

การสร้างคุณภาพสี่หน้าคนความละเอียดสูงยิ่งด้วยการแยกค่าเอกฐานอันดับสูง



นาย กฤษฎา อัครสกุลเกียรติ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรดุษฎีบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2553

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

COLOR FACE SUPER-RESOLUTION RECONSTRUCTION WITH HIGHER-ORDER
SINGULAR VALUE DECOMPOSITION



Mr. Krissada Asavaskulkeit

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

A Dissertation Submitted in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy Program in Electrical Engineering
Department of Electrical Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2010
Copyright of Chulalongkorn University

Thesis Title **COLOR FACE SUPER-RESOLUTION RECONSTRUCTION WITH
HIGHER-ORDER SINGULAR VALUE DECOMPOSITION**

By **Mr. Krissada Asavaskulkeit**


Field of Study **Electrical Engineering**

Thesis Advisor **Associate Professor Somchai Jitapunkul, Dr.Ing.**

Accepted by the Faculty of Engineering, Chulalongkorn University in Partial Fulfillment
of the Requirements for the Doctoral Degree

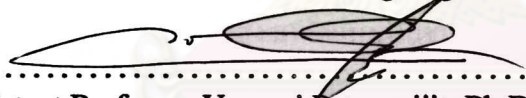
.....  **Dean of the Faculty of Engineering**
(Associate Professor Boonsom Lerdhirunwong, Dr.Ing.)


THESIS COMMITTEE

.....  **Chairman**
(Associate Professor Chedsada Chirungrueng, Ph.D.)

.....  **Thesis Advisor**
(Associate Professor Somchai Jitapunkul, Dr.Ing.)

.....  **Examiner**
(Assistant Professor Nisachon Tangsangiumvisai, Ph.D.)

.....  **External Examiner**
(Assistant Professor Vorapoj Patanavijit, Ph.D.)

.....  **External Examiner**
(Associate Professor Somchart Chokchaitam, Dr.Eng.)

ศูนย์วิจัยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

กฤษฎา อัสวสกุลเกียรติ : การสร้างคืนภาพสีหน้าคนความละเอียดสูงยิ่งด้วยการแยกค่า
 เอกฐานอันดับสูง. (COLOR FACE SUPER-RESOLUTION RECONSTRUCTION
 WITH HIGHER-ORDER SINGULAR VALUE DECOMPOSITION) อ. ที่ปรึกษา
 วิทยานิพนธ์หลัก : รศ. ดร. สมชาย จิตะพันธ์กุล, 100 หน้า.

วิทยานิพนธ์ฉบับนี้เสนอการสร้างคืนเทนเซอร์ภาพสีหน้าคนความละเอียดสูงยิ่งด้วย
 การแยกค่าเอกฐานอันดับสูงบนพื้นฐานของระบบสี่ทั้ง 4 แบบได้แก่ ระบบสี RGB , ระบบสี
 YCbCr , ระบบสีHSV และระบบสีCIELAB จำนวน 2 วิธี วิธีแรกนั้นอาศัยแบบจำลองถดถอย
 ร่วมกับ MPCA เนื่องจากภาพสีหน้าคนจะมีลักษณะที่เหมาะสมกับข้อมูลในปริภูมิเทนเซอร์

เราพบว่าในวิธีดั้งเดิมที่เคยมีการนำเสนอ นั้น ไม่ได้คำนึงถึงความสัมพันธ์ของข้อมูลใน
 แต่ละช่องสัญญาณสี ดังนั้น กระบวนการสร้างคืนภาพมีค่าความผิดพลาดเกิดขึ้น ซึ่งถ้าอาศัยวิธี
 ที่เราได้นำเสนอในวิทยานิพนธ์ฉบับนี้ในแง่ของการ พิจารณาผลกระทบของค่าลักษณะเฉพาะ ,
 สัญญาณรบกวน , จำนวนภาพเทรนนิ่งและความซับซ้อนของอัลกอริทึม พบว่าเราจะสามารถ
 สร้างภาพคืนความละเอียดสูงยิ่งได้ใกล้เคียงกับภาพต้นฉบับ

ในส่วนของวิธีที่สอง ที่เราจะนำเสนอ นั้น เราจะนำ ภาพหน้าคนที่เป็นภาพสี คู่ความ
 ละเอียดต่ำและความละเอียดสูง มาตัดแบ่งเป็นชิ้นย่อยๆ โดยจะพิจารณาในปริภูมิเทนเซอร์ด้วย
 การแยกค่าเอกฐานอันดับสูง ในกระบวนการเทรนนิ่ง จากนั้นกระบวนการสร้างคืนภาพสีหน้า
 คนจะแบ่งเป็นสองขั้นตอนได้แก่ การสร้างภาพคืนในส่วนของโครงสร้างหลักของใบหน้าและ
 ขั้นตอนถัดมาคือการสร้างคืนในส่วนของโครงสร้างที่เป็นชิ้นย่อยๆเฉพาะที่

เราสามารถยืนยันได้จาก ผลการทดลองบนฐานข้อมูลมาตรฐานด้วยภาพสีหน้าคน
 พบว่าวิธีการที่เรานำเสนอขึ้นมาสามารถสร้างคืนภาพได้อย่างมีประสิทธิภาพเพิ่มขึ้นจากวิธี
 ดั้งเดิม แต่การที่จะต้องนำภาพมาตัดออกเป็นชิ้นย่อยๆ ในกระบวนการเทรนนิ่งจะส่งผลให้เกิด
 กระบวนการคำนวณที่ซับซ้อนมากกว่ากระบวนการที่เรานำเสนอในวิธีแรก

ภาควิชา.....วิศวกรรมไฟฟ้า.....ลายมือชื่อนิสิต.....
 สาขาวิชา.....วิศวกรรมไฟฟ้า.....ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก.....
 ปีการศึกษา.....2553.....

##4871881021: MAJOR ELECTRICAL ENGINEERING
KEYWORDS : MPCA / COLOR / HALLUCINATION

KRISSADA ASAVASKULKEIT: COLOR FACE SUPER-RESOLUTION RECONSTRUCTION WITH HIGHER-ORDER SINGULAR VALUE DECOMPOSITION.
THESIS ADVISOR: ASSOC. PROF. SOMCHAI JITAPUNKUL, Dr.Ing., 100 pp.

This dissertation proposes two novel frameworks for color face super-resolution reconstruction with higher-order singular value decomposition in four basic color systems such as RGB , YCbCr , HSV and CIELAB color system. The first framework is based on the linear regression model with MPCA since a color face image can be naturally described as tensors or multi-linear arrays. We find that the traditional method does not consider the correlation of data in each color channel. Therefore, there is an error in the face reconstruction process. In this dissertation, we investigate the performance of our proposed method in sense of effect of number of eigenvalue, effect of noise and complexity respectively and we can reconstruct the reasonable color face images which are compared with the ground truth color face images. In the second framework, we decompose each pair of low and high resolution training face images into a small patches and apply higher-order singular value decomposition in a tensor space. In color face reconstruction process, there are two steps : the first step tends to reconstruct a global face. Next step, the local detail is hallucinated from small overlapped patches. The experimental results from standard color facial database show that our second proposed framework can effectively reconstruct the color face images than the previous method. However, decomposing small patches in the training process will result in a more complicated process than that of the first framework.

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

Department : Electrical Engineering
Field of Study : Electrical Engineering
Academic Year : 2010

Student's Signature
Advisor's Signature

Kri ssada Asavaskulkeit
Somchai Jitapunkul

Acknowledgements

The work contained in this dissertation represents the accumulation of three years of my work made possible only by the collectively support of family, friends, colleagues and mentors. First and foremost, I would like to express my deep gratitude to my advisor Assoc. Dr. Somchai Jitapunkul. It was a great privilege and honor to work and study under his guidance. I would like to thank him for his supports, friendship, empathy and great vision.

My thanks also go to the members of my dissertation committees, Assoc. Dr. Chedsada Chinrungrueng, Assist Dr. Nisachon Tangsangiumvisai, Assoc. Dr. Somchart Chokchaitam and Assist Dr. Vorapoj Patanavijit. I am grateful to thank the AUN/SEED-Net which is mainly supported by the Japanese Government through the Japan International Cooperation Agency (JICA), and partially supported by the ASEAN Foundation. Many thanks to my all colleagues at DSPRL (Digital Signal Processing Research Laboratory) for academic documentary help, technical/ theory information and programming/ data information. Finally, my life has been constantly fulfilled by love and support of my family.

I am extremely grateful to my parents, Piya Asavaskulkeit and Boonsri Asavaskulkeit, for their love, caring and sacrifices, my father and mother for educating and preparing me for my future life since I was a kid. This work is dedicated to them.

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

Contents

	Page
Abstract (Thai)	iv
Abstract (English)	v
Acknowledgements	vi
Contents	vii
List of Tables	ix
List of Figures	xi
Chapter	
I Introduction	1
1.1 Background and Signification of the Research Problems	1
1.2 Literature Review	2
1.3 Objectives	6
1.4 Scope	6
1.5 Expected Prospects	6
1.6 Research Procedure	7
II Basic Background and Related Topics	8
2.1 Principal Component Analysis (PCA) Subspace Analysis	8
2.2 Subspace Learning based on Tensor Analysis	9
2.2.1 Tensor PCA	9
2.3 Multilinear Principal Component Analysis (MPCA)	11
2.3.1 Multilinear Projection of Tensor Objects	12
2.3.2 MPCA Algorithm	12
2.3.3 Full Projection	13
2.3.4 MPCA Versus PCA and 2-D PCA Solutions	15
2.4 TensorFaces: Multilinear Analysis of Facial Images	15
2.5 Color Image Processing	17
2.5.1 Mathematical Definition of Color Matching	18
2.5.2 Mathematics of Color Reproduction	19
2.5.3 Additive Color Systems	20
2.5.4 Subtractive Color Systems	21
2.5.5 Color Spaces	22
2.5.6 Uniform Color Spaces	23
III The Proposed Frameworks	25
3.1 Color Face Hallucination with Linear Regression Model in MPCA	25
3.2 Color Face Super-Resolution Based on Tensor Patch Method	27

Chapter	Page
3.2.0.1 Global Low-Resolution Color Face Image Synthesis	29
3.2.0.2 Local Patch-Based and High-Frequency Residue Recovery in Color Face Image Hallucination	29
IV The Experimental Results	32
4.1 Image Databases	32
4.1.1 FERET Database	32
4.2 Preprocessing	32
4.2.1 Cropping and Resizing	33
4.2.2 Normalization	33
4.3 Experiments and Analysis on Color Face hallucination with TensorPCA subspace	34
4.4 Experiments and Analysis on Color Face hallucination with Linear Regression Model in MPCA	39
4.4.1 Impact of Number of Eigenvalues	39
4.4.2 Impact of training set size	47
4.4.3 Robustness to Noise	51
4.4.4 Complexity	60
4.4.5 Partially Occluded Color Face Images	61
4.5 Experiments and Analysis on Color Face hallucination with Tensor Patch	64
4.5.1 Complexity	75
V Conclusions	77
5.1 Conclusions of The Dissertation	77
5.2 Future Directions	77
References	78
Appendices	84
Appendix A	85
Appendix B	86
Appendix C	99
Vitae	100

List of Tables

Table	Page
4.1 PSNR results of the facial images with traditional PCA and tensorPCA methods in Fig. 4.4 for RGB color model.	37
4.2 PSNR results of the facial images with traditional PCA and tensorPCA methods in Fig. 4.5 for YCbCr color model.	37
4.3 PSNR results of the facial images with traditional PCA and tensorPCA methods in Fig. 4.6 for HSV color model.	37
4.4 PSNR results of the facial images with traditional PCA and tensorPCA methods in Fig. 4.7 for CIELAB color model.	38
4.5 PSNR results of the facial images with traditional PCA and MPCA methods in Fig. 4.8 for RGB color model.	47
4.6 PSNR results of the facial images with traditional PCA and MPCA methods in Fig. 4.9 for YCbCr color model.	47
4.7 PSNR results of the facial images with traditional PCA and MPCA methods in Fig. 4.10 for HSV color model.	48
4.8 PSNR results of the facial images with traditional PCA and MPCA methods in Fig. 4.11 for CIELAB color model.	48
4.9 PSNR results of the facial images with tensorPCA and MPCA methods in Fig. 4.16 for RGB color model.	48
4.10 PSNR results of the facial images with tensorPCA and MPCA methods in Fig. 4.17 for YCbCr color model.	49
4.11 PSNR results of the facial images with tensorPCA and MPCA methods in Fig. 4.18 for HSV color model.	49
4.12 PSNR results of the facial images with tensorPCA and MPCA methods in Fig. 4.19 for CIELAB color model.	49
4.13 comparison of complexity (execution time) with training set size 60 images	60
4.14 comparison of complexity (execution time) with training set size 120 images	60
4.15 comparison of complexity (execution time) with training set size 240 images	61
4.16 comparison of complexity (execution time) with training set size 500 images	61
4.17 PSNR results of the facial images with tensor patches method in Fig. 4.43 - 4.58 for RGB color model.	72
4.18 PSNR results of the facial images with tensor patches method in Fig. 4.43 - 4.58 for YCbCr color model.	73
4.19 PSNR results of the facial images with tensor patches method in Fig. 4.43 - 4.58 for HSV color model.	74
4.20 PSNR results of the facial images with tensor patches method in Fig. 4.43 - 4.58 for CIELAB color model.	74
4.21 comparison of complexity (execution time) between tensor patch method and MPCA method with training set size 60 images	75
4.22 comparison of complexity (execution time) between tensor patch method and MPCA method with training set size 120 images	75
4.23 comparison of complexity (execution time) between tensor patch method and MPCA method with training set size 240 images	76

Table	Page
4.24 comparison of complexity (execution time) between tensor patch method and MPCA method with training set size 500 images	76
1 PSNR results of Fig. 2 with a linear regression model in MPCA method	86
2 PSNR results of Fig. 3 with a linear regression model in MPCA method	87
3 PSNR results of Fig. 4 with a linear regression model in MPCA method	87
4 PSNR results of Fig. 5 with a linear regression model in MPCA method	88
5 PSNR results of Fig. 6 with a linear regression model in tensorPCA method	88
6 PSNR results of Fig. 7 with a linear regression model in tensorPCA method	89
7 PSNR results of Fig. 8 with a linear regression model in tensorPCA method	89
8 PSNR results of Fig. 9 with a linear regression model in tensorPCA method	90



List of Figures

Figure	Page
2.1 The visual illustration of the 1-mode unfolding of a third-order tensor \mathcal{A} to matrix $\mathbf{A}^{(1)}$	13
2.2 Visual illustration of multilinear projection: (a) projection in the 1-mode vector space and (b) 2-mode and 3-mode vectors.	14
2.3 Visual illustration of (a) total scatter tensor, (b) 1-mode eigenvalues, (c) 2-mode eigenvalues, and (d) 3-mode eigenvalues.	15
2.4 Some of the basis vectors resulting from the multilinear analysis of the facial image data tensor \mathcal{D} [1,2].	16
2.5 Cone sensitivities.	19
2.6 CIE RGB and XYZ color matching functions: RGB are shown in dashed lines, and XYZ are shown in solid lines.	20
2.7 Additive color system.	21
2.8 Subtractive color system.	23
3.1 block diagram of color face hallucination with the linear regression model in MPCA	27
4.1 Preprocessing diagram	32
4.2 Cropping image	33
4.3 Example training faces from FERET database in our proposed algorithm	33
4.4 Color hallucinated face images in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent traditional PCA; (d) face hallucination result with 95 percent traditional PCA; (e) face hallucination result with 100 percent traditional PCA; (f) face hallucination result with 90 percent tensorPCA; (g) face hallucination result with 95 percent tensorPCA; (h) face hallucination result with 100 percent tensorPCA;	35
4.5 Color hallucinated face images in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent traditional PCA; (d) face hallucination result with 95 percent traditional PCA; (e) face hallucination result with 100 percent traditional PCA; (f) face hallucination result with 90 percent tensorPCA; (g) face hallucination result with 95 percent tensorPCA; (h) face hallucination result with 100 percent tensorPCA;	35
4.6 Color hallucinated face images in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent traditional PCA; (d) face hallucination result with 95 percent traditional PCA; (e) face hallucination result with 100 percent traditional PCA; (f) face hallucination result with 90 percent tensorPCA; (g) face hallucination result with 95 percent tensorPCA; (h) face hallucination result with 100 percent tensorPCA;	36

Figure	Page
4.7 Color hallucinated face images in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent traditional PCA; (d) face hallucination result with 95 percent traditional PCA; (e) face hallucination result with 100 percent traditional PCA; (f) face hallucination result with 90 percent tensorPCA; (g) face hallucination result with 95 percent tensorPCA; (h) face hallucination result with 100 percent tensorPCA;	36
4.8 Color hallucinated face images in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent traditional PCA; (d) face hallucination result with 95 percent traditional PCA; (e) face hallucination result with 100 percent traditional PCA; (f) face hallucination result with 90 percent MPCA; (g) face hallucination result with 95 percent MPCA; (h) face hallucination result with 100 percent MPCA;	41
4.9 Color hallucinated face images in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent traditional PCA; (d) face hallucination result with 95 percent traditional PCA; (e) face hallucination result with 100 percent traditional PCA; (f) face hallucination result with 90 percent MPCA; (g) face hallucination result with 95 percent MPCA; (h) face hallucination result with 100 percent MPCA;	41
4.10 Color hallucinated face images in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent traditional PCA; (d) face hallucination result with 95 percent traditional PCA; (e) face hallucination result with 100 percent traditional PCA; (f) face hallucination result with 90 percent MPCA; (g) face hallucination result with 95 percent MPCA; (h) face hallucination result with 100 percent MPCA;	42
4.11 Color hallucinated face images in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent traditional PCA; (d) face hallucination result with 95 percent traditional PCA; (e) face hallucination result with 100 percent traditional PCA; (f) face hallucination result with 90 percent MPCA; (g) face hallucination result with 95 percent MPCA; (h) face hallucination result with 100 percent MPCA;	42
4.12 Color hallucinated face images, compared with other traditional methods in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; . .	43

Figure	Page
4.13 Color hallucinated face images, compared with other traditional methods in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; . . .	43
4.14 Color hallucinated face images, compared with other traditional methods in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; . . .	44
4.15 Color hallucinated face images, compared with other traditional methods in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; . . .	44
4.16 Color hallucinated face images, compared between TensorPCA and MPCA in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent tensorPCA; (d) face hallucination result with 95 percent tensorPCA; (e) face hallucination result with 100 percent tensorPCA; (f) face hallucination result with 90 percent MPCA; (g) face hallucination result with 95 percent MPCA; (h) face hallucination result with 100 percent MPCA;	45
4.17 Color hallucinated face images, compared between TensorPCA and MPCA in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent tensorPCA; (d) face hallucination result with 95 percent tensorPCA; (e) face hallucination result with 100 percent tensorPCA; (f) face hallucination result with 90 percent MPCA; (g) face hallucination result with 95 percent MPCA; (h) face hallucination result with 100 percent MPCA;	45
4.18 Color hallucinated face images, compared between TensorPCA and MPCA in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent tensorPCA; (d) face hallucination result with 95 percent tensorPCA; (e) face hallucination result with 100 percent tensorPCA; (f) face hallucination result with 90 percent MPCA; (g) face hallucination result with 95 percent MPCA; (h) face hallucination result with 100 percent MPCA;	46

Figure	Page
4.19 Color hallucinated face images, compared between TensorPCA and MPCA in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent tensorPCA; (d) face hallucination result with 95 percent tensorPCA; (e) face hallucination result with 100 percent tensorPCA; (f) face hallucination result with 90 percent MPCA; (g) face hallucination result with 95 percent MPCA; (h) face hallucination result with 100 percent MPCA;	46
4.20 Color hallucinated face images with training set size 60 images in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	50
4.21 Color hallucinated face images with training set size 120 images in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	50
4.22 Color hallucinated face images with training set size 240 images in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	51
4.23 Color hallucinated face images with noise variance 10^{-6} in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	52

Figure	Page
4.24 Color hallucinated face images with noise variance 2.5×10^{-5} in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-5} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	52
4.25 Color hallucinated face images with noise variance 2.5×10^{-4} in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-4} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	53
4.26 Color hallucinated face images with noise variance 10^{-3} in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-3} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	53
4.27 Color hallucinated face images with noise variance 10^{-6} in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	54
4.28 Color hallucinated face images with noise variance 2.5×10^{-5} in YCBCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	54

Figure	Page
4.29 Color hallucinated face images with noise variance 2.5×10^{-4} in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-4} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	55
4.30 Color hallucinated face images with noise variance 10^{-3} in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-3} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	55
4.31 Color hallucinated face images with noise variance 10^{-6} in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	56
4.32 Color hallucinated face images with noise variance 2.5×10^{-5} in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-5} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	56
4.33 Color hallucinated face images with noise variance 2.5×10^{-4} in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-4} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	57

Figure	Page
4.34 Color hallucinated face images with noise variance 10^{-3} in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-3} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	57
4.35 Color hallucinated face images with noise variance 10^{-6} in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	58
4.36 Color hallucinated face images with noise variance 2.5×10^{-5} in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	58
4.37 Color hallucinated face images with noise variance 2.5×10^{-4} in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-4} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	59
4.38 Color hallucinated face images with noise variance 10^{-3} in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-3} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	59

Figure	Page
4.39 Partially occluded face hallucination results in RGB color model. (a) original HR images; (b) input LR images with noise, motion, blur and left eye occluded; (c) face hallucination result in RGB color model with 100 percent traditional PCA; (d) face hallucination result in RGB color model with bilinear method; (e) face hallucination result in RGB color model with 95 percent MPCA;	62
4.40 Partially occluded face hallucination results in YCbCr color model. (a) original HR images; (b) input LR images with noise, motion, blur and left eye occluded; (c) face hallucination result in YCbCr model system with 100 percent traditional PCA; (d) face hallucination result in YCbCr model system with bilinear method; (e) face hallucination result in YCbCr model system with 95 percent MPCA; . . .	62
4.41 Partially occluded face hallucination results in HSV color model. (a) original HR images; (b) input LR images with noise, motion, blur and left eye occluded in HSV color model; (c) face hallucination result in HSV color model with 100 percent traditional PCA; (d) face hallucination result in HSV color model with bilinear method; (e) face hallucination result in HSV color model with 95 percent MPCA;	63
4.42 Partially occluded face hallucination results in CIELAB color model. (a) original HR images; (b) input LR images with noise, motion, blur and left eye occluded in CIELAB color model; (c) face hallucination result in CIELAB color model with 100 percent traditional PCA; (d) face hallucination result in CIELAB color model with bilinear method; (e) face hallucination result in CIELAB color model with 95 percent MPCA;	63
4.43 Color hallucinated face images in RGB color model. (a) original HR images (30 × 30); (b) input LR images (15 × 15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (5 × 5) patches method and 95 percent MPCA;	65
4.44 Color hallucinated face images in YCbCr color model. (a) original HR images (30 × 30); (b) input LR images (15 × 15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (5 × 5) patches method and 95 percent MPCA;	65

Figure	Page
4.45 Color hallucinated face images in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (5×5) patches method and 95 percent MPCA;	66
4.46 Color hallucinated face images in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (5×5) patches method and 95 percent MPCA;	66
4.47 Color hallucinated face images in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (5×5) patches method and 100 percent MPCA;	67
4.48 Color hallucinated face images in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (5×5) patches method and 100 percent MPCA;	67
4.49 Color hallucinated face images in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (5×5) patches method and 100 percent MPCA;	68
4.50 Color hallucinated face images in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (5×5) patches method and 95 percent MPCA;	68

Figure	Page
4.51 Color hallucinated face images in RGB color model. (a) original HR images (30 × 30); (b) input LR images (15 × 15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (10 × 10) patches method and 95 percent MPCA;	69
4.52 Color hallucinated face images in YCbCr color model. (a) original HR images (30 × 30); (b) input LR images (15 × 15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (10 × 10) patches method and 95 percent MPCA;	69
4.53 Color hallucinated face images in HSV color model. (a) original HR images (30 × 30); (b) input LR images (15 × 15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (10 × 10) patches method and 95 percent MPCA;	70
4.54 Color hallucinated face images in CIELAB color model. (a) original HR images (30 × 30); (b) input LR images (15 × 15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (10 × 10) patches method and 95 percent MPCA;	70
4.55 Color hallucinated face images in RGB color model. (a) original HR images (30 × 30); (b) input LR images (15 × 15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (10 × 10) patches method and 100 percent MPCA;	71
4.56 Color hallucinated face images in YCbCr color model. (a) original HR images (30 × 30); (b) input LR images (15 × 15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (10 × 10) patches method and 100 percent MPCA;	71

Figure	Page
4.57 Color hallucinated face images in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (10×10) patches method and 100 percent MPCA;	72
4.58 Color hallucinated face images in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (10×10) patches method and 100 percent MPCA;	73
1 Some original high-resolution color face images (30×30) for testing.	90
2 Some of experimental results (RGB color model) with a linear regression model in MPCA method. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	91
3 Some of experimental results (YCbCr color model) with a linear regression model in MPCA method. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	92
4 Some of experimental results (HSV color model) with a linear regression model in MPCA method. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	93

Figure	Page	
5	Some of experimental results (CIELAB color model) with a linear regression model in MPCA method. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;	94
6	Some of experimental results (RGB color model) with a linear regression model in tensorPCA method. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent tensorPCA; (d) different image of face hallucination result with 90 percent tensorPCA; (e) face hallucination result with 95 percent tensorPCA; (f) different image of face hallucination result with 95 percent tensorPCA; (g) face hallucination result with 100 percent tensorPCA; (h) different image of face hallucination result with 100 percent tensorPCA;	95
7	Some of experimental results (YCbCr color model) with a linear regression model in tensorPCA method. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent tensorPCA; (d) different image of face hallucination result with 90 percent tensorPCA; (e) face hallucination result with 95 percent tensorPCA; (f) different image of face hallucination result with 95 percent tensorPCA; (g) face hallucination result with 100 percent tensorPCA; (h) different image of face hallucination result with 100 percent tensorPCA;	96
8	Some of experimental results (HSV color model) with a linear regression model in tensorPCA method. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent tensorPCA; (d) different image of face hallucination result with 90 percent tensorPCA; (e) face hallucination result with 95 percent tensorPCA; (f) different image of face hallucination result with 95 percent tensorPCA; (g) face hallucination result with 100 percent tensorPCA; (h) different image of face hallucination result with 100 percent tensorPCA;	97
9	Some of experimental results (CIELAB color model) with a linear regression model in tensorPCA method. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent tensorPCA; (d) different image of face hallucination result with 90 percent tensorPCA; (e) face hallucination result with 95 percent tensorPCA; (f) different image of face hallucination result with 95 percent tensorPCA; (g) face hallucination result with 100 percent tensorPCA; (h) different image of face hallucination result with 100 percent tensorPCA;	98

CHAPTER I

INTRODUCTION

1.1 Background and Signification of the Research Problems

The term super-resolution (SRR) refers to the process of obtaining higher-resolution (HR) images from several lower-resolution (LR) ones, i.e. resolution enhancement. The quality improvement is caused by fractional-pixel displacements between images. Super-resolution allows to overcome the limitations of the imaging system (resolving limit of the sensors) without the need for additional hardware. The reconstruction attempts to take advantage of the additional spatio-temporal data available in the sequence of images portraying the same scene. The fundamental problem addressed in super-resolution is a typical example of an inverse problem, wherein multiple low-resolution (LR) images are used to solve for the original high-resolution (HR) image.

Hallucination or reconstruction is a super-resolution algorithm that uses a different kind of constraint, in addition to the reconstruction constraint. This algorithm attempts to recognize local features in the low-resolution images and then enhances their resolution in an appropriate manner. Moreover, face hallucination is still a very active field of research and challenging because people are so familiar with the face. A small error, e.g. an asymmetry of the eyes, might be significant to human perception, whereas for super resolution of generic images the errors in textured regions, e.g. leaves, are often overlooked. It can be widely applied in many fields ranging from image compression to face identification. Especially in video surveillance, a higher resolution face image with detailed facial features will be obviously significant to raise the systems performance.

Face hallucination with the reconstruction-based methods which try to model the process of image formulation to build the relationship between LR and HR based on reconstruction constraints and smoothness constraints, is quite limited by the number of input LR and usually cannot work well in single-image super-resolution problem. Then, the face hallucination with learning-based methods becomes very popular. These methods use some training set directly or indirectly to reconstruct unknown HR images but a major problem of these methods is the high computation requirement due to complex learning process.

In our frameworks, we concentrate in the color image processing which differs from grayscale image processing because of the redundancy and the complementary information within the color bands. The processing is much more complicated due to the increased dimensionality of the problem and exchanges information from and among all bands. Almost all super-resolution methods to date have been designed to increase the resolution of a single channel (monochromatic) image. A related problem such as color super-resolution (SR), ad-

dresses fusing a set of previously demosaiced color low-resolution frames to enhance their spatial resolution. To date, there is very little work addressing the problem of color SR. The typical solution involves applying monochromatic SR algorithms to each of the color channels independently, while using the color information to improve the accuracy of the motion estimation. Another approach is transforming the problem to a different color space, where chrominance layers are separated from luminance, and SR is applied only to the luminance channel. Both of these methods are suboptimal as they do not fully exploit the correlation across the color bands.

Real data of natural and social sciences is often very high-dimensional especially the color face images. They can be naturally described as tensors or multilinear arrays. In the most of previous works on face representation, the face is represented as a vector in high-dimensional space. However, an image is intrinsically a matrix, or the second order tensor. In vector representation, the face image has to be converted to a vector. A typical way to do this so-called matrix-to-vector alignment is to concatenate all the rows in the matrix together to get a single vector. To acquire such linear transformation, traditional subspace learning methods, Principal Component Analysis (PCA) need to eigen-decomposition of some matrices. Moreover, the learning parameters in PCA is very large. These methods might not acquire good performance when the number of training samples is small. Recently, multilinear algebra, the algebra of higher-order tensors, is applied for analyzing the multifactor structure of image ensembles. Tensorface which is a novel face representation algorithm represents the set of face images by a higher-order tensor and extends traditional PCA to higher-order tensor decomposition [1, 2]. Then, we can apply multilinear principal component analysis (MPCA) to face hallucination. The MPCA performs feature extraction by determining a multilinear projection that captures most of the original tensorial input variation. In this way, the multiple factors related to expression, illumination and pose can be separated from different dimensions of the tensor. In addition, when the MPCA is implemented in the color space RGB, YCbCr, HSV and CIELAB, it can be investigated that there is a correlation between each color channel.

1.2 Literature Review

The super-resolution restoration idea was first presented by Huang et al. [3] in 1984. It is the process of combining multiple low-resolution images to form a higher resolution one. Numerous super-resolution algorithms have been proposed in the literature [4–8]. Most try to produce a super-resolution image from a sequence of low-resolution images [9, 10]. Based on the definition of SRR, the relevant research papers, published in the conferences and journals are comprehensively reviewed and are broadly categorized into two classes [11]. Specifically, the categorization is into the classes of reconstruction-based SRR algorithm and recognition-based SRR algorithm (or hallucination).

This reconstruction-based SRR algorithm does not require images for training there-

fore this algorithm does not depend on observed images but reconstruction-based approach inherits limitations when magnification factor increases. The frequency domain approach is a part of the reconstruction-based method. It makes explicit use of the aliasing that exists in each LR image to reconstruct an HR image [12, 13]. Although the frequency domain methods are intuitively simple and computationally cheap, the observation model is restricted to only global translational motion and blur. Due to the lack of data correlation in the frequency domain, it is also difficult to apply the spatial domain a priori knowledge for regularization. Next, the projection onto convex sets (POCS) approach is proposed to describe an alternative iterative approach and it can also incorporate prior knowledge about the solution into the reconstruction process. With the estimates of registration parameters, this algorithm simultaneously solves the restoration and interpolation problem to estimate the SR image [14]. Since the SRR algorithm is an ill-posed problem from an insufficient number of LR images and ill-conditioned blur operators, the regularized ML approach, called regularization, is proposed to stabilize the inversion of ill-posed problem. The last approach of reconstruction based is a nonuniform interpolation approach. In addition, it is the most intuitive method for SR image reconstruction and a fast super-resolution reconstruction based on a non-uniform interpolation using a frequency domain registration is proposed by Vandewalle et al [15–17].

In recognition-based SRR algorithm (or hallucination), this algorithm require images for training therefore this algorithm depend on observed images but this algorithm have high performance when magnification factor increases [18]. With statistical approach, Baker and Kanade [19] proposed another super-resolution algorithm (hallucination or recognition-based super-resolution) that attempts to recognize local features in the low-resolution image and then enhances their resolution in an appropriate manner. Due to the training database, therefore, this algorithm performance depends on the image type (such as face or character) and this algorithm is not robust enough to be sued in typical surveillance video.

For face identification, especially by human, it is desirable to render a high-resolution face image from the low-resolution one. This technique is called face hallucination or face super-resolution [19]. They infer the high frequency components from a parent structure by recognizing the local features from the training set, but there exists some noise in certain area. The simplest way to increase image resolution is a direct interpolation of input images with such algorithms as nearest neighbor or cubic spline. However, the performance of direct interpolation is usually poor since no new information is added in the process. Some other approaches [6, 20–26] are based on learning from the training set containing high and low-resolution image pairs, with the assumption that high-resolution images are Markov random fields (MRFs) [20, 21, 27]. These methods are more suitable for synthesizing local texture, and are usually applied to generic images without special consideration of the property of face images. Baker and Kanade [19, 28] developed a hallucination method based on the property of face images. Abandoning the MRFs assumption, it infers the high-frequency components from a parent structure by recognizing the local features from the training set. Liu et al. [29] developed a two-step statistical modeling approach integrating global and lo-

cal parameter models. Both methods rely on explicit resolution-reduction-function, which is sometimes unavailable in practice. Wang and Tang develop an efficient face hallucination algorithm using an eigentransformation algorithm [30, 31]. However, the method only utilizes global information without paying attention to local details. Inspired by a well-known manifold learning method, locally linear embedding (LLE), Chang et al. [32] develop the Neighbor Embedding algorithm based on the assumption that the local distribution structure in sample space is preserved in the down-sampling process, where the structure is encoded by patch-reconstruction weights.

To go beyond the current super-resolution techniques which only consider face images under fixed imaging conditions in terms of pose, expression and illumination, these factors are crucial to face analysis and synthesis. Recently, Vasilescu et al. introduce multilinear analysis to face modelling [1, 2] and demonstrate its promising application in computer vision. In the method, equipped with tensor algebra, the multiple factors are unified in the same framework with the coordination between factors expressed in an elegant tensor product form. Wu et al. propose a novel regression model to use tensor principal component analysis (PCA) subspace as the face representation [33], which is a special case of the concurrent subspace analysis and Ayan et al. introduce a learning-based method for super-resolution of face that uses kernel principal component analysis (PCA) for deriving prior knowledge about the face class [34]. In addition, Takahiro et al. present a kernel PCA-based adaptive resolution enhancement method of still images. The proposed method introduces two novel approaches into the kernel PCA-based reconstruction of high frequency components missed from a high-resolution (HR) image [35]. Jia et al. propose a multimodal tensor model for face super-resolution with nonlinear deformations and choose a global image-based tensor to perform synthesis across different facial modalities, and a local patch-based multiresolution tensor for hallucination [36–38]. Ma et al. present a simple and efficient multiview face hallucination (MFH) method to generate high-resolution (HR) multiview faces from a single given low-resolution (LR) one [39].

The problem of decomposing tensors (also called n-way arrays or multidimensional arrays) is approached in a variety of ways by extending the Singular Value Decomposition (SVD), principal components analysis (PCA), and other methods to higher orders; see, e.g., [40–46]. Multilinear analysis is a general extension of traditional linear methods such as PCA or matrix SVD and Lathauwer et al. propose a multilinear generalization of the symmetric eigenvalue decomposition for pair-wise symmetric tensors and investigate how tensor symmetries affect the decomposition [47]. Kolda et al. explore the orthogonal decomposition of tensors (also known as multidimensional arrays or n-way arrays) using two different definitions of orthogonality and present numerous examples to illustrate the difficulties in understanding such decompositions. For example, color images are often stored as a sequence of RGB triplets, i.e., as separate red, green and blue overlays [48].

Note that almost all super-resolution methods to date have been designed to increase the resolution of a single channel (monochromatic) image. A related problem, color SR,

addresses fusing a set of previously demosaiced color LR frames to enhance their spatial resolution. To date, there is very little work addressing the problem of color SR. The typical solution involves applying monochromatic SR algorithms to each of the color channels independently [49, 50], while using the color information to improve the accuracy of motion estimation. Another approach is transforming the problem to a different color space, where chrominance layers are separated from luminance, and SR is applied only to the luminance channel [51]. Both of these methods are suboptimal as they do not fully exploit the correlation across the color bands.

In color image super-resolution, Patil et al. propose efficient registration and wavelet based interpolation technique to yield a color super resolved image from four low resolution color images [52]. Therefore, this technique is efficient and computationally fast having clear perspective of real time implementation. The new algorithm in adaptive color super-resolution reconstruction, robust M-estimation is proposed [53]. Using a robust error norm in the objective function, and adapting the estimation process to each of the low-resolution frames, the proposed method effectively suppresses the outliers due to violations of the assumed observation model, and results in color super-resolution estimates with crisp details and no color artifacts, without the use of regularization.

Because abstractdigital color cameras sample the continuous color spectrum using three or more filters therefore, each pixel represents a sample of only one of the color bands. This arrangement is called a mosaic. To produce a full-resolution color image, the recorded image must be processed to estimate the values of the pixels for all the other color bands. This restoration process is often called demosaicing. Ron et al. proposes method involves two successive steps for color super-resolution with CCD sensors [54]. His technique is to let the edges support the color information, and the color channels support the edges, and thereby achieve better perceptual results than those that are bounded by the sampling theoretical limit. Next, Trussell et al. uses stacked notation to represent the mosaiced image capture and derives the minimum mean square error (MMSE) estimator for the demosaiced image [55]. Farsui et al. propose a fast and robust hybrid method of super-resolution and demosaicing, based on a maximum a posteriori estimation technique by minimizing a multiterm cost function [56]. They used L_1 norm for measuring the difference between the projected estimate of the high-resolution image and each low-resolution image, removing outliers in the data and errors due to possibly inaccurate motion estimation. Bilateral regularization is used for spatially regularizing the luminance component, resulting in sharp edges and forcing interpolation along the edges and not across them. Moreover, an additional regularization term is used to force similar edge location and orientation in different color channels.

In super-resolution of color video Sequences, Nimish et al. propose a new multiframe algorithm to enhance the spatial resolution of frames in video sequences and this technique specifically accounts for the possibility that motion estimation will be inaccurate and compensates for these inaccuracies [57]. In [58], an iterative algorithm for enhancing the reso-

lution of monochrome and color image sequences is proposed and two sets of experiments are presented. First, several different experiments using the same motion estimator but three different data fusion approaches to merge the individual motion fields were performed. Second, estimated high-resolution images using the block matching estimator were compared to those obtained by employing a recursive scheme.

In this dissertation, two frameworks are applied to improve the performance of color face hallucination. In the first framework, we employ multilinear principal component analysis (MPCA) in linear regression model for face reconstruction. In the training set, we compute the MPCA subspace projections for both the HR images and the LR images. Next, the color testing image is hallucinated by back-projection in subspace process.

The second frameworks are the combination LR and HR images in a unified tensor which can be reduced to two parts: a global image-based tensor and a local patch-based multiresolution tensor for incorporating high-resolution image details.

1.3 Objectives

Propose a novel face super-resolution (hallucination) with higher-order tensors for color image. The higher-order tensor can be suitable for color face images. For this reason, it can be overcome curse of dimensionality and it also preserves the significant information when the images are in a feature subspace. The two frameworks of color face super-resolution reconstruction with MPCA are proposed to increase the resolution of hallucination performance. In addition, the complexity in hallucination process can be reduced from our proposed method.

1.4 Scope

1. Develop the face hallucination technique with a linear regression model in MPCA that can reconstruct the reasonable color facial image.
 - (a) Only the color face image of full frontal view faces will be presented to the PCA in each color channel.
 - (b) Only the color face image of full frontal view faces will be presented to the linear regression model with MPCA.
 - (c) The color face image of full frontal view faces with partially occluded will be presented to the linear regression model with MPCA.

1.5 Expected Prospects

1. Acquire a basic knowledge of principal component analysis (PCA) for applying to face hallucination.

2. Obtain the new PCA analysis techniques.
3. Obtain new color face hallucination systems.
4. Publish the international journal or conference papers.
5. Know the advantages and disadvantages of using the proposed MPCA techniques in color face hallucination.
6. Understand the necessity of the MPCA techniques for color face hallucination.

1.6 Research Procedure

1. Study previous research papers relevant to the research works of the dissertation.
2. Develop the novel hallucination techniques.
3. Develop simulation programs.
4. Test the proposed algorithms by using standard face databases such as FERET.
5. Perform the proposed algorithm on a color facial database.
6. Collect and analyze computational results obtained from simulation programs.
7. Summarize the major findings as we found in step 6 and conclude the performance of the proposed framework in all concerned aspects.
8. Publish the international journal or conference papers.
9. Check whether the conclusions meet all the objectives of the research work of the dissertation.
10. Write the dissertation.

CHAPTER II

BASIC BACKGROUND AND RELATED TOPICS

In this chapter, the fundamental knowledge of the two-dimensional subspace analysis algorithm is described. First of all, the traditional 1D subspace is represented in vector form. Next, subspace analysis with tensor PCA and the Multilinear Principal Component Analysis (MPCA) is introduced. Finally, we review the basic knowledge of color system in image processing.

2.1 Principal Component Analysis (PCA) Subspace Analysis

Linear dimensionality reduction techniques have been widely used in pattern recognition and computer vision, such as face hallucination, image retrieval, etc. Principal Components Analysis (PCA) is the one of unsupervised subspace method, which is used to reduce multidimensional data sets to lower dimensions for analysis. Let \mathbf{A} be the m by n matrix of pixels intensity of the image and the image vector, γ , is the vector of \mathbf{A} which was previously transformed by column-stack vectorization. Thus, the dimension of γ is $mn \times 1$. The average of γ can be found as

$$\psi = \frac{1}{N} \sum_{i=1}^N \gamma_i, \quad (2.1)$$

where N is the number of training images. The zero-mean normalization is applied to all image vectors by

$$\phi_i = \gamma_i - \Psi, i = 1, 2, 3, \dots, M, \quad (2.2)$$

where ϕ_i is the i^{th} zero-mean normalized of γ_i . The covariance matrix, \mathbf{C} , of these image vectors can be calculated as

$$\mathbf{C} = \Phi\Phi^T, \quad (2.3)$$

where $\Phi = [\phi_1 \ \phi_2 \ \dots \ \phi_N]$

The PCA is defined as an orthogonal linear transformation that transforms the data to a new coordinate system such that the greatest variance by any projection of the data comes to lie on the principal component directions. This transformation is therefore equivalent to finding the eigenvalue decomposition of the matrix \mathbf{C} .

According to the dimension of ϕ , the dimension of \mathbf{C} will be $mn \times mn$ which can be normally quite large for calculating the eigenvalue decomposition. The number of training samples is normally smaller than the dimension of ϕ then the non-zero eigenvalues of this covariance matrix can be found in another way via a new matrix.

$$\mathbf{L} = \Phi^T\Phi, \quad (2.4)$$

The dimension of \mathbf{L} is only $m \times n$, thus the eigenvalue decomposition of \mathbf{L} can be done easier than \mathbf{C} . The eigenvalue decomposition of \mathbf{L} ,

$$\mathbf{L} = \mathbf{F}\mathbf{\Lambda}\mathbf{F}^T, \quad (2.5)$$

where $\mathbf{\Lambda}$ is the diagonal matrix which contains the eigenvalues of \mathbf{L} and \mathbf{F} contains a set of eigenvectors of \mathbf{L} . Finally, the eigenvectors of \mathbf{C} which correspond to the non-zero eigenvalues of \mathbf{C} can be determined by

$$\mathbf{U} = \Phi\mathbf{F}, \quad (2.6)$$

where \mathbf{U} is the matrix that contains a set of eigenvectors of \mathbf{C} .

The eigenvector associated with the largest eigenvalue has the same direction as the first principal component, the eigenvector associated with the second largest eigenvalue determines the direction of the second principal component, and so on. Since the lower-order principal components often contain the most important aspects of the data, the dimension of projected space can be reduced by retaining those characteristics of the data set that contribute most to its variance, by keeping lower-order principal components and ignoring higher-order ones.

2.2 Subspace Learning based on Tensor Analysis

Recently, multilinear algebra, the algebra of higher-order tensors, was applied for analyzing the multifactor structure of image ensembles [59]. Vasilescu and Terzopoulos have proposed a novel face representation algorithm called Tensorface [60]. Tensorface represents the set of face images by a higher-order tensor and extends traditional PCA to higher-order tensor decomposition. In this way, the multiple factors related to expression, illumination and pose can be separated from different dimensions of the tensor. However, Tensorface still considers each face image as a vector instead of 2-dimensional (2D) tensor. Thus, Tensorface is computationally expensive. Moreover, it does not encode discriminating information, thus it is not optimal for recognition.

2.2.1 Tensor PCA

Let $\mathbf{A} \in \mathbf{R}^{m \times n}$ denote an image of size $m \times n$. Mathematically, \mathbf{A} can be thought of as the 2^{nd} order tensor (or, 2-tensor) in the tensor space $\mathbf{R}^m \otimes \mathbf{R}^n$. Let $(\mathbf{u}_1, \dots, \mathbf{u}_m)$ be a set of orthonormal basis functions of \mathbf{R}^m . Let $(\mathbf{v}_1, \dots, \mathbf{v}_n)$ be a set of orthonormal basis functions of \mathbf{R}^n . Thus, an 2-tensor \mathbf{A} can be uniquely written as:

$$\mathbf{A} = \sum_{ij} (\mathbf{u}_i^T \mathbf{A} \mathbf{v}_j) \mathbf{u}_i \mathbf{v}_j^T, \quad (2.7)$$

This indicates that $\mathbf{u}_i \mathbf{v}_j^T$ forms a basis of the tensor space $\mathbf{R}^m \otimes \mathbf{R}^n$. Define two matrices $U = [\mathbf{u}_1, \dots, \mathbf{u}_m] \in \mathbf{R}^{m \times l_1}$ and $V = [\mathbf{v}_1, \dots, \mathbf{v}_n] \in \mathbf{R}^{n \times l_2}$. Let \mathbf{U} be a subspace of \mathbf{R}^m spanned

by $\mathbf{u}_{i=1}^{l_1}$ and \mathbf{V} be a subspace of \mathbf{R}^n spanned by $\mathbf{v}_{j=1}^{l_2}$. Thus, the tensor product $\mathcal{U} \otimes \mathcal{V}$ is a subspace of $\mathbf{R}^m \otimes \mathbf{R}^n$. The projection of $\mathbf{A} \in \mathbf{R}^{m \times n}$ onto the space $\mathcal{U} \otimes \mathcal{V}$ is $\mathbf{Y} = \mathbf{U}^T \mathbf{A} \mathbf{V} \in \mathbf{R}^{l_1 \times l_2}$

Suppose we have N images, $\mathbf{A}_1, \dots, \mathbf{A}_N \in \mathbf{R}^{m \times n}$. These images belong to k categories C_1, \dots, C_k . For the i -th category, there are n_i images. The mean of each category $\mathbf{M}_i^{\mathbf{A}}$ is computed by taking the average of \mathbf{A} in category i , i.e.,

$$\mathbf{M}_i^{(\mathbf{A})} = \frac{1}{n_i} \sum_{j \in C_i} \mathbf{A}_j \quad (2.8)$$

and the global mean $\mathbf{M}^{(\mathbf{A})}$ is defined as

$$\mathbf{M}^{(\mathbf{A})} = \frac{1}{N} \sum_{j=1}^N \mathbf{A}_j. \quad (2.9)$$

Let $\mathbf{Y}_i = \mathbf{U}^T \mathbf{A}_i \mathbf{V} \in \mathbb{R}^{l_1 \times l_2}$. Likewise, we can define

$$\mathbf{M}_i^{(\mathbf{Y})} = \frac{1}{n_i} \sum_{j \in C_i} \mathbf{Y}_j \quad (2.10)$$

and

$$\mathbf{M}^{(\mathbf{Y})} = \frac{1}{N} \sum_{j=1}^n \mathbf{Y}_j, \quad (2.11)$$

It is easy to check that $\mathbf{M}_i^{\mathbf{Y}} = \mathbf{U}^T \mathbf{M}_i^{(\mathbf{A})} \mathbf{V}$ and $\mathbf{M}^{\mathbf{Y}} = \mathbf{U}^T \mathbf{M}^{(\mathbf{A})} \mathbf{V}$.

The tensor subspace learning problem aims at finding the $(l_1 \times l_2)$ dimensional space $\mathcal{U} \otimes \mathcal{V}$ based on the specific objective functions. Particularly, we will introduce a novel algorithms called TensorPCA in this section.

TensorPCA is fundamentally based on PCA. It tries to project the data to the tensor subspace of maximal variances so that the reconstruction error can be minimized. The objective function of TensorPCA can be described as follows:

$$MAX_{\mathbf{U}, \mathbf{V}} \sum_i \|\mathbf{Y}_i - \mathbf{M}^{(\mathbf{Y})}\|^2 \quad (2.12)$$

Note that we use tensor norm of the difference of two tensors to measure the distance of two images. Since order two tensor is essentially matrix, we use Frobenius norm of a matrix as our 2-d tensor norm.

Since $\|\mathbf{A}\|^2 = \text{tr}(\mathbf{A}\mathbf{A}^T)$, we have

$$\begin{aligned} \sum_{i=1}^n \|\mathbf{Y}_i - \mathbf{M}^{(\mathbf{Y})}\|^2 &= \sum_{i=1}^n \text{tr}((\mathbf{Y}_i - \mathbf{M}^{(\mathbf{Y})})(\mathbf{Y}_i - \mathbf{M}^{(\mathbf{Y})})^T) \\ &= \sum_{i=1}^n \text{tr}(\mathbf{U}^T (\mathbf{Y}_i - \mathbf{M}^{(\mathbf{Y})}) \mathbf{V} \mathbf{V}^T (\mathbf{Y}_i - \mathbf{M}^{(\mathbf{Y})})^T \mathbf{U}) \\ &= \text{tr}(\mathbf{U}^T \sum_{i=1}^n (\mathbf{Y}_i - \mathbf{M}^{(\mathbf{Y})}) \mathbf{V} \mathbf{V}^T (\mathbf{Y}_i - \mathbf{M}^{(\mathbf{Y})})^T \mathbf{U}) \\ &= \text{tr}(\mathbf{U}^T \mathbf{M}_{\mathbf{V}} \mathbf{V}) \end{aligned} \quad (2.13)$$

where $\mathbf{M}_V = \sum_{i=1}^n (\mathbf{U}^T (\mathbf{Y}_i - \mathbf{M}^{(Y)}) \mathbf{V} \mathbf{V}^T (\mathbf{Y}_i - \mathbf{M}^{(Y)})^T \mathbf{U})$. Similarly, $\|\mathbf{A}\|^2 = \text{tr}(\mathbf{A} \mathbf{A}^T)$, so we also have

$$\begin{aligned} &= \text{tr}(\mathbf{V}^T (\sum_{i=1}^n (\mathbf{A}_i - \mathbf{M}^{(A)}) \mathbf{U} \mathbf{U}^T (\mathbf{A}_i - \mathbf{M}^{(A)})^T) \mathbf{V}) \\ &= \text{tr}(\mathbf{V}^T \mathbf{M}_U \mathbf{V}) \end{aligned} \quad (2.14)$$

where $\mathbf{M}_U = \sum_{i=1}^n ((\mathbf{A}_i - \mathbf{M}^{(A)})^T \mathbf{U} \mathbf{U}^T (\mathbf{A}_i - \mathbf{M}^{(A)}))$. Thus, the optimal projection \mathbf{U} should be the eigenvectors of \mathbf{M}_V and the optimal projection \mathbf{V} should be the eigenvectors of \mathbf{M}_U .

One might notice that \mathbf{U} and \mathbf{V} can not be computed independently. In our algorithm, we try to find an optimal coordinate system of $\mathbf{R}^m \otimes \mathbf{R}^n$. That is, we assume that both \mathbf{U} and \mathbf{V} are orthonormal, i.e. $\mathbf{U}^T \mathbf{U} = \mathbf{U} \mathbf{U}^T = \mathbf{I}$ and $\mathbf{V}^T \mathbf{V} = \mathbf{V} \mathbf{V}^T = \mathbf{I}$. In such case,

$$\mathbf{M}'_V = \sum_{i=1}^n ((\mathbf{A}_i - \mathbf{M}^A)(\mathbf{A}_i - \mathbf{M}^A)^T) \quad (2.15)$$

and

$$\mathbf{M}'_U = \sum_{i=1}^n ((\mathbf{A}_i - \mathbf{M}^A)^T (\mathbf{A}_i - \mathbf{M}^A)). \quad (2.16)$$

It is clear that \mathbf{M}'_V no longer depends on \mathbf{V} , and \mathbf{M}'_U no longer depends on \mathbf{U} . Therefore, the matrix \mathbf{U} can be simply computed as the eigenvectors of \mathbf{M}'_V and the matrix \mathbf{V} can be computed as the eigenvectors of \mathbf{M}'_U . Note that, both \mathbf{M}'_U and \mathbf{M}'_V are symmetric, hence their eigenvectors are orthonormal. This is consistent with our assumptions. If we try to reduce the original tensor space to a $l_1 \times l_2$ tensor subspace, we choose the first l_1 column vectors in \mathbf{U} and the first l_2 column vectors in \mathbf{V} .

2.3 Multilinear Principal Component Analysis (MPCA)

For the theoretically inclined reader, it should be noted that there are some recent developments in the analysis of higher order tensors, then this section introduces a new MPCA framework for tensor object dimensionality reduction and feature extraction using tensor representation. This framework is introduced from the perspective of capturing the original tensors variation. It provides a systematic procedure to determine effective representations of tensor objects. This contrasts to previous work such as those reported in [61], where vector, not tensor, representation was used, and the works reported in [59], [62], where matrix representation was utilized. Furthermore, unlike previous attempts, such as the one in [29], design issues of paramount importance in practical applications, such as the initialization, termination, convergence of the algorithm, and the determination of the subspace dimensionality, are discussed in details. The basic idea in MPCA solution to the problem of dimensionality reduction for tensor objects is introduced [33, 63].

2.3.1 Multilinear Projection of Tensor Objects

In this section, we review some basic multilinear concepts used in the MPCA framework development and introduces the multilinear projection of tensor objects for the purpose of dimensionality reduction.

Throughout this paper, the discussion is restricted to real-valued vectors, matrices, and tensors since the targeted applications, such as holistic gait recognition using binary silhouettes, involve real data only. The extension to the complex valued data sets is out of the scope of this work and it will be the focus of a forthcoming research.

An N th-order tensor is denoted as $\mathcal{A} \in \mathbf{R}^{I_1 \times I_2 \times \dots \times I_N}$ for $I_n = 1, \dots, N$. It is addressed by N indices $i_n, n = 1, \dots, N$ and each i_n addresses the n -mode of \mathcal{A} . The n -mode vectors of \mathcal{A} are defined as the I_n -dimensional vectors obtained from \mathcal{A} by varying the index i_n while keeping all the other indices fixed. Unfolding \mathcal{A} along the n -mode is denoted as $\mathbf{A}_{(n)} \in \mathbf{R}^{I_n \times (I_1 \times \dots \times I_{n-1} \times I_{n+1} \times \dots \times I_N)}$ and the column vectors of $\mathbf{A}_{(n)}$ are the n -mode vectors of \mathcal{A} which are illustrated in Fig. 2.1. Let the set of tensors be $\{\mathcal{A}_m, m = 1, \dots, M\}$ and the total scatter of these tensors is defined as

$$\Psi_{\mathcal{A}} = \sum_{m=1}^M \|\mathcal{A}_m - \bar{\mathcal{A}}\|^2, \quad (2.17)$$

where $\bar{\mathcal{A}}$ is the mean tensor calculated as $\bar{\mathcal{A}} = (1/M) \sum_{m=1}^M \mathcal{A}_m$. The n -mode total scatter matrix of these samples can be defined as

$$\mathbf{C}_{\mathbf{A}} = \sum_{m=1}^M (\mathbf{A}_{m(n)} - \bar{\mathbf{A}}_{(n)}) (\mathbf{A}_{m(n)} - \bar{\mathbf{A}}_{(n)})^T, \quad (2.18)$$

where $\mathbf{A}_{m(n)}$ is the n -mode unfolded of \mathcal{A} and $\bar{\mathbf{A}}$ is sample mean. The n -mode unfolded matrix can be illustrated in Fig. 2.1. In Fig. 2.1, a third-order tensor can be unfolded in 1-mode vector.

2.3.2 MPCA Algorithm

A set of M tensor objects $\{\mathcal{X}_m, m = 1, \dots, M\}$ is available for training with each tensor object $\mathcal{X}_m \in \mathbf{R}^{I_1 \times I_2 \times \dots \times I_N}$ assuming values in a tensor space $\mathbf{R}^{I_1} \otimes \mathbf{R}^{I_2} \dots \otimes \mathbf{R}^{I_N}$, where \otimes denotes the Kronecker product. The main objective of MPCA is to define a multilinear transformation $\tilde{\mathbf{U}}^{(n)}$ which denoted $I_n \times P_n$ matrix containing the orthonormal n -mode basis vectors and the matrix $\tilde{\mathbf{U}}^{(n)}$ is n th projection matrix, $n = 1, \dots, N$. It can map the original tensor space $\mathbf{R}^{I_1} \otimes \mathbf{R}^{I_2} \dots \otimes \mathbf{R}^{I_N}$ into a tensor subspace $\mathbf{R}^{P_1} \otimes \mathbf{R}^{P_2} \dots \otimes \mathbf{R}^{P_N}$ with ($P_n < I_n$, for $n = 1, \dots, N$):

We can define the projection of n -mode vector of \mathcal{X}_m as

$$\mathcal{Y}_m = \mathcal{X}_m \times_1 \tilde{\mathbf{U}}^{(1)T} \times_2 \tilde{\mathbf{U}}^{(2)T} \dots \times_N \tilde{\mathbf{U}}^{(N)T}. \quad (2.19)$$

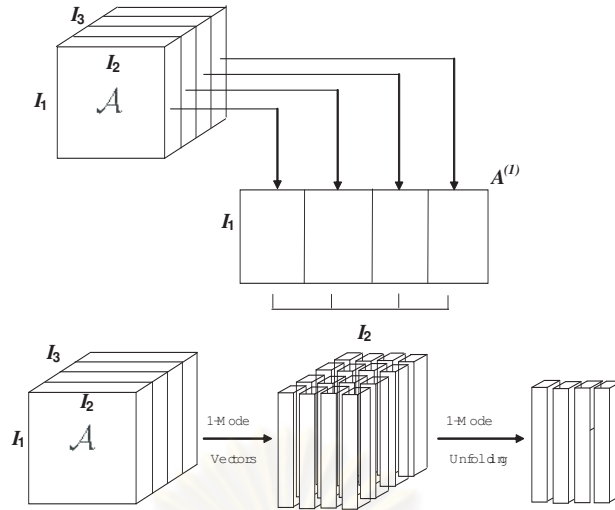


Figure 2.1: The visual illustration of the 1-mode unfolding of a third-order tensor \mathcal{A} to matrix $\mathbf{A}^{(1)}$

The tensor \mathcal{Y}_m can capture most of the variations observed in the original tensor objects, assuming that these variations are measured by the total tensor scatter. The objective of MPCA is the determination of the N projection matrices $\tilde{\mathbf{U}}^{(n)}$ that maximize the total tensor scatter $\Psi_{\mathcal{Y}}$ as

$$\{\tilde{\mathbf{U}}^{(n)}, n = 1, \dots, N\} = \underset{\tilde{\mathbf{U}}^{(1)}, \tilde{\mathbf{U}}^{(2)}, \dots, \tilde{\mathbf{U}}^{(N)}}{\operatorname{argmax}} \Psi_{\mathcal{Y}}. \quad (2.20)$$

There is no known optimal solution which allows for the simultaneous optimization of the N projection matrices. Since the projection to an N th-order tensor subspace consists of N projections to N vector subspaces, N optimization subproblems can be solved by finding $\tilde{\mathbf{U}}^{(n)}$ that maximizes the scatter in the n -mode vector subspace.

The dimensionality P_n for each mode is assumed to be known or predetermined. The matrix $\tilde{\mathbf{U}}^{(n)}$ consists of the P_n eigenvectors corresponding to the largest P_n eigenvalues of the matrix and it can be expressed as

$$\Phi^{(n)} = \sum_{m=1}^M (\mathbf{X}_m^{(n)} - \bar{\mathbf{X}}^{(n)}) \cdot \tilde{\mathbf{U}}_{\Phi^{(n)}} \cdot \tilde{\mathbf{U}}_{\Phi^{(n)}}^T \cdot (\mathbf{X}_m^{(n)} - \bar{\mathbf{X}}^{(n)})^T, \quad (2.21)$$

where

$$\tilde{\mathbf{U}}_{\Phi^{(n)}} = (\tilde{\mathbf{U}}^{(n+1)} \otimes \tilde{\mathbf{U}}^{(n+2)} \otimes \dots \otimes \tilde{\mathbf{U}}^{(1)} \otimes \tilde{\mathbf{U}}^{(2)} \otimes \dots \otimes \tilde{\mathbf{U}}^{(n-1)}). \quad (2.22)$$

The optimization of $\tilde{\mathbf{U}}_{\Phi^{(n)}}$ depends on the projections in other modes, so there is no closed-form solution. Therefore an iterative procedure is proposed to solve (2.22). The projection matrices are calculated one by one, keeping all the others fixed (local optimization).

2.3.3 Full Projection

The term full projection refers to the multilinear projection for MPCA with $P_n = I_n$ for $n = 1, \dots, N$. In this case, we can see that $\tilde{\mathbf{U}}_{\Phi^{(n)}} \cdot \tilde{\mathbf{U}}_{\Phi^{(n)}}^T$ is an identity matrix. As a

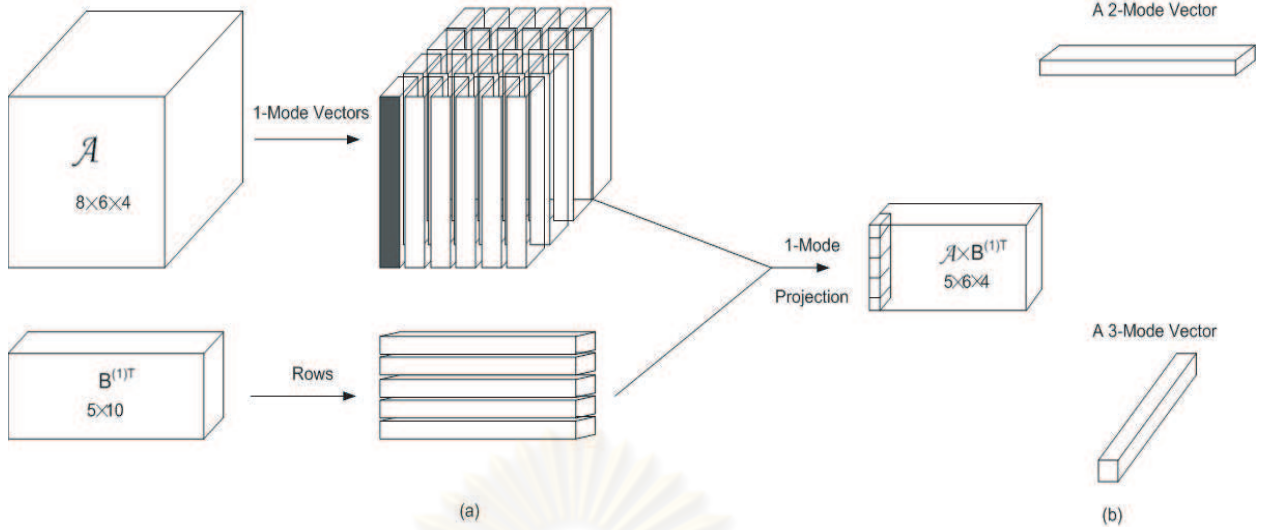


Figure 2.2: Visual illustration of multilinear projection: (a) projection in the 1-mode vector space and (b) 2-mode and 3-mode vectors.

result, $\Phi^{(n)}$ reduces to $\Phi^{(n)*} = \sum_{m=1}^M (\mathbf{X}_{m(n)} - \bar{\mathbf{X}}_{(n)}) (\mathbf{X}_{m(n)} - \bar{\mathbf{X}}_{(n)})^T$, with $\Phi^{(n)}$ determined by the input tensor samples only and independent of other projection matrices. The optimal $\tilde{\mathbf{U}}^{(n)} = \mathbf{U}^{(n)*}$ is then obtained as the matrix comprised of the eigenvectors of $\Phi^{(n)*}$ directly without iteration, and the total scatter $\Psi_{\mathcal{X}}$ in the original data is fully captured. However, there is no dimensionality reduction through this full projection. From the properties of eigendecomposition, it can be concluded that if all eigenvalues per mode are distinct, the full projection matrices (corresponding eigenvectors) are also distinct and that the full projection is unique (up to sign).

To interpret the geometric meanings of the n -mode eigenvalues, the total scatter tensor $\mathcal{Y}_{var}^* \in \mathbf{R}^{I_1 \times I_2 \times \dots \times I_N}$ of the full projection is introduced as an extension of the total scatter matrix. Each entry of the tensor \mathcal{Y}_{var}^* is defined as

$$\mathcal{Y}_{var}^*(i_1, i_2, \dots, i_N) = \sum_{m=1}^M [(\mathcal{Y}_m^* - \bar{\mathcal{Y}}^*)(i_1, i_2, \dots, i_N)]^2 \quad (2.23)$$

where

$$\mathcal{Y}_m^*(i_1, i_2, \dots, i_N) = \mathcal{X}_m \times_1 \mathbf{U}^{(1)*T} \dots \times_N \mathbf{U}^{(N)*T} \quad (2.24)$$

and

$$\bar{\mathcal{Y}}^* = (1/M) \sum_{m=1}^M \mathcal{Y}_m^* \quad (2.25)$$

Using the previous definition, it can be shown that for the so-called full projection ($P_n = I_n$ for all n), the i_n th n -mode eigenvalue $\lambda_{i_n}^{(n)*}$ is the sum of all the entries of the i_n th n -mode slice of \mathcal{Y}_{var}^*

$$\lambda_{i_n}^{(n)*} = \sum_{i=1}^{I_1} \dots \sum_{i_{n-1}=1}^{I_{n-1}} \sum_{i_{n+1}=1}^{I_{n+1}} \dots \sum_{i_N=1}^{I_N} \mathcal{Y}_{var}^*(i_1, \dots, i_{n-1}, i_n, i_{n+1}, \dots, i_N) \quad (2.26)$$

In this paper, the eigenvalues are all arranged in a decreasing order. Fig. 2.3 shows visually what the n -mode eigenvalues represent. In this graph, third-order tensors, e.g., short sequences (four frames) of images with size 6×5 , are projected to a tensor space of size $6 \times 5 \times 4$ (full projection) so that a total scatter tensor $\mathcal{Y}_{var}^* \in \mathbf{R}^{6 \times 5 \times 4}$ is obtained.

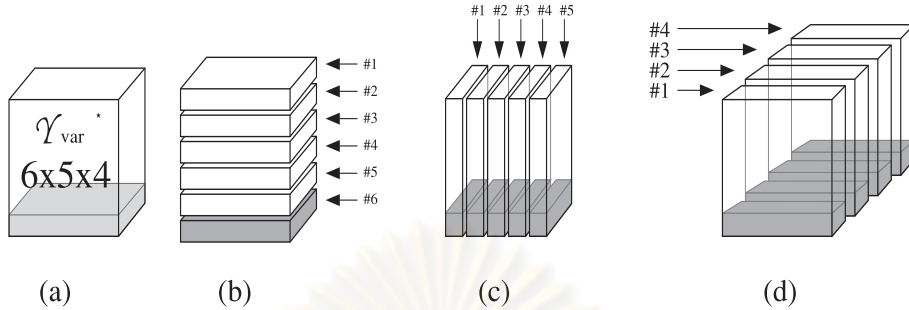


Figure 2.3: Visual illustration of (a) total scatter tensor, (b) 1-mode eigenvalues, (c) 2-mode eigenvalues, and (d) 3-mode eigenvalues.

2.3.4 MPCA Versus PCA and 2-D PCA Solutions

It is not difficult to see that the MPCA framework generalizes not only the classical PCA solution but also a number of the so-called 2-D PCA algorithms.

Indeed, for $N = 1$, the input samples are vectors $\mathbf{x}_m \in \mathbf{R}^{I_1}$. There is only one mode and MPCA is reduced to PCA. For dimensionality reduction purposes, only one projection matrix \mathbf{U} is needed in order to obtain $\mathbf{y}_m = \mathbf{x}_m \times_1 \mathbf{U} = \mathbf{U}^T \mathbf{x}_m$. In this case, there is only one $\Phi^{(n)} = \Phi^{(1)} = \sum_{m=1}^M (\mathbf{x}_m - \bar{\mathbf{x}}) \cdot (\mathbf{x}_m - \bar{\mathbf{x}})^T$, which is the total scatter matrix of the input samples in PCA [36]. The projection matrix maximizing the total scatter (variation) in the projected space is determined from the eigenvectors of $\Phi^{(1)}$. Thus, MPCA subsumes PCA.

In the so-called 2-D PCA solutions, input samples are treated as matrices, in other words second-order tensors. Two (left and right) projection matrices are sought to maximize the captured variation in the projected space. The proposed MPCA algorithm is equivalent to the 2-D PCA solution of [62], with the exception of the initialization procedure and termination criterion. Other 2-D PCA algorithms such as those discussed in [64] can be viewed as variations of the method in [62] and thus they can be considered special cases of MPCA when second-order tensors are considered.

2.4 TensorFaces: Multilinear Analysis of Facial Images

Multilinear algebra offers a natural approach to the analysis of the multifactor structure of image ensembles and to addressing the difficult problem of disentangling the constituent factors or modes then an image formation depends on scene geometry, viewpoint, and illumination conditions [60]. We apply multilinear analysis to the facial image data using the N -mode decomposition algorithm described in Section 2.3.

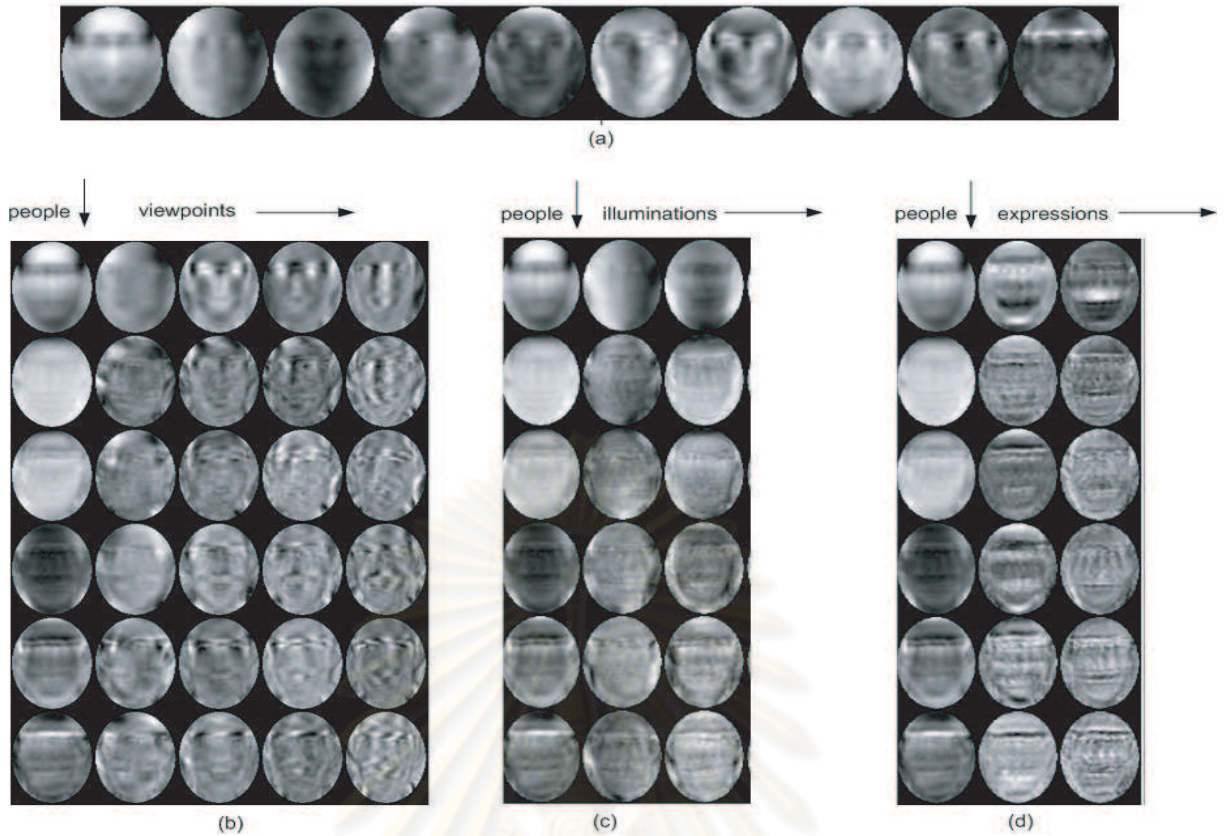


Figure 2.4: Some of the basis vectors resulting from the multilinear analysis of the facial image data tensor \mathcal{D} [1, 2].

In a concrete application of our multilinear image analysis technique, we employ the Weizmann face database of 28 male subjects photographed in 15 different poses under 4 illuminations performing 3 different expressions. The 5-mode decomposition of \mathcal{D} is

$$\mathcal{D} = \mathcal{Z} \times_1 \mathbf{U}_{people} \times_2 \mathbf{U}_{views} \times_3 \mathbf{U}_{illums} \times_4 \mathbf{U}_{expres} \times_5 \mathbf{U}_{pixels}. \quad (2.27)$$

Using a global rigid optical flow algorithm, we roughly aligned the original 512×352 pixel images relative to one reference image. The images were then decimated by a factor of 3 and cropped as shown in Fig. 2.4, yielding a total of 7943 pixels per image within the elliptical cropping window. Our facial image data tensor \mathcal{D} is a $28 \times 5 \times 3 \times 3 \times 7943$ tensor. The number of modes is $N = 5$. The core tensor \mathcal{Z} governs the interaction between the factors represented in the 5 mode matrices: The 28×28 mode matrix \mathbf{U}_{people} spans the space of people parameters, the 5×5 mode matrix \mathbf{U}_{views} spans the space of viewpoint parameters, the 3×3 mode matrix \mathbf{U}_{illums} spans the space of illumination parameters and the 3×3 mode matrix \mathbf{U}_{expres} spans the space of expression parameters. The 7943×7943 mode matrix \mathbf{U}_{pixels} orthonormally spans the space of images.

Our multilinear analysis, which we call TensorFaces, subsumes linear, PCA analysis or conventional eigenfaces. Each column of \mathbf{U}_{pixels} is an eigenimage. These eigenimages are identical to conventional eigenfaces [13, 17], since the former were computed by performing

an SVD on the mode-5 flattened data tensor \mathcal{D} which yields the matrix \mathbf{D}_{pixels} whose columns are the vectorized images. To further show mathematically that PCA is a special case of our multilinear analysis, we write the latter in terms of matrix notation. A matrix representation of the N -mode SVD can be obtained by unfolding \mathcal{D} and \mathcal{Z} as follows:

$$\mathbf{D}_{(n)} = \mathbf{U}_{(n)} \mathbf{Z}_{(n)} (\mathbf{U}_{(n-1)} \otimes \dots \otimes \mathbf{U}_{(1)} \otimes \mathbf{U}_{(N)} \otimes \dots \otimes \mathbf{U}_{(n+2)} \otimes \mathbf{U}_{(n+1)})^T, \quad (2.28)$$

Using 2.28 we can express the decomposition of \mathcal{D} as

$$\underbrace{\mathbf{D}_{(pixels)}}_{\text{imagedata}} = \underbrace{\mathbf{U}_{(pixels)}}_{\text{basisvectors}} \underbrace{(\mathbf{Z}_{(pixels)} \mathbf{U}_{(expres)} \otimes \mathbf{U}_{(illums)} \otimes \mathbf{U}_{(views)} \otimes \mathbf{U}_{(people)})^T}_{\text{coefficients}} \quad (2.29)$$

The above matrix product can be interpreted as a standard linear decomposition of the image ensemble, where the mode matrix $\mathbf{U}_{(pixels)}$ is the PCA matrix of basis vectors and the associated matrix of coefficients is obtained as the product of the flattened core tensor times the Kronecker product of the people, viewpoints, illuminations, and expressions mode matrices. Thus, as we stated above, our multilinear analysis subsumes linear, PCA analysis.

The advantage of multilinear analysis is that the core tensor \mathcal{Z} can transform the eigenimages present in the matrix $\mathbf{U}_{(pixels)}$ into eigenmodes, which represent the principal axes of variation across the various modes (people, viewpoints, illuminations, expressions) and represents how the various factors interact with each other to create an image. This is accomplished by simply forming the product $\mathcal{Z} \times_5 \mathbf{U}_{(pixels)}$. By contrast, PCA basis vectors or eigenimages represent only the principal axes of variation across images. To demonstrate, Fig. 2.4 illustrates in part the results of the multilinear analysis of the facial image tensor \mathcal{D} . Fig. 2.4(a) shows the first 10 PCA eigenimages contained in $\mathbf{U}_{(pixels)}$. Fig. 2.4(b) illustrates some of the eigenmodes in the product $\mathcal{Z} \times_5 \mathbf{U}_{(pixels)}$. A few of the lower-order eigenmodes are shown in the three arrays. The labels at the top of each array indicate the names of the horizontal and vertical modes depicted by the array. Note that the basis vector at the top left of each panel is the average over all people, viewpoints, illuminations, and expressions, and that the first column of eigenmodes (people mode) is shared by the three arrays.

2.5 Color Image Processing

Over the last three decades, we have seen several important contributions in the field of color image processing. While there have been many early papers that address various aspects of color images, it is only recently that a more complete understanding of color vision, colorimetry, and color appearance has been applied to the design of imaging systems and image processing methodologies. The first contributions in this area were those that changed the formulation of color signals from simple algebraic equations to matrix representation [65], [66], [67]. More powerful use of the matrix algebraic representation was presented in [68], where set theoretic methods were introduced to color processing. The first overview extending signal processing concepts to color was presented in IEEE Signal

Processing Magazine in 1993 [69]. This was followed by a special issue on color image processing in IEEE Transactions on Image Processing in July 1997, where a complete review of the state of the art at that time was found in [70, 71].

At this point, the focus of the issue shifts from hardware and system centric to image processing techniques that form the backbone of many color systems and applications. The first of these is discussed in Detection and Classification of Edges in Color Images [72], where the emphasis is placed on detecting discontinuities, i.e., transitions from one region to another, instead of similarities in a given image. One of the most fundamental steps in many applications and in the design of image processing techniques is to ensure that a given image is optimized for noise and enhanced in quality. This is outlined in the article Vector Filtering for Color Imaging [73], where the authors discuss and compare the various techniques outlining their strength, areas of improvements, and future research directions. Finally, the article titled Digital Color Halftoning [74] provides a complete review of the methods employed by printers to reproduce color images and the challenges they face in ensuring that these images are free of visual artifacts.

2.5.1 Mathematical Definition of Color Matching

A vector space approach to describing color is useful for expressing and solving complex problems in color imaging. For this reason, we will use this notation to describe the fundamentals of color matching. Let the $N \times 3$ matrix $\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3]$ represent the response of the eye, where the N vectors, \mathbf{s}_i , correspond to the response of the i th type sensor (cone) in Fig. 2.5. A given visible spectrum can be represented by an N vector, \mathbf{f} , a function whose value is radiant energy. Hence, the response of the sensors to the input spectrum is a three vector, \mathbf{c} , obtained by

$$\mathbf{c} = \mathbf{S}^T \mathbf{f} \quad (2.30)$$

Two visible N -vectors spectra \mathbf{f} and \mathbf{g} are said to have the same color if they appear the same to a human observer. In our linear model, this implies that if \mathbf{f} and \mathbf{g} represent different spectral distributions, they portray equivalent colors if

$$\mathbf{S}^T \mathbf{f} = \mathbf{S}^T \mathbf{g} \quad (2.31)$$

From this, it can be easily seen that many different spectra can result in the same color appearance to a given observer. This fascinating phenomena is known as metamerism (meh tam er ism), and the two spectra are termed as metamers. In essence, metamerism is basically color aliasing and can be described by generalizing the well-known Shannon sampling theorem frequently encountered in communications and digital signal processing. It should be noted, however, that the level of metamerism may vary across various observers, dependent on their individual cone sensitivities.

In practice, it is desirable to have a matrix of color matching functions that are non-negative, so they can be physically realized as optical filters. This problem was addressed

by the Commission Internationale de l'Éclairage (CIE), in 1931, yielding the XYZ color matching functions shown in Fig. 2.6 as solids lines. Hence, the matrix \mathbf{A} can now be used to represent these functions. The Y value was chosen to be the luminous efficiency function, making it equivalent to the photometric luminance value. This standardization led to the precise definition of colorimetric quantities, such as tristimulus values and chromaticity.

The term tristimulus values refers to the vector of values obtained from a radiant spectrum, \mathbf{r} , by $\mathbf{t} = [X, Y, Z]^T = \mathbf{A}^T \mathbf{r}$ (we recognize the inconsistency of denoting the elements of \mathbf{t} by X, Y, Z , but since the color world still uses the X, Y, Z terms, we use it here). The chromaticity is then obtained by normalizing the tristimulus values yielding

$$\begin{aligned} x &= X/(X + Y + Z) \\ y &= Y/(X + Y + Z) \\ z &= Z/(X + Y + Z) \end{aligned} \quad (2.32)$$

Since $x + y + z = 1$, any two chromaticity coordinates are sufficient to characterize the chromaticity of a spectrum. In general, as a matter of convention, the x and y terms are used as the standard.

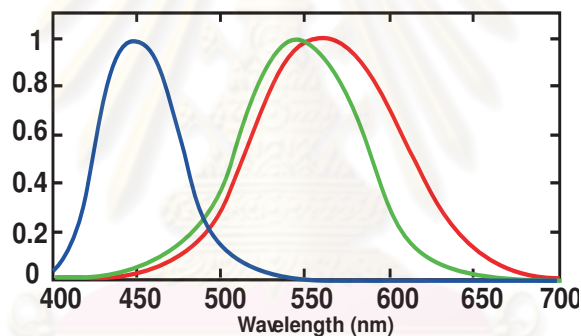


Figure 2.5 Cone sensitivities.

2.5.2 Mathematics of Color Reproduction

To reproduce a color image, it is necessary to generate new vectors in N space (spectral space) from those obtained by a given multispectral sensor. Since the eye can be represented as a three-channel sensor, it is most common for a multispectral sensor to use three types of filters. Hence, the characteristics of the resulting multispectral response functions associated with the input and output devices are critical aspects for color reproduction. Output devices can be characterized as being additive or subtractive. Additive devices, such as cathode ray tubes (CRTs), produce light of varying spectral composition as viewed by the human observer. On the other hand, subtractive devices, such as ink-jet printers, produce filters that attenuate portions of an illuminating spectrum. We will discuss both types in the following, clearly highlighting their differences.

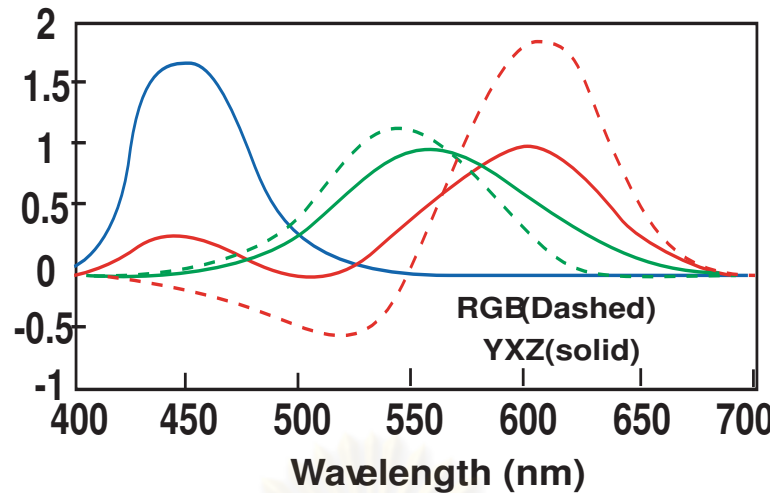


Figure 2.6: CIE RGB and XYZ color matching functions: RGB are shown in dashed lines, and XYZ are shown in solid lines.

2.5.3 Additive Color Systems

In additive devices, various colors are generated by combining light sources with different wavelengths. These light sources are known as primary. An example of this is illustrated in Fig. 2.7. In the Fig. 2.7, it can be easily seen that cyan, magenta, yellow, and white are generated by combining blue and green; red and blue; red and green; and red, green, and blue, respectively. The red, green, and blue channels of an example color image are also shown for illustration purposes. Other colors can be generated by varying the intensities of the red, green, and blue primaries. For instance, the screen of a television, or CRT, is covered with phosphoric dots that are clustered in groups. Each group contains these primary colors: red, green, and blue, which are combined in a weighted fashion to produce a wide range of colors. Additive color systems are characterized by their corresponding multispectral output response. For instance, a three-color monitor is represented by the $N \times 3$ matrix, \mathbf{E} , which serves the same purpose as the primaries in the color matching experiment. The amount of each primary is controlled by a three-vector \mathbf{c} . The spectrum of the output is then computed as follows:

$$\mathbf{f} = \mathbf{E}\mathbf{c}, \quad (2.33)$$

Hence, the tristimulus values associated with a standard observer who is viewing the screen are given by

$$\mathbf{t} = \mathbf{A}^T \mathbf{f} = \mathbf{A}^T \mathbf{E}\mathbf{c}, \quad (2.34)$$

There are several challenges that need to be considered when dealing with additive systems. One is to choose the control values so that the output matches the intended target values. This is not feasible for all possible colors due to the power limitations of the output device.

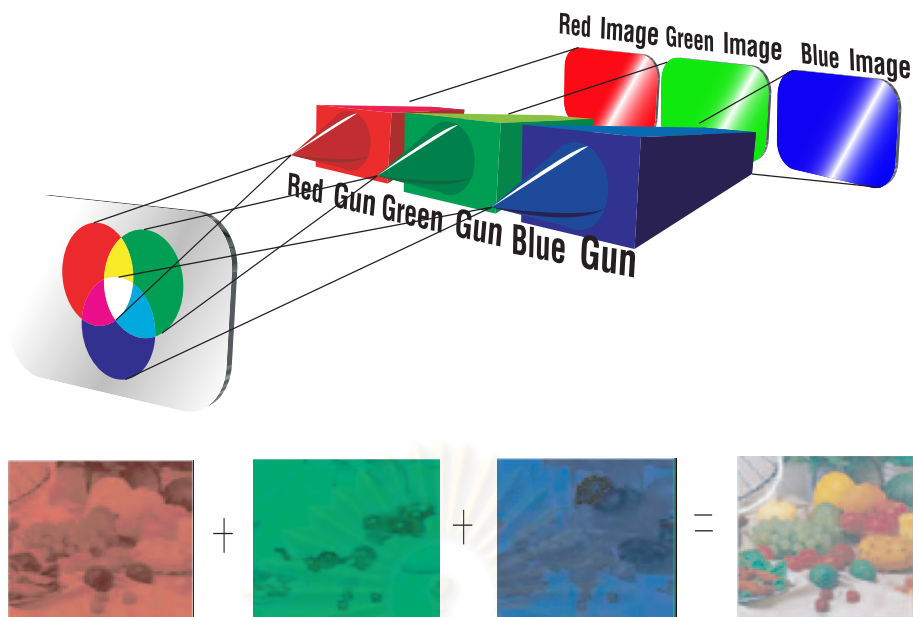


Figure 2.7 Additive color system.

Furthermore, the control values cannot be negative, i.e., we cannot produce negative light. In addition, we have the question of estimating the best values of c from some recorded data.

2.5.4 Subtractive Color Systems

Subtractive systems are characterized by the fundamental property that color is obtained by removing (subtracting) selected portions of a source spectrum. This is illustrated in Fig. 2.8, where cyan, magenta, and yellow colorants are used to absorb, in this respect, the red, green, and blue spectral components from white light. The cyan, magenta, and yellow channels of a color image are also shown for illustration purposes. Hence, each colorant absorbs its complementary color and transmits the remainder of the spectrum. The amount of light removed, by blocking or absorption, is determined by the concentration and material properties of the colorant. The color is generally presented on a transmissive medium like transparencies or on a reflective medium like paper. While the colorant for subtractive systems may be inks, dyes, wax, or toners, the same mathematical representation outlined in previously can be used to approximate them. The main property of interest for imaging in subtractive systems is the optical density. The transmission of an optically transmissive material is defined as the ratio of the intensity of the light that passes through the material to the intensity of the source. This is illustrated by

$$T = \frac{I_{out}}{I_{in}}. \quad (2.35)$$

As a result, the optical density is defined by

$$d = -\log_{10}(T). \quad (2.36)$$

and is related to the physical density of the material. The inks can be characterized by their density spectra, the $N \times 3$ matrix \mathbf{D} . Hence, the spectrum that is seen by the observer is the product of an illumination source, the transmission of the ink, and the reflectance of the paper. Since the transmissions of the individual inks reduce the light proportionately, the output at each wavelength, λ , is given by

$$g(\lambda) = l(\lambda)t_1(\lambda)t_2(\lambda)t_3(\lambda) \quad (2.37)$$

where $t_i(\lambda)$ is the transmission of the i th ink and $l(\lambda)$ is the intensity of the illuminant. For simplification, the reflectance of the paper is assumed perfect and is assigned the value of 1.0. The transmission of a particular colorant is related logarithmically to the concentration of the ink on the page. The observed spectrum is obtained mathematically by

$$g = \mathbf{L}[10^{-\mathbf{D}\mathbf{c}}] \quad (2.38)$$

where \mathbf{L} is a diagonal matrix representing an illuminant spectrum and \mathbf{c} is the concentration of the colorant. The concentration values are held between zero and unity and the matrix of density spectra, \mathbf{D} , represents the densities at the maximum concentration. The exponential term is computed componentwise, i.e.,

$$10^r = [10^{r_1} 10^{r_2} \dots 10^{r_N}]^T. \quad (2.39)$$

This simple model ignores nonlinear interactions between colorant layers. For a reflective medium, the model requires an additional diagonal matrix, which represents the reflectance spectrum of the surface. For simplicity, this can be conceptually included in the illuminant matrix \mathbf{L} . The actual process for subtractive color reproduction is much more complicated and cannot, in general, be comprehensively modeled by the equations described here. Hence, these systems are usually characterized by look-up tables (LUTs) that capture their input-output relationships empirically. The details of handling device characterizations via LUTs are described in [75].

2.5.5 Color Spaces

The proper use and understanding of color spaces is necessary for the development of color image processing methods that are optimal for the human visual system. Many algorithms have been developed that process in an RGB color space without ever defining this space in terms of the CIE color matching functions, or even in terms of the spectral responses of R, G, and B. Such algorithms are nothing more than multichannel image processing techniques applied to a three-band image, since there is no accounting for the perceptual aspect of the problem. To obtain some relationship with the human visual system, many color image processing algorithms operate on data in hue, saturation, lightness (HSL) spaces. Commonly, these spaces are transformations of the aforementioned RGB color space and hence have no visual meaning until a relationship is established back to a CIE color space. To further confuse the issue, there are many variants of these color spaces, including hue saturation

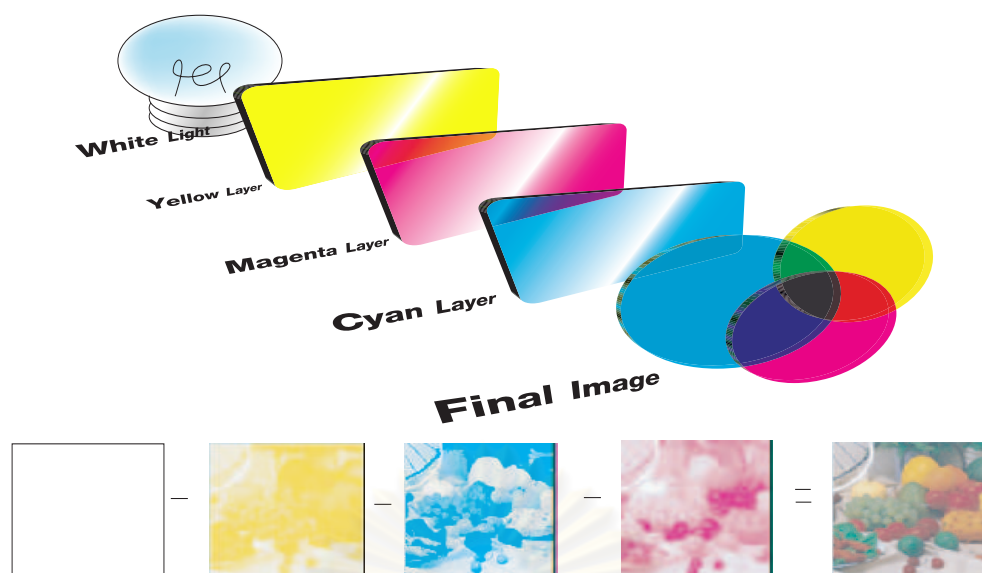


Figure 2.8 Subtractive color system.

value (HSV), hue saturation intensity (HSI), and hue chroma intensity (HCI), some of which have multiple definitions in terms of transforming from RGB. Since color spaces are of such importance and a subject of confusion, we will discuss them in details.

There are two primary aspects of a color space that make it more desirable and attractive for use in color devices: 1) its computational expediency in transforming a given set of data to the specific color space and 2) conformity of distances of color vectors in the space to that observed perceptually by a human subject, i.e., if two colors are far apart in the color space, they look significantly different to an observer with normal color vision. Unfortunately, these two criteria are antagonistic. The color spaces that are most suited for measuring perceptual differences require complex computation, and vice versa.

2.5.6 Uniform Color Spaces

It is well publicized that the psychovisual system is nonlinear and extremely complex. It cannot be modeled by a simple function. The sensitivity of the system depends on what is being observed and the purpose of the observation. A measure of sensitivity that is consistent with the observations of arbitrary scenes is well beyond our present capabilities. However, much work has been done to determine human color sensitivity in matching two color fields that subtend only a small portion of the visual field. In fact, the color matching functions (CMFs) of Figure 2.6 are more accurately designated by the solid angle of the field of view that was used for their measurement. A two-degree field of view was used for those CMFs.

It is well known that mean square error is, in general, a poor measure of error in any phenomenon involving human judgment. A common method of treating the nonuniform error problem is to transform the space into one where Euclidean distances are more closely correlated with perceptual ones. As a result, the CIE recommended, in 1976, two transformations in an attempt to standardize measures in the industry. Neither of these standards

achieve the goal of a uniform color space. However, the recommended transformations do reduce the variations in the sensitivity ellipses by a large degree. In addition, they have another major feature in common: the measures are made relative to a reference white point. By using the reference point, the transformations attempt to account for the adaptive characteristics of the visual system. The first of these transformation is the CIELAB space defined by

(2.40)

$$L^* = 116\left(\frac{Y}{Y_n}\right)^{\frac{1}{3}} - 16 \quad (2.41)$$

$$a^* = 500\left[\left(\frac{X}{X_n}\right)^{\frac{1}{3}} - \left(\frac{Y}{Y_n}\right)^{\frac{1}{3}}\right] \quad (2.42)$$

$$b^* = 200\left[\left(\frac{Y}{Y_n}\right)^{\frac{1}{3}} - \left(\frac{Z}{Z_n}\right)^{\frac{1}{3}}\right] \quad (2.43)$$

for $\left(\frac{X}{X_n}\right), \left(\frac{Y}{Y_n}\right), \left(\frac{Z}{Z_n}\right) > 0.01$. The values X_n, Y_n, Z_n are the CIE tristimulus values of the reference white under the reference illumination, and X, Y, Z are the tristimulus values, which are to be mapped to the CIELAB color space. This maps the reference white to $(L^*, a^*, b^*) = (100, 0, 0)$. The requirement that the normalized values be greater than 0.01 is an attempt to account for the fact that at low illumination the cones become less sensitive and the rods (monochrome receptors) become active. Hence, a linear model is used at low light levels.

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

CHAPTER III

THE PROPOSED FRAMEWORKS

In this chapter, the two frameworks were proposed for improving the performance of face hallucination. In Section 3.1, the first framework, *Color Face Hallucination with Linear Regression Model in MPCA*, take the advantage of the MPCA. The second framework, *Color Face Super-Resolution Based on Tensor Patches Method*, was introduced in Section 3.2.

3.1 Color Face Hallucination with Linear Regression Model in MPCA

The goal of this part is to propose a novel hallucination reconstruction, using the MPCA for color face image. Given a training image set $\{\mathcal{X}^H, \mathcal{X}^L\}$, where $\mathcal{X}^H = \{\mathcal{X}_i^h\}_{i=1, \dots, K}$, $\mathcal{X}^L = \{\mathcal{X}_i^l\}_{i=1, \dots, K}$ and K is the number of training image. The training color face images can be defined as $\mathcal{X}_i^h \in \mathbf{R}^{I_1} \otimes \mathbf{R}^{I_2} \otimes \mathbf{R}^{I_3}$ and $\mathcal{X}_i^l \in \mathbf{R}^{J_1} \otimes \mathbf{R}^{J_2} \otimes \mathbf{R}^{J_3}$ which are the HR color face image and LR color face image, respectively.

In this paper, we propose our method in many color models such as RGB, YCbCr, HSV and CIELAB. For example, the training image sets \mathcal{X}_i^l and \mathcal{X}_i^h can be described in RGB model as $\mathcal{X}_i^h \in \mathbf{R}^{I_1 \times I_2 \times 3}$ and $\mathcal{X}_i^l \in \mathbf{R}^{J_1 \times J_2 \times 3}$. In addition, the index $I_1 \times I_2 \times 3$ is an array of color pixels, where each color pixel is a triplet corresponding to the red, green, and blue components of an RGB image.

Following standard multilinear algebra, any tensor can be expressed as the product

$$\mathcal{Y}_i^h = \mathcal{X}_i^h \times_1 \tilde{\mathbf{U}}^{h(1)T} \times_2 \tilde{\mathbf{U}}^{h(2)T} \times_3 \tilde{\mathbf{U}}^{h(3)T} \quad (3.1)$$

and

$$\mathcal{Y}_i^l = \mathcal{X}_i^l \times_1 \tilde{\mathbf{U}}^{l(1)T} \times_2 \tilde{\mathbf{U}}^{l(2)T} \times_3 \tilde{\mathbf{U}}^{l(3)T} \quad (3.2)$$

where $\mathcal{Y}_i^h \in \mathbf{R}^{P_1} \otimes \mathbf{R}^{P_2} \otimes \mathbf{R}^{P_3}$ and $\mathcal{Y}_i^l \in \mathbf{R}^{Q_1} \otimes \mathbf{R}^{Q_2} \otimes \mathbf{R}^{Q_3}$, with $(P_1 < I_1, P_2 < I_2$ and $P_3 < 3)$ as the index of HR training set and $(Q_1 < J_1, Q_2 < J_2$ and $Q_3 < 3)$ as the index of LR training set. The tensor \mathcal{Y}_i can capture most of the variations observed in the original tensor objects, assuming that these variations are measured by the total scatter. Therefore, two sets of MPCA subspace projection are obtained, which are $\mathcal{Y}_i^h = [y_{r,s,t}^h]_i$ and $\mathcal{Y}_i^l = [y_{r,s,t}^l]_i$ respectively. In addition, we use $[y_{r,s,t}]$ to represent a tensor with $y_{r,s,t}$ as its (r, s, t) -th entry.

One can see that if the sets of $\tilde{\mathbf{U}}^{(1)}, \tilde{\mathbf{U}}^{(2)}, \tilde{\mathbf{U}}^{(3)}$ are disjoint sets of orthonormal vectors then the correlation between the decomposition coefficients can be suppressed. From the model $\mathcal{Y}_i^h = f(\mathcal{Y}_i^l)$. When a generative model is used, f is actually a probability. Thus we can consider the conditional probability $P(\mathcal{Y}_i^h | \mathcal{Y}_i^l)$. When a new testing color face image

(LR) \mathcal{X}^l is provided, the HR MPCA subspace projection is given by:

$$\mathcal{Y}^h = \underset{\mathcal{Y}}{\operatorname{argmax}} P(\mathcal{Y}|\mathcal{Y}^l). \quad (3.3)$$

The HR color face image \mathcal{X}^h can be reconstructed by back-projection from the MPCA subspace into the image tensor space as

$$\mathcal{X}^h = \mathcal{Y}^h \times_1 \tilde{\mathbf{U}}^{h(1)} \times_2 \tilde{\mathbf{U}}^{h(2)} \times_3 \tilde{\mathbf{U}}^{h(3)} \quad (3.4)$$

Because each individual coefficient can be estimated separately, we have

$$\hat{y}_{r,s,t}^h = \underset{y_{r,s,t}}{\operatorname{argmax}} P(y_{r,s,t}|\mathcal{Y}^l), \quad (3.5)$$

From the assumption of low-correlation between the coefficients in \mathcal{Y}^l , we can simplify this probability in (3.5) as

$$P(y_{r,s,t}|\mathcal{Y}^l) \approx P(y_{r,s,t}|y_{1,1,1}^l) \cdot P(y_{r,s,t}|y_{Q_1, Q_2, Q_3}^l). \quad (3.6)$$

We can also rewrite $[y_{r,s,t}^l]$ into a vector form as

$$P(y_r|\mathcal{Y}^l) \approx \prod_{p=1}^{Q_1 Q_2 Q_3} P(y_r|y_p^l). \quad (3.7)$$

We use Gaussian to model the probability in (3.7) as:

$$P(y_r|y_p^l) \approx c \exp\left\{-\frac{(y_r - w_{r,p} y_p^l)^2}{2}\right\}, \quad (3.8)$$

where c is a constant. This Gaussian model evaluates the weighted distance between the projection coefficients. Equation (3.7) can be rewritten as

$$P(y_r|\mathcal{Y}^l) \approx c \exp\left\{-\sum_{p=1}^{Q_1 Q_2 Q_3} \frac{(y_r - w_{r,p} y_p^l)^2}{2}\right\}. \quad (3.9)$$

The Maximum Likelihood estimate of (3.9) is given by

$$\hat{y}_r^h = \underset{y_r}{\operatorname{argmax}} \log P(y_r|\mathcal{Y}^l). \quad (3.10)$$

We can express in a linear regression model as [76]

$$\hat{y}_r^h = \sum_{p=1}^{Q_1 Q_2 Q_3} w'_{r,p} y_p^l, \quad (3.11)$$

where

$$w'_{r,p} = \frac{w_{r,p}}{Q_1 Q_2 Q_3}. \quad (3.12)$$

We can calculate the value of $w'_{r,p}$ from the HR and LR of training sets. Each training image provides one equation to find $w'_{r,p}$ ($p = 1, \dots, Q_1 Q_2 Q_3$) and the column vector formed by

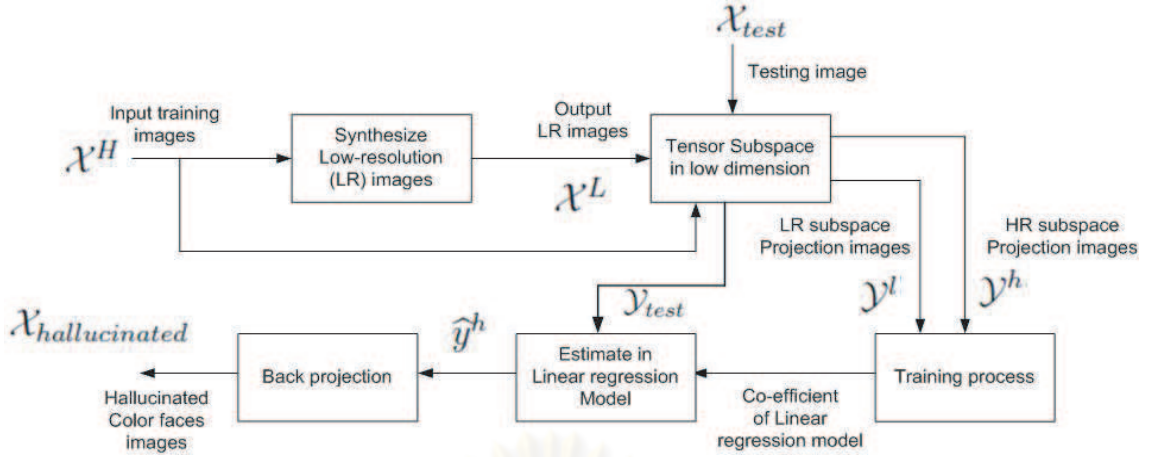


Figure 3.1: block diagram of color face hallucination with the linear regression model in MPCA

the r th projection coefficients for the HR images be \mathbf{y}_r^h . In the same way, the column vector formed by the p th projection coefficients for the corresponding LR set be \mathbf{y}_p^l and we can state as:

$$\mathbf{y}_r^h = [\mathbf{y}_1^l, \dots, \mathbf{y}_{Q_1 Q_2 Q_3}^l] \mathbf{w}_r, \quad (3.13)$$

where

$$\mathbf{w}_r = [w'_{r,1}, \dots, w'_{r,Q_1 Q_2 Q_3}]^T. \quad (3.14)$$

After that \mathbf{w}_r can be given by an Least-Square (LS) estimate:

$$\mathbf{w}_r = [\mathbf{y}_1^l, \dots, \mathbf{y}_{Q_1 Q_2 Q_3}^l]^+ \mathbf{y}_r^h. \quad (3.15)$$

3.2 Color Face Super-Resolution Based on Tensor Patch Method

In this section, we apply HOSVD tensor patch within the well-known framework for color face hallucination. A tensor structure provides a powerful mechanism to incorporate information and interaction of these image ensembles of multiple modalities at different resolutions. More precisely, given a training dataset of high-resolution face images, we blur and subsample them with different Gaussian filters and sub-sampling factors, while keeping the image size unchanged, so to generate a set of low-resolution training face images. To further improve the modeling accuracy, we uniformly decompose these face images into overlapped image blocks, and then obtain a hierarchical ensemble containing block-wise face images at low- and high-resolution. With these training data in place, we can construct a seventh-order tensor \mathcal{D} . We use HOSVD to decompose \mathcal{D} into

$$\mathcal{D} = \mathcal{Z} \times_1 \mathbf{U}_{idens} \times_2 \mathbf{U}_{pixel1} \times_3 \mathbf{U}_{pixel2} \times_4 \mathbf{U}_{patch1} \times_5 \mathbf{U}_{patch2} \times_6 \mathbf{U}_{color} \times_7 \mathbf{U}_{resos} \quad (3.16)$$

where tensor \mathcal{D} groups these block-wise training images into a tensor structure, and the core tensor \mathcal{Z} governs the interactions between the seven mode factors. In (3.16), mode matrix \mathbf{U}_{idens} spans the parameter space of identity, \mathbf{U}_{pixel1} and \mathbf{U}_{pixel2} span the space of pixel in x-axis and y-axis, \mathbf{U}_{patch1} and \mathbf{U}_{patch2} span the space of patch in x-axis and y-axis, \mathbf{U}_{color} span the spaces of color space and \mathbf{U}_{resos} span the space of resolution.

To model high-resolution details for the purpose of face hallucination, we uniformly decompose the low- and high-resolution face images into small overlapped patches, and perform tensor modeling at patch level. We can hallucinate high-resolution image data with all the decomposed patches. The final high-resolution face images are compositions of their corresponding overlapped small patches.

We suppose that \mathbf{H}_1 is the high-resolution color images, \mathbf{S}_1 is its low-resolution correspondences to be synthesized and \mathbf{L}_1 is any low-resolution color face input images. The task comes as finding the maximum *a posteriori* (MAP) estimation of \mathbf{H}_1 given \mathbf{L}_1 which can be formulated as

$$\{\mathbf{H}_{1MAP}\} = \operatorname{argmax}_{\mathbf{H}_1, \mathbf{S}_1} \log P(\mathbf{H}_1, \mathbf{S}_1 | \mathbf{L}_1) \quad (3.17)$$

By applying Bayes rule, we have

$$P(\mathbf{H}_1, \mathbf{S}_1 | \mathbf{L}_1) = P(\mathbf{H}_1 | \mathbf{S}_1, \mathbf{L}_1) P(\mathbf{S}_1 | \mathbf{L}_1) \quad (3.18)$$

During the sequential processes of our face hallucination, the high-resolution face image is independently reconstructed. Based on the synthesized low-resolution image, the above expression can be state as

$$\begin{aligned} P(\mathbf{H}_1, \mathbf{S}_1 | \mathbf{L}_1) &= P(\mathbf{H}_1 | \mathbf{S}_1) P(\mathbf{S}_1 | \mathbf{L}_1) \\ &= P(\mathbf{S}_1 | \mathbf{H}_1) P(\mathbf{H}_1) P(\mathbf{S}_1 | \mathbf{L}_1). \end{aligned} \quad (3.19)$$

The high-resolution image is naturally composed from the two part:

$$\mathbf{H} = \mathbf{H}^{lm} + \mathbf{H}^h, \quad (3.20)$$

where \mathbf{H}^{lm} represents face images containing low- and middle-frequency information, and \mathbf{H}^h contains high-frequency part. Since \mathbf{H}^{lm} contributes the main part of after blurring and subsampling, then the probability $P(\mathbf{S} | \mathbf{H})$ can be approximated as $P(\mathbf{S} | \mathbf{H}^{lm})$. Based on (3.20), we also have $P(\mathbf{H}) = P(\mathbf{H}^h | \mathbf{H}^{lm}) P(\mathbf{H}^{lm})$, and the estimation of \mathbf{H} given \mathbf{H}^{lm} is equivalent to the estimation of \mathbf{H}^h given \mathbf{H}^{lm} , we then reformulate probability $P(\mathbf{S}_1 | \mathbf{H}_1) P(\mathbf{H}_1)$ as

$$\begin{aligned} P(\mathbf{S}_1 | \mathbf{H}_1) P(\mathbf{H}_1) &= P(\mathbf{S}_1 | \mathbf{H}_1^{lm}) P(\mathbf{H}_1^h | \mathbf{H}_1^{lm}) P(\mathbf{H}_1^{lm}). \\ &= P(\mathbf{H}_1^{lm} | \mathbf{S}_1) P(\mathbf{H}_1 | \mathbf{H}_1^{lm}). \end{aligned} \quad (3.21)$$

We can rewrite probability $P(\mathbf{S}_1 | \mathbf{L}_1)$ as

$$P(\mathbf{S}_1 | \mathbf{L}_1) = P(\mathbf{L}_1 | \mathbf{S}_1) P(\mathbf{S}_1). \quad (3.22)$$

Probabilities $P(\mathbf{L}_1|\mathbf{S}_1)P(\mathbf{S}_1)$, $P(\mathbf{H}_1^{lm}|\mathbf{S}_1)$ and $P(\mathbf{H}_1|\mathbf{H}_1^{lm})$ sequentially constrain \mathbf{S}_1 , \mathbf{H}_1^{lm} and \mathbf{H}_1 . This leads to a two-step sequential solution. In the first step, by using a global image-based tensor, we can synthesize the low-resolution \mathbf{S}_1 . In the second step, after obtaining \mathbf{S}_1 , the \mathbf{H}_1^{lm} and \mathbf{H}_1 containing low-frequency, middle-frequency and high-frequency information can be computed using the local patch-based tensor. In addition, the final high-resolution \mathbf{H}_1 is computed by maximizing $P(\mathbf{H}_1|\mathbf{H}_1^{lm})$.

3.2.0.1 Global Low-Resolution Color Face Image Synthesis

In this section, the synthesis \mathbf{S}_1 is computed by maximizing probability $P(\mathbf{L}_1|\mathbf{S}_1)P(\mathbf{S}_1)$. Since \mathbf{L}_1 and \mathbf{S}_1 are the low-resolution given and synthesized face images with the same modality, we regard their relationship as Gaussian

$$P(\mathbf{L}_1|\mathbf{S}_1) = \frac{1}{f} \exp\{-\|\mathbf{S}_1 - \mathbf{L}_1\|^2/\lambda\} \quad (3.23)$$

where f is a normalization constant and λ scales the variance.

In (3.16), if we index into its basis subtensor at a particular modality m^1 , then the subtensor containing the individual image data as in (3.16) can be approximated by $\mathcal{G} = \mathcal{Z}_G \times_1 \mathbf{U}_{idens} \times_2 \mathbf{U}_{pixel1} \times_3 \mathbf{U}_{pixel2}$ and we get $\mathcal{G}_{m^1} = \mathcal{B}_{\mathcal{G}_{m^1}} \times_1 \mathbf{V}^T$. We unfold it into matrix representation and it becomes $\mathbf{G}_{m^1}^{(1)T} = \mathbf{B}_{\mathcal{G}_{m^1}^{(1)T}} \mathbf{V}$. Suppose $\mathbf{G}_{m^1}^{(1)T}$ correspond to color face images \mathbf{S}_1 , then we substitute for \mathbf{S}_1 in (3.23) resulting in

$$P(\mathbf{L}_1|\mathbf{S}_1) = \frac{1}{f} \exp\{-\|\mathbf{B}_{\mathcal{G}_{m^1}^{(1)T}} \mathbf{V} - \mathbf{L}_1\|^2/\lambda\} \quad (3.24)$$

In reality the given low-resolution \mathbf{L}_1 and \mathbf{S}_1 synthesized have the same modality. By setting $\mathbf{B}_{\mathcal{G}_{m^1}^{(1)T}} \mathbf{V} = \mathbf{L}_1$, we maximize (3.24) and approximately compute

$$\mathbf{V} = (\mathbf{B}_{\mathcal{G}_{m^1}^{(1)}} \mathbf{B}_{\mathcal{G}_{m^1}^{(1)T}})^{-1} \mathbf{B}_{\mathcal{G}_{m^1}^{(1)T}} \mathbf{L}_1 \quad (3.25)$$

where $(\mathbf{B}_{\mathcal{G}_{m^1}^{(1)}} \mathbf{B}_{\mathcal{G}_{m^1}^{(1)T}})^{-1} \mathbf{B}_{\mathcal{G}_{m^1}^{(1)T}}$ is the pseudoinverse of $\mathbf{B}_{\mathcal{G}_{m^1}^{(1)T}}$.

3.2.0.2 Local Patch-Based and High-Frequency Residue Recovery in Color Face Image Hallucination

To obtain their hallucinated high-resolution, we maximize $P(\mathbf{H}_1^{lm}|\mathbf{S}_1)$ using the local patch-based multiresolution tensor. The inference of \mathbf{H}_1^{lm} from \mathbf{S}_1 is independent. In the following, we take \mathbf{H}_1^{lm} as an example to illustrate this second process.

Since the training local multiresolution tensor is constructed from small overlapped patches, we decompose the synthesized \mathbf{S}_1 uniformly in the same way as decomposing training data, and factorize the likelihood $P(\mathbf{H}_1^{lm}|\mathbf{S}_1)$ at patch level as

$$P(\mathbf{H}_1^{lm}|\mathbf{S}_1) = \prod_{p_1, p_2=1}^N P(\mathbf{H}_{1_{p_1, p_2}}^{lm} | \mathbf{S}_{1_{p_1, p_2}}). \quad (3.26)$$

Assuming \mathbf{A} is the blurring and subsampling operator connecting $\mathbf{H}_{1_{p_1, p_2}}^{lm}$ and $\mathbf{S}_{1_{p_1, p_2}}$ in an imaging observation model, we regard these processes as Gaussian, therefore

$$P(\mathbf{H}_1^{lm} | \mathbf{S}_1) = \prod_{p_1, p_2=1}^N \frac{1}{w} \exp\{-\|\mathbf{A}\mathbf{H}_{1_{p_1, p_2}}^{lm} - \mathbf{S}_{1_{p_1, p_2}}\|^2 / \beta\} \quad (3.27)$$

where w is a normalization constant and β scales the variance.

Suppose the local multiresolution tensor in (3.16) has a basis tensor

$$\mathcal{B}_{\mathcal{L}} = \mathcal{Z}_{\mathcal{L}} \times_2 \mathbf{U}_{pixel1} \times_3 \mathbf{U}_{pixel2} \times_4 \mathbf{U}_{patch1} \times_5 \mathbf{U}_{patch2} \times_6 \mathbf{U}_{color} \times_7 \mathbf{U}_{resos}. \quad (3.28)$$

We index into this basis tensor at a particular resolution r and patch position p_1 and p_2 , yielding a basis subtensor

$$\mathcal{B}_{\mathcal{L}_{r, p_1, p_2}} = \mathcal{Z}_{\mathcal{L}} \times_2 \mathbf{U}_{pixel1} \times_3 \mathbf{U}_{pixel2} \times_6 \mathbf{U}_{color} \times_4 \mathbf{V}_{p_1}^T \times_5 \mathbf{V}_{p_2}^T \times_7 \mathbf{V}_r^T. \quad (3.29)$$

Then as described in Section (2), the subtensor containing the pixel data for that particular patch can be approximated as $\mathcal{D}_{r, p_1, p_2} = \mathcal{B}_{\mathcal{L}_{r, p_1, p_2}} \times_1 \mathbf{V}^T$, and its unfolded representation is $\mathbf{D}_{r, p_1, p_2}^{(1)T} = \mathbf{B}_{\mathbf{L}_{r, p_1, p_2}}^{(1)T} \mathbf{V}$. Similarly, we can obtain a subtensor for resolution r' of the same patch position, which is $\mathbf{L}_{r', p_1, p_2}^{(1)T} = \mathbf{B}_{\mathbf{L}_{r', p_1, p_2}}^{(1)T} \tilde{\mathbf{V}}$. Suppose $\mathbf{L}_{r, p_1, p_2}^{(1)T}$ and $\mathbf{L}_{r', p_1, p_2}^{(1)T}$ correspond to $\mathbf{S}_{1_{p_1, p_2}}$ and $\mathbf{H}_{1_{p_1, p_2}}^{lm}$, respectively; we substitute them in (3.27) as

$$P(\mathbf{H}_1^{lm} | \mathbf{S}_1) = \prod_{p_1, p_2=1}^N \frac{1}{w} \exp\{-\|\mathbf{A}\mathbf{B}_{\mathbf{L}_{r', p_1, p_2}}^{(1)T} \tilde{\mathbf{V}} - \mathbf{B}_{\mathbf{L}_{r, p_1, p_2}}^{(1)T} \mathbf{V}\|^2 / \beta\}. \quad (3.30)$$

We optimize the parameter $\tilde{\mathbf{V}}$ based on the construction properties of the local multiresolution patch tensor, which suggests that the relation between $\mathbf{B}_{\mathbf{L}_{r', p_1, p_2}}^{(1)T} \tilde{\mathbf{V}}$ and $\mathbf{B}_{\mathbf{L}_{r, p_1, p_2}}^{(1)T} \mathbf{V}$ observes a basic imaging observation model through the blurring and subsampling operator \mathbf{A} . This is consistent with the uniqueness of the identity parameter vector in a tensor space as well. By setting $\tilde{\mathbf{V}} = \mathbf{V}$, we can approximately compute $\mathbf{H}_{1_{p_1, p_2}}^{lm}$ as

$$\mathbf{H}_{1_{p_1, p_2}}^{lm} = \mathbf{B}_{\mathbf{L}_{r', p_1, p_2}}^{(1)T} \Psi \mathbf{S}_{1_{p_1, p_2}} \quad (3.31)$$

where Ψ is the pseudoinverse of $\mathbf{B}_{\mathbf{L}_{r, p_1, p_2}}^{(1)T}$ and is equal to $(\mathbf{B}_{\mathbf{L}_{r, p_1, p_2}}^{(1)T} \mathbf{B}_{\mathbf{L}_{r, p_1, p_2}}^{(1)T})^{-1} \mathbf{B}_{\mathbf{L}_{r, p_1, p_2}}^{(1)T}$. After reconstructing all the patches at different positions, the final hallucinated color face image \mathbf{H}_1^{lm} is simply a composition of the corresponding hallucinated small patches.

We recover the highest frequency part by patch learning from the high-resolution training data. The inference of \mathbf{H}_1 from \mathbf{H}_1^{lm} is independent. In the following, we take \mathbf{H}_1 as an example to illustrate how to hallucinate the final high-resolution face images.

We use a MRFs to model the \mathbf{H}_1 to be inferred. By decomposing into $\mathbf{H}_{1_q}^{lm}$ square patches

$$\begin{aligned} P(\mathbf{H}_1 | \mathbf{H}_1^{lm}) &= P(\mathbf{H}_1^{lm} | \mathbf{H}_1) P(\mathbf{H}_1) \\ &= \prod_{q=1}^Q P(\mathbf{H}_{1_q}^{lm} | \mathbf{H}_{1_q}) P(\mathbf{H}_1). \end{aligned} \quad (3.32)$$

The difference between \mathbf{H}_1 and \mathbf{H}_1^{lm} is the high-frequency band information. Since the high-frequency information depends on the lower-frequency band, we use the Laplacian image $\mathbf{L}_{\mathbf{H}_1^{lm}}$ of \mathbf{H}_1^{lm} to represent the middle-frequency band. To infer \mathbf{H}_1 , we use the sum of squared differences of Laplacian images as metrics to model $\prod_{q=1}^Q P(\mathbf{H}_{1q}^{lm} | \mathbf{H}_{1q})$ as

$$\prod_{q=1}^Q P(\mathbf{H}_{1q}^{lm} | \mathbf{H}_{1q}) \propto \prod_{q=1}^Q \exp\{-\|\mathbf{L}_{\mathbf{H}_{1q}^{lm}} - \mathbf{L}_{\mathbf{H}_{1q}^{(t)}}\|^2\} \quad (3.33)$$

where $\mathbf{L}_{\mathbf{H}_{1q}^{(t)}}$ are the Laplacian images from high-resolution training face images. Comparing the Laplacian images $\mathbf{L}_{\mathbf{H}_{1q}^{lm}}$ with $\{\mathbf{L}_{\mathbf{H}_{1q}^{(t)}}\}_{t=1}^T$ from the training dataset, the patch $\mathbf{H}_{1q}^{(t)}$ with $\mathbf{L}_{\mathbf{H}_{1q}^{(t)}}$ closest to $\mathbf{L}_{\mathbf{H}_{1q}^{lm}}$ is the most probable to be chosen as \mathbf{H}_{1q} . Since we model the high-resolution image as a MRFs, based on the HammersleyClifford theorem, $P(\mathbf{H}_1)$ is a product $\prod_{\mathbf{H}_{1q}, \mathbf{H}_{1\bar{q}}} \Phi(\mathbf{H}_{1q}, \mathbf{H}_{1\bar{q}})$ of compatibility functions $\Phi(\mathbf{H}_{1q}, \mathbf{H}_{1\bar{q}})$ over all neighboring pairs, where $\mathbf{H}_{1q}, \mathbf{H}_{1\bar{q}}$ are one of the neighboring patch pairs in a 4-neighbor system.

The compatibility function $\Phi(\mathbf{H}_{1q}, \mathbf{H}_{1\bar{q}})$ is defined using the similarity of pixel values on the overlapping area of the neighboring patches:

$$\Phi(\mathbf{H}_{1q}, \mathbf{H}_{1\bar{q}}) \propto \exp\{-\|O_{\mathbf{H}_{1q}} - O_{\mathbf{H}_{1\bar{q}}}\|^2\}, \quad (3.34)$$

where $O_{\mathbf{H}_{1q}}$ denotes the pixels of patch \mathbf{H}_{1q} overlapping with neighboring patch $\mathbf{H}_{1\bar{q}}$, and vice versa for $O_{\mathbf{H}_{1\bar{q}}}$. We illustrate this 4-neighbor system and the corresponding patch overlapping relations, then \mathbf{H}_1 estimated as

$$\operatorname{argmax}_{\mathbf{H}_1} \prod_{q=1}^Q P(\mathbf{H}_{1q}^{lm} | \mathbf{H}_{1q}) \prod_{(q, \bar{q})} \Phi(\mathbf{H}_{1q}, \mathbf{H}_{1\bar{q}}). \quad (3.35)$$

Solving probabilistic (3.35) to obtain \mathbf{H}_1 is not a trivial task. We use the iterated conditional modes (ICM) algorithm [77]. More specifically, we maximize $P(\mathbf{H}_{1q}^{lm} | \mathbf{H}_{1q})$ for all patch positions $q \in \{1, \dots, Q\}$ to yield the initial maximum likelihood estimate of . Based on this initial estimate, we then pick a random patch position q and update the estimate of \mathbf{H}_{1q} using the current estimates of its neighbors $\mathbf{H}_{1\bar{q}}$ by maximizing $P(\mathbf{H}_{1q}^{lm} | \mathbf{H}_{1q}) \prod_{(q, \bar{q})} \Phi(\mathbf{H}_{1q}, \mathbf{H}_{1\bar{q}})$. We repeat this random patch selection and updating process until converging to the final high-resolution image \mathbf{H}_1 .

CHAPTER IV

THE EXPERIMENTAL RESULTS

4.1 Image Databases

We used face images from a subset of FERET databases to form two data sets for training and testing images in four color models, which are RGB, YCbCr, HSV and CIELAB. The experiments are conducted with a large number of face images from FERET data set [78,79] and other collections, which consist of many different races, illuminations and types of face images.

4.1.1 FERET Database

The FERET database [78,79] contains 1199 individuals and 365 duplicate sets of images. There are images per subject, one for each of the following facial expressions or configurations: centerlight, with glasses, happy, left-light, without glasses, normal, right-light, sad, sleepy, surprised, and wink. All sample images of one person from the FERET database are shown in Fig. 4.3.

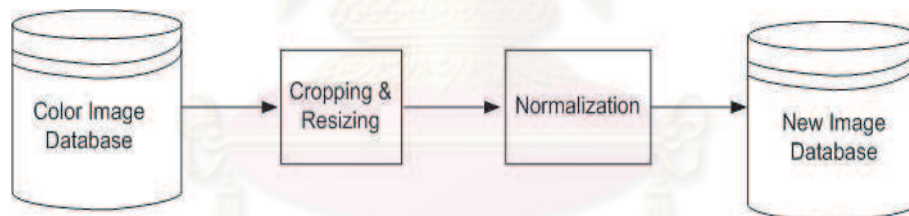


Figure 4.1 Preprocessing diagram

4.2 Preprocessing

In some databases, we notice that the background, some possible transformations of the object (scaling, rotation and translation) and sensor-dependent variations (for example, automatic gain control calibration and bad lens points) could undermine the face hallucination performance. This impact can be minimized by cropping and normalization. The preprocessing of this dissertation is following to the diagram in Fig. 4.1.

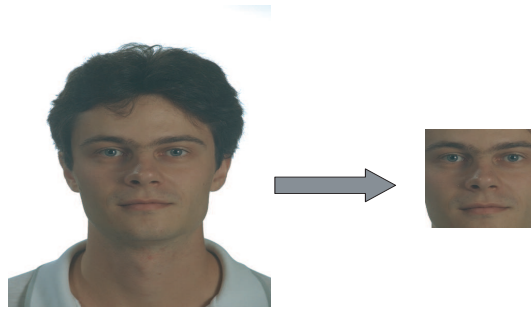


Figure 4.2 Cropping image

4.2.1 Cropping and Resizing

In this dissertation, the cropping procedure was manually implemented by human. Each image was manually cropped and resized to 30×30 pixels. By attempting to align images such that the faces are the same size, in the same position and at the same orientation. Specifically, the image is scaled and translated to make the eye coordinates coincident with pre-specified locations in the output. After cropping, all of the images were resized to same dimensions by linear interpolation. The cropping for a sample image on FERET database was shown in Fig. 4.2.



Figure 4.3 Example training faces from FERET database in our proposed algorithm

4.2.2 Normalization

The normalization is to compensate for intensity variations. By

$$\mathbf{A}' = \frac{\mathbf{A}}{\|\text{vec}(\mathbf{A})\|} \quad (4.1)$$

where \mathbf{A} is the original image matrix, \mathbf{A}' is the normalized image matrix and $\|\text{vec}(\mathbf{A})\|$ represents to the norm of the vectorization of the image matrix.

4.3 Experiments and Analysis on Color Face hallucination with TensorPCA subspace

In this section, we experimentally evaluate our proposed technique by using TensorPCA methods. In our experiments, we randomly select 500 normal expression images of different persons on the same light condition and other 50 images are used for testing. According to demand, we manually crop the interesting region of the faces and unify the images to the size of (30×30) .

In the degradation process, each testing image (LR) is introduced with Gaussian blur with variance 1 and resized by down-sampling 2:1 (15×15), then we add Gaussian noise with variance 10^{-6} . To establish a standard training data set, we aligned these face images manually by hand, marking the location of 3 points: the centers of the eyeballs and the lower tip of the nose. These 3 points define an affine warp, which is used to warp the images into a canonical form. We use Peak signal-to-noise ratio (PSNR) to evaluate the performance of the facial reconstruction.

We compare the hallucination results between the traditional PCA and tensorPCA by vary both of the number of principal component from 90 to 100 percent PCA. The experimental results are shown in Fig. 4.4-4.7: (a) original HR images (30×30), (b) input LR images (15×15) with noise, motion and blur in LR images, (c)-(e) face hallucination result with 90, 95 and 100 percent traditional PCA respectively, (f)-(h) face hallucination result with 90, 95 and 100 percent tensorPCA respectively.

From the traditional PCA method, the color face image has to be converted to a vector representation. Then, we can see that from the hallucinated results in Fig. 4.4-4.7 (c)-(e), it can hardly maintain global smoothness and visual rationality, especially on location around color face contour and margin of the nose and the mouth. In addition, the results have some noise around the eyes and mouth. On the other hand, the hallucinated results from the tensorPCA method in Fig. 4.4-4.7 (f)-(h) can reconstruct the reasonable color face images which are compared with the ground truth color face images in Fig. 4.4-4.7 (a). In Fig. 4.4-4.7, the outcomes we get in HSV color space show that the color distort from the original HR images. Additionally, the details in our hallucination results such as eyes, noses, lips and eyebrows quite differ from the original HR images.

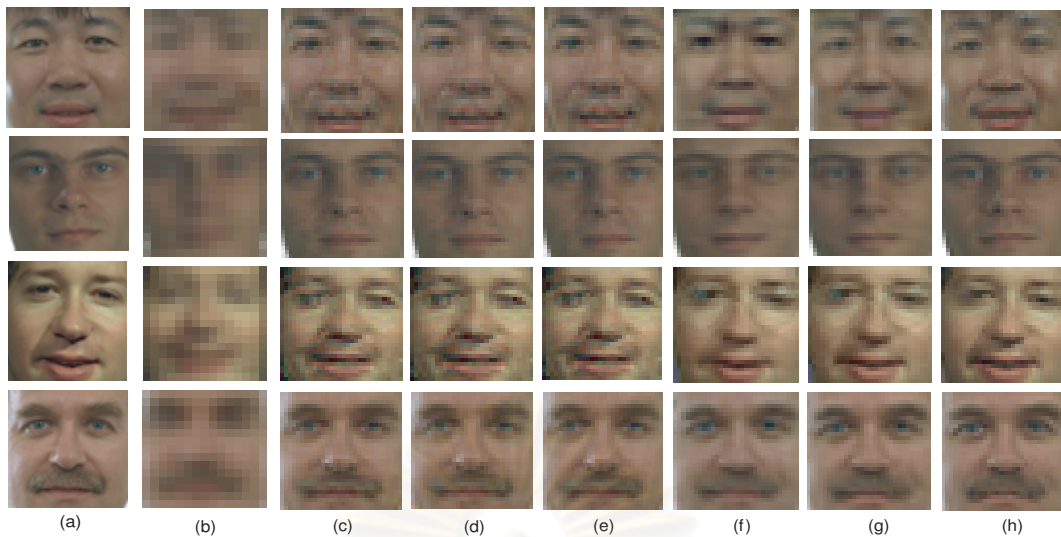


Figure 4.4: Color hallucinated face images in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent traditional PCA; (d) face hallucination result with 95 percent traditional PCA; (e) face hallucination result with 100 percent traditional PCA; (f) face hallucination result with 90 percent tensorPCA; (g) face hallucination result with 95 percent tensorPCA; (h) face hallucination result with 100 percent tensorPCA;

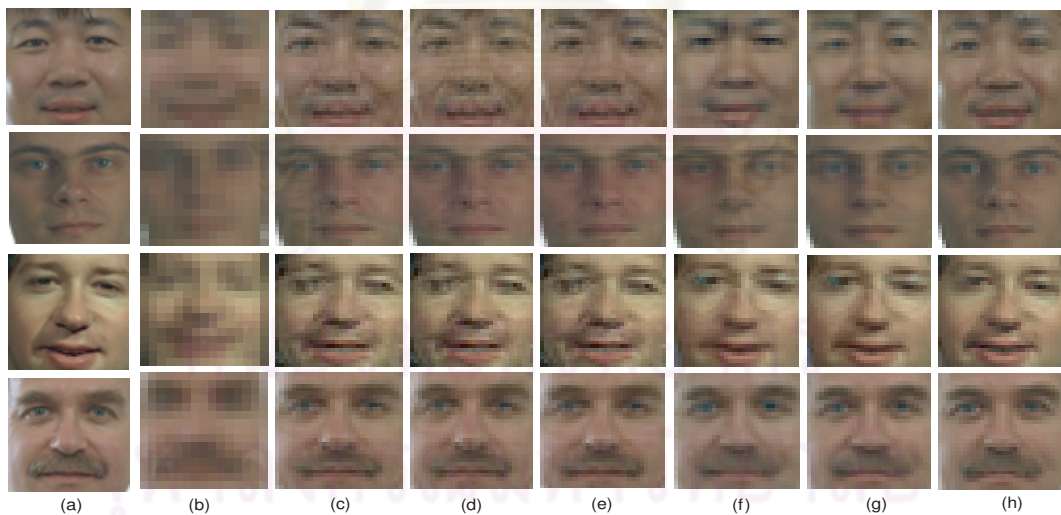


Figure 4.5: Color hallucinated face images in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent traditional PCA; (d) face hallucination result with 95 percent traditional PCA; (e) face hallucination result with 100 percent traditional PCA; (f) face hallucination result with 90 percent tensorPCA; (g) face hallucination result with 95 percent tensorPCA; (h) face hallucination result with 100 percent tensorPCA;

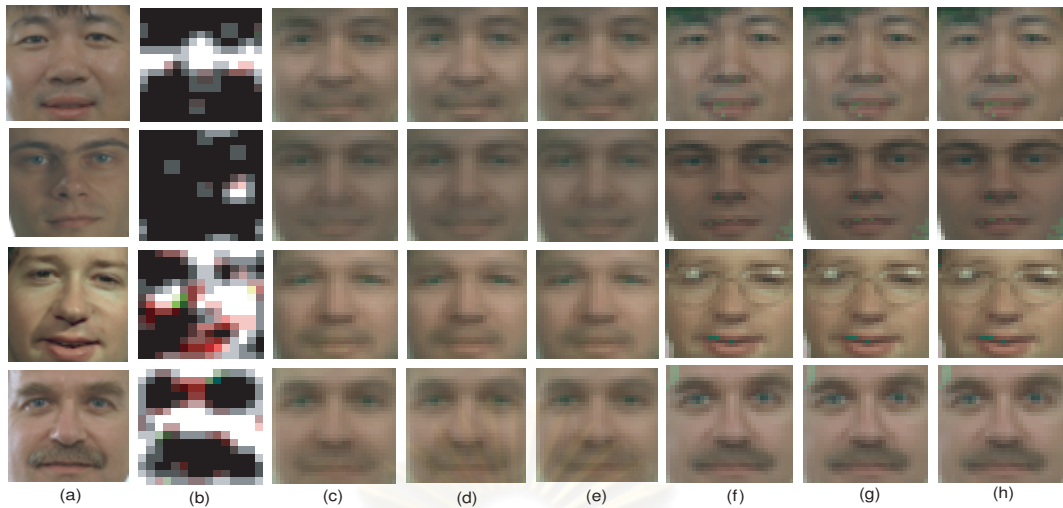


Figure 4.6: Color hallucinated face images in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent traditional PCA; (d) face hallucination result with 95 percent traditional PCA; (e) face hallucination result with 100 percent traditional PCA; (f) face hallucination result with 90 percent tensorPCA; (g) face hallucination result with 95 percent tensorPCA; (h) face hallucination result with 100 percent tensorPCA;



Figure 4.7: Color hallucinated face images in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent traditional PCA; (d) face hallucination result with 95 percent traditional PCA; (e) face hallucination result with 100 percent traditional PCA; (f) face hallucination result with 90 percent tensorPCA; (g) face hallucination result with 95 percent tensorPCA; (h) face hallucination result with 100 percent tensorPCA;

Table 4.1: PSNR results of the facial images with traditional PCA and tensorPCA methods in Fig. 4.4 for RGB color model.

Case	first row	second row	third row	fourth row
1) 90 percent PCA	24.05 dB	30.13 dB	23.40 dB	23.98 dB
2) 95 percent PCA	24.13 dB	30.39 dB	23.77 dB	24.09 dB
3) 100 percent PCA	24.34 dB	30.67 dB	23.98 dB	24.32 dB
4) 90 percent tensorPCA	26.63 dB	35.63 dB	27.61 dB	29.88 dB
5) 95 percent tensorPCA	30.96 dB	36.69 dB	28.33 dB	30.60 dB
6) 100 percent tensorPCA	31.18 dB	40.04 dB	28.61 dB	30.82 dB

Table 4.2: PSNR results of the facial images with traditional PCA and tensorPCA methods in Fig. 4.5 for YCbCr color model.

Case	first row	second row	third row	fourth row
1) 90 percent PCA	24.13 dB	30.21 dB	23.61 dB	24.07 dB
2) 95 percent PCA	24.25 dB	30.51 dB	23.98 dB	24.22 dB
3) 100 percent PCA	24.58 dB	30.88 dB	24.22 dB	24.64 dB
4) 90 percent tensorPCA	26.71 dB	36.08 dB	27.67 dB	30.09 dB
5) 95 percent tensorPCA	30.98 dB	36.87 dB	28.45 dB	30.58 dB
6) 100 percent tensorPCA	31.32 dB	40.13 dB	28.64 dB	30.78 dB

Table 4.3: PSNR results of the facial images with traditional PCA and tensorPCA methods in Fig. 4.6 for HSV color model.

Case	first row	second row	third row	fourth row
1) 90 percent PCA	22.86 dB	26.85 dB	22.51 dB	22.77 dB
2) 95 percent PCA	23.02 dB	27.04 dB	22.87 dB	22.91 dB
3) 100 percent PCA	23.24 dB	27.63 dB	23.06 dB	23.12 dB
4) 90 percent tensorPCA	26.58 dB	30.91 dB	27.60 dB	29.34 dB
5) 95 percent tensorPCA	27.19 dB	31.05 dB	27.79 dB	29.51 dB
6) 100 percent tensorPCA	27.58 dB	31.12 dB	27.85 dB	29.60 dB

Table 4.4: PSNR results of the facial images with traditional PCA and tensorPCA methods in Fig. 4.7 for CIELAB color model.

Case	first row	second row	third row	fourth row
1) 90 percent PCA	24.42 dB	30.66 dB	23.82 dB	24.13 dB
2) 95 percent PCA	24.67 dB	31.03 dB	24.01 dB	24.36 dB
3) 100 percent PCA	24.93 dB	31.78 dB	24.34 dB	24.79 dB
4) 90 percent tensorPCA	29.48 dB	36.41 dB	27.90 dB	30.45 dB
5) 95 percent tensorPCA	31.54 dB	37.03 dB	28.28 dB	31.42 dB
6) 100 percent tensorPCA	31.97 dB	40.29 dB	28.60 dB	31.63 dB

In Table. 4.1-4.4, we show the PSNR results from the hallucinated images in Fig. 4.4-4.7. We can see that the hallucination method with tensorPCA has more PSNR values compared with the traditional PCA method. In RGB color model, at the same percentage of eigenvalues 90, 95 and 100, tensorPCA method can give more higher the PSNR values than the traditional PCA about 2.58-5.90 dB, 4.56-6.83 dB and 4.63-9.37 dB respectively. However, in HSV color space the PSNR results in Table. 4.1-4.4, the tensor PCA method can also give more higher the PSNR values than the traditional PCA about 3.72-6.57 dB, 4.01-6.60 dB and 3.49-6.48 dB at the same percentage of eigenvalues 90, 95 and 100 respectively. Moreover, the PSNR results in CIELAB color space are the best performance among color spaces.

4.4 Experiments and Analysis on Color Face hallucination with Linear Regression Model in MPCA

In this experiment, the MPCA is used to perform principal component analysis of the training images and we investigate the performance of our proposed method in sense of impact of number of eigenvalues, impact of training set size, robustness to noise and complexity respectively.

4.4.1 Impact of Number of Eigenvalues

The MPCA is used to perform principal component analysis of the training images. Some sample results which compared between the tensorPCA and MPCA are shown in Fig. 4.8-4.11. The figures are organized as followed: column (a) the original HR (30×30) color images; (b) input the LR color images (15×15) with noise, motion and blur; (c)-(e) face hallucination results with Traditional PCA method; (f)-(h) face hallucination results with linear regression model in MPCA and the number of eigenvalue is varied from 90 to 100 percent. Compared with the input image and the traditional PCA method result, the hallucinated face images from the linear regression model in MPCA have much clearer detail features. As shown in Fig. 4.8-4.11 (c), (d) and (e), with traditional PCA method, we can observe that dirty disturbance in the global reconstructed images and the results have some noise around the eyes and mouth. In addition, all the PSNR results in different both color space and the number of PCA are shown in Table. 4.5-4.8. As we can see, the PSNR values from the MPCA method are significantly higher than the traditional PCA method.

Additionally, we compare between the hallucination method with linear regression model in MPCA and other traditional methods such as bilinear interpolation and Liu method [80]. All the results are shown in Fig. 4.12-4.15: (a) the original HR (30×30) color images; (b) input the LR color images (15×15) with noise, motion and blur; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method and (e)-(g) face hallucination results with linear regression model in MPCA and the number of eigenvalue is varied from 90 to 100 percent. We can see that the performance of hallucination by our proposed method is much better. Likewise, the hallucination results from bilinear interpolation method, which are displayed in Fig. 4.12-4.15 (c), cannot reconstruct the facial images because this method is unable to solve noise, motion and blur problems.

Noticed from the figures, the performance of our proposed algorithm depends on the number of the PCA. We can remark from the facial results in Fig. 4.8-4.15 that they tend to produce sharper facial features, clear eyelids and mouth. In particular, if our algorithm is implemented with 100 percent of PCA, the results will become similar to the original HR facial images. The outcomes we get in HSV color space show that the color distort from the original HR images. Additionally, the details in our hallucination results such as eyes, noses,

lips and eyebrows quite differ from the original HR images.

We also compare the performance which are shown in Fig. 4.16-4.19 between tensor-PCA and MPCA method. Since the the tensorPCA method does not realize the correlation between each color channel in a color system. For this reason, in Table. 4.9-4.12, in each hallucinated facial image from our proposed technique has more the PSNR values than the tensorPCA method about 0.2-0.3 dB. All the PSNR results in different both color space and the number of PCA are shown in Table. 4.9-4.10. We can see that our method has the highest PSNR values compared with other methods on all test faces. The PSNR results in CIELAB color space are the best performance among color spaces. However, in HSV color space the PSNR results in Table. 4.9-4.12 are less favorable than other color spaces.



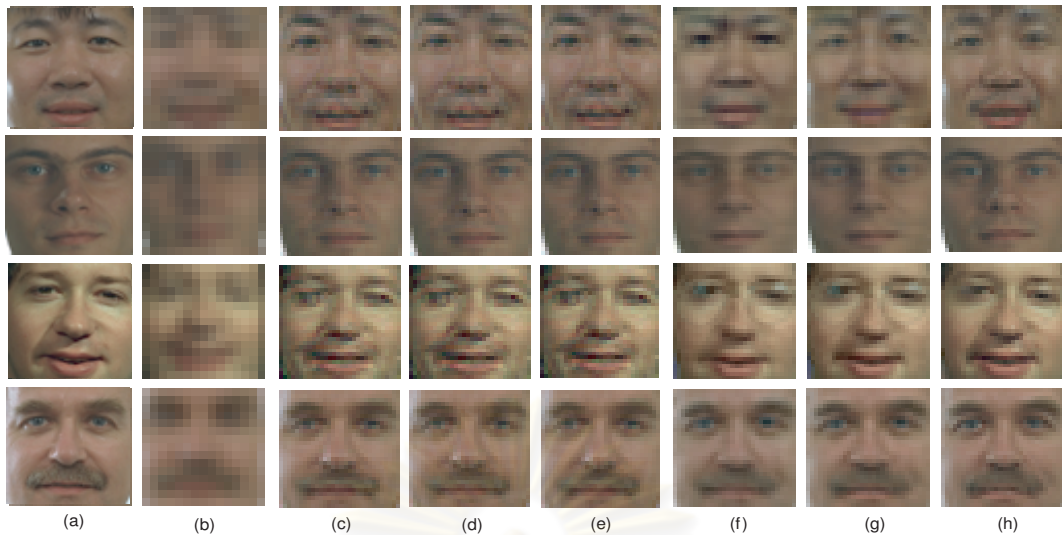


Figure 4.8: Color hallucinated face images in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent traditional PCA; (d) face hallucination result with 95 percent traditional PCA; (e) face hallucination result with 100 percent traditional PCA; (f) face hallucination result with 90 percent MPCA; (g) face hallucination result with 95 percent MPCA; (h) face hallucination result with 100 percent MPCA;

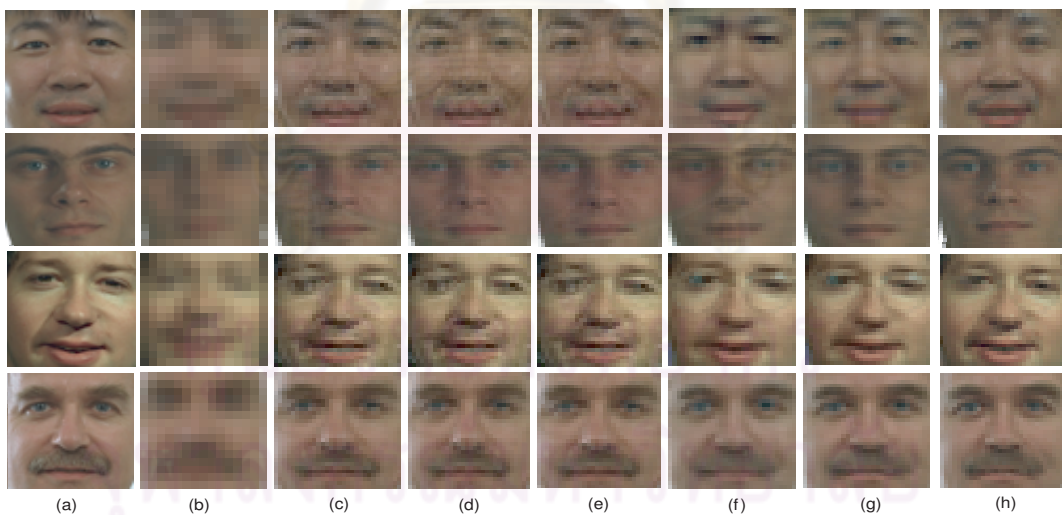


Figure 4.9: Color hallucinated face images in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent traditional PCA; (d) face hallucination result with 95 percent traditional PCA; (e) face hallucination result with 100 percent traditional PCA; (f) face hallucination result with 90 percent MPCA; (g) face hallucination result with 95 percent MPCA; (h) face hallucination result with 100 percent MPCA;



Figure 4.10: Color hallucinated face images in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent traditional PCA; (d) face hallucination result with 95 percent traditional PCA; (e) face hallucination result with 100 percent traditional PCA; (f) face hallucination result with 90 percent MPCA; (g) face hallucination result with 95 percent MPCA; (h) face hallucination result with 100 percent MPCA;



Figure 4.11: Color hallucinated face images in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent traditional PCA; (d) face hallucination result with 95 percent traditional PCA; (e) face hallucination result with 100 percent traditional PCA; (f) face hallucination result with 90 percent MPCA; (g) face hallucination result with 95 percent MPCA; (h) face hallucination result with 100 percent MPCA;



Figure 4.12: Color hallucinated face images, compared with other traditional methods in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA;

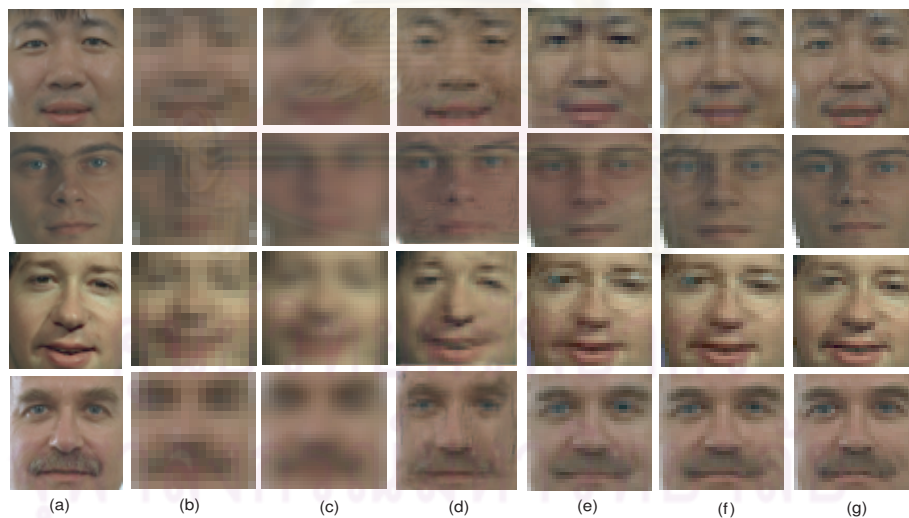


Figure 4.13: Color hallucinated face images, compared with other traditional methods in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA;

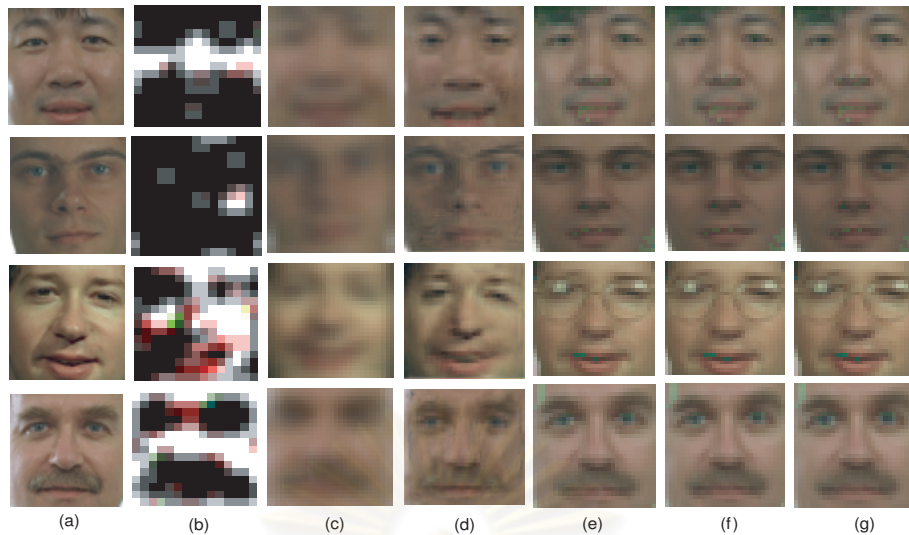


Figure 4.14: Color hallucinated face images, compared with other traditional methods in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA;



Figure 4.15: Color hallucinated face images, compared with other traditional methods in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA;

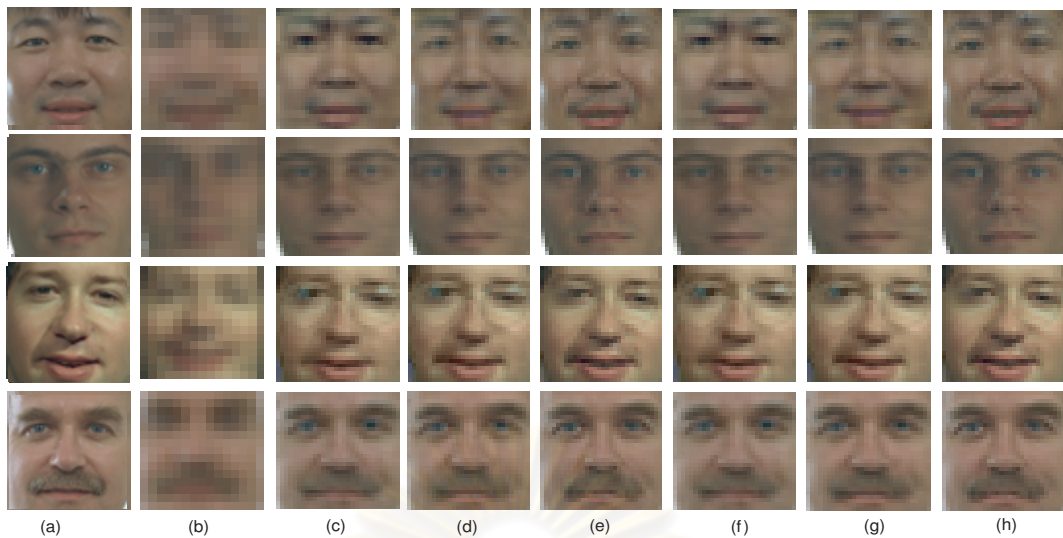


Figure 4.16: Color hallucinated face images, compared between TensorPCA and MPCA in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent tensorPCA; (d) face hallucination result with 95 percent tensorPCA; (e) face hallucination result with 100 percent tensorPCA; (f) face hallucination result with 90 percent MPCA; (g) face hallucination result with 95 percent MPCA; (h) face hallucination result with 100 percent MPCA;



Figure 4.17: Color hallucinated face images, compared between TensorPCA and MPCA in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent tensorPCA; (d) face hallucination result with 95 percent tensorPCA; (e) face hallucination result with 100 percent tensorPCA; (f) face hallucination result with 90 percent MPCA; (g) face hallucination result with 95 percent MPCA; (h) face hallucination result with 100 percent MPCA;

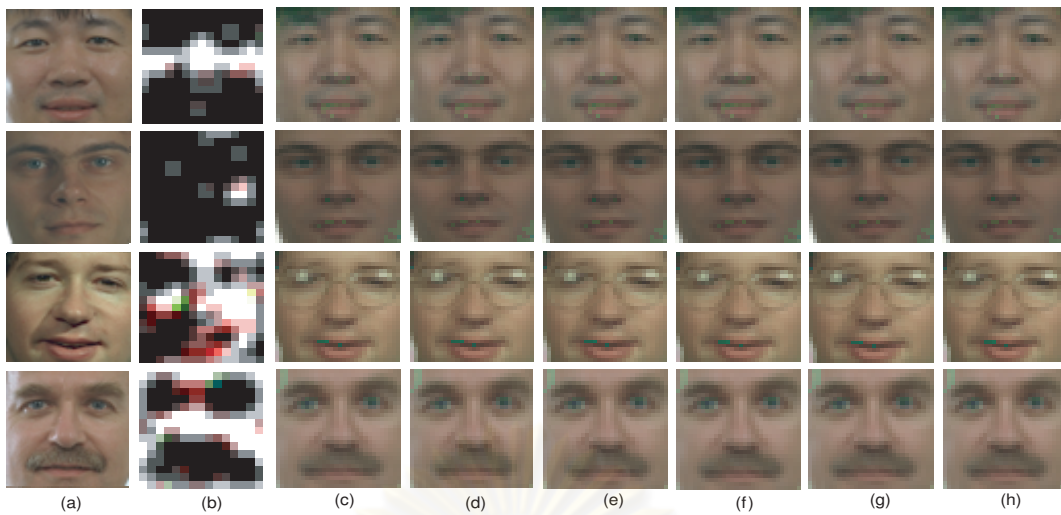


Figure 4.18: Color hallucinated face images, compared between TensorPCA and MPCA in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent tensorPCA; (d) face hallucination result with 95 percent tensorPCA; (e) face hallucination result with 100 percent tensorPCA; (f) face hallucination result with 90 percent MPCA; (g) face hallucination result with 95 percent MPCA; (h) face hallucination result with 100 percent MPCA;



Figure 4.19: Color hallucinated face images, compared between TensorPCA and MPCA in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent tensorPCA; (d) face hallucination result with 95 percent tensorPCA; (e) face hallucination result with 100 percent tensorPCA; (f) face hallucination result with 90 percent MPCA; (g) face hallucination result with 95 percent MPCA; (h) face hallucination result with 100 percent MPCA;

Table 4.5: PSNR results of the facial images with traditional PCA and MPCA methods in Fig. 4.8 for RGB color model.

Case	first row	second row	third row	fourth row
1) 90 percent PCA	24.05 dB	30.13 dB	23.40 dB	23.98 dB
2) 95 percent PCA	24.13 dB	30.39 dB	23.77 dB	24.09 dB
3) 100 percent PCA	24.34 dB	30.67 dB	23.98 dB	24.32 dB
4) 90 percent MPCA	26.88 dB	35.91 dB	27.89 dB	30.14 dB
5) 95 percent MPCA	31.15 dB	36.91 dB	28.53 dB	30.78 dB
6) 100 percent MPCA	31.35 dB	40.23 dB	28.74 dB	30.98 dB

Table 4.6: PSNR results of the facial images with traditional PCA and MPCA methods in Fig. 4.9 for YCbCr color model.

Case	first row	second row	third row	fourth row
1) 90 percent PCA	24.13 dB	30.21 dB	23.61 dB	24.07 dB
2) 95 percent PCA	24.25 dB	30.51 dB	23.98 dB	24.22 dB
3) 100 percent PCA	24.58 dB	30.88 dB	24.22 dB	24.64 dB
4) 90 percent MPCA	27.03 dB	36.38 dB	27.93 dB	30.13 dB
5) 95 percent MPCA	31.24 dB	37.04 dB	28.57 dB	30.69 dB
6) 100 percent MPCA	31.47 dB	40.29 dB	28.84 dB	30.88 dB

4.4.2 Impact of training set size

In Fig. 4.20 - 4.22 show examples of hallucinated face images based on a different number of training samples. There is not much difference between the results using 120 and 240 training samples. This shows that our hallucination algorithm can achieve satisfactory results even based on a relatively small training set. However, when the training set is too small, 60 training samples, many individual characteristics cannot be rendered.

Table 4.7: PSNR results of the facial images with traditional PCA and MPCA methods in Fig. 4.10 for HSV color model.

Case	first row	second row	third row	fourth row
1) 90 percent PCA	22.86 dB	26.85 dB	22.51 dB	22.77 dB
2) 95 percent PCA	23.02 dB	27.04 dB	22.87 dB	22.91 dB
3) 100 percent PCA	23.24 dB	27.63 dB	23.06 dB	23.12 dB
4) 90 percent MPCA	26.85 dB	31.02 dB	27.71 dB	29.41 dB
5) 95 percent MPCA	27.28 dB	31.18 dB	27.87 dB	29.59 dB
6) 100 percent MPCA	27.75 dB	31.22 dB	27.96 dB	29.63 dB

Table 4.8: PSNR results of the facial images with traditional PCA and MPCA methods in Fig. 4.11 for CIELAB color model.

Case	first row	second row	third row	fourth row
1) 90 percent PCA	24.42 dB	30.66 dB	23.82 dB	24.13 dB
2) 95 percent PCA	24.67 dB	31.03 dB	24.01 dB	24.36 dB
3) 100 percent PCA	24.93 dB	31.78 dB	24.34 dB	24.79 dB
4) 90 percent MPCA	29.59 dB	36.48 dB	28.01 dB	30.58 dB
5) 95 percent MPCA	31.66 dB	37.11 dB	28.35 dB	31.52 dB
6) 100 percent MPCA	32.08 dB	40.35 dB	28.69 dB	31.69 dB

Table 4.9: PSNR results of the facial images with tensorPCA and MPCA methods in Fig. 4.16 for RGB color model.

Case	first row	second row	third row	fourth row
1) 90 percent tensorPCA	26.63 dB	35.63 dB	27.61 dB	29.88 dB
2) 95 percent tensorPCA	30.96 dB	36.69 dB	28.33 dB	30.60 dB
3) 100 percent tensorPCA	31.18 dB	40.04 dB	28.61 dB	30.82 dB
4) 90 percent MPCA	26.88 dB	35.91 dB	27.89 dB	30.14 dB
5) 95 percent MPCA	31.15 dB	36.91 dB	28.53 dB	30.78 dB
6) 100 percent MPCA	31.35 dB	40.23 dB	28.74 dB	30.98 dB

Table 4.10: PSNR results of the facial images with tensorPCA and MPCA methods in Fig. 4.17 for YCbCr color model.

Case	first row	second row	third row	fourth row
1) 90 percent tensorPCA	26.71 dB	36.08 dB	27.67 dB	30.09 dB
2) 95 percent tensorPCA	30.98 dB	36.87 dB	28.45 dB	30.58 dB
3) 100 percent tensorPCA	31.32 dB	40.13 dB	28.64 dB	30.78 dB
4) 90 percent MPCA	27.03 dB	36.38 dB	27.93 dB	30.13 dB
5) 95 percent MPCA	31.24 dB	37.04 dB	28.57 dB	30.69 dB
6) 100 percent MPCA	31.47 dB	40.29 dB	28.84 dB	30.88 dB

Table 4.11: PSNR results of the facial images with tensorPCA and MPCA methods in Fig. 4.18 for HSV color model.

Case	first row	second row	third row	fourth row
1) 90 percent tensorPCA	26.58 dB	30.91 dB	27.60 dB	29.34 dB
2) 95 percent tensorPCA	27.19 dB	31.05 dB	27.79 dB	29.51 dB
3) 100 percent tensorPCA	27.58 dB	31.12 dB	27.85 dB	29.60 dB
4) 90 percent MPCA	26.85 dB	31.02 dB	27.71 dB	29.41 dB
5) 95 percent MPCA	27.28 dB	31.18 dB	27.87 dB	29.59 dB
6) 100 percent MPCA	27.75 dB	31.22 dB	27.96 dB	29.63 dB

Table 4.12: PSNR results of the facial images with tensorPCA and MPCA methods in Fig. 4.19 for CIELAB color model.

Case	first row	second row	third row	fourth row
1) 90 percent tensorPCA	29.48 dB	36.41 dB	27.90 dB	30.45 dB
2) 95 percent tensorPCA	31.54 dB	37.03 dB	28.28 dB	31.42 dB
3) 100 percent tensorPCA	31.97 dB	40.29 dB	28.60 dB	31.63 dB
4) 90 percent MPCA	29.59 dB	36.48 dB	28.01 dB	30.58 dB
5) 95 percent MPCA	31.66 dB	37.11 dB	28.35 dB	31.52 dB
6) 100 percent MPCA	32.08 dB	40.35 dB	28.69 dB	31.69 dB

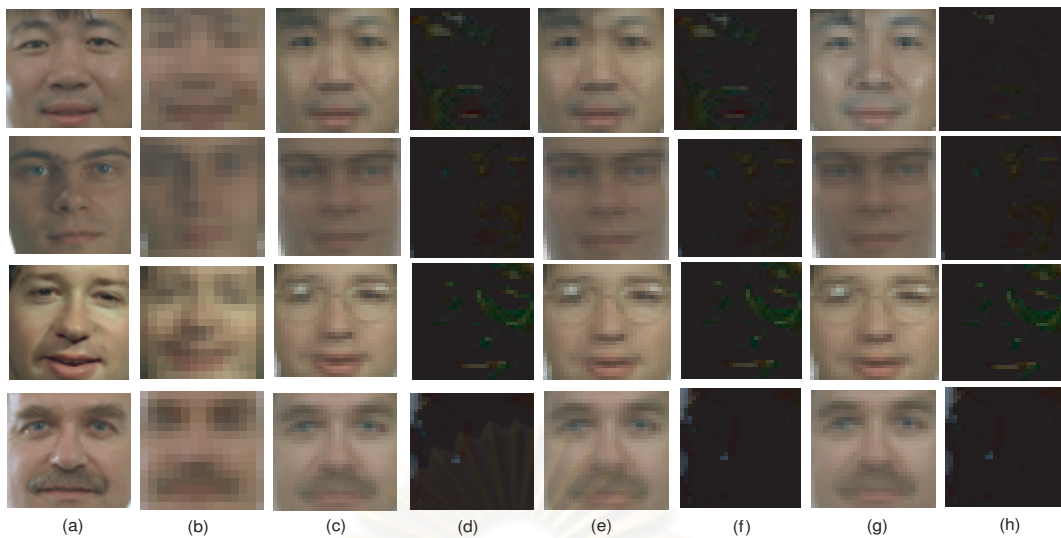


Figure 4.20: Color hallucinated face images with training set size 60 images in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

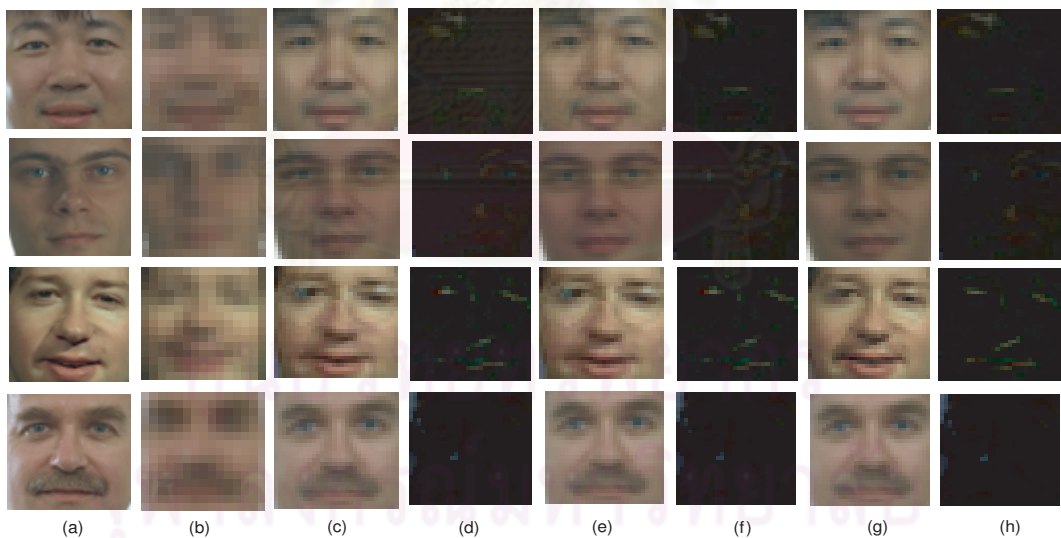


Figure 4.21: Color hallucinated face images with training set size 120 images in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

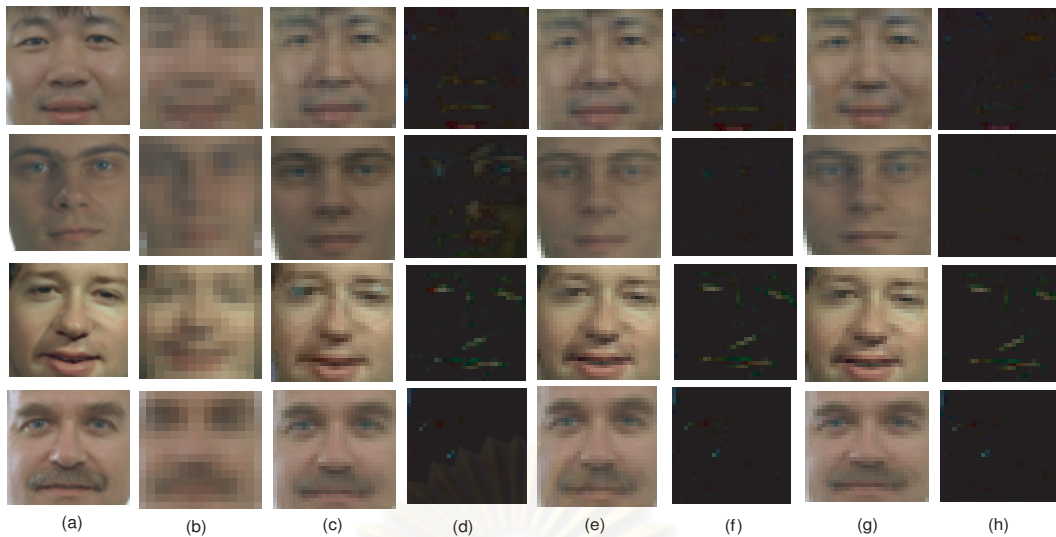


Figure 4.22: Color hallucinated face images with training set size 240 images in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

4.4.3 Robustness to Noise

In this section, we add zero mean Gaussian noise with variance 10^{-6} , 2.5×10^{-5} , 2.5×10^{-4} and 10^{-3} to low-resolution face images. All the results are shown in Fig. 4.23-4.38: (a) the original HR (30×30) color images; (b) input the LR color images (15×15) with different noise variance ; (c), (e) and (g) face hallucination result with MPCA method; (d), (f) and (h) different image of face hallucination results. We can observe that the reconstructed color face images can remove most of the noise distortion and retain most of the facial characteristics.

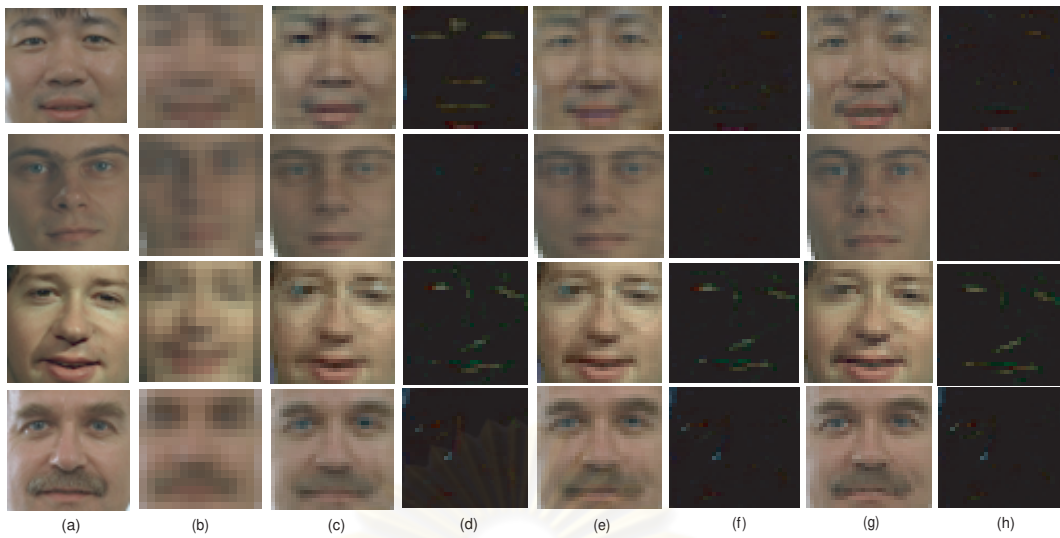


Figure 4.23: Color hallucinated face images with noise variance 10^{-6} in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

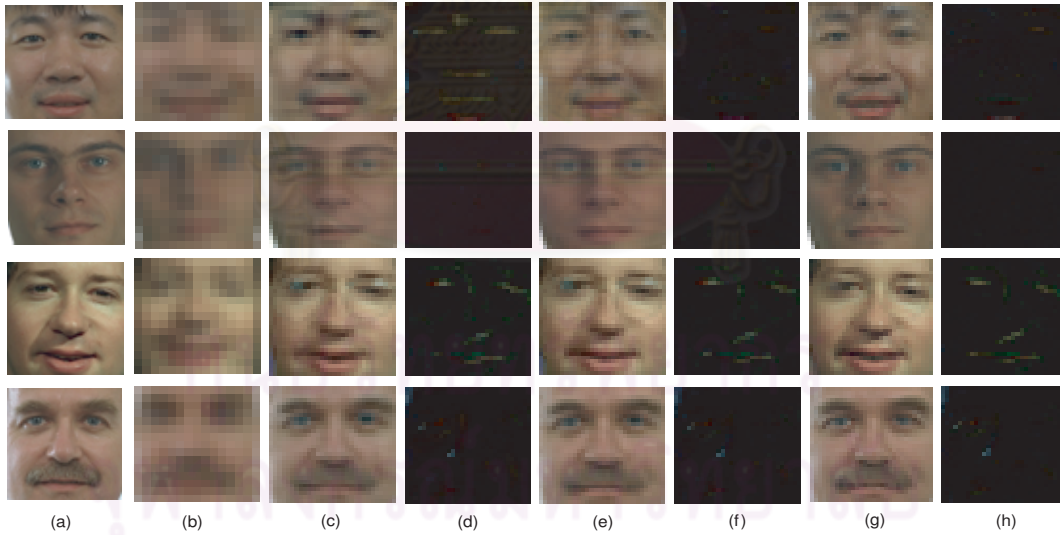


Figure 4.24: Color hallucinated face images with noise variance 2.5×10^{-5} in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-5} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

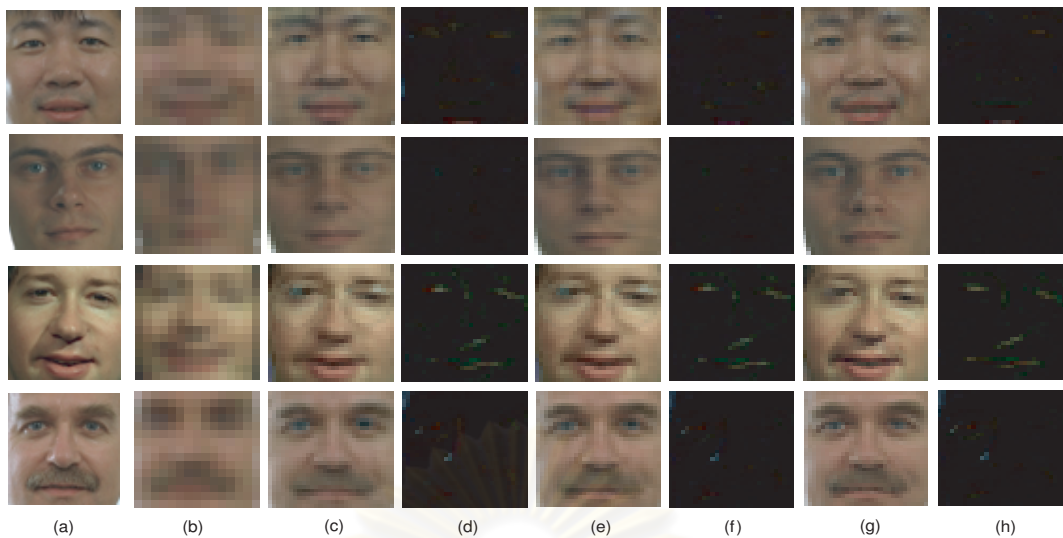


Figure 4.25: Color hallucinated face images with noise variance 2.5×10^{-4} in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-4} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

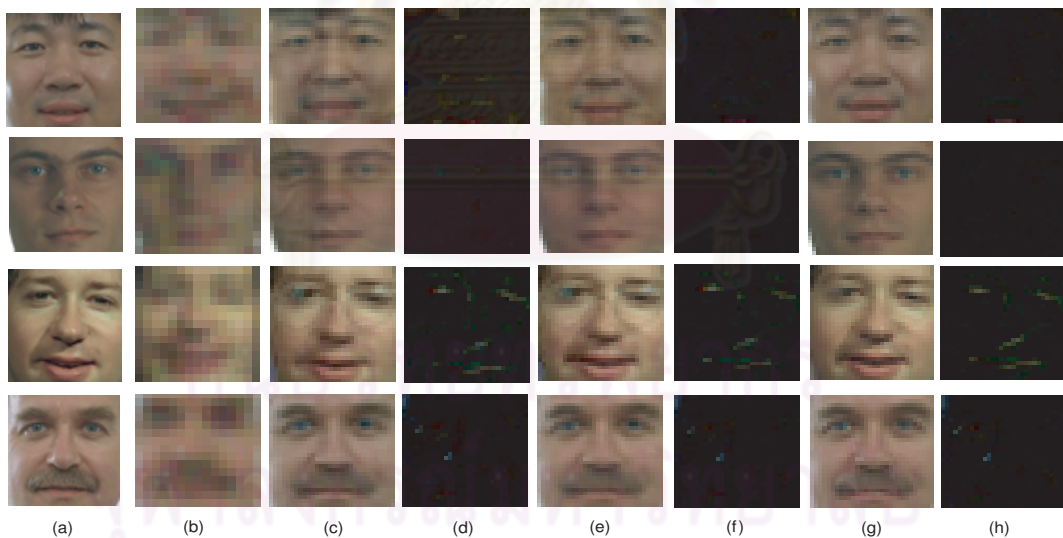


Figure 4.26: Color hallucinated face images with noise variance 10^{-3} in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-3} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

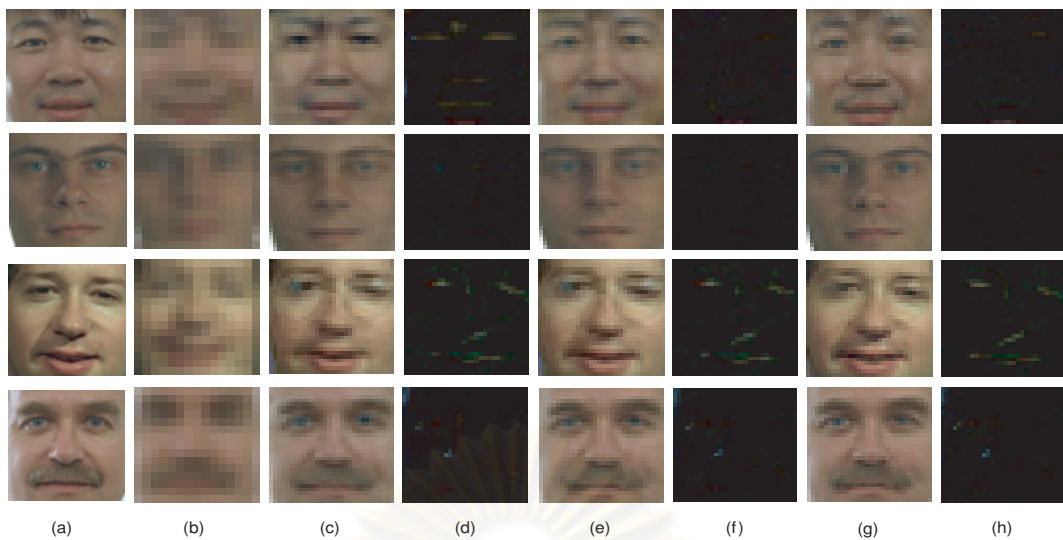


Figure 4.27: Color hallucinated face images with noise variance 10^{-6} in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

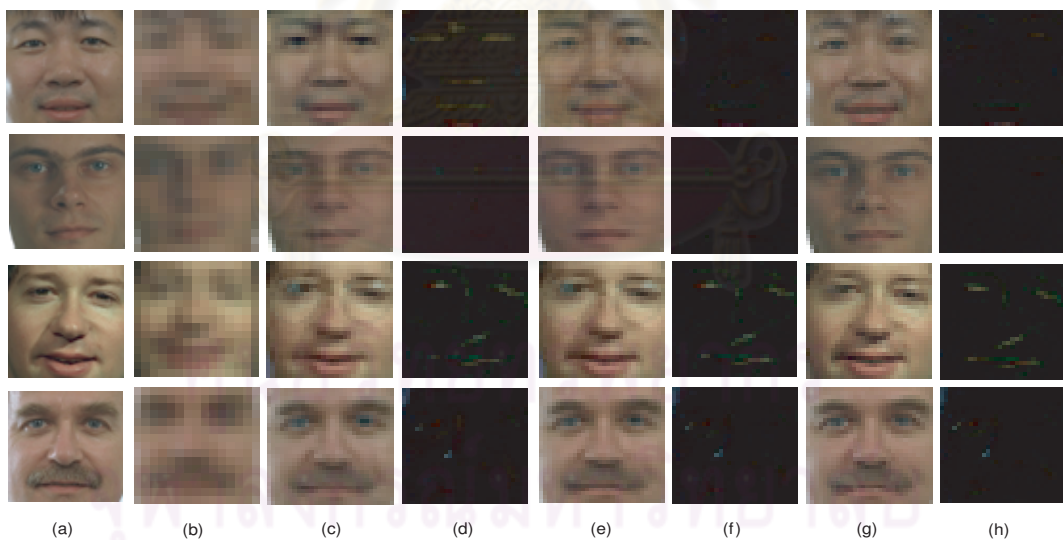


Figure 4.28: Color hallucinated face images with noise variance 2.5×10^{-5} in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

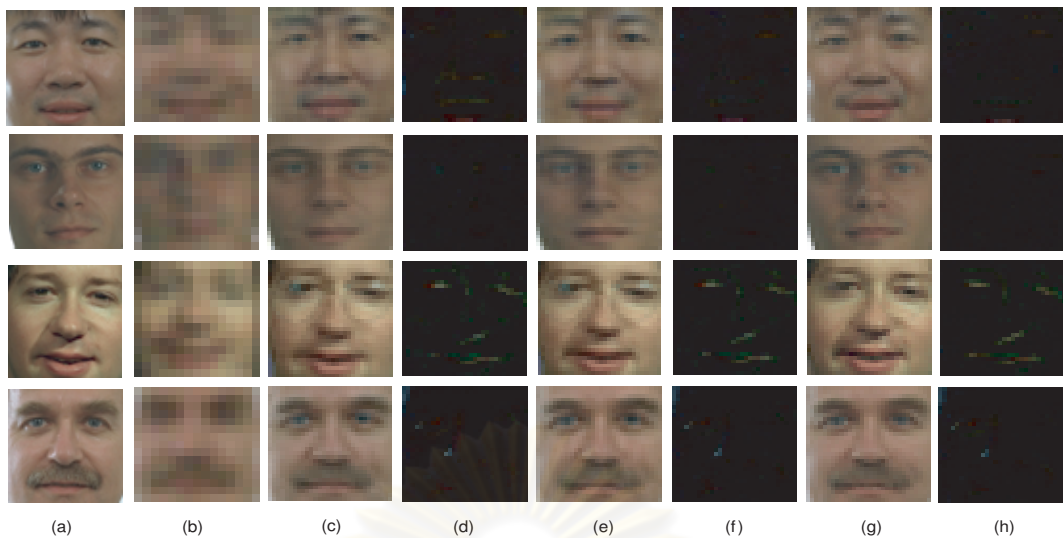


Figure 4.29: Color hallucinated face images with noise variance 2.5×10^{-4} in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-4} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

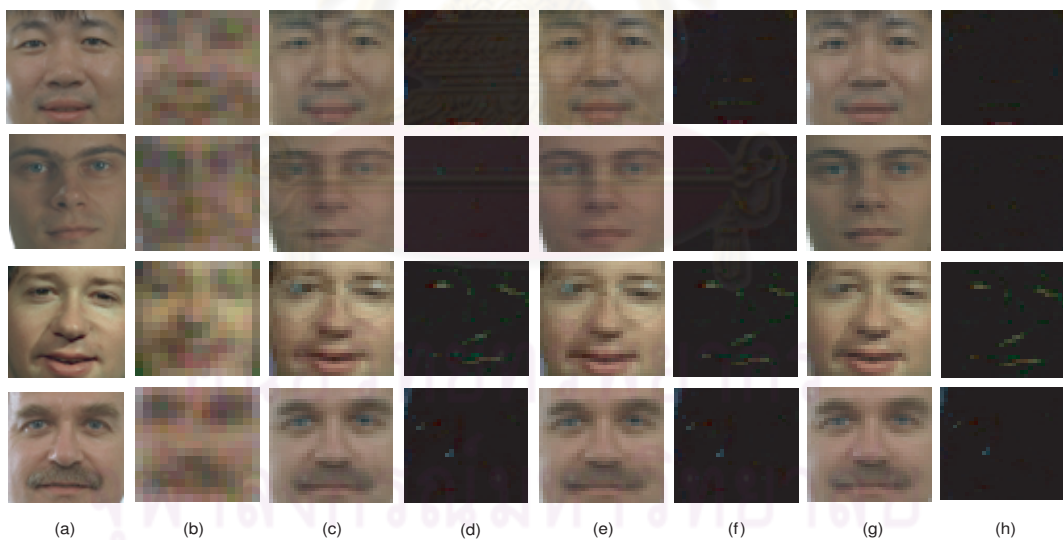


Figure 4.30: Color hallucinated face images with noise variance 10^{-3} in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-3} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

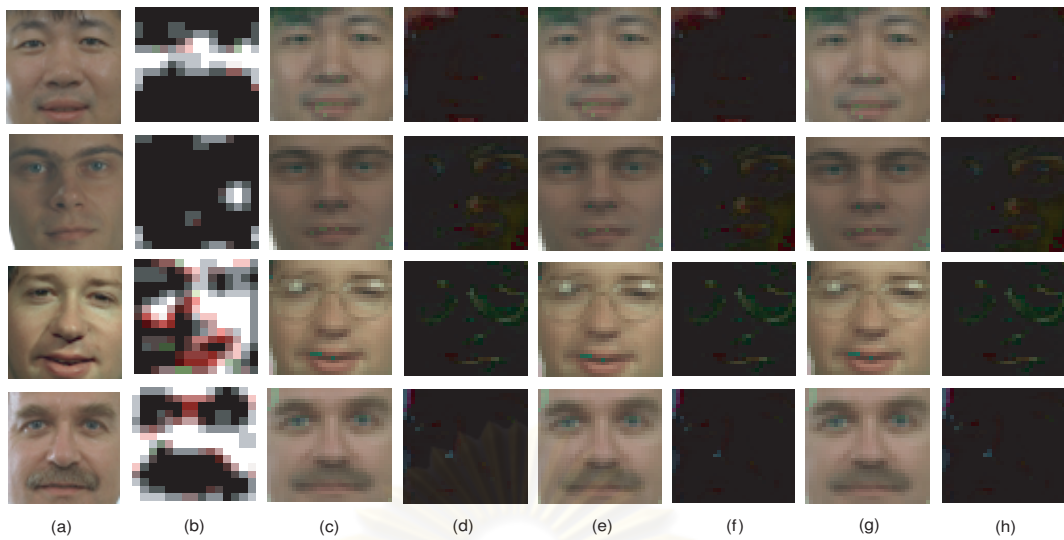


Figure 4.31: Color hallucinated face images with noise variance 10^{-6} in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

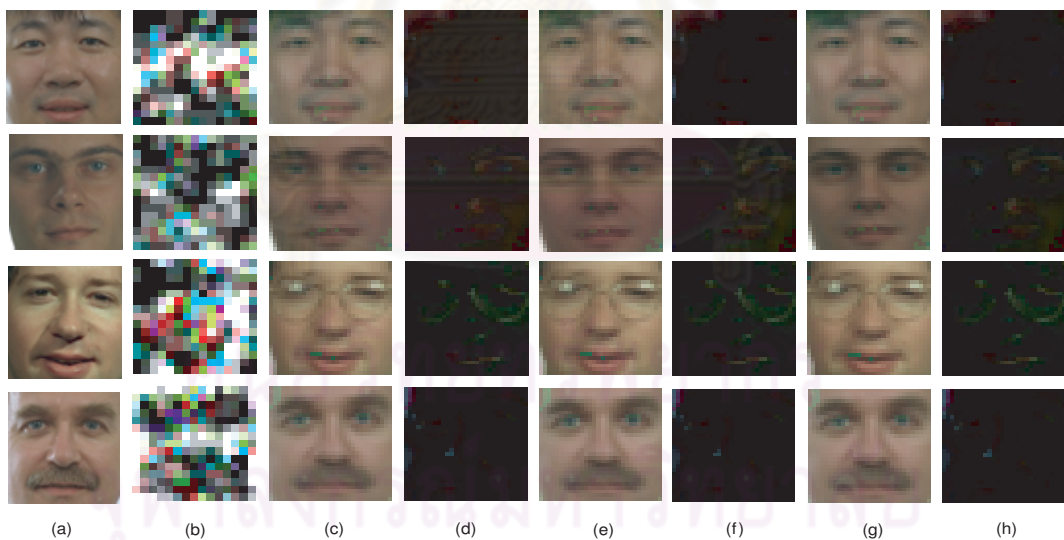


Figure 4.32: Color hallucinated face images with noise variance 2.5×10^{-5} in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-5} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

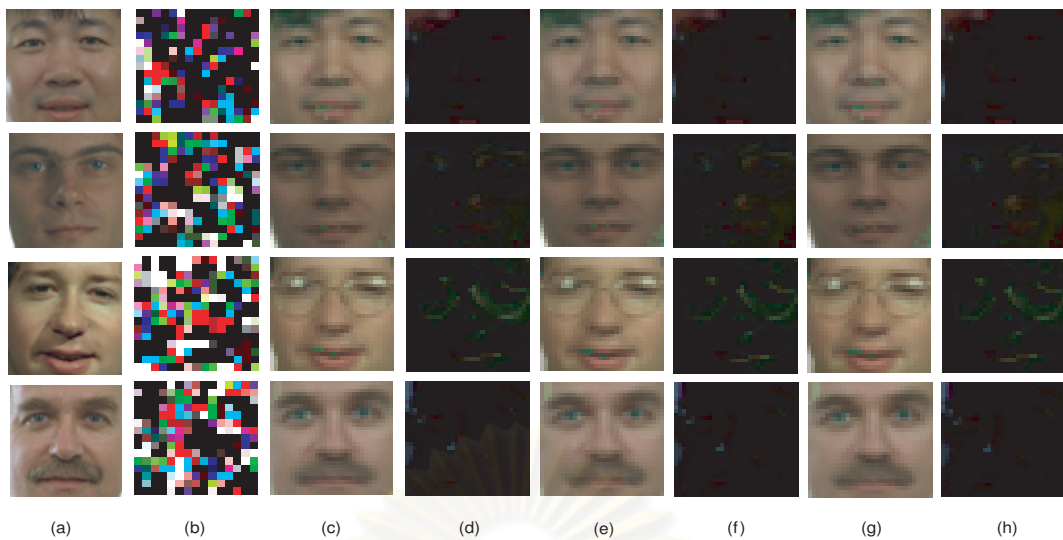


Figure 4.33: Color hallucinated face images with noise variance 2.5×10^{-4} in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-4} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

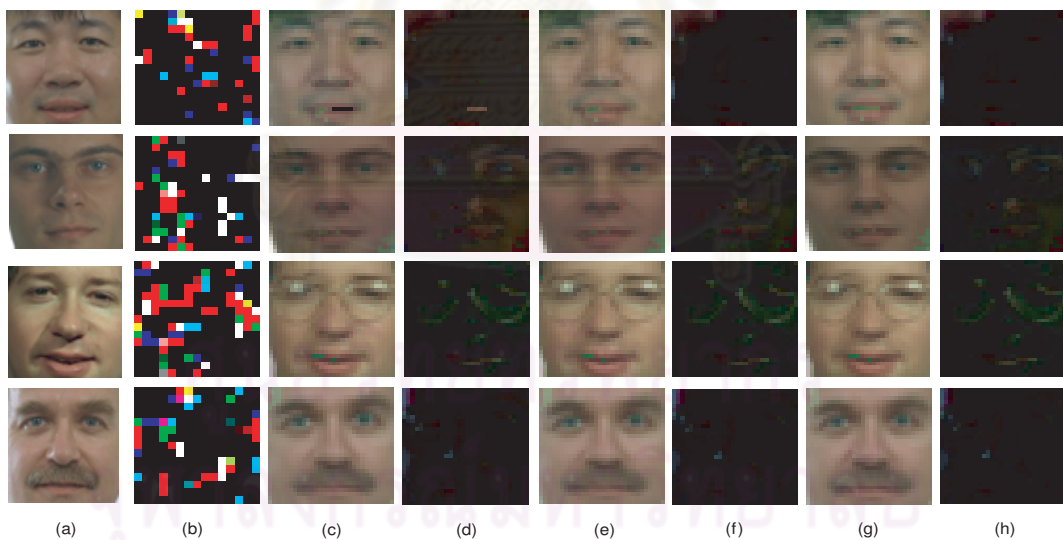


Figure 4.34: Color hallucinated face images with noise variance 10^{-3} in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-3} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

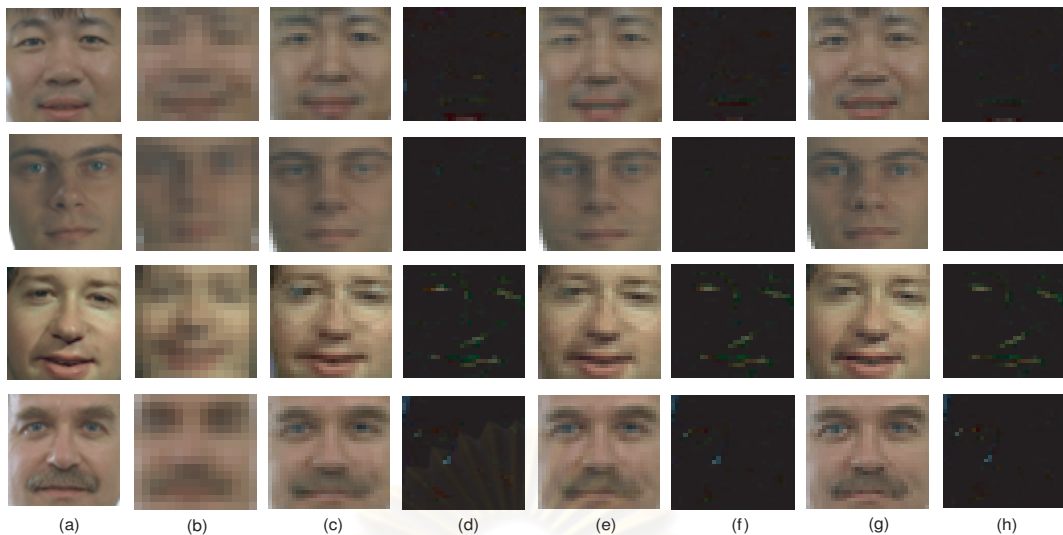


Figure 4.35: Color hallucinated face images with noise variance 10^{-6} in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

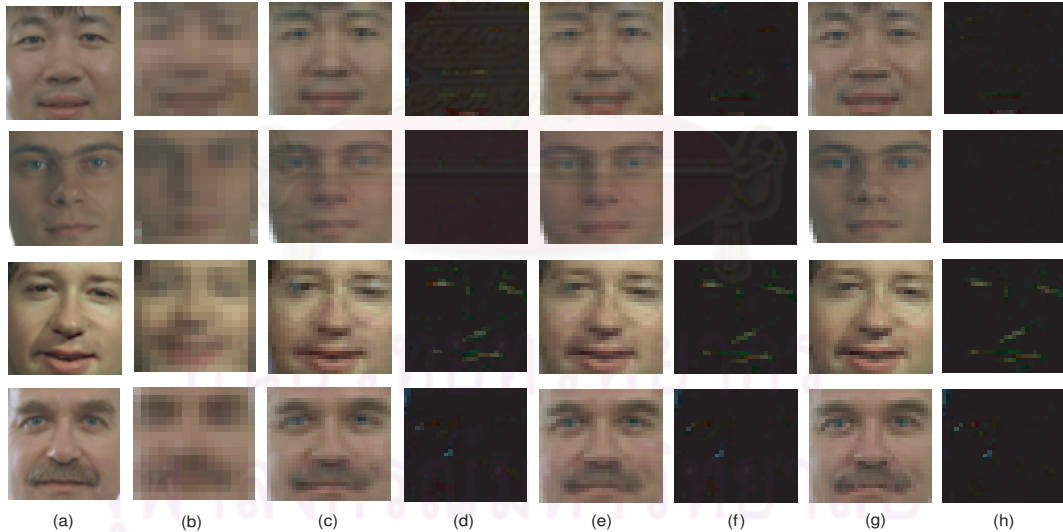


Figure 4.36: Color hallucinated face images with noise variance 2.5×10^{-5} in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-6} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

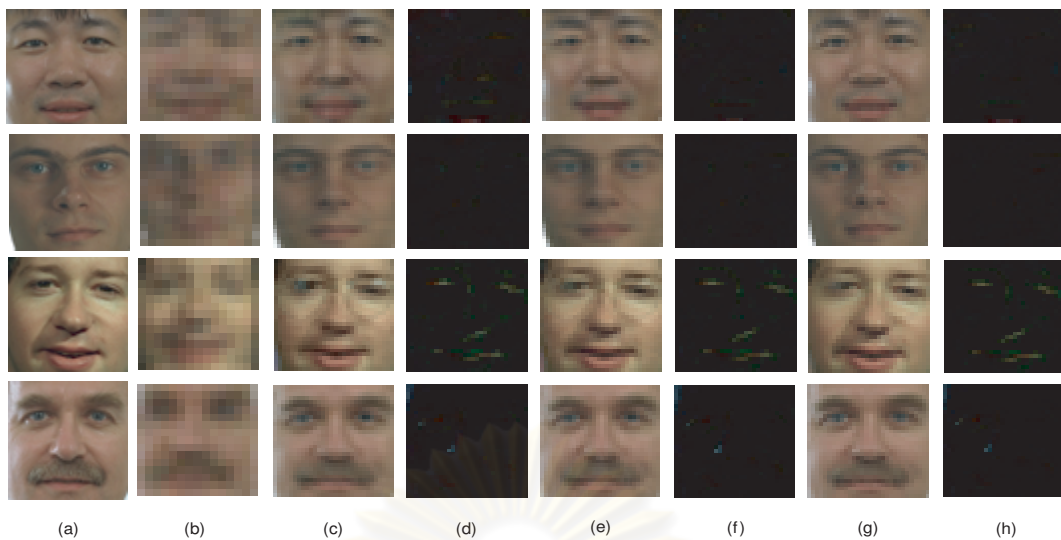


Figure 4.37: Color hallucinated face images with noise variance 2.5×10^{-4} in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-4} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

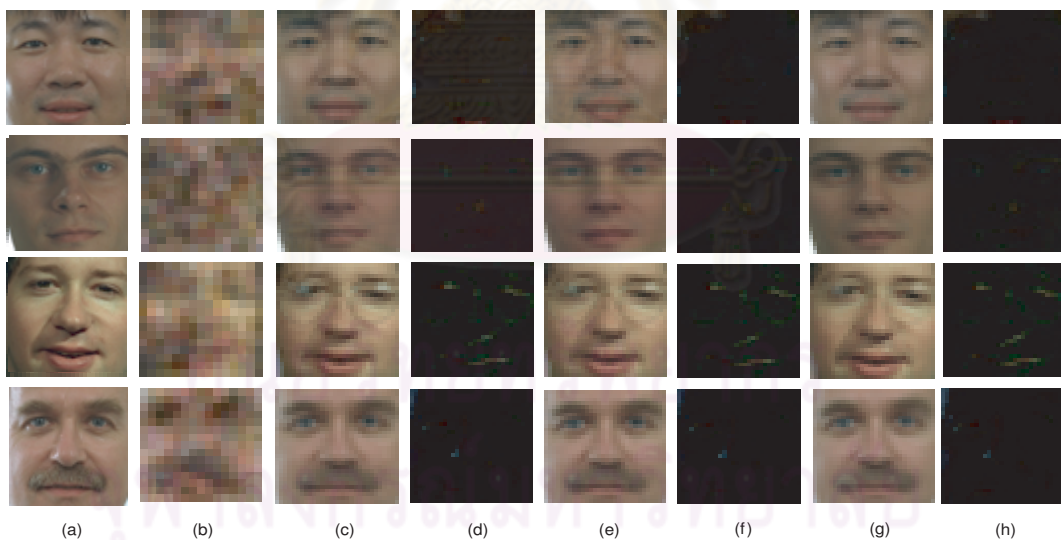


Figure 4.38: Color hallucinated face images with noise variance 10^{-3} in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise variance 10^{-3} , motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

Table 4.13 comparison of complexity (execution time) with training set size 60 images

Case	90 percent	95 percent	100 percent
1) <i>MPCA method</i>	14.403683 s.	14.676264 s.	15.586928 s.
2) <i>tensorPCA method</i>	14.756412 s.	14.935647 s.	15.894112 s.
3) <i>normalize</i>	1.0244	1.0176	1.0195

Table 4.14 comparison of complexity (execution time) with training set size 120 images

Case	90 percent	95 percent	100 percent
1) <i>MPCA method</i>	28.424068 s.	29.345926 s.	31.406217 s.
2) <i>tensorPCA method</i>	31.327327 s.	31.368005 s.	32.325575 s.
3) <i>normalize</i>	1.1021	1.0176	1.0292

4.4.4 Complexity

With regard to the computational complexity, we compare the execution time between our method and tensorPCA in RGB color model [33]. We test this experiment on a desktop-computer which is implemented on Microsoft Windows XP Professional 64 bits (version 2003), Intel(R) Core(TM)2 CPU 6600 with 2.8 GH and 3 GB of RAM and the result can be shown in Table. 4.13 - 4.16. For hallucination in MPCA method with 60, 120, 240 and 500 training sample images, the time in this simulation is about 14.40-15.58 seconds, 28.42-31.40 seconds, 58.70-65.27 seconds and 120.60-132.60 seconds respectively. However, with 60, 120, 240 and 500 training sample images, the method in [33] has a total time of 14.75-15.89 seconds, 31.32-32.32 seconds, 61.01-67.99 seconds and 132.99-133.69 respectively.

Moreover, in Table. 4.13 - 4.16, the tensorPCA method in independent channel color takes 1.01-1.1 times our algorithm time to implement similar results. The complexity of our algorithm, color face hallucination with MPCA, is less than the method in [33] because the MPCA can simultaneously reduce the dimension of data tensor (color face images) in PCA processes.

Table 4.15 comparison of complexity (execution time) with training set size 240 images

Case	90 percent	95 percent	100 percent
1) <i>MPCA method</i>	57.800732 s.	58.412280 s.	65.276072 s.
2) <i>tensorPCA method</i>	61.019110 s.	64.011103 s.	67.990804 s.
3) <i>normalize</i>	1.0556	1.0958	1.0415

Table 4.16 comparison of complexity (execution time) with training set size 500 images

Case	90 percent	95 percent	100 percent
1) <i>MPCA method</i>	120.603640 s.	122.589145 s.	132.603396 s.
2) <i>tensorPCA method</i>	132.994203 s.	133.472162 s.	133.696668 s.
3) <i>normalize</i>	1.1027	1.0887	1.0082

4.4.5 Partially Occluded Color Face Images

In this experiment, the regression model with MPCA also provides an ability to deal with the partially occluded face image and the blocking effect is used. Since the MPCA subspace analysis is still a holistic analysis method, we use a block for the partially occluded image patch. The experimental results are shown in Fig. 4.39-4.42 (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion, blur and left eye occluded; (c) face hallucination result with 100 percent traditional PCA; (d) face hallucination result with bilinear method and (e) face hallucination result with 95 percent MPCA. We can see that the traditional PCA and bilinear interpolation method are not suitable for hallucination facial images which are partially occluded. On the other hand, the linear regression model in MPCA can reconstruct realistic color face images.

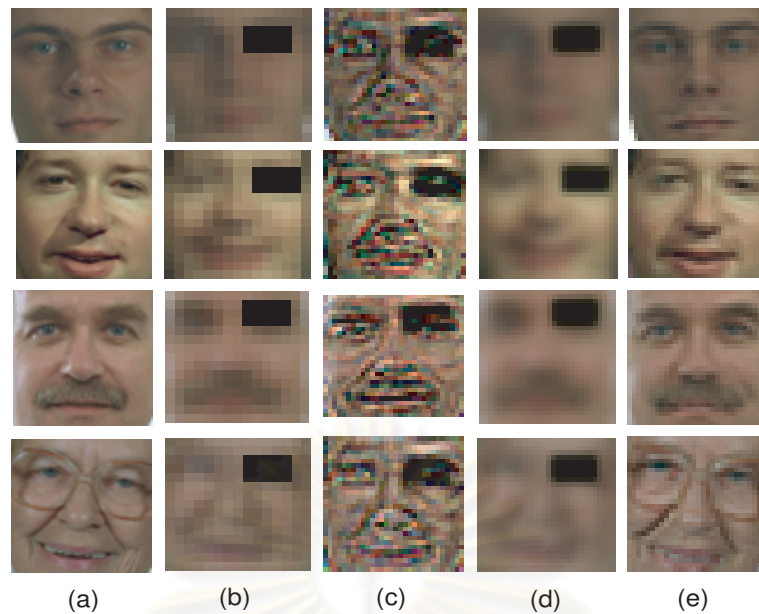


Figure 4.39: Partially occluded face hallucination results in RGB color model. (a) original HR images; (b) input LR images with noise, motion, blur and left eye occluded; (c) face hallucination result in RGB color model with 100 percent traditional PCA; (d) face hallucination result in RGB color model with bilinear method; (e) face hallucination result in RGB color model with 95 percent MPCA;

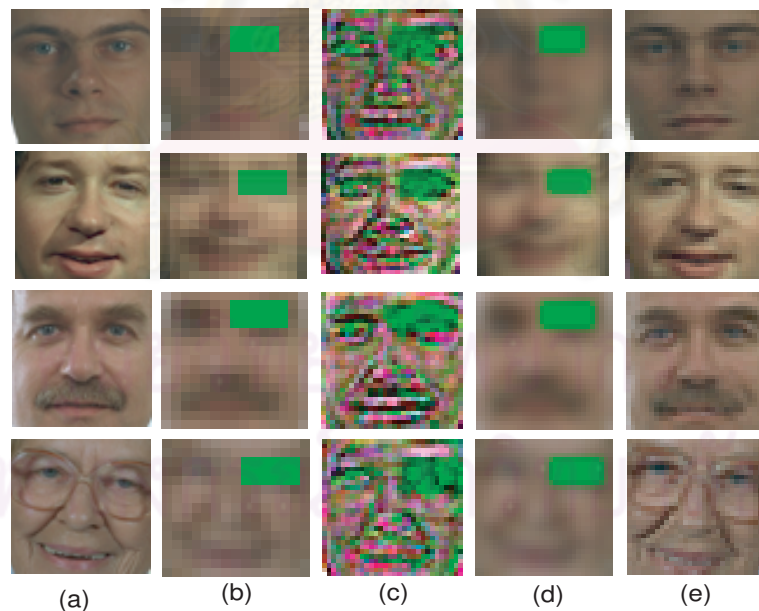


Figure 4.40: Partially occluded face hallucination results in YCbCr color model. (a) original HR images; (b) input LR images with noise, motion, blur and left eye occluded; (c) face hallucination result in YCbCr model system with 100 percent traditional PCA; (d) face hallucination result in YCbCr model system with bilinear method; (e) face hallucination result in YCbCr model system with 95 percent MPCA;

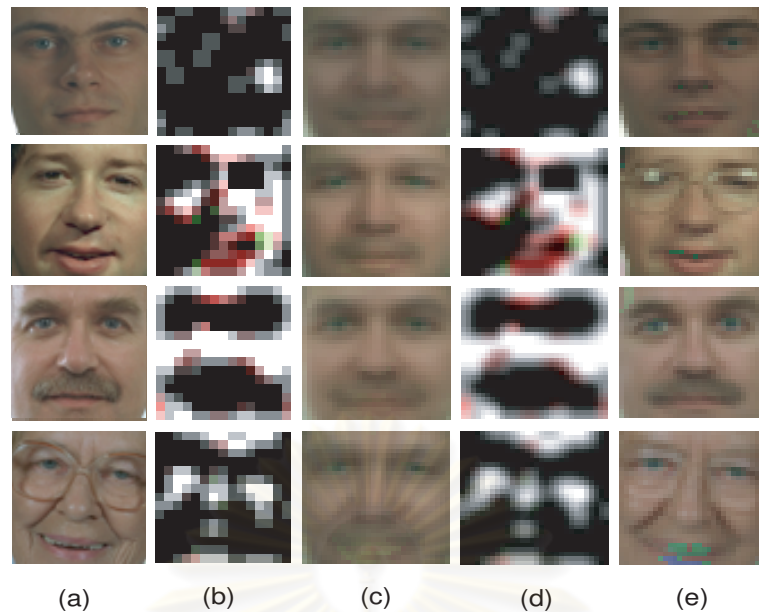


Figure 4.41: Partially occluded face hallucination results in HSV color model. (a) original HR images; (b) input LR images with noise, motion, blur and left eye occluded in HSV color model; (c) face hallucination result in HSV color model with 100 percent traditional PCA; (d) face hallucination result in HSV color model with bilinear method; (e) face hallucination result in HSV color model with 95 percent MPCA;

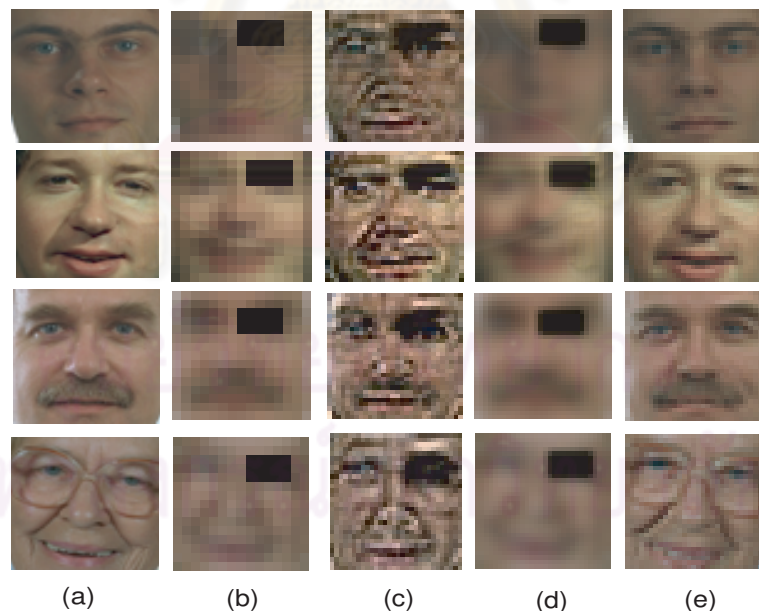


Figure 4.42: Partially occluded face hallucination results in CIELAB color model. (a) original HR images; (b) input LR images with noise, motion, blur and left eye occluded in CIELAB color model; (c) face hallucination result in CIELAB color model with 100 percent traditional PCA; (d) face hallucination result in CIELAB color model with bilinear method; (e) face hallucination result in CIELAB color model with 95 percent MPCA;

4.5 Experiments and Analysis on Color Face hallucination with Tensor Patch

In this experiment, we choose FERET database for a set of simulated experiments. There are 500 color faces for training set and other 50 facial images for testing. To establish a standard training dataset, we use a face image size of (30×30) and align the data manually by marking the location of three points: the centers of mouth and two eyes. These three points define an affine warp, which was used to warp the images into a canonical form. For all the 500 high-resolution facial images in the training dataset, we blurred and subsampled them to obtain their low-resolution (15×15) samples.

We decompose each of 500 pairs of low and high resolution training face images into (10×10) and (5×5) patches which overlapped horizontally and vertically with each other by 2 pixel and 4 pixel (the patch size and overlapping size were experimentally determined). We also quantify our performance by evaluating the peak signal-to-noise ratio (PSNR) between the ground truth face images and the hallucinated images.

Some experimental results are given in Fig. 4.43-4.58: (a) original HR images (30×30) , (b) input LR images (15×15) with noise, motion and blur in LR images, (c) face hallucination result with bilinear interpolation method, (d) face hallucination result with Liu method, (e)-(g) face hallucination result with 90 - 100 percent MPCA and (h) face hallucination result with Tensor patch method. The results in column (h) shows that tensor patches technique is good at hallucinating and reproducing details of local face regions, but poor at detail around an eye. We also give in Table. 4.17-4.20 the PSNR values between the hallucinated face images with linear regression model in MPCA and those face hallucination results with tensor patches in Fig. 4.43-4.58. Table 4.43-4.58 show that our proposed approach outperforms all the other face super-resolution techniques in terms of PSNR.

We compare our method with difference patch size, some example results are presented in Fig. 4.43-4.58. Compared with the results in Fig. 4.43-4.50 (h) using tensor patches technique with size (5×5) , the hallucinated results in Fig. 4.51-4.58 (h) tensor patches technique with size (10×10) then the bigger patch size can produce better color facial results than small patch size. In addition, with the same value of percent MPCA, the tensor patch method with size (10×10) can give higher PSNR values than size (5×5) about 0.19 - 1.24 dB.



Figure 4.43: Color hallucinated face images in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (5×5) patches method and 95 percent MPCA;



Figure 4.44: Color hallucinated face images in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (5×5) patches method and 95 percent MPCA;



Figure 4.45: Color hallucinated face images in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (5×5) patches method and 95 percent MPCA;



Figure 4.46: Color hallucinated face images in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (5×5) patches method and 95 percent MPCA;



Figure 4.47: Color hallucinated face images in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (5×5) patches method and 100 percent MPCA;



Figure 4.48: Color hallucinated face images in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (5×5) patches method and 100 percent MPCA;



Figure 4.49: Color hallucinated face images in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (5×5) patches method and 100 percent MPCA;



Figure 4.50: Color hallucinated face images in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (5×5) patches method and 95 percent MPCA;



Figure 4.51: Color hallucinated face images in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (10×10) patches method and 95 percent MPCA;



Figure 4.52: Color hallucinated face images in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (10×10) patches method and 95 percent MPCA;



Figure 4.53: Color hallucinated face images in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (10×10) patches method and 95 percent MPCA;



Figure 4.54: Color hallucinated face images in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (10×10) patches method and 95 percent MPCA;

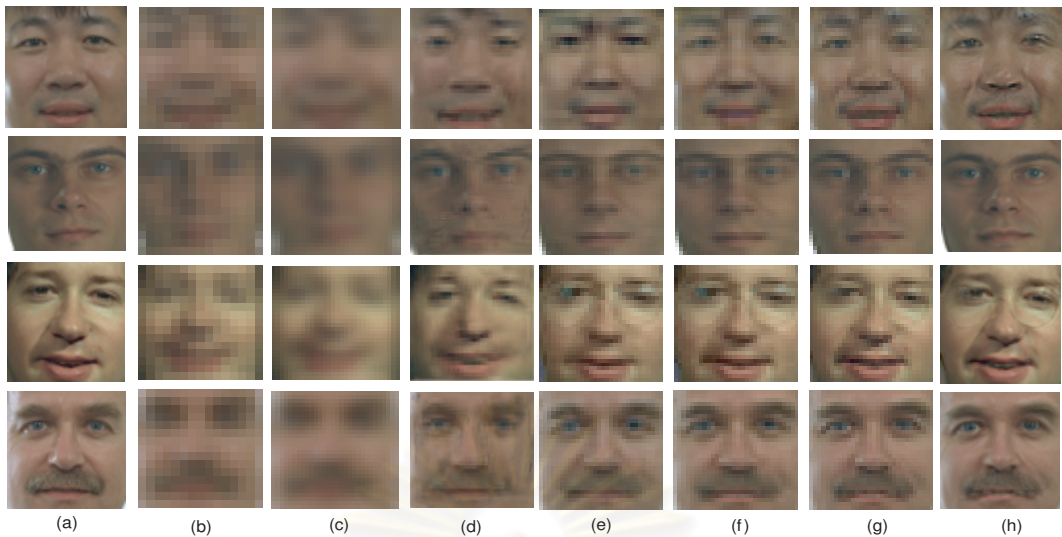


Figure 4.55: Color hallucinated face images in RGB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (10×10) patches method and 100 percent MPCA;



Figure 4.56: Color hallucinated face images in YCbCr color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (10×10) patches method and 100 percent MPCA;



Figure 4.57: Color hallucinated face images in HSV color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (10×10) patches method and 100 percent MPCA;

Table 4.17: PSNR results of the facial images with tensor patches method in Fig. 4.43 - 4.58 for RGB color model.

Case	first row	second row	third row	fourth row
1) 90 percent MPCA	26.88 dB	35.91 dB	27.89 dB	30.14 dB
2) 95 percent MPCA	31.15 dB	36.91 dB	28.53 dB	30.78 dB
3) 100 percent MPCA	31.35 dB	40.23 dB	28.74 dB	30.98 dB
4) patch (5×5) with 95 percent MPCA	34.19 dB	41.15 dB	32.32 dB	35.69 dB
5) patch (5×5) with 100 percent MPCA	34.68 dB	42.23 dB	32.45 dB	35.78 dB
6) patch (10×10) with 95 percent MPCA	35.28 dB	41.86 dB	32.67 dB	35.87 dB
7) patch (10×10) with 100 percent MPCA	35.92 dB	42.71 dB	32.85 dB	36.26 dB



Figure 4.58: Color hallucinated face images in CIELAB color model. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with bilinear interpolation method; (d) face hallucination result with Liu method; (e) face hallucination result with 90 percent MPCA; (f) face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) face hallucination result with Tensor (10×10) patches method and 100 percent MPCA;

Table 4.18: PSNR results of the facial images with tensor patches method in Fig. 4.43 - 4.58 for YCbCr color model.

Case	first row	second row	third row	fourth row
1) 90 percent MPCA	27.03 dB	36.38 dB	27.93 dB	30.13 dB
2) 95 percent MPCA	31.24 dB	37.04 dB	28.57 dB	30.69 dB
3) 100 percent MPCA	31.47 dB	40.29 dB	28.84 dB	30.88 dB
4) patch (5×5) with 95 percent MPCA	34.59 dB	41.27 dB	32.39 dB	35.73 dB
5) patch (5×5) with 100 percent MPCA	35.10 dB	42.35 dB	32.55 dB	35.84 dB
6) patch (10×10) with 95 percent MPCA	34.87 dB	41.88 dB	32.66 dB	35.91 dB
7) patch (10×10) with 100 percent MPCA	35.32 dB	42.78 dB	32.93 dB	36.34 dB

Table 4.19: PSNR results of the facial images with tensor patches method in Fig. 4.43 - 4.58 for HSV color model.

Case	first row	second row	third row	fourth row
1) 90 percent <i>MPCA</i>	26.85 dB	31.02 dB	27.71 dB	29.41 dB
2) 95 percent <i>MPCA</i>	27.28 dB	31.18 dB	27.87 dB	29.59 dB
3) 100 percent <i>MPCA</i>	27.75 dB	31.22 dB	27.96 dB	29.63 dB
4) patch (5 × 5) with 95 percent <i>MPCA</i>	33.20 dB	31.25 dB	31.22 dB	34.57 dB
5) patch (5 × 5) with 100 percent <i>MPCA</i>	33.36 dB	31.31 dB	31.24 dB	34.61 dB
6) patch (10 × 10) with 95 percent <i>MPCA</i>	33.32 dB	31.41 dB	31.28 dB	34.70 dB
7) patch (10 × 10) with 100 percent <i>MPCA</i>	33.41 dB	31.49 dB	31.35 dB	34.78 dB

Table 4.20: PSNR results of the facial images with tensor patches method in Fig. 4.43 - 4.58 for CIELAB color model.

Case	first row	second row	third row	fourth row
1) 90 percent <i>MPCA</i>	29.59 dB	36.48 dB	28.01 dB	30.58 dB
2) 95 percent <i>MPCA</i>	31.66 dB	37.11 dB	28.35 dB	31.52 dB
3) 100 percent <i>MPCA</i>	32.08 dB	40.35 dB	28.69 dB	31.69 dB
4) patch (5 × 5) with 95 percent <i>MPCA</i>	35.21 dB	41.31 dB	32.39 dB	35.75 dB
5) patch (5 × 5) with 100 percent <i>MPCA</i>	36.15 dB	42.40 dB	32.48 dB	35.89 dB
6) patch (10 × 10) with 95 percent <i>MPCA</i>	35.47 dB	41.52 dB	32.71 dB	35.97 dB
7) patch (10 × 10) with 100 percent <i>MPCA</i>	36.39 dB	42.82 dB	32.96 dB	36.44 dB

Table 4.21: comparison of complexity (execution time) between tensor patch method and MPCA method with training set size 60 images

Case	90 percent	95 percent	100 percent
1) <i>MPCA method</i>	14.403683 s.	14.676264 s.	15.586928 s.
2) <i>tensor patch size</i> (10×10)	32.445231 s.	39.667863 s.	59.889969 s.
3) <i>tensor patch size</i> (5×5)	253.547854 s.	287.634373 s.	304.315276 s.

Table 4.22: comparison of complexity (execution time) between tensor patch method and MPCA method with training set size 120 images

Case	90 percent	95 percent	100 percent
1) <i>MPCA method</i>	28.424068 s.	29.345926 s.	31.406217 s.
2) <i>tensor patch size</i> (10×10)	68.749302 s.	77.146453 s.	111.857586 s.
3) <i>tensor patch size</i> (5×5)	512.846542 s.	574.463359 s.	664.599841 s.

4.5.1 Complexity

In this section, we compare the execution time between the tensor patch method and MPCA method. The execution time results are come from a desktop-computer which is implemented on Microsoft Windows XP Professional 64 bits (version 2003), Intel(R) Core(TM)2 CPU 6600 with 2.8 GH and 3 GB of RAM and the result can be shown in Table. 4.21 - 4.24.

For hallucination in MPCA method with 60, 120, 240 and 500 training sample images, the time in this simulation is about 14.40-15.58 seconds, 28.42-31.40 seconds, 58.70-65.27 seconds and 120.60-132.60 seconds respectively but the execution time of tensor patch method with size (10×10) is about 32.44-59.88 seconds, 68.74-111.85 seconds, 146.70-223.09 seconds and 290.16-461.16 seconds respectively. In addition, in Table. 4.21 - 4.24, we can see that the execution time of tensor patch method with size (5×5) will increase rapidly from 253.54 seconds to 2798.165270 seconds when training sample increase from 60 to 500 images.

Table 4.23: comparison of complexity (execution time) between tensor patch method and MPCA method with training set size 240 images

Case	90 percent	95 percent	100 percent
1) <i>MPCA method</i>	57.800732 s.	58.412280 s.	65.276072 s.
2) <i>tensor patch size</i> (10×10)	146.709354 s.	159.184390 s.	223.091005 s.
3) <i>tensor patch size</i> (5×5)	1014.458788 s.	1143.303170 s.	1244.944058 s.

Table 4.24: comparison of complexity (execution time) between tensor patch method and MPCA method with training set size 500 images

Case	90 percent	95 percent	100 percent
1) <i>MPCA method</i>	120.603640 s.	122.589145 s.	132.603396 s.
2) <i>tensor patch size</i> (10×10)	290.167969 s.	344.809926 s.	461.160870 s.
3) <i>tensor patch size</i> (5×5)	2513.275331 s.	2659.911861 s.	2798.165270 s.

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

CHAPTER V

CONCLUSIONS

This chapter summarizes the works presented in this dissertation including conclusions and future directions.

5.1 Conclusions of The Dissertation

In this dissertation, the improved frameworks of color face hallucination are proposed. Firstly, this dissertation proposed a novel face hallucination with linear regression model in MPCA for the general color model such as RGB color model, YCbCr color model, HSV color model and CIELAB color model to improve the performance of the system. Since multilinear principal component analysis (MPCA) is more suitable for face representation than traditional method, like PCA. For better performance in super-resolution reconstruction task, higher-order tensor still be necessary.

Secondly, we apply higher-order singular value decomposition (HOSVD) in tensor space. We formulate a unified tensor in tensor patches which can be reduced to two parts: a global image-based tensor and a local patch-based multiresolution tensor for incorporating high-resolution image details. Our experiments show not only performance superiority over existing benchmark face super-resolution techniques, but also novelty of our approach in color face super-resolution.

5.2 Future Directions

- Several parameters (such as the number of standard face images, the number of shifting image and the number of classifiers) are still manually specified. The optimal values are found by experiments for the best hallucination result. Nevertheless, automatic parameter specification is necessary for the practical applications in the future research.
- Based on using the tensor MPCA subspace with regression model, we will directly perform our technique across different modality and under changing illumination conditions.
- For applications in practical scenarios where faces captured in raw color images are normally nonfrontal views at low resolution, we will develop a face hallucination algorithm for reconstruction reasonable nonfrontal facial images.

References

- [1] Vasilescu, M. and Terzopoulos, D. Multilinear analysis of image ensembles: Tensorfaces. In Proc. of European Conference on Computer Vision (ECCV) (2002): 447–460.
- [2] Vasilescu, M. and Terzopoulos, D. Multilinear subspace analysis of image ensembles. In Proc. of Computer Vision and Pattern Recognition (CVPR) 2 (2003): 93–99.
- [3] Huang, T. S. and Tsan, R. Y. Multiple frame image restoration and registration. In Advances in Computer Vision and Image Processing 1 (1982): 317–339.
- [4] Schultz, R. and Stevenson, R. A bayesian approach to image expansion for improved definition. IEEE Trans. Image Processing 3 , 3 (1994): 233–242.
- [5] Schultz, R. and Stevenson, R. Extraction of high-resolution frames from video sequences. IEEE Trans. Image Processing 5 , 6 (1996): 996–1011.
- [6] Elad, M. and Feuer, A. Restoration of a single superresolution image from several blurred, noisy, and undersampled measured images. IEEE Trans. Image Processing 6 , 12 (1997): 1646–1658.
- [7] Elad, M. and Feuer, A. Superresolution restoration of an image sequence: adaptive filtering approach. IEEE Trans. Image Processing 8 , 3 (1999): 387–395.
- [8] Elad, M. and Feuer, A. Super-resolution reconstruction of image sequences. IEEE Trans. on Pattern Analysis and Machine Intelligence 21 , 9 (1999): 817–834.
- [9] Patti, A., Sezan, M., and Tekalp, A. Super-resolution video reconstruction with arbitrary sampling lattices and nonzero aperture time. IEEE Trans. Image Processing 6 , 8 (1997): 1064–1076.
- [10] Basle, B., Blake, A., and Zisserman, A. Motion deblurring and super-resolution from an image sequence. Proc. Fourth European Conf. Computer Vision (1996): 573–581.
- [11] Patanavijit, V. and Jitapunkul, S. A lorentzian stochastic estimation for a robust iterative multiframe super-resolution reconstruction with lorentzian-tikhonov regularization. EURASIP Journal on Advances in Signal Processing 2007 (2007): 21.
- [12] Kim, S., Bose, N., and Valenzuela, H. Recursive reconstruction of high resolution image from noisy undersampled multiframes. IEEE Trans. Acoustics, Speech and Signal Process 38 (1990): 1013–1027.

- [13] Kim, S. and Wen-Yu, S. Recursive high-resolution reconstruction of blurred multiframe images. IEEE Trans. Image Process. 2 (1993): 534–539.
- [14] Combettes, P. and Civanlar, M. The foundations of set theoretic estimation. Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP) 4 , Apr. (1991): 2921–2924.
- [15] Vandewalle, P., Ssstrunk, S., and Vetterli, M. Double resolution from a set of aliased images. Proc. SPIE Electronic Imaging 2004: Sensors and Camera Systems for Scientific, Industrial, and Digital Photography Applications V (Jan. 2004): 374–382.
- [16] Vandewalle, P., Sbaiz, L., Vetterli, M., and Ssstrunk, S. Super-resolution from highly undersampled images. Proc. IEEE Int. Conf. on Image Processing (ICIP) 1 (Sept. 2005): 889–892.
- [17] Vandewalle, P., Ssstrunk, S., and Vetterli, M. A frequency domain approach to registration of aliased images with application to super resolution. EURASIP J. Appl. Signal Process. (2006): 14.
- [18] Capel, D. P. and Zisserman, A. Super-resolution from multiple views using learnt image models. In Proc. IEEE Int. Conf. on Computational Vision Pattern Recognition (CVPR) (2001): 627–634.
- [19] Baker, S. and Kanade, T. Hallucinating faces. IEEE Int. Conf. Automatic Face and Gesture Recognition.
- [20] Hardie, R., Barnard, K., and Armstrong, E. Joint map registration and high-resolution image estimation using a sequence of undersampled images. IEEE Trans. Image Processing 6 (June 1977): 1621–1633.
- [21] Bonet, J. D. Multiresolution sampling procedure for analysis and synthesis of texture images. Proc. SIGGRAPH 97 (August 1997): 361–368.
- [22] Bonet, J. D. and Viola, P. A nonparametric multi-scale statistical model for neutral images. Proc. Advances Neural Inf. Process.
- [23] Kim, S. and Su, W. Recursive reconstruction of high-resolution reconstruction of blurred multiframe images. IEEE Trans. Image Processing 2 (June 1993): 534–539.
- [24] Penev, P. S. and Sirovich, L. The global dimensionality of face space. Proc. IEEE Int. Conf. Automatic Face and Gesture Recognition (2000): 264–270.
- [25] Ma, X., Zhang, J., and Qi, C. Position-based face hallucination method. IEEE International Conference on Multimedia and Expo, 2009 (ICME). (2009): 290–293.

- [26] Ma, X., Zhang, J., and Qi, C. An example-based two-step face hallucination method through coefficient learning. Image Analysis and Recognition :Lecture Notes in Computer Science. 5627 (2009): 471–480.
- [27] Freeman, W. T. and Pasztor, E. C. Learning low-level vision. IEEE Int. Conf. Computer Vision (ICCV).
- [28] Baker, S. and Kanade, T. Limits on super-resolution and how to break them. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI). 24 , 9 (2002): 1167–1183.
- [29] Liu, C., Zhang, C., and Shum, H. A two-step approach to hallucinating faces: Global parametric model and local nonparametric model. In Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR) (2001): 192–198.
- [30] Wang, X. and Tang, X. Hallucinating face by eigentransformation. IEEE Trans. on Systems, Man and Cybernetics 35 , 3 (2005): 425–434.
- [31] Tang, X. and Wang, X. Face sketch recognition. IEEE Trans. on Circuits, Systems, and Video Technology 14 , 1 (2004): 50–57.
- [32] Chang, H., Yeung, D., and Xiong, Y. Super-resolution through neighbor embedding. In Proc. of SIGGRAPH 1 (2004): 275–282.
- [33] Wu, J. and Trivedi, M. M. A regression model in tensorpca subspace for face image super-resolution reconstruction. In Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR) 3 (2006): 627–630.
- [34] Chakrabarti, A., Rajagopalan, A., and Chellappa, R. Super-resolution of face images using kernel pca-based prior. IEEE Trans. on Multimedia 9 (2007): 888–892.
- [35] Ogawa, T. and Haseyama, M. Kernel pca-based resolution enhancement approach of still images using different levels of pyramid structure. In Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP) 9 (April 2008): 1293–1296.
- [36] Jia, K. and Gong, S. Multi-modal tensor face for simultaneous superresolution and recognition. In Proc. IEEE Int. Conf. Computer Vision (ICCV) (2005): 1683–1690.
- [37] Jia, K. and Gong, S. Multi-resolution tensor for facial expression hallucination. In Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR) (2006): 395–402.
- [38] Jia, K. and Gong, S. Generalized face super-resolution. IEEE Trans. Image Processing 17 , 6 (2008): 873–886.
- [39] Ma, X., Huang, H., Wang, S., and Qi, C. A simple approach to multiview face hallucination. Signal Processing Letters, IEEE 17 (June 2010): 579–582.

- [40] Carroll, J. and Chang, J. Analysis of individual differences in multidimensional scaling via an n-way generalization of eckart-young decomposition. Psychometrika 35 (1970): 283–319.
- [41] A. Kapteyn, H. N. and T.Wansbeek An approach to n-mode components analysis. Psychometrika 51 (1986): 269–275.
- [42] Kroonenbergand, P. M. and Leeuw, J. Principal component analysis of three-mode data by means of alternating least squares algorithms. Psychometrika 45 (1980): 69–97.
- [43] Kruskal, J. B. Three-way arrays: Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. Linear Algebra Appl 18 (1977): 95–138.
- [44] Leibovici, D. and Sabatier, R. A singular value decomposition of a k-way array for principal component analysis of multiway data, pta-k. Linear Algebra Appl (1998): 307–329.
- [45] Zhangand, T. and Golub, G. H. Some mathematical notes on three-mode factor analysis. SIAM J. Matrix Anal. Appl 23 (2001): 534–550.
- [46] Tucker, L. R. Some mathematical notes on three-mode factor analysis. Psychometrika 31 (1966): 279–311.
- [47] Lathauwer, L., Moor, B., and Vandewalle, J. Multilinear singular value tensor decompositions. SIAM Journal on Matrix Analysis and Applications 21 (April 2008): 1253–1278.
- [48] Kolda, T. G. Orthogonal tensor decompositions.. SIAM Journal on Matrix Analysis and Applications 23 (July 2001): 243–255.
- [49] Shah, N. R. and Zakhor, A. Resolution enhancement of color video sequences. IEEE Trans. Image Proc. 8 (June 1999): 879–885.
- [50] Tom, B. C. and Katsaggelos, A. Resolution enhancement of monochrome and color video using motion compensation. IEEE Trans. Image Proc. 10 (Feb 2001): 278–287.
- [51] Irani, M. and Peleg, S. Improving resolution by image registration. CVGIP: Graph. Models Image Proc. 53 (1991): 231–239.
- [52] Patil, V. H., Bormane, D. S., and Patil, H. K. Color super resolution image reconstruction. In Proc. IEEE Int. Conf. on Computational Intelligence and Multimedia Application 6 (2007): 366–370.
- [53] El-Yamany, N. A. and Papamichalis, P. E. Robust color image superresolution: An adaptive m-estimation framework. EURASIP Journal on Image and Video Proc.

- [54] Kimmel, R. Demosaicing: Image reconstruction from color ccd samples. IEEE Trans. Image Proc. 8 (Sep 1999): 1221–1228.
- [55] Trussell, H. and Hartwig, R. E. Mathematics for demosaicking. IEEE Trans. Image Proc. 11 (April 2002): 485–492.
- [56] Farsiu, S., Elad, M., and Milanfar, P. Multiframe demosaicing and super-resolution of color images. IEEE Trans. Image Proc. 15 (Jan 2006): 141–159.
- [57] Shah, N. R. and Zakhor, A. Resolution enhancement of color video sequences. IEEE Trans. Image Proc. 8 (June 1999): 879–885.
- [58] Tom, B. C. and Katsaggelos, A. K. Resolution enhancement of monochrome and color video using motion compensation. IEEE Trans. Image Proc. 10 (Feb. 2001): 278–287.
- [59] Yang, J., Zhang, D., Frangi, A. F., , and Yang, J. Two-dimensional pca: A new approach to appearance-based face representation and recognition. IEEE Trans. Pattern Anal. Mach. Intell 26 (Jan. 2004): 131-137.
- [60] Vasilescu, M. A. O. and Terzopoulos, D. Multilinear subspace analysis for image ensembles. In Proc. of Computer Vision and Pattern Recognition (CVPR) 2 (June 2003): 93–99.
- [61] Vasilescu, M. A. O. and Terzopoulos, D. Multilinear analysis of image ensembles: Tensor-faces. In Proc. 7th Eur. Conf. Comput. Vis (ECCV) (May 2002): 447–460.
- [62] Ye, J., Janardan, R., and Li, Q. Gpca: An efficient dimension reduction scheme for image compression and retrieval. In Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (2004): 354-363.
- [63] Lu, H., Plataniotis, K. N. K., and Venetsanopoulos, A. N. MPCA: Multilinear principal component analysis of tensor objects. IEEE Transactions on Neural Networks 19 , 1 (2008): 18–39.
- [64] Yang, J., Zhang, D., and Frangi, A. F. Two-dimensional pca: A new approach to appearance-based face representation and recognition. IEEE Trans. Pattern Anal. Mach. Intell 26 (Jan. 2004): 131-137.
- [65] Cohen, J. and Kappauf, W. Metameric color stimuli, fundamental metamers, and wyszeckis metameric blacks. Amer. J. Psychol 95 , 4 (1982): 537-564.
- [66] Horn, B. Exact reproduction of colored images. Computer Vision, Graph. Image Processing 26 (1984): 135-167.

- [67] Wandell, B. The synthesis and analysis of color images. IEEE Trans. Pattern Anal. Machine Intell 9 (Jan. 1987): 2-13.
- [68] Trussell, H. Applications of set theoretic methods to color systems. Color Res. Applicat 16 (Feb. 1991): 31-41.
- [69] Trussell, H. Dsp solutions run the gamut for color systems. IEEE Signal Processing Mag 10 (Apr. 1993): 8-23.
- [70] Sharma, G. and Trussell, H. Digital color processing. IEEE Trans. Image Processing 6 , 7 (1997): 901-932.
- [71] Plataniotis, K. N. and Venetsanopoulos, A. N. Color image processing and applications. Berlin, Germany: Springer-Verlag.
- [72] Koschan, A. and Abidi, M. Detection and classification of edges in color images. IEEE Signal Processing Mag 22 (Jan. 2005): 64-73.
- [73] Lukac, R., Smolka, B., Martin, K., Plataniotis, K., and Venetsanopoulos, A. Vector filtering for color imaging. IEEE Signal Processing Mag 22 (Jan. 2005): 74-86,.
- [74] Baqai, F., Lee, J., Agar, A., and Allebach, J. Digital color halftoning. IEEE Signal Processing Mag 22 (Jan. 2005): 87-96.
- [75] Vrhel, M. and Trussell, H. Color printer characterization in MATLAB. International Conference on Image Processing (ICIP) 1 (Sept. 2002): 457-460.
- [76] Kay and M., S. Fundamentals of statistical signal processing: Estimation theory.. Prentice Hall (1993): .
- [77] Hardie, R. C., Barnard, K. J., and Armstrong, E. E. Joint map registration and high-resolution image estimation using a sequence of undersampled images. IEEE Trans. Image Processing 6 (Dec. 1997): 1621-1633.
- [78] Phillips, P. J., Wechsler, H., Huang, J., and Rauss, P. The FERET database and evaluation procedure for face recognition algorithms. Image and Vision Computing 16 , 5.
- [79] Phillips, P. J., Moon, H., Rizvi, S. A., and Rauss, P. J. The FERET evaluation methodology for face recognition algorithms. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI). 22.
- [80] Liu, C., Shum, H. Y., and Zhang, C. S. A two-step approach to hallucinating faces: Global parametric model and local nonparametric model. In Proc. of IEEE computer society conference on computer vision and pattern recognition (2001): 192–198.



Appendices

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

Appendix A

List of Abbreviations

2DPCA	Two-Dimensional Principal Component Analysis
HOSVD	High Order Singular Value Decomposition
LR	Low Resolution
HR	High Resolution
ICCA	Image Cross-Covariance Analysis
LDA	Linear Discriminant Analysis
MPCA	Multilinear Principal Component Analysis
NN	The nearest neighbor classifier
PCA	Principal Component Analysis
RSM	Random Subspace Method
SAR	Synthetic Aperture Radar
SSS	Small Sample Size Problem
SVD	Singular Value Decomposition
HSV	Hue, Saturation, Value
LLE	Locally Linear Embedding
MRF	Markov Random Field
MFH	multiview face hallucination

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

Appendix B

Some Experimental Results

Table 1 PSNR results of Fig. 2 with a linear regression model in MPCA method

Case	90 percent	95 percent	100 percent
1) <i>the 1st row</i>	31.49 dB	31.52 dB	32.57 dB
2) <i>the 2nd row</i>	31.40 dB	31.43 dB	31.61 dB
3) <i>the 3rd row</i>	24.61 dB	24.67 dB	26.89 dB
4) <i>the 4th row</i>	27.45 dB	27.52 dB	27.87 dB
5) <i>the 5th row</i>	35.60 dB	35.61 dB	35.71 dB
6) <i>the 6th row</i>	26.31 dB	26.39 dB	26.77 dB
7) <i>the 7th row</i>	27.24 dB	27.40 dB	29.61 dB
8) <i>the 8th row</i>	34.34 dB	34.55 dB	34.68 dB
9) <i>the 9th row</i>	36.53 dB	36.67 dB	37.33 dB
10) <i>the 10th row</i>	42.74 dB	42.79 dB	43.56 dB
11) <i>the 11th row</i>	28.28 dB	28.30 dB	28.63 dB
12) <i>the 12th row</i>	33.69 dB	33.71 dB	33.96 dB
13) <i>the 13th row</i>	26.56 dB	26.73 dB	26.85 dB
14) <i>the 14th row</i>	31.90 dB	31.96 dB	32.25 dB

Table 2 PSNR results of Fig. 3 with a linear regression model in MPCA method

Case	90 percent	95 percent	100 percent
1) <i>the 1st row</i>	31.52 dB	31.58 dB	32.62 dB
2) <i>the 2nd row</i>	31.43 dB	31.45 dB	31.69 dB
3) <i>the 3rd row</i>	24.67 dB	24.73 dB	26.96 dB
4) <i>the 4th row</i>	27.49 dB	27.55 dB	27.95 dB
5) <i>the 5th row</i>	35.61 dB	35.63 dB	35.75 dB
6) <i>the 6th row</i>	26.36 dB	26.44 dB	26.90 dB
7) <i>the 7th row</i>	27.25 dB	27.47 dB	29.70 dB
8) <i>the 8th row</i>	34.39 dB	34.58 dB	34.86 dB
9) <i>the 9th row</i>	36.62 dB	36.70 dB	37.45 dB
10) <i>the 10th row</i>	42.75 dB	42.83 dB	43.58 dB
11) <i>the 11th row</i>	28.35 dB	28.39 dB	28.77 dB
12) <i>the 12th row</i>	33.70 dB	33.74 dB	34.06 dB
13) <i>the 13th row</i>	26.59 dB	26.75 dB	26.95 dB
14) <i>the 14th row</i>	31.99 dB	32.03 dB	32.35 dB

Table 3 PSNR results of Fig. 4 with a linear regression model in MPCA method

Case	90 percent	95 percent	100 percent
1) <i>the 1st row</i>	30.22 dB	30.35 dB	30.87 dB
2) <i>the 2nd row</i>	30.13 dB	30.24 dB	30.54 dB
3) <i>the 3rd row</i>	23.56 dB	23.61 dB	23.97 dB
4) <i>the 4th row</i>	26.24 dB	26.39 dB	26.98 dB
5) <i>the 5th row</i>	35.12 dB	35.24 dB	35.54 dB
6) <i>the 6th row</i>	25.86 dB	25.97 dB	26.25 dB
7) <i>the 7th row</i>	26.89 dB	27.06 dB	27.66 dB
8) <i>the 8th row</i>	31.03 dB	31.51 dB	32.18 dB
9) <i>the 9th row</i>	33.37 dB	33.92 dB	34.85 dB
10) <i>the 10th row</i>	41.12 dB	41.56 dB	41.94 dB
11) <i>the 11th row</i>	27.52 dB	27.78 dB	28.01 dB
12) <i>the 12th row</i>	31.76 dB	32.12 dB	32.72 dB
13) <i>the 13th row</i>	26.08 dB	26.23 dB	26.54 dB
14) <i>the 14th row</i>	30.67 dB	30.86 dB	31.15 dB

Table 4 PSNR results of Fig. 5 with a linear regression model in MPCA method

Case	90 percent	95 percent	100 percent
1) <i>the 1st row</i>	31.58 dB	31.61 dB	32.75 dB
2) <i>the 2nd row</i>	31.57 dB	31.63 dB	31.88 dB
3) <i>the 3rd row</i>	24.70 dB	24.78 dB	26.87 dB
4) <i>the 4th row</i>	27.53 dB	27.58 dB	27.92 dB
5) <i>the 5th row</i>	35.64 dB	35.73 dB	35.94 dB
6) <i>the 6th row</i>	26.43 dB	26.52 dB	26.98 dB
7) <i>the 7th row</i>	27.47 dB	27.71 dB	28.13 dB
8) <i>the 8th row</i>	34.48 dB	34.67 dB	34.96 dB
9) <i>the 9th row</i>	36.68 dB	36.75 dB	37.49 dB
10) <i>the 10th row</i>	42.79 dB	42.87 dB	43.64 dB
11) <i>the 11th row</i>	28.37 dB	28.41 dB	28.83 dB
12) <i>the 12th row</i>	33.75 dB	33.78 dB	34.19 dB
13) <i>the 13th row</i>	26.65 dB	26.88 dB	27.24 dB
14) <i>the 14th row</i>	31.98 dB	32.11 dB	32.52 dB

Table 5 PSNR results of Fig. 6 with a linear regression model in tensorPCA method

Case	90 percent	95 percent	100 percent
1) <i>the 1st row</i>	31.19 dB	31.22 dB	32.27 dB
2) <i>the 2nd row</i>	31.06 dB	31.09 dB	31.21 dB
3) <i>the 3rd row</i>	24.31 dB	24.33 dB	26.52 dB
4) <i>the 4th row</i>	27.20 dB	27.32 dB	27.47 dB
5) <i>the 5th row</i>	35.34 dB	35.41 dB	35.62 dB
6) <i>the 6th row</i>	26.02 dB	26.03 dB	26.39 dB
7) <i>the 7th row</i>	27.11 dB	27.32 dB	29.48 dB
8) <i>the 8th row</i>	34.26 dB	34.32 dB	34.38 dB
9) <i>the 9th row</i>	36.03 dB	36.09 dB	36.56 dB
10) <i>the 10th row</i>	42.63 dB	42.71 dB	43.13 dB
11) <i>the 11th row</i>	28.14 dB	28.23 dB	28.51 dB
12) <i>the 12th row</i>	33.07 dB	33.29 dB	33.78 dB
13) <i>the 13th row</i>	26.29 dB	26.45 dB	26.68 dB
14) <i>the 14th row</i>	31.53 dB	31.72 dB	32.18 dB

Table 6 PSNR results of Fig. 7 with a linear regression model in tensorPCA method

Case	90 percent	95 percent	100 percent
1) <i>the 1st row</i>	31.36 dB	31.44 dB	32.53 dB
2) <i>the 2nd row</i>	31.14 dB	31.32 dB	31.44 dB
3) <i>the 3rd row</i>	24.49 dB	24.52 dB	26.78 dB
4) <i>the 4th row</i>	27.36 dB	27.42 dB	27.68 dB
5) <i>the 5th row</i>	35.50 dB	35.61 dB	35.71 dB
6) <i>the 6th row</i>	26.19 dB	26.25 dB	26.63 dB
7) <i>the 7th row</i>	26.89 dB	27.11 dB	28.95 dB
8) <i>the 8th row</i>	34.03 dB	34.31 dB	34.56 dB
9) <i>the 9th row</i>	36.43 dB	36.52 dB	37.01 dB
10) <i>the 10th row</i>	42.21 dB	42.42 dB	43.19 dB
11) <i>the 11th row</i>	28.09 dB	28.27 dB	28.63 dB
12) <i>the 12th row</i>	33.25 dB	33.47 dB	33.87 dB
13) <i>the 13th row</i>	26.18 dB	26.42 dB	26.74 dB
14) <i>the 14th row</i>	31.27 dB	31.78 dB	32.04 dB

Table 7 PSNR results of Fig. 8 with a linear regression model in tensorPCA method

Case	90 percent	95 percent	100 percent
1) <i>the 1st row</i>	30.14 dB	30.25 dB	30.56 dB
2) <i>the 2nd row</i>	30.01 dB	30.11 dB	30.26 dB
3) <i>the 3rd row</i>	23.31 dB	23.44 dB	23.62 dB
4) <i>the 4th row</i>	26.01 dB	26.22 dB	26.45 dB
5) <i>the 5th row</i>	34.51 dB	34.77 dB	34.92 dB
6) <i>the 6th row</i>	25.64 dB	25.71 dB	25.97 dB
7) <i>the 7th row</i>	26.21 dB	26.83 dB	27.19 dB
8) <i>the 8th row</i>	30.92 dB	31.20 dB	31.85 dB
9) <i>the 9th row</i>	33.18 dB	33.54 dB	34.17 dB
10) <i>the 10th row</i>	40.68 dB	40.94 dB	41.37 dB
11) <i>the 11th row</i>	27.36 dB	27.52 dB	27.78 dB
12) <i>the 12th row</i>	31.61 dB	31.99 dB	32.32 dB
13) <i>the 13th row</i>	25.89 dB	25.97 dB	26.02 dB
14) <i>the 14th row</i>	30.11 dB	30.24 dB	30.79 dB

Table 8 PSNR results of Fig. 9 with a linear regression model in tensorPCA method

Case	90 percent	95 percent	100 percent
1) <i>the 1st row</i>	31.47 dB	31.53 dB	32.38 dB
2) <i>the 2nd row</i>	31.32 dB	31.49 dB	31.74 dB
3) <i>the 3rd row</i>	24.56 dB	24.62 dB	26.31 dB
4) <i>the 4th row</i>	27.48 dB	27.52 dB	27.88 dB
5) <i>the 5th row</i>	35.47 dB	35.55 dB	35.86 dB
6) <i>the 6th row</i>	26.38 dB	26.49 dB	26.78 dB
7) <i>the 7th row</i>	27.41 dB	27.63 dB	27.99 dB
8) <i>the 8th row</i>	34.16 dB	34.56 dB	34.85 dB
9) <i>the 9th row</i>	36.29 dB	36.44 dB	37.08 dB
10) <i>the 10th row</i>	42.61 dB	42.75 dB	43.59 dB
11) <i>the 11th row</i>	28.08 dB	28.26 dB	28.73 dB
12) <i>the 12th row</i>	33.38 dB	33.57 dB	33.93 dB
13) <i>the 13th row</i>	26.58 dB	26.62 dB	27.01 dB
14) <i>the 14th row</i>	31.69 dB	32.05 dB	32.46 dB

Figure 1 Some original high-resolution color face images (30×30) for testing.

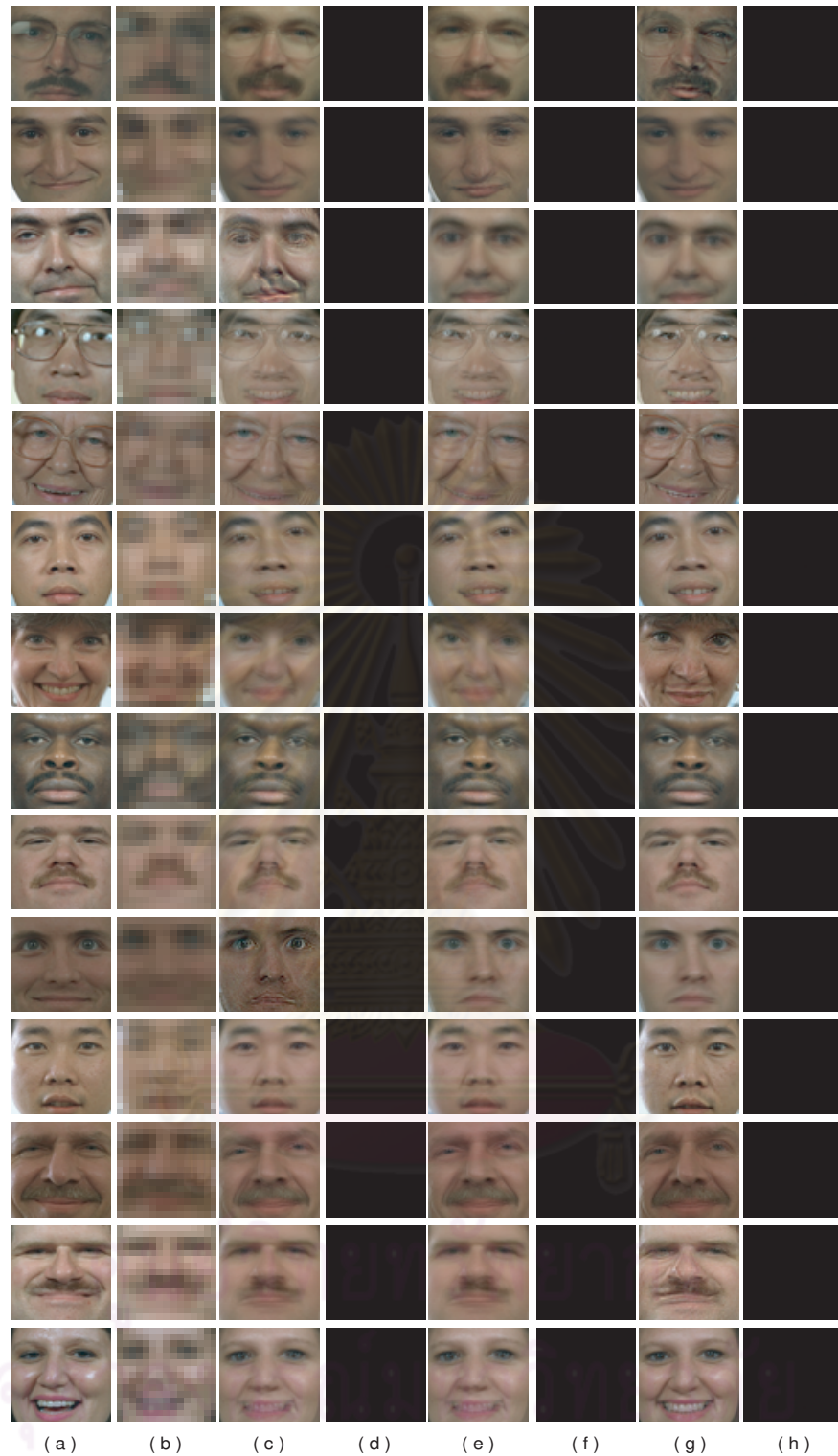


Figure 2: Some of experimental results (RGB color model) with a linear regression model in MPCA method. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

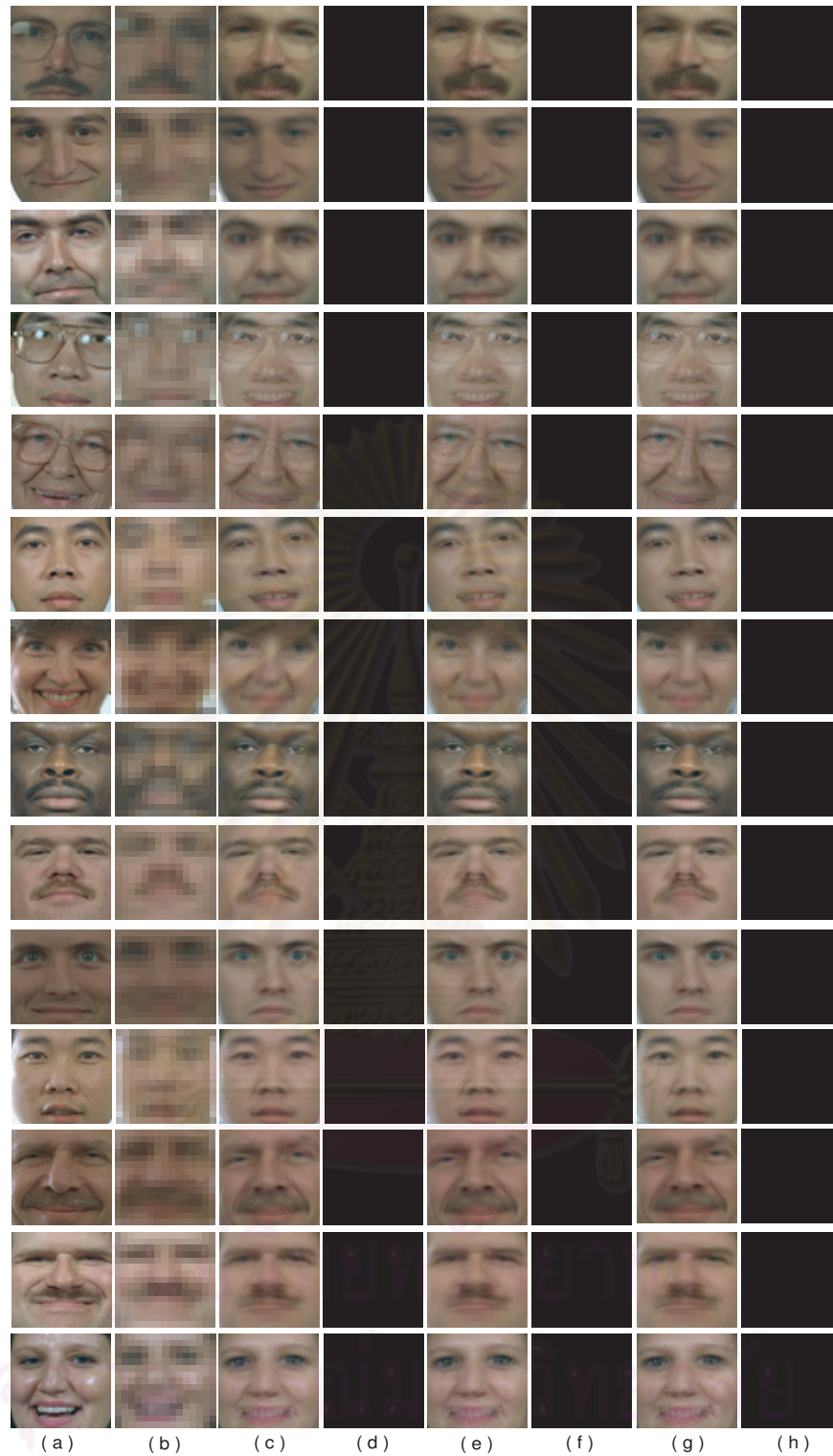


Figure 3: Some of experimental results (YCbCr color model) with a linear regression model in MPCA method. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

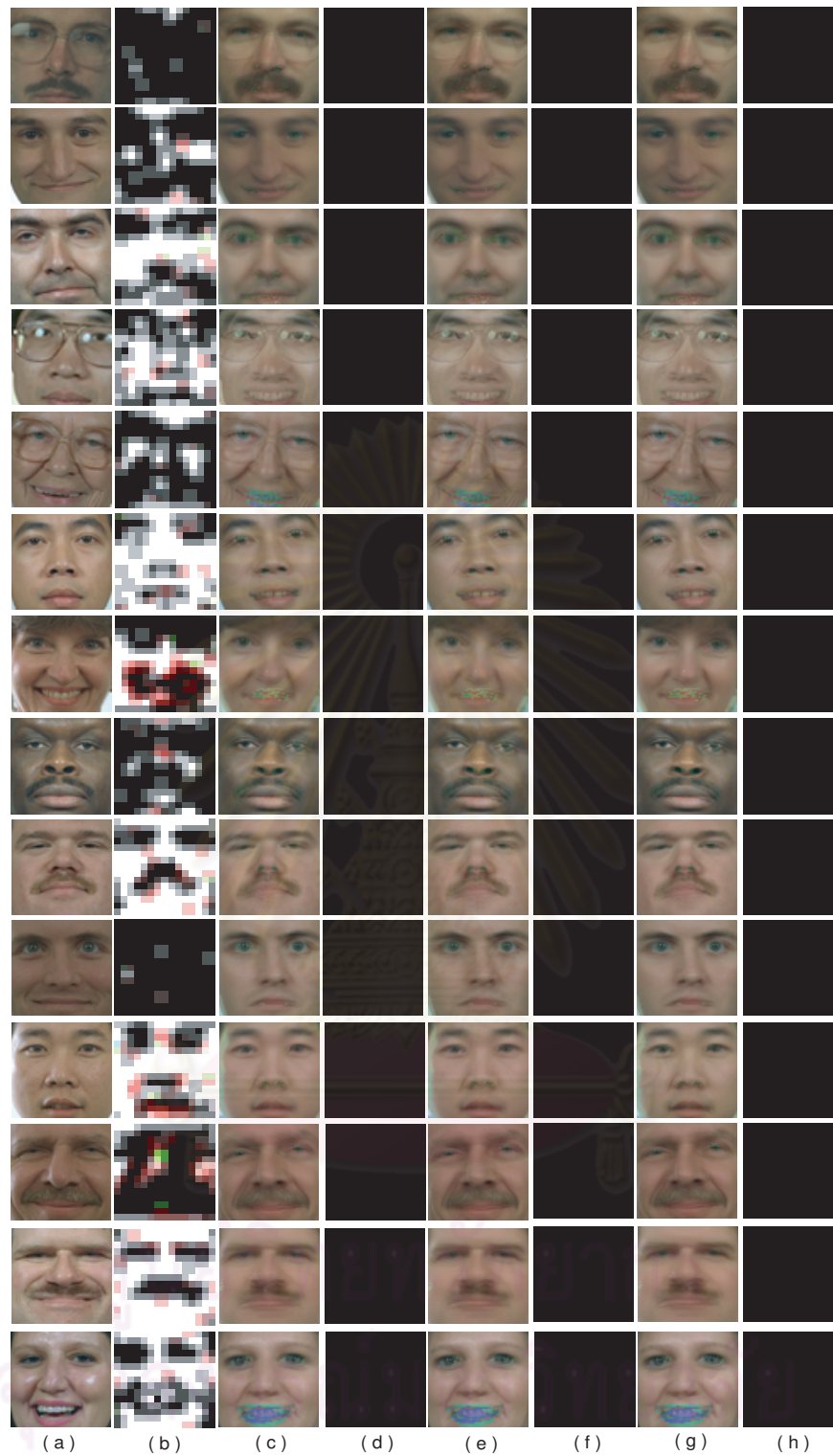


Figure 4: Some of experimental results (HSV color model) with a linear regression model in MPCA method. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

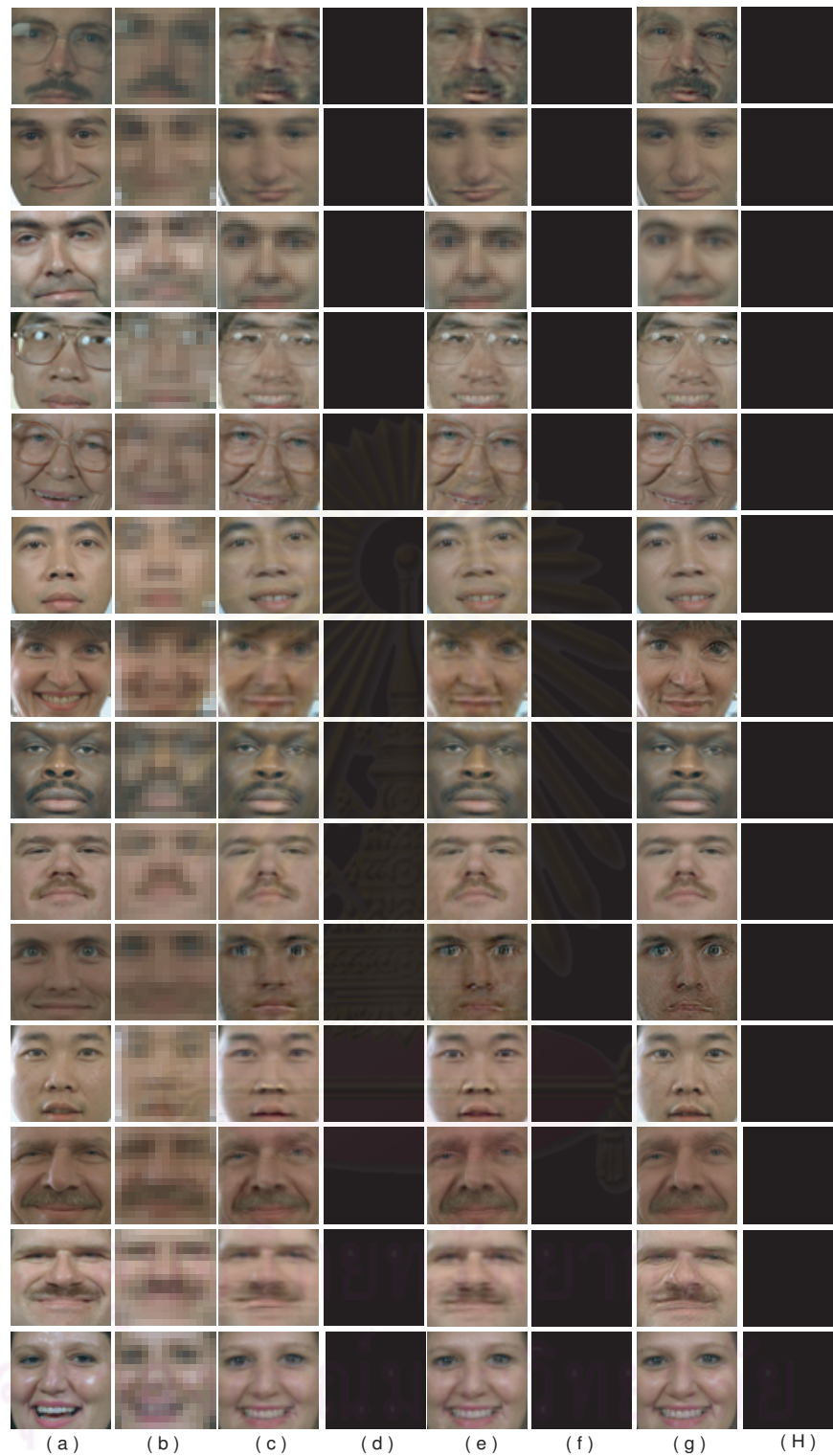


Figure 5: Some of experimental results (CIELAB color model) with a linear regression model in MPCA method. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent MPCA; (d) different image of face hallucination result with 90 percent MPCA; (e) face hallucination result with 95 percent MPCA; (f) different image of face hallucination result with 95 percent MPCA; (g) face hallucination result with 100 percent MPCA; (h) different image of face hallucination result with 100 percent MPCA;

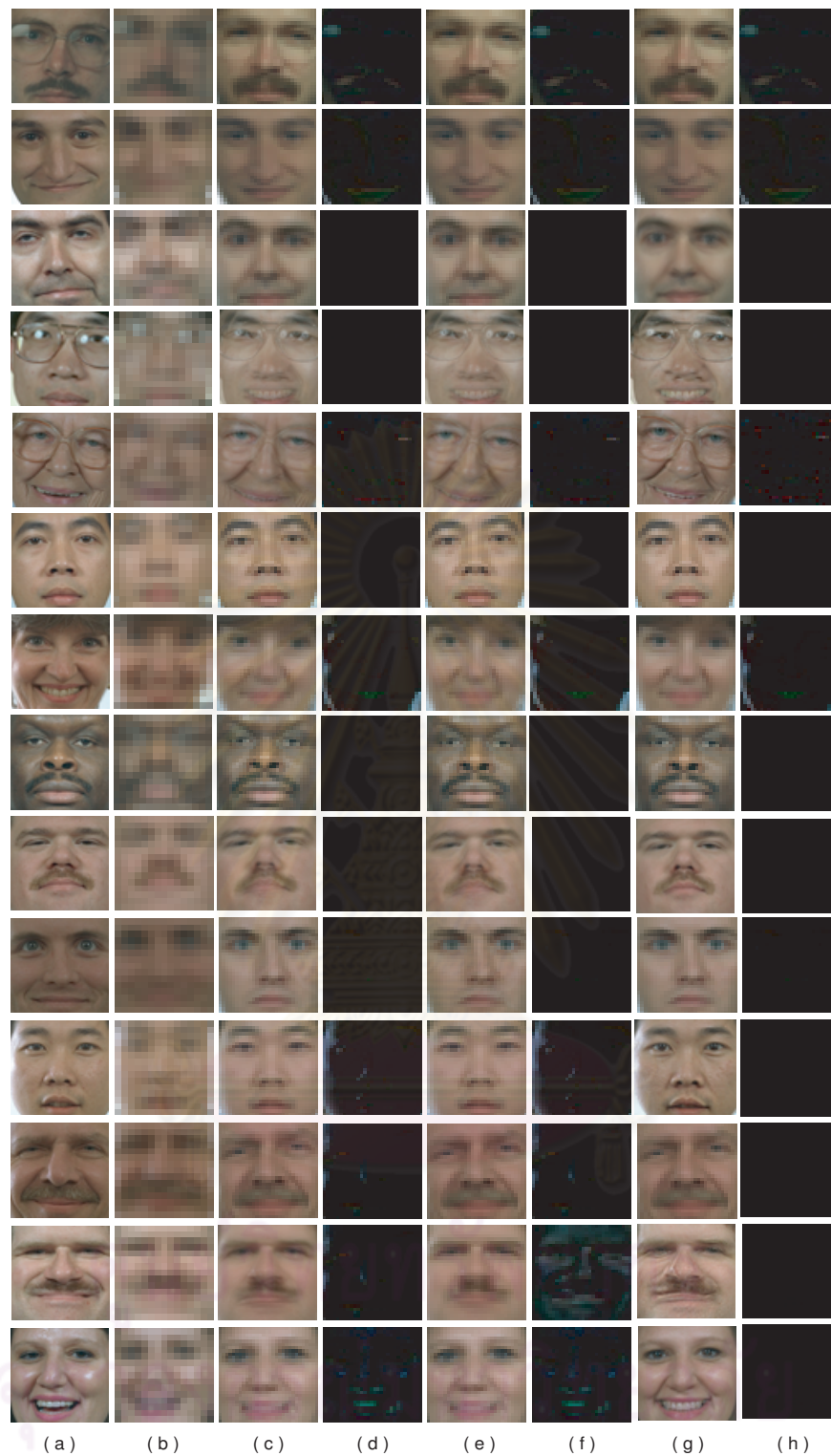


Figure 6: Some of experimental results (RGB color model) with a linear regression model in tensorPCA method. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent tensorPCA; (d) different image of face hallucination result with 90 percent tensorPCA; (e) face hallucination result with 95 percent tensorPCA; (f) different image of face hallucination result with 95 percent tensorPCA; (g) face hallucination result with 100 percent tensorPCA; (h) different image of face hallucination result with 100 percent tensorPCA;

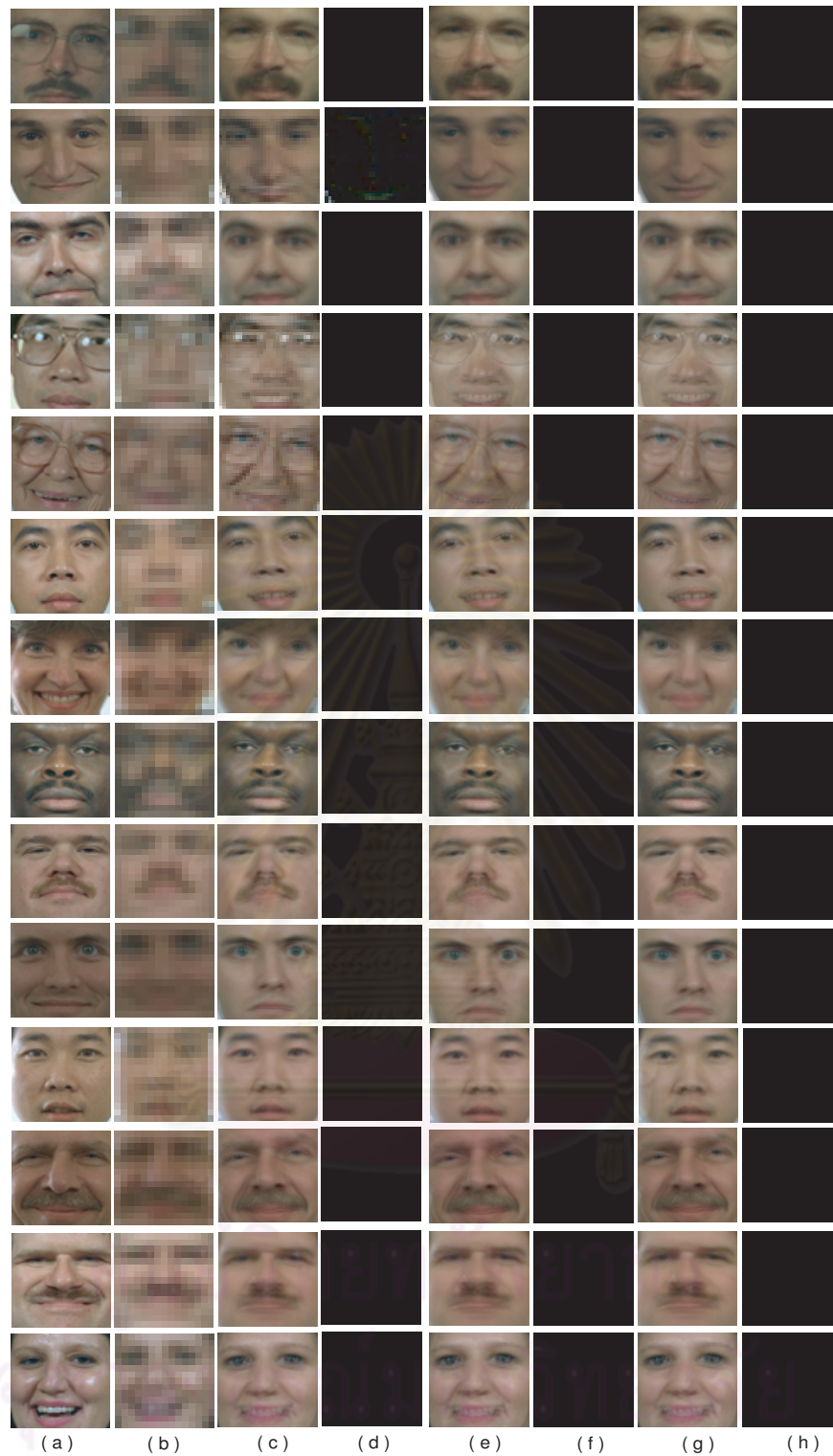


Figure 7: Some of experimental results (YCbCr color model) with a linear regression model in tensorPCA method. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent tensorPCA; (d) different image of face hallucination result with 90 percent tensorPCA; (e) face hallucination result with 95 percent tensorPCA; (f) different image of face hallucination result with 95 percent tensorPCA; (g) face hallucination result with 100 percent tensorPCA; (h) different image of face hallucination result with 100 percent tensorPCA;

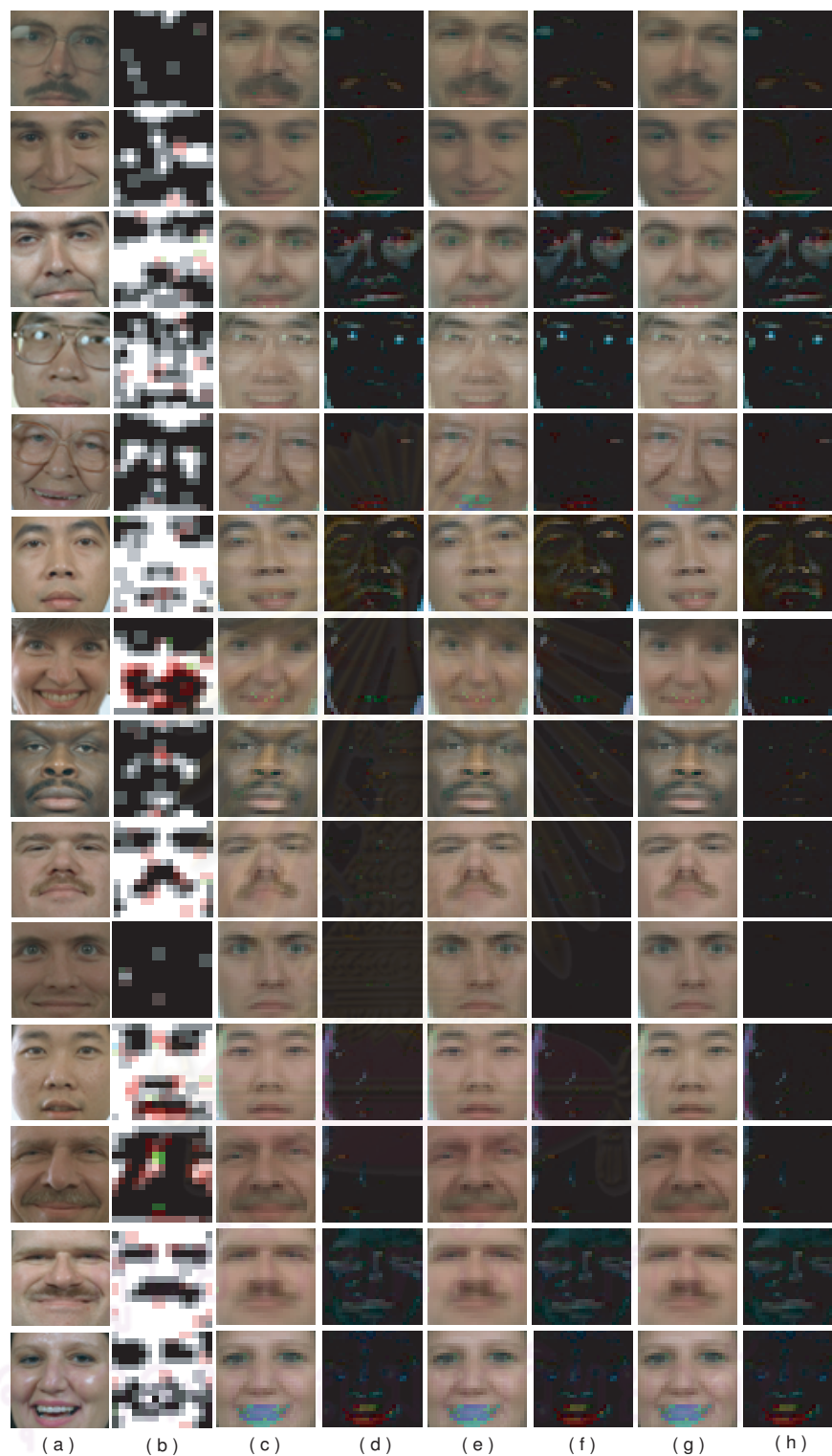


Figure 8: Some of experimental results (HSV color model) with a linear regression model in tensorPCA method. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent tensorPCA; (d) different image of face hallucination result with 90 percent tensorPCA; (e) face hallucination result with 95 percent tensorPCA; (f) different image of face hallucination result with 95 percent tensorPCA; (g) face hallucination result with 100 percent tensorPCA; (h) different image of face hallucination result with 100 percent tensorPCA;

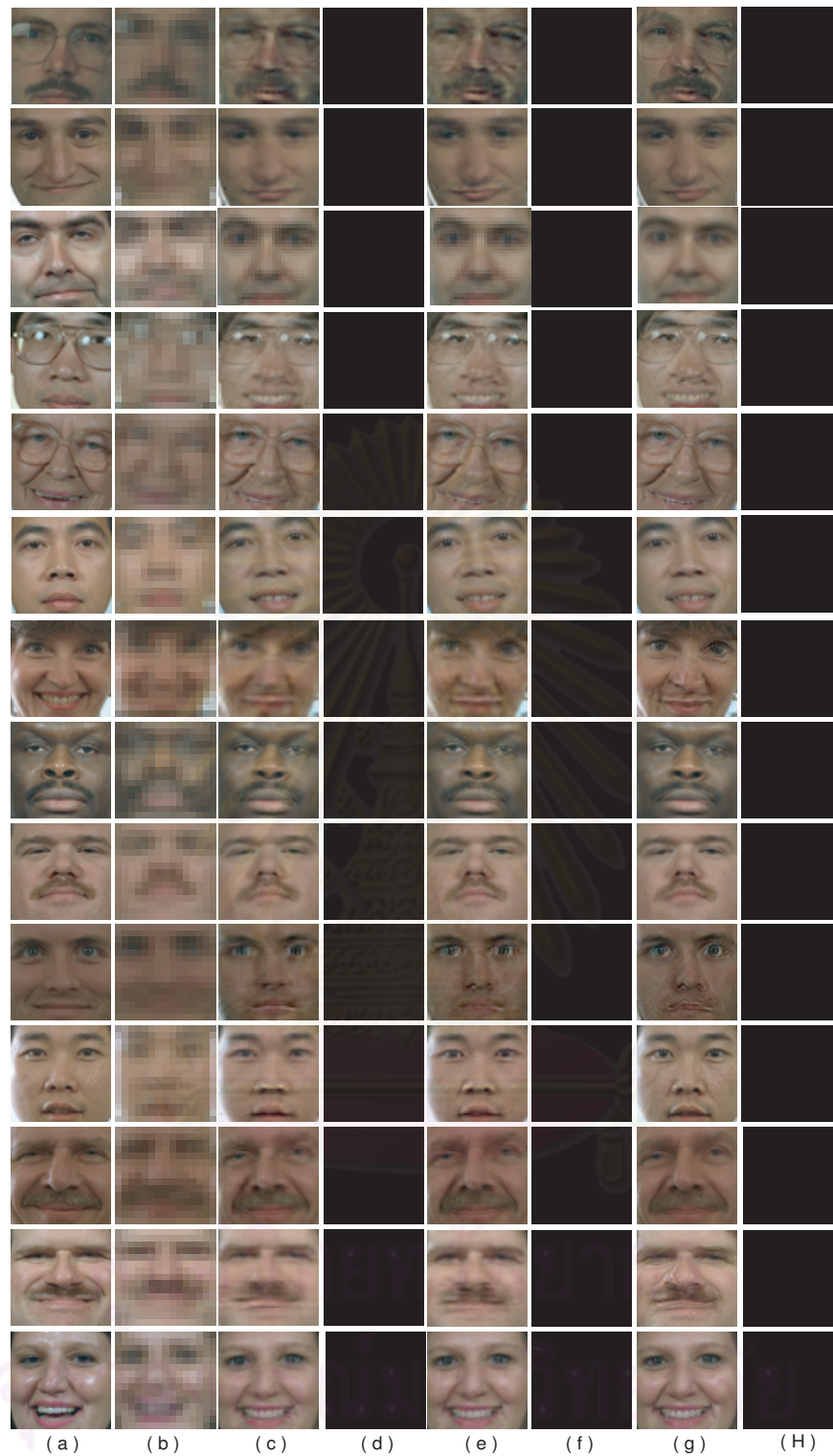


Figure 9: Some of experimental results (CIELAB color model) with a linear regression model in tensorPCA method. (a) original HR images (30×30); (b) input LR images (15×15) with noise, motion and blur in LR images; (c) face hallucination result with 90 percent tensorPCA; (d) different image of face hallucination result with 90 percent tensorPCA; (e) face hallucination result with 95 percent tensorPCA; (f) different image of face hallucination result with 95 percent tensorPCA; (g) face hallucination result with 100 percent tensorPCA; (h) different image of face hallucination result with 100 percent tensorPCA;

Appendix C

Publications and Presentations

Krissada Asavaskulkeit and Somchai Jitapunkul,

“Generalized color face hallucination with linear regression model in MPCA,” In preparation to submit to IEICE Transaction on Information and Systems.

Krissada Asavaskulkeit and Somchai Jitapunkul,

“A Color Face Hallucination with a Linear Regression Model in MPCA,” Proceeding on the 2009 International Conference on Computer Engineering and Applications (ICCEA 2009), pp. 100–104, Manila, Philippine, 6-8 June 2009.

Krissada Asavaskulkeit and Somchai Jitapunkul,

“Performance Evaluation of Color Face Hallucination with a Linear Regression Model in MPCA,” Proceeding on the 2009 International Conference on Image Processing, Computer Vision, Pattern Recognition (IPCV 2009), pp. 387–392, Las Vegas Nevada, USA, 13-16 July 2009.

Krissada Asavaskulkeit and Somchai Jitapunkul,

“The Color Face Hallucination with the Linear Regression Model and MPCA in HSV Space,” Proceeding on the 16th International Conference on Systems, Signals and Image Processing (IWSSIP 2009), Chalkida, Greece, 18-20 June 2009.

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

Vitae

Krissada Asavaskulkeit was born in Bangkok, Thailand in 1981. He received the B.Eng. and M.Eng. degrees from the Department of Electrical Engineering at the Chulalongkorn University, Bangkok, Thailand, in 2001 and 2004 respectively. He is currently pursuing the Doctoral degree in electrical engineering at Chulalongkorn University, Bangkok, Thailand, since 2005. His research areas are digital signal processing in image processing including face hallucination and super-resolution reconstruction.



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย