calculated. In this work, we propose two methods to achieve this goal.

Disparity from Sub-Aperture Scan (SAS)

By opening only a sub-section of the lens aperture to allow light pass through, a sub-aperture image I_A can be captured; and by changing different sub-aperture locations, a sequence of images I_A^v ($v = 1, 2, \dots$) that observes different stereo angles could be captured by the camera. Since only a small part of the aperture is opened for each image, a vignetting rectification procedure needs to be carried out before they can be used for further processing.

With the assumption of intensity conservation among different LF sub-view images, applying the chain rule for differentiation, we have:

$$\frac{\partial \tilde{I}_A^v}{\partial x} \frac{\partial x}{\partial v} + \frac{\partial \tilde{I}_A^v}{\partial y} \frac{\partial y}{\partial v} + \frac{\partial \tilde{I}_A^v}{\partial v} = 0, \quad (5.7)$$

5.3. LIGHT FIELD ACQUISITION AND RECONSTRUCTION

where $\tilde{I}_A^v(x, y)$ is the vignetting rectified pixel intensity at (x, y) of the sub-aperture image with view index v . $\frac{\partial \tilde{I}_A^v}{\partial x}$ and $\frac{\partial \tilde{I}_A^v}{\partial y}$ are the image's horizontal and vertical gradient respectively; and $\frac{\partial \tilde{I}_A^v}{\partial v}$ is the gradient between the scan sequence. Using Horn's method [84], the horizontal and vertical optical flow $P_X = \frac{\partial x}{\partial v}$ and $P_Y = \frac{\partial y}{\partial v}$ could be estimated. P_X and P_Y are then combined estimate the unit disparity dp :

$$P = \text{sign}(P_X) \max(|P_X|, |P_Y|). \quad (5.8)$$

The max operator is adopted for the concern of texture-less regions. In these regions at certain directions the estimation might fail but the max operator might help to use other directions to remedy.

The perspective flow P is then discretized and mapped into a new S -bin image: $\Theta(P)$, where $\Theta(\cdot)$ denotes the mapping operation.

Based on Assumption 4.1 that pixels in a small patch area share the same disparity, we set the patch disparity to be equal to the disparity of the pixel with largest intensity gradient inside that patch. The reason being that optical flow estimation is more precise for areas with strong intensity variation.

Disparity from Normalized Fluctuation (NF)

Instead of using binary sub-aperture masks, optimized random mask patterns can also be used to calculate the disparity map. Suppose image I_M^k is the image taken with aperture modulation mask M_V^k . k is the mask index. According to Eqn. (5.3):

$$I_M^k = \Phi^k L. \quad (5.9)$$

5.3. LIGHT FIELD ACQUISITION AND RECONSTRUCTION

First, measurements (at least 2) need to be normalized by their mask's mean transmission rate:

$$\tilde{I}_M^k = \frac{I_M^k}{(\sum_{v=1}^V M_V^k(v))/V}, \quad (5.10)$$

and then still using the assumption of intensity conservation:

$$\frac{\partial \tilde{I}_M^k}{\partial x} \frac{\partial x}{\partial k} + \frac{\partial \tilde{I}_M^k}{\partial y} \frac{\partial y}{\partial k} + \frac{\partial \tilde{I}_M^k}{\partial k} = 0, \quad (5.11)$$

Horn's method [84] is again used to estimate the term $\frac{\partial x}{\partial k}$ and $\frac{\partial y}{\partial k}$ between two normalized measurements: $\tilde{I}_M^{k_1}$ and $\tilde{I}_M^{k_2}$. However $\frac{\partial x}{\partial k}$ and $\frac{\partial y}{\partial k}$ needs to be normalized again before they can be used as disparity map. This time, however, they are normalized by the modulation mask's horizontal and vertical summation deviation:

$$P_X = \frac{\partial x}{\partial k} / (\sum_v M_H(v)(M_V^{k_1}(v) - M_V^{k_2}(v))), \quad (5.12)$$

$$P_Y = \frac{\partial y}{\partial k} / (\sum_v M_V(v)(M_V^{k_1}(v) - M_V^{k_2}(v))). \quad (5.13)$$

Matrices M_H and M_V are defined in Eqn. (4.1).

We compare the two methods hereby introduced to calculate the perspective flow: Sub-Aperture Scan can obviously produce better flow quality, however it also rely heavily on the correctness of vignetting correction pattern; the normalized fluctuation approach produces slightly worse flow estimation, however the difference is still acceptable. The normalized fluctuation approach uses optimized mask patterns throughout the whole process, therefore produces higher reconstruction quality outputs. This will be validated in Section 5.5.

5.3.2 Light Field Reconstruction

To recover the light field data L from its measurements X , Eqn. (5.3) poses an inverse problem in which the number of measurements ($N^v \cdot K$) is significantly smaller than the number of unknowns ($N^v \cdot V$). Since the light field data has a strong spatio-angular correlation, it can be sparsely represented by a few signal examples [71]. We choose to implement sparse coding technique for the inverse problem. With the proposed perspective-shifted LF dictionary D_{LF} , the LF acquisition process in Eqn. (5.3) can be rewritten as:

$$X = \Phi L = \Phi D_{LF} \alpha. \quad (5.14)$$

Since D_{LF} comprises of perspective-shifted atoms at different disparities, the dictionary can be divided into multiple segments: $D_{LF} = [D_{LF}^1, D_{LF}^2, \dots, D_{LF}^S]$. The atoms that have been shifted at the same disparity are grouped as one segment. The coding coefficient α can also be divided into multiple segments, each for one dictionary segment: $\alpha = [\alpha^1, \alpha^2, \dots, \alpha^S]$. Based on our assumption that pixels in a small image patch have the same disparity, the coding coefficient vector α should have non-zero elements over only one dictionary segment.

To be specific, for coefficient segments α^i , $i = 1, 2, \dots, S$ whose indices i are different from the calculated perspective flow map: $i \neq \Theta(P)$, $\alpha^i = 0$; and for $i = \Theta(P)$, the following optimization problem is to be solved:

$$\begin{aligned} & \text{minimize } \|\alpha^{\Theta(P)}\|_0^0 \\ & \text{subject to } \|X - \Phi D_{LF}^{\Theta(P)} \alpha^{\Theta(P)}\|_2^2 \leq \varepsilon \end{aligned} \quad (5.15)$$

To solve Eqn. (5.15), greedy Orthogonal Matching Pursuit (OMP) algorithm could be applied in a similar manner when it is used in image denoising [52].

Figure 5.3: Proposed camera system for light field acquisition.

When OMP is directly applied on D_{LF} , which has an extremely large dimension $((N^v \cdot V) \times (S \cdot K^c))$, the coding process will be extremely time consuming. However due to the perspective flow detection process, the actual LF dictionary used for coding is reduced to a much lower dimension $((N^v \cdot V) \times K^c)$. This results in a much more efficient LF reconstruction algorithm both in terms of computation complexity, and reconstruction quality which we show in Section 5.5.

5.4 A Hardware Implementation

A hardware implementation of LF acquisition system based on coded aperture will be introduced in this section. A complete system schematic drawing is shown in Fig. 5.3. The system comprises the following components:

1. $2\times$ relay lens (aspherized achromatic Lens with 50mm EFL);
2. $2\times$ right angle prisms;
3. Polarizing beam-splitter (designed wavelength 420-680nm);
4. LCoS spatial modulator (1280×1024 pixels);

5.4. A HARDWARE IMPLEMENTATION

Light from the scene is first collected and converged by the front imaging lens, and then the light passes through a relay lens and becomes collimated. The polarizing beam-splitter (PBS) will bend s -polarized light towards the LCoS (Liquid Crystal on Silicon) module, and let p -polarized light pass through. The LCoS is a reflective device that can change the polarization angle of incident light beams. By setting the corresponding pixel values on LCoS, the polarization angle at that specific aperture location can be dynamically changed. When the reflected light beam passes through the PBS again, different polarization angles translate to different transmission rate. Finally the second relay lens will converge the light and focus onto the camera sensor (or DSLR viewfinder).

By changing the pattern displayed on LCoS, the system aperture can be changed. By showing only a small fraction of the whole aperture, parallax will be maximized.

For SAS, the whole scanning process will be fast. In our experimental set-up, 20ms is used for taking one such image. Theoretically, two scans will be enough for the disparity calculation, which will take only 40ms. For a better disparity estimation, more pictures can be taken which will of course take longer time. According to our experiment, five quick scans will suffice to produce a good enough perspective flow estimation, which takes around 0.1s. This sub-aperture scanning process is somewhat similar to most DSLR camera's auto focus function (which in essence is also a scanning process), and when the scanning is finished, subsequent image(s) will be taken using optimized aperture mask, and all the measurement(s) will later be used together with the disparity flow estimation for the LF reconstruction.

The hardware set up of the LF camera is shown in Fig. 5.4, which will later be used for experimental evaluation of the proposed LF acquisition and reconstruction scheme.

5.4. A HARDWARE IMPLEMENTATION

Figure 5.4: The hardware implementation for concept proof for the proposed system. Right angle prisms are currently not used in our current stage of implementation, however the concept of sub-aperture scanning and LF reconstruction is validated using this set-up.

5.5 Evaluation and Analysis

In order to evaluate the efficiency of the proposed LF acquisition and reconstruction scheme, experiments are designed and carried out both on simulated LF data, and on images acquired via the designed hardware.

Following the steps introduced in Section 4.2, we train a perspective-shifted LF dictionary D_{LF} with the following parameters:

- train the global image dictionary $D^c \in \mathbb{R}^{256 \times 400}$, (image patch size $\sqrt{N^c} \times \sqrt{N^c} = 16 \times 16$, and dictionary atom number $K^c = 400$) based on multiple natural images. The reason of setting 16×16 as the patch size is for the consideration of the shifting process later: a 16×16 image patch leaves enough margin for a smaller patch of 8×8 to shift across. The atom number 400 is chosen to guarantee the dictionary redundancy.
- shift and extract patches of size $\sqrt{N^v} \times \sqrt{N^v} = 8 \times 8$ from each atom in D^c with adjacent view disparity value dp ;
- go through all adjacent view disparity value dp in the 13×1 vector: $[-3.0, -2.5, -2.0, -1.5, -1.0, -0.5, 0, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0]$ to create a LF dictionary $D_{LF} \in \mathbb{R}^{(64 \cdot 25) \times (13 \cdot 400)} = \mathbb{R}^{1600 \times 5200}$.

5.5.1 Evaluation of the Sub-aperture Scan Process

We evaluate how different numbers of sub-aperture scan can affect the final LF reconstruction quality. The LF data *Buddha* [1] is used for the evaluation, and the results are shown in Fig. 5.5. As expected, the reconstruction quality grows with the increase of aperture scans. More scans lead to better estimation of the disparity, and therefore a more precise choice of the LF dictionary segment for each patch;

Figure 5.5: Evaluation of aperture scan number and measurement number’s effect on final reconstruction quality for SAS. The data is collected on there LF images *Buddha*, *Dragons*, and *Maria* with 5×5 views.

Also obvious from the Fig. 5.5, the reconstruction quality grows with the increase of measurement numbers. It is difficult to tell which factor has more impact on the final reconstruction quality, however it is possible to conclude: A scan number of 5 ensures a high reconstruction quality (which allows a 20 millisecond exposure time for each sub-aperture image); and that the reconstruction quality grows slow after measurement number 4.

5.5.2 Measurement Mask Pattern Evaluation

According to Section 5.2.2, a measurement mask is chosen among a pool of uniform randomly distributed values with a large neighborhood correlation. In this subsection we validate this assumption.

Experiments are carried out on LF data *Buddha*, *Dragons*, and *Maria*. 70 uniform

Figure 5.6: Evaluation of the modulation mask quality with respect to the mask elements' total neighborhood correlation defined in Eqn. (5.5) for the LF images *Buddha*, *Dragons*, and *Maria* respectively in (a), (b), and (c).

distributed random mask patterns are generated for each LF data, these masks are then used to evaluation the LF acquisition and reconstruction process. SAS method is used with 5 sub-aperture scans, and only 1 measurement is used for the reconstruction. The reconstruction performance PSNR are shown in Fig. 5.6.

As can be seen, the reconstruction quality (vertical axis) generally increases with the neighborhood correlation value (horizontal value). Therefore, this experiment validates the optimization method we used for the choosing of an efficient coding mask pattern.

5.5.3 Quantitative Evaluation of the Light Field Reconstruction

In this sub-section we carry out quantitative evaluation of the light field reconstruction algorithms herein proposed and compare them with several other state-of-the-art methods. The reconstruction methods we choose to compare are listed below. All the coding masks used are optimized using the method introduced in Section 5.2.2, and the reconstruction are all based on the proposed perspective-shifted LF dictionary

5.5. EVALUATION AND ANALYSIS

$$D_{LF} \in \mathbb{R}^{1600 \times 5200}.$$

1. Coded Aperture (CA) [5]: during LF acquisition, an optimized aperture mask is installed at the lens aperture plan, which codes each sub-aperture view with a constant rate. No disparity information will be used during reconstruction.
2. Coded Sensor-Plane (CS) [4]: during acquisition, an optimized mask is installed at the plane closely in front of the sensor, which codes each directional light ray with a different constant for each pixel. No disparity information will be used during reconstruction.
3. Proposed Sub-Aperture Scan method (SAS): sub-aperture scan is performed, during which 5 sub-aperture images are taken for the calculation of disparity information (introduced in Section 5.3.1). The disparity will be used to extract a dictionary segment for each patch.
4. Proposed Normalized Fluctuation method (NF): during acquisition, an optimized mask is installed at the lens aperture plane. The NF method introduced in 5.3.1 will be used for the disparity calculation, which will be used to choose the right dictionary disparity segment during LF reconstruction.

A complete table of reconstruction performance measured in PSNR on simulated LF data *Maria*, *Buddha* [1], and *Dragons* [4] are listed in Table 5.1. The average performance data of the three are drawn in Fig. 5.7. As can be seen from the figure, the proposed LF acquisition and reconstruction method SAS and NF shows obvious advantages over state-of-the-art methods. Several conclusions could be drawn:

- Both of the SAS and NF method herein proposed produce much higher reconstruction quality especially when measurement number is less than 4;

5.5. EVALUATION AND ANALYSIS

- The SAS method produces outstanding reconstruction quality using only one measurement (the aperture scan process can be considered as a preliminary setting-up procedure similar to that of the auto-focus detection on a conventional DSLR that leads to a final shot). The SAS produces much better reconstruction than its competing methods (4dB+) when only one measurement is taken.
- The NF method, although only working with at least two images, produces the highest reconstruction quality at all possible measurement ranges with good margin.
- Both the SAS and NF are much quicker than that of CA and CS (more than 1/5 of CA and 1/20 of CS), and demands much less working memory for computation.

To generalize, the proposed dictionary structure, and its reconstruction scheme have a great advantage over [71] in the following aspects:

1. Much less dictionary training complexity: in [71], the dictionary training process takes more than 10 hours on a 24-core processor; the proposed method is only the complexity of a normal 2D image dictionary;
2. Much faster reconstruction: As only a fraction of the dictionary will be used for the reconstruction, the calculation complexity will be dramatically reduced in the order of several minutes on a single core processor as compared to [71], where a spatial 480×270 and angular 5×5 LF reconstruction takes 18 hours on a 8-core workstation with parallel implementation;
3. Disparity will be strictly preserved during the reconstruction, which lead to higher reconstruction fidelity.

Figure 5.7: LF reconstruction performance comparison for competing methods. Both reconstruction PSNR and computation time are evaluated. The curves and bars are average of three synthetic LF data *Buddha*, *Maria* and *Dragons*.

5.5.4 Quantitative Evaluation of Disparity Restoration

The most critical information for light field data is the disparity information across all perspective views. Therefore, the restoration of this information is an important evaluation criteria for LF reconstruction.

We use the method introduced in [27] to calculate the LF disparity. In [27] a voting scheme is used for the disparity estimation. And, instead of using all disparity values for the evaluation, we only select scene points whose edge confidence value C_e is above a threshold (in our experiment, we set as 0.05). Any disparity estimation below that threshold can be considered as inaccurate [27], therefore they are excluded from this evaluation.

The results are shown in Table 5.2. This experiment validates the efficiency of the

5.5. EVALUATION AND ANALYSIS

Table 5.1: Quantitative evaluation of LF reconstruction for synthesized LF data *Maria*, *Buddha* [1], and *Dragons* [4] with different number of measurements. The four entries for each unit are (from top to bottom): (1) coded aperture method [5]; (2) coded sensor-plane method; [4]; (3) sub-aperture scan method using 5 scan images; (4) normalized fluctuation method. Best performance, i.e. highest PSNR (in dB) and shortest time, are put in bold for each unit. Compression ratio is calculated as measurement no./reconstructed angular sub-view no.

Measurement No. /Compression Ratio	1/ 0.04		2/ 0.08		3/ 0.12		4/ 0.16		5/ 0.20		6/ 0.24		7/ 0.28	
	PSNR	Time	PSNR	Time	PSNR	Time	PSNR	Time	PSNR	Time	PSNR	Time	PSNR	Time
Maria (741 × 741)	28.05	1498	30.78	1943	31.38	2219	32.19	1498	32.30	1722	32.33	2001	32.45	2236
	28.02	5106	29.95	5450	30.85	5865	31.15	6024	31.42	6144	31.54	6220	31.64	6361
	32.45	268	32.52	287	32.42	297	32.58	299	32.59	393	32.56	560	32.55	804
	/	/	34.51	272	34.70	268	34.89	278	34.86	275	34.92	278	34.93	276
Buddha (768 × 768)	29.58	1226	32.66	1219	33.21	1293	33.56	1428	33.68	1638	33.75	1922	33.79	2238
	30.04	3682	31.97	4925	32.75	4927	32.96	4597	33.17	4435	33.28	5071	33.36	4843
	33.64	217	33.65	240	33.56	251	33.61	263	33.55	360	33.53	535	33.54	722
	/	/	33.35	273	33.55	272	33.75	272	33.70	270	33.76	264	33.82	263
Dragons (593 × 840)	27.56	735	30.64	762	31.16	849	31.84	972	31.91	1168	31.95	1368	32.02	1734
	28.16	2610	29.99	3209	30.63	3136	30.85	3448	31.06	3545	31.18	3221	31.26	3382
	31.93	195	32.00	210	31.97	214	32.04	227	31.99	305	31.98	421	32.00	575
	/	/	31.84	182	32.11	169	32.47	174	32.43	177	32.55	176	32.63	172
Average	28.40	1153	31.36	1308	31.92	1454	32.53	1299	32.63	1509	32.67	1764	32.75	2069
	28.74	3799	30.64	4528	31.41	4643	31.65	4690	31.88	4708	32.00	4837	32.08	4862
	32.68	227	32.72	246	32.65	254	32.74	263	32.71	353	32.69	505	32.70	700
	/	/	33.23	242	33.45	236	33.70	241	33.66	241	33.74	240	33.79	237

5.6. CONCLUSION

Table 5.2: Quantitative evaluation of LF disparity reconstruction PSNR (in dB) for synthesized LF data *Maria*, *Buddha* [1], and *Dragons* [4] with different number of measurements.

Scan No.	2	3	4	5	6	7
Buddha	53.31	53.22	53.84	53.80	53.86	53.90
Maria	57.06	56.15	57.66	57.75	57.79	57.83
Dragons	53.61	53.51	53.98	53.92	54.03	54.06
Average	54.66	54.63	55.16	55.15	55.23	55.27

proposed method in LF disparity restoration.

5.5.5 Qualitative Evaluation of the Reconstructed Light Field

In this sub-section we give visual demonstrations of the proposed LF acquisition/reconstruction schemes, and the output of the designed LF camera.

Fig. 5.8 shows 9 sub-view images from the reconstructed 5×5 LF data using our camera. The LF images are reconstructed using only 2 measurements with the NF method. Red dotted lines are added across the views to demonstrate the disparity.

To demonstrate the reconstruction details, we zoom in to some occlusion areas of the reconstructed LF from different perspective view points. The details are shown in Fig. 5.9 and Fig. 5.10 for the reconstructed LF *Buddha* and *Dragon*.

Fig. 5.11 and Fig. 5.12 are two examples of refocusing based on the reconstructed LF data using synthesized data *Buddha*, and using our proposed camera, respectively.

5.6 Conclusion

In this chapter a compressed sensing framework is proposed for the sampling and reconstruction of a high resolution light field based on a coded aperture camera. First, a LF dictionary based on perspective-shifting is proposed for sparse representation

5.6. CONCLUSION

Figure 5.8: 9 of the reconstructed 5×5 angular views with measurements taken using the proposed LF camera. The NF method is used, 2 measurements are taken. Red dotted lines are added to demonstrate the disparity between different reconstructed views.

Figure 5.9: LF reconstruction details for *Buddha*. Areas in red rectangles are zoomed-in for better observation. Areas in blue rectangles show perspective differences from different LF views. The NF method is used, 2 measurements are taken with reconstruction PSNR 33.35.

Figure 5.10: LF reconstruction details for *Dragons*. Areas in red rectangles are zoomed-in for better observation. Areas in blue rectangles show perspective differences from different LF views. The NF method is used, 2 measurements are taken with reconstruction PSNR 31.84.

Figure 5.11: Refocusing demonstration on the reconstructed light field based on synthetic data *Buddha*. 5 sub-aperture scans and only 1 extra measurement is used for the reconstruction. The reconstruction PSNR is 33.64 dB. For better illustration of the focus change, the details in blue and red rectangles in (a), (b) are zoomed in below in (c) and (d) respectively.

Figure 5.12: Refocusing demonstration on the reconstructed light field based on measurements from our proposed LF camera. The NF method is used, and 2 measurements are taken. For better illustration of the focus change, the details in blue and red rectangles in (a), (b) are zoomed in below in (c) and (d) respectively.

5.6. CONCLUSION

of the highly correlated light field. Then, two separate methods, i.e., Sub-Aperture Scan (SAS) and Normalized Fluctuation (NF) are proposed to acquire/calculate the scene disparity, which will be used during the LF reconstruction with the proposed disparity-aware dictionary. At last, a hardware implementation of the proposed LF acquisition/reconstruction scheme is carried out. Both quantitative and qualitative evaluation show the proposed methods produce state-of-the-art performances in both reconstruction quality and computation efficiency.

Chapter 6

Light Field Data Post-Processing: Compression and Depth Estimation

6.1 Introduction

6.1.1 Light Field Compression

The light field (LF) is a function that describes the intensities of light rays in all possible propagation directions. The LF (Light field) data is extremely bulky, even for ones that are discrete in the angular dimension. Light field compression therefore is an important issue in its storage as well as transmission.

For compression, the LF is usually treated as a 3D structured data, and video compression methods could be applied by treating the angular indices as time axis in a video. In [21] the authors modified the video compression techniques to code the four-dimensional data efficiently; they also proposed disparity-compensated image

6.1. INTRODUCTION

prediction among the LF images. Other works can also be found that lie within the scope of a video codec, such as in [22], where the authors introduced the full inter-prediction scheme in HEVC into intra-prediction for the LF data. Kundu et al. [23] exploits inter-view correlation by applying homography and 2D warping for view prediction. Outside of the scheme of “disparity compensation”, sakamoto et al. [24] proposed to reconstruct the light field from multi-focus images, which contains mostly low frequency components and therefore easier to compress.

The strong correlations within a LF lies not only among the LF views, but also between the compensation elements between adjacent LF views, i.e., there exists a linear relationship between compensation elements among the LF views. This suggests that higher compression ratio and performances could be achieved if the interior structure among all the views could be exploited. We took another approach which bases on sparse representation over a learnt light field dictionary [85]. The Light Field Dictionary proposed in Section 4.2 is employed, and a sparse coding procedure based on LF disparity voting is proposed. The designed structure within the LF atoms greatly boosts its compression efficiency. Experiments and comparisons have been carried out which show that the proposed *LFD codec* produces state-of-the-art compression performances for the LF data.

6.1.2 Scene Depth Estimation Based on Light Field Data

The challenge of depth inference from light field data is similar to that of traditional stereo vision. Instead of a pair of input images, the angular space of the light field is usually densely sampled, it enables the formation of a high resolution Epipolar-Plane Image (EPI). By finding the slopes of the lines on the EPI, the depth map can be calculated.

6.1. INTRODUCTION

Wanner et al. used a structure tensor based on local gradients to estimate the direction of lines on EPI [3]. The tensor is easy and highly efficient to implement. Kim et al. [27] proposed a scoring mechanism for all the hypothetical disparities for each scene point, and choose the highest scored disparity as their estimation. This method proves to be more accurate than the previous however slightly more time and memory consuming.

The output depth map estimation is usually noisy especially for smooth and texture-less regions. Some filtering or interpolation is always needed for a satisfactory depth map output [13,25,26]. Notably, Wanner et al. [3] used a energy minimization approach combined with state-of-the-art convex optimization algorithms to calculate high quality depth maps, however the computational complexity is quite high. A “fine-to-coarse” approach was implemented in Kim’s recent work [27], where disparity estimations from lower resolution EPI are interpolated then used to estimate the missing data in the higher resolution EPI. Kim’s method produces state-of-the-art quality depth map outputs, with efficient implementation (in an order of 1 minute for a 2M pixel image).

He et al. proposed the “Guided Filter” in [28], where they use a guidance image to filter another image. Inspired by this idea, we propose an adaptive guided filtering (AGF) algorithm for disparity/depth map optimization. The center view from the light field data will be used to provide the scene object contour and texture information. The filter is able to preserve the disparity edges, smooth the regions without influence of the image texture, and reject the data entries with low estimation confidence. Experiment shows AGF is much faster in implementation as compared to other variational based algorithms, and produces competitive visual results.

6.2 *LFD Codec* Based on Sparse Coding Over A Light Field Dictionary

We propose to compress the light field using the perspective-shifted LF dictionary D_{LF} proposed in Section 4.2. The complete system diagram of the codec is shown in Fig. 6.1.

The Coder

The disparity map of the light field images P are first calculated, the output of which will then be used for sparse coding using the disparity-aware LF dictionary D_{LF} introduced in Sec. 5.2. The disparity map, alone with the sparse coding atom indices are entropy coded. The sparse coding coefficients are quantized and then coded into the output file. The entropy codebook as well as the LF dictionary D_{LF} become part of the codec.

The decoder

The decoder first decodes the sparse coding atom indices and the disparity map according to the entropy codebook, and then the LF images are reconstructed using the LF dictionary D_{LF} and the quantized sparse coding coefficients.

The two key procedures of the proposed codec are disparity calculation, and sparse coding. Each will be elaborated in the following sub-sections.

6.2.1 Disparity Map Voting

In order to fully take advantage of the LF dictionary D_{LF} that has distinctive disparity segments $[D_{LF}^1, D_{LF}^2, \dots, D_{LF}^S]$, the disparity of the LF needs to be calculated first.

6.2. *LFD CODEC* BASED ON SPARSE CODING OVER A LIGHT FIELD DICTIONARY

Figure 6.1: System illustration of the proposed *LFD codec*.

For a higher accuracy output, however at the cost of more complex computation, we adopt the voting method introduced in [27] for this task. By stacking the LF views either in horizontal or vertical direction into a 3D data structure, an EP (Epi-Polar) image $E_Y(x, v)$ can be produced by slicing the structure along the view index direction v and the spatial axis direction x , at a fixed position on the other spatial axis $y = Y$. In Chapter 2, a visual illustration of how an EP image is formed has been given. For a certain scene point at (x', Y, v_c) , and for each possible disparity value dp , its EP line comprises the following set of points on $E_Y(x, v)$:

$$\{(x, v) | x = x' + (v_c - v)dp, v = 1, \dots, N_V\}, \quad (6.1)$$

where $v_c = \frac{1+N_V}{2}$ is the central view index. The collection of all EPI pixel intensities along the line $\{(x, v)\}$, and all admissible dp variations form the set:

$$\mathfrak{R}_Y(x', v, dp) = \{E_Y(x' + (v_c - v)dp, v) | v = 1, \dots, N_V\}. \quad (6.2)$$

If one certain disparity value dp' is true for x' , with the assumption of a Lambertian surface, the intensity elements in $\mathfrak{R}_Y(v, dp')$ should be identical. A score is calculated for each admissible disparity values dp to evaluate the proximity of intensities along the EP line using Eqn. (6.3). dp' will be chosen for the (x', Y, v_c) when it produces

6.2. LFD CODEC BASED ON SPARSE CODING OVER A LIGHT FIELD DICTIONARY

the highest score.

$$S_Y(x', dp) = \frac{1}{|\mathfrak{R}|} \sum_v \left(1 - \left\| \frac{\Re_Y(x', v, dp) - E_Y(x', v_c)}{H} \right\|^2 \right), \quad (6.3)$$

$$P(x', Y) = \arg \max_{dp} S_Y(x', dp). \quad (6.4)$$

Along with $P(x', Y)$, an edge confidence $C_e(x', Y)$ can also be calculated by measuring the EPI intensity variance in a 1D local window Ω ($\Omega = 9$):

$$Ce(x', Y) = \sum_{x \in \Omega} \|E_Y(x, v_c) - E_Y(x', v_c)\|^2. \quad (6.5)$$

Repeat this process for all x 's and all Y s, a complete disparity map P and confidence map Ce will be generated for the light field.

6.2.2 Light Field Sparse Coding

The light field data has a strong spatio-angular correlation, it can be sparsely represented by a few signal examples [71] [39]. With the perspective-shifted LF dictionary D_{LF} , the light field L can be represented as:

$$L = D_{LF}\alpha. \quad (6.6)$$

As previously explained, the LF dictionary can be divided into multiple segments $D_{LF} = [D_{LF}^1, D_{LF}^2, \dots, D_{LF}^S]$: the atoms that have been shifted at the same disparity are grouped as one segment. The coding coefficient α can also be divided into multiple segments $\alpha = [\alpha^1, \alpha^2, \dots, \alpha^S]$, each corresponds to a dictionary segment. Based on our assumption that pixels in a small image patch have the same disparity, the coding

6.3. ADAPTIVE GUIDED FILTERING ALGORITHM FOR LIGHT FIELD DEPTH ESTIMATION

coefficient vector α should have non-zero elements over only one dictionary segment.

To be specific, for coefficient segments α^i , $i = 1, 2, \dots, S$ whose indices i are different from the calculated perspective flow map: $i \neq \Theta(P)$, $\alpha^i = 0$; and for $i = \Theta(P)$, the following optimization problem is to be solved:

$$\begin{aligned} & \text{minimize } \|\alpha^{\Theta(P)}\|_0^0, \\ & \text{subject to } \|L - D_{LF}^{\Theta(P)} \alpha^{\Theta(P)}\|_2^2 \leq \varepsilon. \end{aligned} \tag{6.7}$$

Here $\Theta(\cdot)$ is a linear operator that quantizes the disparity map P into indices that matches the disparities in D_{LF} . The solution to Eqn. (6.7) is largely similar to that of Eqn. (5.15). Similar coding strategy will be used here, in which the original LF dictionary $((N^v \cdot V) \times (S \cdot K^c))$ will be reduced to $((N^v \cdot V) \times K^c)$ for each disparity segment in $\Theta(P)$.

6.3 Adaptive Guided Filtering Algorithm for Light Field Depth Estimation

Based on the noisy input disparity/depth map (calculated the same way as introduced in Section 6.2.1), we aim to design a filter that can:

- When there are disparity changes, sharpen the edges based on the corresponding input guidance image edges;
- Smooth the disparity area with low variation, resilient from influences of guidance image textures;
- Remove the low confidence disparity values, and interpolate them with correct ones;

6.3. ADAPTIVE GUIDED FILTERING ALGORITHM FOR LIGHT FIELD DEPTH ESTIMATION

6.3.1 Definition of Filter Kernel

In this sub-section we elaborate the kernel of the proposed adaptive guided filter. First of all, an assumption is made for the filter:

Assumption 6.1. A small patch of the ground truth disparity map can be expressed as affine transform of its patch value in the RGB space [28]:

$$q_i = a_k I_i + b_k, \forall i \in w_k. \quad (6.8)$$

In Eqn. (6.8) a_k , b_k are affine transform coefficients for the patch centered at pixel k , both are constant in a patch window of size w_k . I_i is the input guidance RGB image, which is the central view of the light field. q_i is the output disparity value as a patch.

This linear model conforms to our design goals that depth value changes only exist where RGB intensity changes also exist (as $\nabla q_i = a_k \nabla I_i$); and for the textured object surface with no disparity variation, RGB intensity variations should not introduce any artifact in the disparity map (in which case $a_k = 0$).

In order to determine the coefficient values a_k and b_k for each image patch, the following optimization target function is used:

$$\{a_k, b_k\} = \arg \min_{a_k, b_k} \sum_k \sum_{i \in w_k} (\beta_i (a_k I_i + b_k - p_i)^2 + \varepsilon a_k^2), \quad (6.9)$$

where $\beta_i (a_k I_i + b_k - p_i)^2$ is the fidelity term, p_i is the input noisy disparity value at pixel i ; β_i is the confidence value for the pixel i taken from the confidence map Ce

6.3. ADAPTIVE GUIDED FILTERING ALGORITHM FOR LIGHT FIELD DEPTH ESTIMATION

defined in Eqn. (6.5), and it is thresholded according to:

$$\beta_i = \begin{cases} 0 & Ce(i) \leq C_{th} \\ Ce(i) & Ce(i) > C_{th} \end{cases}. \quad (6.10)$$

According to the definition of β_i , different weights will be assigned to input disparity values during the regression, based on their estimation confidence Ce . More confident values will be assigned more weight during regression, and vice versa. What's more, a lowest confidence threshold is set at C_{th} , below which the disparity value will be treated as outliers.

The second term εa_k^2 is the regularization term which prevents a_k from getting too large. It prevents the output disparity from being over adapted to the guidance image texture [28], causing false disparity variations. In our implementation, we set $\varepsilon = 10^{-7}$.

The target function (6.9) is quadratic, therefore the solution can be given by linear regression:

$$a_k = \frac{\sum_{i \in w_k} (\beta_i p_i I_i - \tilde{p}_k \beta_i I_i)}{\sum_{i \in w_k} \beta_i (I_i^2 - \tilde{I}_k I_i) + \varepsilon} \quad (6.11)$$

where $\tilde{I}_k, \tilde{p}_k$ are the confidence weighted average of the guidance image patch and disparity map patch centred at pixel k respectively.

One pixel will be included in multiple patches during regression, and the final output disparity value will be calculated as average of the filtering output from all patches the pixel has been involved with:

$$q_i = \frac{1}{|wn_i|} \sum_{k: i \in w_k} (a_k I_i + b_k), \quad (6.12)$$

6.3. ADAPTIVE GUIDED FILTERING ALGORITHM FOR LIGHT FIELD DEPTH ESTIMATION

where $|wn_i|$ is the number of patches that include the pixel i . Because of the patch size varying mechanism we are about to introduce in the next sub-section, each pixel's $|wn_i|$ value might be different.

6.3.2 The Patch Window Size Variation Mechanism

For a further improved filter performance, we vary the patch window size w_i for each pixel based on its confidence value.

$$w_i = (1 - \frac{\beta_i}{\beta_{\max}})(W_{\max} - W_{\min}) + W_{\min}, \quad (6.13)$$

where $W_{\max}$, $W_{\min}$ are the minimum and maximum patch window sizes, which are set to 3 and 11 respectively in our implementation, and $\beta_{\max} = \max_i \beta_i$.

The idea behind this mechanism is straightforward: high confident values doesn't need much filtering, a small patch window is useful for keeping its original value; however those values of lower confidence require a larger patch size and more neighbouring samples to provide disparity clues for interpolation.

An illustration of the AGF filtering process is given in Fig. 6.2. Fig. 6.2(a) is the center view of the light field image *Maria* [1] (9×9 views, each view 926×926 pixels). Fig. 6.2(b) zooms in the disparity map region in red rectangle of (a), the dark pixels are the ones with estimation confidence lower than the threshold C_{th} , and they have been removed before the disparity map is fed as input to AGF. Fig. 6.2(c) is the filter output for the zoomed in region. Fig. 6.2(d) shows the ground truth disparity map provided by [1]. We can see the edges of the disparity map is preserved according to the original guidance RGB image, the missing low confidence pixels are interpolated with filter output values. The final disparity map is visually satisfying.

We conclude this sub-section by giving an intuitive understanding of the proposed

6.3. ADAPTIVE GUIDED FILTERING ALGORITHM FOR LIGHT FIELD DEPTH ESTIMATION

Figure 6.2: AGF processing result for the light field *Maria*. (b) is the zoomed in disparity map calculated based on EPI, low confidence values are removed. (c) is the AGF output disparity map. (d) is the ground truth disparity map imported from [1].

adaptive guided filter:

Low-variance-smoothing: For pixels with smooth neighbourhood in the guidance image I_i (disparity confidence low), the regularization term εa_k^2 will force a_k to approach zero, and b_k will equal the weighted average of all the eligible disparity values within the patch, which smooths the region.

Edge-preservation: For pixels with high variation in its I_i neighbourhood (disparity confidence high), the regularization effect from εa_k^2 will diminish, disparity value will be matted according to the edges in the guidance image. Furthermore, because of the patch window size varying mechanism, the patch size will be small for these regions with high variations, thus a shaper edge is expected because of less averaging in Eqn. (6.12).

Outlier-exclusion: Because of the weighting parameter β_i in the fidelity term, questionable disparity entries will be rejected automatically ($\beta_i = 0$), and strict conformance will be given to the high confidence entries (β_i large) during regression.

6.4 Model Evaluation

6.4.1 Evaluation of the *LFD Codec*

In order to evaluate the efficiency of the proposed light field compression scheme, experiments are designed and carried out to illustrate the efficiency of the proposed codec, and comparison is drawn with other state-of-the-art codecs.

Following the steps introduced in Sec. 5.2, a perspective-shifted LF dictionary D_{LF} is prepared as following:

- A global image dictionary $D^c \in \mathbb{R}^{256 \times 400}$, (image patch size $\sqrt{N^c} \times \sqrt{N^c} = 16 \times 16$, and dictionary atom number $K^c = 400$) based on multiple natural images.
- With a given adjacent view disparity value dp' , shift and extract image patches of size $\sqrt{N^v} \times \sqrt{N^v} = 8 \times 8$ from each atom in D^c ;
- For all adjacent view disparity values in the 13×1 vector: $[-3.0, -2.5, -2.0, -1.5, -1.0, -0.5, 0, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0]$, repeat the previous step to create a LF dictionary $D_{LF} \in \mathbb{R}^{(64 \cdot 25) \times (13 \cdot 400)} = \mathbb{R}^{1600 \times 5200}$.

The LF images are decomposed into non-overlapping patches of 8×8 for sparse coding. Three bench mark LF images *Maria*, *Buddha* [1], and *Dragons* [4] are used for the evaluation. To simplify the implementation, their angular resolutions all have been reduced to 5×5 . Note that when the LF angular resolution increases, the proposed *LFD codec* can produce equivalent reconstruction PSNR, with proportionally lower bit-ratio. Since angular resolution only affects the choice of LF dictionary which belongs to the codec. The increase of angular dimension will not increase the output file size.

6.4. MODEL EVALUATION

Figure 6.3: Comparison of LF reconstruction PSNR for different transform models with different percentages of non-zero coefficients. The horizontal axis represents the percentage of non-zero coefficients, and vertical axis represents LF reconstruction PSNR. The three methods in comparison are: 2D, and 3D Wavelet Transform, and the proposed LFD compression model.

Figure 6.4: Comparison of LF reconstruction PSNR with *H.264* codec at different compression rates for different bench mark light field images. The horizontal axis represents bit-rate (in bpp), and vertical axis represents reconstruction PSNR.

Comparison with 2D and 3D Wavelet Transform with Equivalent Percentage of Non-zero Coefficients

We compare the reconstruction PSNR of the *LFD* codec with 2D and 3D Wavelet transform models. Both 2D and 3D models adopt the Wavelet filter *db1* and 5-level decomposition. Only a limited percentage of the largest transform coefficients are kept as non-zero, the rest are thresholded to be zero.

Experiments are carried out on three bench mark LF images, and the LF reconstruction PSNR for the three models are shown in Fig. 6.3. As can be seen from the

6.4. MODEL EVALUATION

Figure 6.5: Comparison of LF disparity reconstruction PSNR between *H.264* codec with proposed *LFD* codec at different bit-rates. The horizontal axis represents bit-rate (in bpp), and vertical axis represents LF disparity reconstruction PSNR.

figure, the proposed *LFD* codec produces higher quality restorations with equivalent percentage of non-zero coefficients.

Comparison with *H.264* Codec

We compare the LF reconstruction PSNR of the proposed *LFD* codec with *H.264* codec at different compression rates. Experiments are carried out on three benchmark LF images. The results are shown in Fig. 6.4. As can be seen from the figure, the proposed *LFD* codec outperforms *H.264* codec at equivalent compression ratio.

Evaluation on Disparity Information Preservation

The most critical information for light field data is the disparity information across all perspective views. Therefore, the preservation of this information is an important evaluation criterion for all LF codecs.

Instead of using all disparity values for the evaluation, we only select scene points whose edge confidence value is above a threshold $Ce > 0.05$ (Ce defined in Eqn. (6.5), out of intensity scale 255). Any disparity estimation below that threshold can be considered as inaccurate [27], therefore they are excluded from this evaluation.

Figure 6.6: AGF processing result for the light field *Maria* and *MonasRoom*. (a) the input images, (b) the confidence thresholded disparity maps, (c) the AGF processing results.

The results are shown in Fig. 6.5. As can be seen from the figure, the compression ratio does not play an important role in the preservation of disparity information. However the proposed *LFD codec* outperforms the *H.264 codec*.

6.4.2 Evaluation of the AGF Model for Estimation of Scene Depth

We first conduct experiment on two sets of light field images *Maria* (9×9 , 926×926 pixels each view), and *MonasRoom* (9×9 views, each view 768×768 pixels) [1]. Fig. 6.6 shows the result. The first column are the center view of the input light field images. Second column are the disparity maps that were fed into the adaptive guided filter, the pixels below the confidence threshold C_{th} have been taken out. The last row show the filtering result. We can see the interpolated areas are smooth, and the

Figure 6.7: AGF processing result for the light field *Buddha*. (a) is input image, (b) is the disparity map calculated based on EPI. (c) is the processing result using GCL proposed in [3]. (d) is our result.

edges are well preserved.

Comparison has been made between our algorithm with the Globally Consistent Labeling (GCL) method proposed in [3] on the light field image *Buddha* (9×9 views, with each view 768×768 pixels). Our output is visually competitive, while GCL reported a computation time of around 10 minutes, the proposed filtering algorithm can finish within 1 second.

6.5 Conclusion

In this Chapter, light field data post-processing techniques are investigated. First, a light field compression codec (*LFD Codec*) is proposed. The codec is based on a disparity-aware light field dictionary introduced in Section 4.2. LF disparity voting information are involved in the sparse coding procedure. Experiments and comparisons have been carried out which shows that the proposed *LFD codec* produces state-of-the-art compression performances for the LF data both in terms of scene reconstruction and disparity preservation qualities.

Then, an adaptive guided filtering (AGF) algorithm is proposed for the refinement of noisy disparity map calculated based on EPI images of the light field data. A guidance image is used to provide the image contour and texture information. The filter is able to preserve the disparity edges, smooth the regions without influence of the image texture, and reject the data entries with low confidence during coefficients regression. Experiment shows AGF is much faster in implementation as compared to other variational or hierarchical based optimization algorithms (at least 100 times faster than the GCL [3]), and produces competitive visual results.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

Structure exists in all forms of visual data in the nature. In this thesis, we focus on modeling these data structures using sparse representation techniques over a redundant dictionary, which has been proven as an efficient tool in numerous visual signal processing applications. We propose several types of redundant dictionaries with specially designed structures that adapts to the unique application scenarios. Different sparse coding and dictionary training strategies are investigated. Considerations are given to challenging issues such as multi-scale dictionary cross-scale interactions, dictionary disparity segment de-correlation, perspective-shifted dictionary sparse coding acceleration etc. All efforts aim to provide a more powerful frame for representing complicated visual data structures. A detailed description of each separate work will be give as follows.

In Chapter 3, a multi-scale sparse representation scheme is proposed to explore the potential of cross-scale sparsity in application of visual signal processing. A multi-

7.1. CONCLUSION

scale dictionary (MD) structure is designed. A Cross-scale Matching Pursuit (CMP) algorithm is proposed for multi-scale sparse coding. Two dictionary learning methods: Cross-scale Cooperative Learning MD/CCL), and Cross-scale Atom Clustering (MD/CAC) are proposed with each focusing on one of the two important attributes of an efficient multi-scale dictionary: the similarity, and uniqueness of corresponding atoms in different scales. We analyze and compare their different advantages in the application of image denoising under different noise levels, where both methods produce state-of-the-art denoising results.

In Chapter 4, a light field dictionary (LFD) based perspective shifting of central view atom is proposed for sparse representation of the high dimensional light field data. A simple light field compressed sensing framework based on aperture coding is introduced to evaluate the efficiency of the proposed LFD. The extreme dimensionality of the LFD poses great challenge for light field reconstruction. A two-stage coding algorithm is proposed which uses the Winner-Take-All (WTA) hashing strategy to narrow down the search range for light field sparse coding. the algorithm proves to be able to increase the coding efficiency by almost three times and keeping the reconstruction quality almost the same with original OMP coding.

In Chapter 5, a compressed sensing framework is proposed for the sampling and reconstruction of a high resolution light field based on a coded aperture camera. First, a LF dictionary based on perspective-shifting is proposed for sparse representation of the highly correlated light field. Then, two separate methods, i.e., Sub-Aperture Scan (SAS) and Normalized Fluctuation (NF) are proposed to acquire/calculate the scene disparity, which will be used during the LF reconstruction with the proposed disparity-aware dictionary. At last, a hardware implementation of the proposed LF acquisition/reconstruction scheme is carried out. Both quantitative and qualitative

7.2. FUTURE WORK

evaluation show the proposed methods produce state-of-the-art performances in both reconstruction quality and computation efficiency.

In Chapter 6, light field data post-processing techniques are investigated. First, a light field compression codec (*LFD Codec*) is proposed. The codec is based on a disparity-aware light field dictionary introduced in Section 4.2. LF disparity voting information are involved in the sparse coding procedure. Experiments and comparisons have been carried out which shows that the proposed *LFD codec* produces state-of-the-art compression performances for the LF data both in terms of scene reconstruction and disparity preservation qualities.

Then, still in Chapter 6, an adaptive guided filtering (AGF) algorithm is proposed for the refinement of noisy disparity map calculated based on EPI images of the light field data. A guidance image is used to provide the image contour and texture information, the filter is able to preserve the disparity edges, smooth the regions without influence of the image texture, and reject the data entries with low confidence during coefficients regression. Experiment shows AGF is much faster in implementation as compared to other variational or hierarchical based optimization algorithms (at least 100 times faster than the GCL [3]), and produces competitive visual results.

7.2 Future Work

7.2.1 Toward A More Flexible Multi-scale Structure

The Multi-scale dictionary herein proposed in this thesis has quite a few limitations in its structures and training methodologies that could be improved for a even better cross-scale interaction and dictionary performances.

Based on the multi-scale dictionary training method introduced in Chapter 3,

7.2. FUTURE WORK

the dictionary scale number is fixed before it is actually trained with input data samples, this makes it lacks scale adaptivity for different images with different cross-scale features. Some images with its energy mostly centralized in a narrow frequency band might be better to use MDs with smaller scale numbers; and vice versa, for dictionaries with a more spread frequency spectra, a larger scale number is required. Therefore, an adaptive scheme is required along the learning process of the dictionary.

As concluded in Chapter 3, a multi-scale structure without sufficient cross-scale interaction during dictionary learning would lead to a deteriorated dictionary performance. However does insufficient or unnecessary dictionary scale could lead to deteriorated dictionary performance, this needs to be further investigated.

The multi-scale structure is beneficial not only in applications such as image denoising. More could be done such as in object recognition, image classification, where a multi-scale could be beneficial obviously. Such possibility could be investigated further in the future.

7.2.2 Light Field Capturing and Data Processing

The proposed LF dictionary in Chapter 4 is based on perspective shifting of the centre view, therefore it has limitations in describing irregular patterns with occlusions and specularities. However, since the LF is reconstructed from over-lapping patches, over-lapped areas are actually average of different patches with different disparities, some irregular patterns are able to be reconstructed. However, this remains as limitation of the proposed method. More research be done on this direction. To be specific, a coupled dictionary that combines dictionaries of regular light field patterns and specular patterns could be investigated.

Both SAS and NF method proposed in Chapter 5, have their limited performance

7.2. FUTURE WORK

with dynamic scenes, since the disparity information is sensitive to scene content change. This limitation can be alleviated by shortening the exposure time for each scan, however this deteriorates the output image quality.

The main limitations of the current compressed sensing cameras are the degree of freedoms in designing the individual elements of the sensing matrix Φ . Due to the physical implementation restrains (Φ has to be the concatenation of multiple constant diagonal matrices in Section 4.2), the current forms of sensing matrices are still far from a powerful frame for the capture of the light field. This requires more hardware innovations. New designs in CMOS sensors and camera structures that can resolve such complications of sensing matrix optimization are possible future research directions in compressed light field sensing.

Publication List

Journal Publications

1. Jie Chen, and Lap-Pui Chau, "A Rain Pixel Recovery Algorithm for Videos With Highly Dynamic Scenes," IEEE Transactions on Image Processing, vol. 23, no. 3, 2014.
2. Jie Chen, and Lap-Pui Chau, "Multi-Scale Dictionary Learning via Cross-Scale Cooperative Learning and Atom Clustering for Visual Signal Processing," IEEE Transactions on Circuits and Systems for Video Technology, vol. 25, no. 9, 2015.
3. J. Chen; L. P. Chau, "Light Field Compressed Sensing Over A Disparity-Aware Dictionary," IEEE Transactions on Circuits and Systems for Video Technology, vol.PP, no.99, 2015
4. Jie Chen, Junhui Hou, and Lap-Pui Chau, "Light Field Compression based on A Perspective-Shifted Dictionary and 3D DCT Transform," under review.

Conference Publications

1. Jie Chen, and Lap-Pui Chau. "Heavy Haze Removal in A Learning Framework," IEEE International Symposium on Circuits and Systems, 2015.

7.2. FUTURE WORK

2. Jie Chen, and Lap-Pui Chau. "A Novel Adaptive Histogram Adjustment Method for Heavy Haze Removal in A Learning Framework," ITS Asia Pacific Forum, 2015.
3. Jie Chen, and Lap-Pui Chau. "A Fast Adaptive Guided Filtering Algorithm for Light Field Depth Interpolation," IEEE International Symposium on Circuits and Systems, pp. 2281-2284, 2014.
4. Jie Chen, Alexander Matyasko, and Lap-Pui Chau. "A Light Field Sparse Representation Structure and Its Fast Coding Technique," IEEE International Conference on Digital Signal Processing, pp. 214-218, 2014.
5. Cheen-Hau Tan, Jie Chen, and Lap-Pui Chau. "Dynamic scene rain removal for moving cameras," IEEE International Conference on Digital Signal Processing, pp. 372-376, 2014.
6. Jie Chen, and Lap-Pui Chau. "Rain removal from dynamic scene based on motion segmentation," IEEE International Symposium on Circuits and Systems, pp. 2139-2142, 2013.
7. Jie Chen, and Lap-Pui Chau. "An Enhanced Window-Variant Dark Channel Prior for Depth Estimation Using Single Foggy Image," IEEE International Conference on Image Processing, pp. 3508-3512, 2013.
8. Junhui Hou, Lap-Pui Chau, Ying He, Jie Chen, and Nadia Magnenat-Thalmann. "Human motion capture data recovery via trajectory-based sparse representation." IEEE International Conference on Image Processing, pp. 709-713. 2013.

Bibliography

- [1] S. Wanner, S. Meister, and B. Goldluecke, “Datasets and benchmarks for densely sampled 4d light fields,” in *Vision, Modeling & Visualization*. The Eurographics Association, 2013.
- [2] J. Mairal, G. Sapiro, and M. Elad, “Multiscale sparse image representation with learned dictionaries,” in *IEEE International Conference on Image Processing*, vol. 3, 2007.
- [3] S. Wanner and B. Goldluecke, “Globally consistent depth labeling of 4d light fields,” in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012.
- [4] K. Marwah, G. Wetzstein, Y. Bando, and R. Raskar, “Compressive light field photography using overcomplete dictionaries and optimized projections,” *ACM Transactions on Graphics*, vol. 32, no. 4, p. 46, 2013.
- [5] A. Ashok and M. A. Neifeld, “Compressive light field imaging,” in *Proc. SPIE*, 2010.
- [6] B. Ophir, M. Lustig, and M. Elad, “Multi-scale dictionary learning using wavelets,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, 2011.

BIBLIOGRAPHY

- [7] R. Yan, L. Shao, and Y. Liu, “Nonlocal hierarchical dictionary learning using wavelets for image denoising,” *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4689–4698, Dec 2013.
- [8] K. Skretting and K. Engan, “Image compression using learned dictionaries by rls-dla and compared with k-svd,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2011.
- [9] J. Hughes, D. Rockmore, and Y. Wang, “Bayesian learning of sparse multiscale image representations,” *IEEE Transactions on Image Processing*, vol. 22, no. 12, 2013.
- [10] J. Mairal, G. Sapiro, and M. Elad, “Learning multiscale sparse representations for image and video restoration,” DTIC Document, Tech. Rep., 2007.
- [11] F. E. Ives, “Parallax stereogram and process of making same.” Apr. 14 1903, uS Patent 725,567.
- [12] G. Lippmann, “La photographie intégrale,” *Academie des Sciences*, vol. 146, pp. 446–451, 1908.
- [13] E. H. Adelson and J. Y. Wang, “Single lens stereo with a plenoptic camera,” *IEEE transactions on pattern analysis and machine intelligence*, 1992.
- [14] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan, “Light field photography with a hand-held plenoptic camera,” *Computer Science Technical Report CSTR*, 2005.
- [15] A. Manakov, J. Restrepo, O. Klehm, R. Hegedus, E. Eisemann, H.-P. Seidel, and I. Ihrke, “A reconfigurable camera add-on for high dynamic range, multispectral,

BIBLIOGRAPHY

- polarization, and light-field imaging,” *ACM Transactions on Graphics*, vol. 32, no. 4, pp. 47–1, 2013.
- [16] T. Georgiev and C. Intwala, “Light field camera design for integral view photography,” *Adobe System, Inc*, 2006.
- [17] C.-K. Liang, T.-H. Lin, B.-Y. Wong, C. Liu, and H. H. Chen, “Programmable aperture photography: multiplexed light field acquisition,” *ACM Transactions on Graphics (TOG)*, vol. 27, no. 3, p. 55, 2008.
- [18] S. D. Babacan, R. Ansorge, M. Luessi, P. R. Matarán, R. Molina, and A. K. Katsaggelos, “Compressive light field sensing,” *IEEE Transactions on Image Processing*, vol. 21, no. 12, 2012.
- [19] H. Nagahara, C. Zhou, T. Watanabe, H. Ishiguro, and S. K. Nayar, “Programmable aperture camera using lcos,” in *Computer Vision–ECCV 2010*. Springer, 2010.
- [20] M. Hosseini Kamal, M. Golbabaee, and P. Vandergheynst, “Light field compressive sensing in camera arrays,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*. Ieee, 2012, pp. 5413–5416.
- [21] M. Magnor and B. Girod, “Data compression for light-field rendering,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 3, pp. 338–343, 2000.
- [22] Y. Li, M. Sjostrom, R. Olsson, and U. Jennehag, “Efficient intra prediction scheme for light field image compression,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2014, pp. 539–543.

BIBLIOGRAPHY

- [23] T. Sakamoto, K. Kodama, and T. Hamamoto, “A study on efficient compression of multi-focus images for dense light-field reconstruction,” in *IEEE Visual Communications and Image Processing*. IEEE, 2012, pp. 1–6.
- [24] —, “A novel scheme for 4-d light-field compression based on 3-d representation by multi-focus images,” in *IEEE International Conference on Image Processing*. IEEE, 2012, pp. 2901–2904.
- [25] T. Stich, A. Tevs, and M. A. Magnor, “Global depth from epipolar volumes—a general framework for reconstructing non-lambertian surfaces.” in *3DPVT*. Cite-seer, 2006.
- [26] T. E. Bishop and P. Favaro, “Full-resolution depth map estimation from an aliased plenoptic light field,” in *ACCV*. Springer, 2011.
- [27] C. Kim, H. Zimmer, Y. Pritch, A. Sorkine-Hornung, and M. Gross, “Scene reconstruction from high spatio-angular resolution light fields,” *To appear ACM Transactions on Graphics. (Proc. SIGGRAPH)*, 2013.
- [28] K. He, J. Sun, and X. Tang, “Guided image filtering,” in *ECCV 2010*. Springer, 2010.
- [29] S. G. Mallat, *A wavelet tour of signal processing: the Sparse way*. Amsterdam ; Boston : Elsevier /Academic Press., 2009.
- [30] A. Boggess and F. J. Narcowich, *A first course in wavelets with Fourier analysis*. Upper Saddle River, N.J. : Pearson Education, 2001.
- [31] I. Daubechies, *Ten Lectures on Wavelets*. Society for Industrial and Applied Mathematics, 1992.

BIBLIOGRAPHY

- [32] A. Pizurica and W. Philips, “Estimating the probability of the presence of a signal of interest in multiresolution single- and multiband image denoising,” *IEEE Transactions on Image Processing*, vol. 15, no. 3, 2006.
- [33] A. Pizurica, W. Philips, I. Lemahieu, and M. Acheroy, “A joint inter- and intra-scale statistical model for bayesian wavelet based image denoising,” *IEEE Transactions on Image Processing*, vol. 11, no. 5, 2002.
- [34] L. Sendur and I. Selesnick, “Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency,” *IEEE Transactions on Signal Processing*, vol. 50, no. 11, 2002.
- [35] E. Candès and D. Donoho, “Ridgelets: A key to higher-dimensional intermittency?” *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 357, no. 1760, 1999.
- [36] J.-L. Starck, E. Candes, and D. Donoho, “The curvelet transform for image denoising,” *IEEE Transactions on Image Processing*, vol. 11, no. 6, 2002.
- [37] E. Le Pennec and S. Mallat, “Sparse geometric image representations with bandelets,” *IEEE Transactions on Image Processing*, vol. 14, no. 4, 2005.
- [38] V. Chappelier and C. Guillemot, “Oriented wavelet transform for image compression and denoising,” *IEEE Transactions on Image Processing*, vol. 15, no. 10, 2006.
- [39] M. Elad, “Sparse and redundant representation modeling—what next?” *IEEE Signal Processing Letters*, vol. 19, no. 12, pp. 922–928, 2012.

BIBLIOGRAPHY

- [40] E. Candes, J. Romberg, and T. Tao, “Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information,” *IEEE Transactions on Information Theory*, vol. 52, no. 2, 2006.
- [41] Y. Nesterov, A. Nemirovskii, and Y. Ye, *Interior-point polynomial algorithms in convex programming*. SIAM, 1994, vol. 13.
- [42] J. A. Nelder and R. Mead, “A simplex method for function minimization,” *The computer journal*, vol. 7, no. 4, pp. 308–313, 1965.
- [43] L. T. Watson and R. T. Haftka, “Modern homotopy methods in optimization,” *Computer Methods in Applied Mechanics and Engineering*, vol. 74, no. 3, pp. 289–305, 1989.
- [44] T. E. Bishop and P. Favaro, “The light field camera: Extended depth of field, aliasing, and superresolution,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 5, 2012.
- [45] M. Aharon, M. Elad, and A. Bruckstein, “K-svd: An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Transactions on Signal Processing*, vol. 54, no. 11, 2006.
- [46] S. S. Chen, D. L. Donoho, and M. A. Saunders, “Atomic decomposition by basis pursuit,” *SIAM journal on scientific computing*, vol. 20, no. 1, pp. 33–61, 1998.
- [47] R. Rubinstein, A. M. Bruckstein, and M. Elad, “Dictionaries for sparse representation modeling,” *IEEE Proceedings: Special Issue on Applications of Sparse Representation and Compressive Sensing*, vol. 98, no. 6, 2010.

BIBLIOGRAPHY

- [48] J. Portilla, V. Strela, M. Wainwright, and E. Simoncelli, “Image denoising using scale mixtures of gaussians in the wavelet domain,” *IEEE Transactions on Image Processing*, vol. 12, no. 11, 2003.
- [49] G. Peyre, “A review of adaptive image representations,” *Selected Topics in Signal Processing, IEEE Journal of*, vol. 5, no. 5, 2011.
- [50] M. Elad, R. Goldenberg, and R. Kimmel, “Low bit-rate compression of facial images,” *IEEE Transactions on Image Processing*, vol. 16, no. 9, 2007.
- [51] J. Hou, L. Chau, Y. He, D. T. P. Quynh, and N. Magnenat-Thalmann, “Dynamic 3-d facial compression using low rank and sparse decomposition,” in *ACM SIGGRAPH Asia, Singapore*, 2012.
- [52] M. Elad and M. Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” *IEEE Transactions on Image Processing*, vol. 15, no. 12, 2006.
- [53] H. Kwon, Y.-W. Tai, and S. Lin, “Data-driven depth map refinement via multi-scale sparse representation,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 159–167.
- [54] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang, “Coupled dictionary training for image super-resolution,” *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3467–3478, 2012.
- [55] J. Hou, L. Chau, Y. He, and N. Magnenat-Thalmann, “Human motion capture data recovery via trajectory-based sparse representation,” in *IEEE International Conference on Image Processing*, 2013.

BIBLIOGRAPHY

- [56] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, “Robust face recognition via sparse representation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.
- [57] Y. Wu, B. Ma, M. Yang, Y. Jia, and J. Zhang, “Metric learning based structural appearance model for robust visual tracking,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 5, 2014.
- [58] C. H. Chan and J. Kittler, “Sparse representation of (multiscale) histograms for face recognition robust to registration and illumination problems,” in *IEEE International Conference on Image Processing*,. IEEE, 2010, pp. 2441–2444.
- [59] F. Zhu and L. Shao, “Weakly-supervised cross-domain dictionary learning for visual recognition,” *International Journal of Computer Vision*, vol. 109, no. 1-2, pp. 42–59, 2014.
- [60] Q. Liu, J. Luo, S. Wang, M. Xiao, and M. Ye, “An augmented lagrangian multi-scale dictionary learning algorithm,” *EURASIP Journal on Advances in Signal Processing*, vol. 2011, no. 1, 2011.
- [61] M. Aharon and M. Elad, “Sparse and redundant modeling of image content using an image-signature-dictionary,” *SIAM Journal on Imaging Sciences*, vol. 1, no. 3, 2008.
- [62] Y. Pati, R. Rezaifar, and P. Krishnaprasad, “Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition,” in *Asilomar Conference on Signals, Systems and Computers*, nov 1993.
- [63] D. P. Bertsekas, “Nonlinear programming,” 1999.
- [64] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.

BIBLIOGRAPHY

- [65] N. Kulkarni, P. Nagesh, R. Gowda, and B. Li, “Understanding compressive sensing and sparse representation-based super-resolution,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 5, pp. 778–789, 2012.
- [66] X. Lu, Y. Yuan, and P. Yan, “Image super-resolution via double sparsity regularized manifold learning,” *IEEE transactions on circuits and systems for video technology*, vol. 23, no. 12, 2013.
- [67] D. G. Dansereau, O. Pizarro, and S. B. Williams, “Linear volumetric focus for light field cameras,” *ACM Transactions on Graphics*, vol. 34, no. 2, p. 15, 2015.
- [68] T. E. Bishop and P. Favaro, “Plenoptic depth estimation from multiple aliased views,” in *ICCV Workshops*. IEEE, 2009, pp. 1622–1629.
- [69] S.-C. Chan, K.-T. Ng, Z.-F. Gan, K.-L. Chan, and H.-Y. Shum, “The plenoptic video,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 12, pp. 1650–1659, 2005.
- [70] K. Venkataraman, D. Lelescu, J. Duparré, A. McMahon, G. Molina, P. Chatterjee, R. Mullis, and S. Nayar, “Picam: an ultra-thin high performance monolithic camera array,” *ACM Transactions on Graphics*, vol. 32, no. 6, 2013.
- [71] K. Marwah, G. Wetzstein, A. Veeraraghavan, and R. Raskar, “Compressive light field photography,” in *ACM SIGGRAPH 2012 Posters*. ACM, 2012.
- [72] J. Chen, A. Matyasko, and L.-P. Chau, “A light field sparse representation structure and its fast coding technique,” in *International Conference on Digital Signal Processing*. IEEE, 2014, pp. 214–218.
- [73] A. Rajaraman and J. D. Ullman, *Mining of massive datasets*. Cambridge University Press, 2012.

BIBLIOGRAPHY

- [74] J. Yagnik, D. StreLOW, D. A. Ross, and R.-s. Lin, “The power of comparative reasoning,” in *IEEE International Conference on Computer Vision*, 2011.
- [75] D. Tschopp and S. Diggavi, “Approximate nearest neighbor search through comparisons,” *arXiv preprint: 0909.2194*, 2009.
- [76] Y.-P. Wang, L.-C. Wang, D.-H. Kong, and B.-C. Yin, “High-resolution light field capture with coded aperture,” *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5609–5618, 2015.
- [77] T. E. Bishop, S. Zanetti, and P. Favaro, “Light field superresolution,” in *IEEE International Conference on Computational Photography*. IEEE, 2009, pp. 1–9.
- [78] X. Lin, Y. Liu, J. Wu, and Q. Dai, “Spatial-spectral encoded compressive hyperspectral imaging,” *ACM Transactions on Graphics*, vol. 33, no. 6, p. 233, 2014.
- [79] X. Lin, J. Suo, G. Wetzstein, Q. Dai, and R. Raskar, “Coded focal stack photography,” in *IEEE International Conference on Computational Photography*. IEEE, 2013, pp. 1–9.
- [80] L. Shi, H. Hassanieh, A. Davis, D. Katabi, and F. Durand, “Light field reconstruction using sparsity in the continuous fourier domain,” *ACM Transactions on Graphics*, vol. 34, no. 1, p. 12, 2014.
- [81] S. F. Ray, *Applied photographic optics: Lenses and optical systems for photography, film, video, electronic and digital imaging*. Focal Press, 2002.
- [82] S. Lyu, “Estimating vignetting function from a single image for image authentication,” in *Proceedings of the 12th ACM workshop on Multimedia and security*. ACM, 2010, pp. 3–12.

BIBLIOGRAPHY

- [83] J. M. Duarte-Carvajalino and G. Sapiro, “Learning to sense sparse signals: Simultaneous sensing matrix and sparsifying dictionary optimization,” *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1395–1408, 2009.
- [84] B. K. Horn and B. G. Schunck, “Determining optical flow,” *Artificial Intelligence*, 1981.
- [85] O. Bryt and M. Elad, “Compression of facial images using the k-svd algorithm,” *Journal of Visual Communication and Image Representation*, vol. 19, no. 4, pp. 270–282, 2008.