Human age estimation based on facial images

Zhao, Wei

2017

Zhao, H. (2017). Human age estimation based on facial images. Doctoral thesis, Nanyang Technological University, Singapore.

<http://hdl.handle.net/10356/72826>

<https://doi.org/10.32657/10356/72826>

Human Age Estimation Based on Facial Images

ZHAO WEI

School of Electrical and Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

2017

Acknowledgments

First and foremost, I would like to express my sincerest gratitude to my supervisor Associate Professor Wang Han, who has given me guidance, encouragement and care all the way throughout my Ph.D life at Nanyang Technological University (NTU). Without his continuous support, I wouldn't have had such a great opportunity to study and research in the excellent environment; also, without his dedication for me, I wouldn't have been the man who I am today. Under his supervision, I have not only learned the professional knowledge, principles and attitudes towards research, but also been influenced by his kindness and thinking of others. So I really feel fortunate to have so many wonderful memories with him.

Besides, I am thankful to the School of Electrical and Electronic Engineering (EEE), NTU for awarding me the research scholarship and offering modern facilities and great environment for my research and life.

I would like to extend my thanks to my teammates and labmates Dr. Zhou Lubing, Dr. Mou Wei, Sui Jiacheng, Dr. Huang Jiangshuai, Mou Xiaozheng, Dr. Shin Bok-Suk, Dr. Wang Shiping, Lim Kart Leong, Yang Shuai, Yuan Shenghai, Soner Ulun, Xi Tao, Zhang Handuo for their inspiring discussions and useful suggestions during the process of my research. I am also grateful to Dr. Guo Fanghong, Kou Fei, Jiang Jun, Jiang Wentao, Wang Yuanzhe, Yue Yufeng, Wu Songwei, Yang Chule, Wang Hongwei, Zhang Jun, Liu Ting, Li Dan, Chen Zhuozhu, Dr. Ye Maojiao, Hu Zelin, Zhao Rui, Wei Zhe, Gong Fanming, Jiang Xiaoyue, Zeng Yijie and Zhou Xiaosong who share plenty of advices, experiences and happy moments with me. I am also indebted to Wu Bin, Li Yongjian, Yu Lei and Yang Jinghan for their friendship and advices during these years.

Finally, I would like to dedicate this thesis to my family. I am deeply grateful to my mother Tian Yun and my father Zhao Shukun for all their constant love, support, encouragement, endeavor, persistence, patience and witness during the process of my growth, which deserve my whole lifetime to appreciate and return. Many thanks goes to my hobbies: my piano, singing, music, basketball, night running, gym, table tennis, travelling, photography and cooking for their loyal companion regardless of my condition, mood and time.

Contents

Acknowledgments	i
Abstract	vi
List of Tables	xii
1 Introduction	1
1.1 Background and Motivations	1
1.2 Main Contributions of the Thesis	6
1.3 Organization of the Thesis	7
2 Literature Review	9
2.1 Human’s Face Aging Process	9
2.2 Aging Representations and Models for Face Images	11
2.2.1 Anthropometric Model	13
2.2.2 Active Appearance Models (AAM)	17
2.2.3 AGing pattErn Subspace (AGES)	17
2.2.4 Age Manifold	19
2.2.5 Appearance-based Models	21
2.3 Age Estimation Algorithms	25
2.3.1 Classification	25
2.3.2 Regression	27
2.3.3 Hybrid of Classification and Regression	29
2.3.4 Ordinal Ranking Based Methods	31

3	Ordinal Ranking Based Approaches for Facial Age Estimation	34
3.1	Multi-Feature Extreme Ordinal Ranking Machine (MFEORM) for Facial Age Estimation	34
3.1.1	Introduction	34
3.1.2	Background on Regression Theory and Kernel Ridge Regression .	35
3.1.3	Randomized Single Hidden-layer Feedforward Neural Networks . .	38
3.1.4	Proposed Method — Multi-feature Extreme Ordinal Ranking Machine (MFEORM)	43
3.1.5	Experiments	45
3.1.6	Concluding Remarks	51
3.2	Advanced Ordinal Ranking for Facial Age Estimation	52
3.2.1	Introduction	52
3.2.2	Proposed Advanced Ordinal Ranking in facial age estimation . . .	54
3.2.3	Experiments	56
3.2.4	Concluding Remarks	58
4	Strategic Decision-making Learning from Label Distributions for Facial Age Estimation	60
4.1	Introduction	60
4.2	Label Distribution Learning and Its Decision-Making Criterion	63
4.3	Strategic Decision-making Label Distribution Learning (SDM-LDL) for Facial Age Estimation	68
4.4	Experiments	75
4.4.1	Experimental Environment Settings	75
4.4.2	Experimental Results	75
4.5	Concluding Remarks	90

5	Conclusions and Future Research	92
5.1	Conclusions and Discussions	92
5.2	Future Works	95
A	Salient Object Detection with Spatiotemporal Background Prior for Video	98
A.1	Introduction	98
A.2	Our approach	104
A.2.1	Temporal background priors estimation	105
A.2.2	Spatial background priors estimation	109
A.2.3	Fusion of spatiotemporal background priors	110
A.2.4	Graph-based saliency detection	111
A.3	Experiments and Analysis	116
A.3.1	Datasets with simple scenes: SegTrack and SegTrack v2	117
A.3.2	Datasets with complex scenes: Freiburg-Berkeley Motion Segmentation Dataset	122
A.4	Concluding Remarks	125
	Publications	127
	References	128

Abstract

With the rapid development of computer vision, pattern recognition and biometrics, more and more attention has been paid to computer-based human facial age estimation, which is widely used when an individual's age needs to be obtained without specifically identifying other irrelevant personal information. Thus it has broad computer vision applications including security surveillance, forensics, biometrics, human-computer interaction (HCI), electronic customer information management, age-specific precision marketing (like age-based visual advertisement), entertainment and so on. There are quantities of realistic scenarios where facial age estimation can be put to good use: with a monitoring camera the juveniles can be warned or even stopped from purchasing cigarettes or banned drugs from vending machines; the underage can be prevented from entering wine bars and buying alcohol drinks; the elders can be cautioned when they want to try some high-risk rides in theme parks, like roller coasters and forest adventures; children's entry into harmful websites can also be restrained which contain violence, pornography or other restricted contents, etc.

However, there is no doubt that this task is tough and challenging. The difficulties of computer-based facial age estimation are reflected in the following aspects. 1) Difference of aging process: different people have their own living environment, ethnic group, gender, lifestyle, social contact, health condition and even gene diversity, which all together determine the speed of aging. 2) Shape or texture: different forms of aging will emerge in different age levels. For example, from infancy to adolescence, the craniofacial growth (shape growth) is the main change. However, from adult period to old age, the craniofacial change decreases remarkably and skin transformation (texture change) would be the

most prominent change. 3) Data insufficiency: nowadays only a very limited number of aging datasets are available, especially which can cover all the age range. 4) Disturbance: under normal conditions, almost all the people are eager for youth and tend to show their younger face, especially for females. So the final age estimation results will be largely interfered by using cosmetics, accessories and even plastic surgery, which can obtain the therapeutic or cosmetic reformation of appearance and tissue.

Consequently, judging from the significance of both research and application, facial age estimation is important for the computer vision community. The goal of this thesis is to present various estimation algorithms in different perspectives to handle this challenging research topic.

In the past several years, a lot of facial age estimation algorithms have been put forward, some of which are able to obtain rather satisfying performance. Among them, fundamentally, most of the traditional facial age estimation approaches are based on classification, regression, the hybrid of the two and ordinal ranking problem. However, most of the existing facial age estimation methods are originated from solving complicated mathematical optimization problems and thus consume huge quantities of time in the training process. To refrain from such algorithm complexity while maintaining a high estimation accuracy, a multi-feature extreme ordinal ranking machine (MFEORM) is proposed for facial age estimation, which combines multi-feature extraction, ordinal hyperplanes ranking and extreme learning machine's faster learning speed together for both high estimation accuracy and efficiency, in which better performance has been obtained. Experimental results clearly demonstrate that the proposed approach can sharply reduce the runtime (at most nearly one hundred times faster) while achieving comparable or better estimation performances than the state-of-the-art approaches. The inner properties of MFEORM are further explored with more superiorities.

The second contribution of this thesis is driven by ordinal hyperplanes ranking, which is one of the state-of-the-art algorithms that achieve top performance in facial age estimation. By further experiments, it is found that this approach is overly dependent

on the idealizations which underlie the ranking rule. This lack of robustness can result in unnecessary erroneous estimation deviations which degrade the performance. So two approaches are proposed with new ranking rules which combine the classifier accuracy and the obtained label in each binary classification substep for ranking criteria. As a result, these two approaches minimize the deviations (normal error variance) of binary classifiers. Also, the extreme learning machine is utilized, taking full advantages of its high learning speed and accuracy. Experimental results from public datasets are presented to demonstrate that the proposed algorithms reduce the mean absolute error and improve age estimation performance while reducing the runtime significantly.

Thirdly, label distribution learning (LDL) is another state-of-the-art methodology in facial age estimation. LDL actually depicts the age label from a different perspective: it takes the age of each facial image instance as a label distribution with a series of age labels rather than the single chronological age label which is commonly used. Since adjacent age labels are not separated entities and they have a close relationship with each other, LDL can better take into account the inter-relationship among neighboring age labels. However, this methodology is deficient in its simple decision-making criterion: the final predicted age is only selected at the one with maximum description degree. In many cases, different age labels may have very similar description degrees. Consequently, blindly deciding the estimated age by virtue of the highest description degree would miss or neglect other valuable age labels which may contribute much to the final predicted age. In this thesis, a strategic decision-making label distribution learning algorithm (SDM-LDL) is proposed with a series of strategies specialized for different types of age label distribution. Experimental results show the superiority and validity of all the proposed strategic decision-making learning algorithms over the existing label distribution learning and other single-label learning algorithms for facial age estimation. The inner properties of SDM-LDL are further explored with more superiorities.

List of Figures

1.1	Albert Einstein’s facial images for aging process (from Internet)	2
2.1	Craniofacial growth from birth to adulthood [1, 2]	9
2.2	Main changes of facial skin textures from adulthood to old age (ten years per sketch) [3, 4]	10
2.3	Flowchart of a typical facial age estimation system	12
2.4	The six distance ratios of facial landmarks proposed by Kwon and Lobo [5]	14
2.5	The eight distance ratios of facial feature points by Ramanathan and Chel- lappa [1]	16
2.6	The vectorization and schematic diagram of the aging pattern in AGES [6]. “m” stands for the missing facial images in the corresponding age. . .	18
2.7	Performance comparison of different linear manifold algorithms for facial age estimation [7]	20
2.8	Classification results of age classification using LBP [8]	22
3.1	A typical example of single-layer feedforward neural network(SLFN) . . .	39
3.2	Facial image examples of one subject in sequential order of different age values in the FG-NET database	46
3.3	Flow chart of ordinal hyperplane ranking for facial age estimation	52
3.4	Ideal and different deviated conditions for the ranking rule of ordinal ranking	53
3.5	Cumulative Scores (CS) of compared age estimation algorithms	59
4.1	Three different types of label distribution	62

4.2	Two primary age label distributions (a), (b) and multiple age labels with same description degrees (c) for the chronological age a	64
4.3	Histogram illustrations for the example of Table 4.1 and 4.2: (a) corresponds to Table 4.1; (b) and (c) correspond to Table 4.2	74
4.4	The line chart (variation tendency) of MAEs with respect to different standard deviations for the original LDL algorithm and the proposed SDM-LDL algorithms with different strategies when age label distribution is generated as Gaussian-like	77
4.5	Different bottom length when generating the triangle age label distribution for the chronological age a	82
4.6	The line chart (variation tendency) of MAEs with respect to different bottom length for the original LDL algorithm and the proposed SDM-LDL algorithms with different strategies when age label distribution is generated as triangle	84
4.7	Different number of age labels when generating age label distribution as multi-label with equal description degrees for the chronological age a	86
4.8	The line chart (variation tendency) of MAEs with respect to different number of age labels for the original LDL and the proposed SDM-LDL with different strategies when age label distribution is generated as multi-label with equal description degrees	88
A.1	What are good background priors?	100
A.2	Framework of the proposed approach	105

A.3	One frame instance to show the computation of saliency values based on the obtained spatiotemporal background priors from Sec. A.2.3: (a) the input frame in a video; (b) extracting superpixels from (a); (c) and (d) are the graphs constructed by appearance and motion information respectively, in which black dots represent background prior superpixels; (e) is the appearance saliency map based on geodesic distances; after setting a threshold, the appearance-based binary salient region (ABSR) as query can be obtained like (f); by using manifold ranking, motion saliency map (g) can be obtained from (d) and (f); final saliency map (h) is generated as the fusion of (e) and (g).	112
A.4	Quantitative comparisons between the proposed method and other six existing approaches using SegTrack and SegTrack v2 datasets as the benchmark of simple scenes: (a) and (c) are precision-recall curves for SegTrack and SegTrack v2 respectively by setting the thresholds from 0 to 255 for obtained saliency maps; (b) and (d) are mean absolute errors (MAE) for SegTrack and SegTrack v2 respectively.	118
A.5	Visualization effects for the compared methods using SegTrack and SegTrack v2 as the benchmark of simple scenes.	120
A.6	Quantitative comparisons of PR curve and MAE between the proposed method and other six existing approaches using FBMS-59 datasets as the benchmark of complex scenes.	123
A.7	Some typical examples of complex scenes and the corresponding visualization effects using the compared methods: (a) multiple salient objects; (b) partial occlusion; (c) complete occlusion; (d) complex background; (e) color similarity between salient object and background; (f) light disturbance.	124

List of Tables

3.1	FG-NET's age level distribution	47
3.2	Consumed time of the compared MFEORM and MFOR	48
3.3	MAEs of compared MEFORM and other popular age estimation algorithms on FG-NET database	49
3.4	Different regularization coefficient with fixed kernel parameter in MFEORM	49
3.5	Different kernel parameter with fixed regularization coefficient in MFEORM	50
3.6	Comparison of different number of feature models	50
3.7	Comparison of different dimensions' impact	51
3.8	MAEs and runtime of the compared ordinal-ranking-based age estimation algorithms	57
3.9	MAEs of compared popular age estimation algorithms	57
4.1	An example of the obtained age label distribution for a new facial image x' (with the authentic age of 16)	72
4.2	Based on Table 4.1, the age label distribution with descending-order description degrees and successive differences di	72
4.3	The comparison of deviations from authentic age based on the example of Table 4.1	73

4.4	MAEs of the compared original LDL algorithm without decision-making strategy and the proposed SDM-LDL algorithms with different strategies on the condition that age label distribution is generated as Gaussian-like (σ from 1 to 10; for Strategy 1 and 2, best results are shown in the table along with the value of N then; for every σ , the optimal MAEs are marked in bold)	76
4.5	The impacts of different values of N (2-10) on MAE using the proposed SDM-LDL algorithms with Strategy 1 and 2 when age label distribution is generated as Gaussian-like (σ from 1 to 10; for every σ , the optimal MAEs are marked in bold)	80
4.6	MAEs of the compared original LDL algorithm without decision-making strategy and the proposed SDM-LDL algorithms with different strategies on the condition that age label distribution is generated as triangle (bottom length from 4 to 30; for Strategy 1 and 2, best results are shown in the table along with the value of N then; for every bottom length, the optimal MAEs are marked in bold)	81
4.7	The impacts of different values of N (2-10) on MAE using the proposed SDM-LDL algorithms with Strategy 1 and 2 when age label distribution is generated as triangle (bottom length from 4 to 30; for every value of bottom length, the optimal MAEs are marked in bold)	85
4.8	MAEs of the compared original LDL algorithm without decision-making strategy and the proposed SDM-LDL algorithms with different strategies on the condition that age label distribution is generated as multi-label with equal description degrees (number of age labels from 3,5,7,... to 15; for Strategy 1 and 2, best results are shown in the table along with the value of N then; for every number of age labels, the optimal MAEs are marked in bold)	87

4.9	MAEs of the compared three different patterns in generating age label distribution (the controlled variables in each pattern are traversed within the whole given range and best results are reported; for every row in the table, the optimal MAEs are marked in bold)	89
4.10	MAEs of different facial age estimation algorithms	90

Chapter 1

Introduction

1.1 Background and Motivations

As is known to all, human faces can imply different kinds of information, such as origin, appearance, emotional tendency, health condition, part of social properties and so on. Among them, age is an important personal attribute, which can be roughly extrapolated from face images. For instance, most people are able to realize the aging process from a series of Albert Einstein's facial images, just as Figure 1.1 shows; well-trained salespersons can distinguish most of their customers' ages with only minor deviation. However, such professionals are obviously scarce compared to so many actual application scenarios. Also in social networking, the age shown from facial images is treated as a vital semantic information [9, 10]. So a question comes up: can a computer do the same thing with high efficiency and the same or even more competitive performance when compared to human beings? For the past few years, with the rapid development of computer vision, pattern recognition and biometrics, the above question's answer has been given by facial age estimation. Fundamentally, age estimation can be divided into two "branches": to determine the exact age (e.g., 23 years old) or the age range (e.g., old, middle-aged, young or baby). This thesis basically aims at the former goal, which most of the existing age estimation algorithms mainly focus on.

Firstly, four key concepts are listed to give a clear definition and differentiation of human age.

Figure 1.1: Albert Einstein’s facial images for aging process (from Internet)

- 1) *Appearance age*: the individual age shown on the appearance of a person’s face. In other words, it is the answer to the question “how old a person looks like from his/her face (appearance)”.
- 2) *Actual age (Chronological age)*: the authentic individual age.
- 3) *Estimated age*: the individual age estimated by computers using certain algorithms.
- 4) *Perceived age*: the individual age measured by testees from the facial images.

Among the above concepts, the actual age is usually treated as the ground truth, which is basically accordant with the appearance age. Also, the estimated age and the perceived age are based on the appearance age.

Facial age estimation has many practical applications in the real world. For biometrics, facial age estimation belongs to one type of “soft” biometrics traits [11], which act as the supplementary information of a person’s identity. As a result, it can be used to reinforce the basic features of biometrics, like fingerprint, face, iris, palm print, DNA, retina and so on, aiming at enhancing the overall performance of a fundamental biometrics system. For example, when combined with face recognition system, facial age estimation is able to detect the passengers’ ages who cross the border with relatively high accuracy. Also, with the development of our information era and higher demand of privacy information security, more and more attention has been paid to surveillance monitoring and security control [12]. One good case in point is that more and more money

transfer frauds on automatic teller machines happen among the people with a certain age group, which can be early warned and even blocked by age estimation system. What's more, facial age estimation can be embedded in the intelligent care unit of the health care system for customized services, such as communicating with patients of different age levels for respective preferences and needs. For human-computer interaction (HCI), facial age estimation also plays an important role. It can automatically change the displayed contents to the user according to his or her age. In this way, the HCI system is able to meet the preferences of all ages. One instance is that in the supermarket, facial age estimation can be incorporated into the intelligent shopping trolley so as to provide the corresponding recommendations for consumers in different ages. In forensics, by using the estimated age of facial images from the gallery database, automatic age estimation system can filtrate potential suspects efficiently without spending lots of manpower and material resources. Another application of facial age estimation is the electronic customer relationship management (ECRM) [13]. Actually it is a strategy for management which utilizes information technology and multimedia interacting to efficiently manage all relationships with customers. More specifically, it combines the intrinsic resources of enterprises and the extrinsic business strategies to grasp and further satisfy what customers need. Since it varies in consuming preferences, habits, expectations for different age groups, the enterprises can get more business opportunities through knowing customers' specific requirements and satisfying these customized needs. However, the most difficult task of all the procedures is to access and analyze customers' personal information. This would require enterprises to build a long-term relationship with customers and cost a great deal of time and money. For instance, Walmart wants to get the information of what proportions of different age groups prefer to purchase coke or coffee; an advertising company needs to grasp the knowledge of the target specific spectators for corresponding advertisements in accordance with respective age groups; an Internet service provider wants to obtain which age group shows more interests in their newly-released service plans; also a clothing store might display a sportswear or a professional

suit for the passing young boys or grown men respectively on the monitor outside the store. Apparently, the above-mentioned cannot be realized because of personal privacy. Nevertheless, if we can take full advantage of the automatic facial age estimation system with the help of a camera, the demographic information can be obtained without invading personal privacy by taking photos of customers' faces and marking the age groups or even exact ages automatically. Although there are rich practical applications, the existing knowledge from perception areas tells us that there exists plenty of challenges realizing automatic computer-based facial age estimation with high accuracy and efficiency. Unlike other face-oriented problems, the difficulties of computer-based facial age estimation are reflected in the following aspects:

- 1) Difference of aging process: Different people have their own living environment, ethnic group, lifestyle, social contact, health condition and gene diversity, which all together determine the speed of aging. Also, there might exist other influential factors that cause advanced face sign. For example, overly-exposure to the sunlight will lead to solar aging which is the outcome of ultraviolet radiation. This effect will reflect on the texture of face. Particularly, Stone [4] has found that genetic predisposition, emotional stress, smoking, disease, drastic weight changes, extreme climate events all lead to faster aging process. Thus, the aging process of our face is individualized and not controllable [6, 14, 15]. Furthermore, the ranking of ordinal age labels is intractable because there exists much uncertainty within different age labels [16, 17].
- 2) Distinctions between males and females: In reality, females are inclined to use more makeups and accessories than males, so it is difficult to simulate a unified face aging pattern for males and females [18]. Usually females' face would comply with younger appearances than they actually are. To the best of our knowledge, there's still no perfect way to obtain a universal discriminative feature for facial age estimation

while reducing personal differences. Consequently, final estimation results will be largely interfered.

- 3) Shape or texture: Different forms of aging will emerge in different age levels. For example, from infancy to adolescence, the craniofacial growth (shape growth) is the main change. However, from adult period to old age, the craniofacial change decreases remarkably and skin transformation (texture change) would be the most prominent change.
- 4) Data insufficiency: A very limited number of aging databases are available, among which only a small proportion are open to public. Another problem is that the majority of aging databases cannot cover all the age range: some are in lack of young or old age levels; some are unevenly distributed for all ages. Although more data can be collected through Internet image mining, it is not easy task to obtain large datasets which contain huge quantities of subjects with facial images distributing evenly for all age labels.

In recent years, more and more researchers show interest in 1) how a computer can handle the problem of human age estimation based on facial image; 2) what approaches can be used to overcome the difficulties and improve the estimation accuracy. Also, some industrial companies have developed facial age estimation systems and integrate them into their products, such as Microsoft, NEC Laboratories and VideoMining. Following this tendency, this thesis mainly focus on the facial aging models and age estimation algorithms, proposing several new algorithms to further improve the estimation accuracy and efficiency. Besides, some defects have been found for certain existing estimation algorithms and thus been revised; new algorithms are proposed for the corresponding ranking rules and decision-making strategies.

1.2 Main Contributions of the Thesis

The main contributions of this thesis are summarized as follows:

- 1) Multi-feature extreme ordinal ranking machine (MFEORM) for facial age estimation is proposed, which combines the advantages of multi-feature space, age's natural characteristics of ordinal information and extreme learning machine's rapid learning rate. MFEORM achieves similar or even better performances and simultaneously takes much less time compared to state-of-the-art methods due to its low training(learning) complexity. The proposed approach avoids tediously conducting the iterative computation for mathematical optimization problem and improves efficiency.

The experiments are conducted comprehensively and thoroughly from the internal & external aspects respectively and find out more about MFEORM's particular characteristics and advantages. Moreover, further properties are explored: a) the influence of different number of feature models on the final results; b) the influence of number of dimensions (after PCA dimension-reduction) on the final results.

- 2) Two new approaches of ranking rules for ordinal hyperplane ranking (AOR) in facial age estimation have been proposed, which effectively overcome the accidental classifying deviations in each binary classification subproblem and improve age estimation performances.

Comprehensive comparisons are made with respect to the Mean Absolute Error (MAE), Cumulative Score (CS) and also the consumed time for different popular age estimation approaches (including AOR), especially those based on ordinal ranking, aiming at evaluating the overall estimation performance and efficiency (normally neglected by common age estimation algorithms) of these methods.

- 3) The above two aspects of contributions depict the age label as single label. From another point of view, every age label has a certain “implicit” relationship with its adjacent age labels. So each age label can also be treated as multiple adjacent labels described by different “description degrees”, by which age estimation can be solved with label distribution learning (LDL). However, the original LDL suffers from their simple decision-making criterion which only blindly chooses the age label with maximum description degree as the final predicted age and fails to take the characteristics of aging process into account. Therefore, a series of strategic decision-making algorithms are proposed for label distribution learning (SDM-LDL) in facial age estimation to effectively solve this problem.

Comprehensive experiments for three different kinds of age label distribution (Gaussian-like, triangle and multi-label with equal description degrees) prove the validity, superiority and robustness of the proposed SDM-LDL algorithms against the original LDL and other existing facial age estimation algorithms. In addition, the respective advantages and properties for each strategy in SDM-LDL are summarized.

1.3 Organization of the Thesis

The reminder of this thesis is organized as follows: Chapter 2 gives a brief literature review about human’s face aging process, different aging representations (models) for face images and existing age estimation algorithms respectively. Chapter 3 introduces two ordinal ranking based approaches for facial age estimation. The first approach (Sec. 3.1) is MFEORM, achieving comparable or better estimation performances than traditional approaches while sharply reducing the runtime (even up to nearly one hundred times faster). The inner properties of MFEORM are further explored with more superiorities. In Sec. 3.2, the defects of ordinal-ranking-based algorithms have been found out during our experiments and the advanced ordinal ranking (AOR) algorithm is further put forward with better performance for facial age estimation. Chapter 4 depicts the age

estimation problem from another perspective — label distribution, fully taking advantages of age’s characteristics and the implicit relationship between different ages. A new label-distribution-learning-based algorithm with multiple dedicated decision rules for age estimation is then proposed in this chapter. Finally, summary and future work are concluded in Chapter 5. In the real-world age estimation scenarios where the background account for the majority, huge image data with large dimensionality is hard to process. So in Appendix A, salient object detection is used to solve this problem and a novel framework for salient object detection is demonstrated using spatiotemporal background priors.

Chapter 2

Literature Review

2.1 Human's Face Aging Process

Generally speaking, human's aging is a slow but irreversible process with strong personalized influential factors, though some chemicals like retinoids might bring minor photoaging effects. Although people have difference in gene, gender, health, working environment, living style, weather condition, sociality and so on, which, as a result, cause different aging process and different forms of representation, there still exists uniform characteristics of changes [19]. From the perspective of anthropometry or biology, two phases of human's life have distinct facial changes and aging process.

Figure 2.1: Craniofacial growth from birth to adulthood [1, 2]

Figure 2.2: Main changes of facial skin textures from adulthood to old age (ten years per sketch) [3, 4]

In the early age, namely from birth to adulthood, the leading change on faces is the craniofacial growth [20]. More specifically, the facial outline is gradually getting expanded with the size of the face larger and larger, as Fig. 2.1 shows. During this stage, along with the cranium growth, the forehead slopes back and shrinks, reserving more spaces on the cranium surface. Meanwhile, the features on the face (e.g., nose, eyes, mouth and ears) enlarge their dimensions so as to occupy the gap spaces. The chin tends to be more protruding while cheeks expand outwards for more areas. Pores might spread out and expand. Hairs and the mustache have the tendency to be thicker with skin color changing slightly. On the contrary, No much changes on the facial skin happen compared to those craniofacial changes [19].

However, as shown in Fig. 2.2, in the stage from adult to old, the dominant facial change is the aging of the skin texture instead of shape changes, which tend to be milder. With the age increasing, the facial skin will gradually become darker, thinner and less elastic because of collagen's loss and gravity effects. Biologic aging also makes spots and wrinkles turn out on the face. In the parts of deep connection, e.g., eyelids, chin, cheeks and nose, the elasticity of soft tissues and muscles tend to be weak with aggravated fat deposition. In other part of the face, fat is inclined to shrink or be assimilated. These prominent changes result in the skin's sagging with the outward manifestation of dropping cheek, double chin and drooping eye bags. Although the craniofacial change isn't that

sharp from adult to old compared to the one in the childhood, it is still perceptible, especially for faces of women. The face would have a tendency of changing from a U-like shape or reverse triangle to the rectangle or trapezoid shape [3, 4]. The skeleton under the skin might also collapse which would be a promoting factor for the aging of the skin, like droops, creases and wrinkles. It is likely that facial aging in this period makes the facial muscle out of control so as to incur the unintentional variations of face behaviors and movements [20].

Facial aging can be interpreted from another perspective of aesthetics and perception. Two aspects of faces are involved, which are attractiveness and distinctiveness. In fact, these two aspects are inversely connected. More specifically, a more attractive and typical face tends to have less distinctiveness, which was proposed by Rhodes et al. in 1996 [21]. After this theory, researchers further put forward the theory that facial aging has a certain relationship with distinctiveness and attractiveness [22]. They used the European descent young males and females as their experimental data, after which they found that older faces showed more distinctiveness but less attractiveness. When they increased the degree of facial caricature, the degree of attractiveness lowered with the adding distinctiveness and meanwhile the face seemed older. Langlois et al. [23] presented the hypothesis that the composite average face would be the most attractive. It was also shown [24] that combining different faces can eliminate facial defects such as irregularities, pimples, asymmetries, creases and make the face appear to be clear, younger and smooth, especially with the involvement of baby's face. O'Toole et al. [25, 26] treated the average face as the baseline and the distance between an individual face and the baseline as the distinctiveness. This defined distance is largely correlated with facial aging.

2.2 Aging Representations and Models for Face Images

Figure 2.3: Flowchart of a typical facial age estimation system

The flow chart of a typical facial age estimation system is shown in Fig. 2.3. As can be seen, it mainly consists of three key steps, namely pre-processing, feature extraction with dimension reduction and estimation using certain algorithms. Particularly, pre-processing is a series of basic and straightforward image processing methods, which will not be introduced in this thesis. Next, the aging representations and models for face images will be detailed.

2.2.1 Anthropometric Model

The origination of this model comes from the craniofacial research based on mathematical model which is called the cardioidal strain transformation [27]. This mathematical model is used for simulating the head's growing process from infancy to adulthood. In detail,

$$\alpha' = \alpha \tag{2.1}$$

$$r' = [1 + m * (1 - \cos\alpha)] * r \tag{2.2}$$

where r is the circle radius, m is an increasing parameter with time increasing, α is the angle from the axis. So the combination of (r', α') demonstrates the growing head along with time flowing.

Later a more integrated explanation of face anthropometry was published by L. Farkas [28], which was based on different facial size and proportion measurements collected from a certain number of feature facial points from childhood to adulthood. But later researchers gradually gave up the pure mathematical model and began to focus on various ratios of distance between facial feature points (landmarks). Kwon and Lobo [5, 29] firstly published the anthropometry-related paper in the application of facial age classification.

Figure 2.4: The six distance ratios of facial landmarks proposed by Kwon and Lobo [5]

In the aging representation, they selected six distance ratios of facial landmarks (shown in Fig. 2.4), which could be treated as the judging criteria to distinguish children from adults. Then after several years, Ramanathan and Chellappa [1] calculated eight distance ratios between specified facial feature points for the age modeling in young age (before adulthood), as shown in Fig. 2.5.

Because of the human's physiological character that the craniofacial changes mainly happen from infancy to adulthood rather than adult aging (e.g., from 20 to 80), this anthropometric model has irreparable deficiency of estimating through the whole age levels. In other words, this model can merely be utilized in the scenarios where only young ages are involved or where the separation between young ages and the elderly is needed. For example, Kwon and Lobo [5, 29] realized the distinction of young teenagers from old adults by calculating the crinkles from different parts of the face, such as near the eyes, on the forehead and so on. The crinkles were located by curve detection in the respective part of the facial images.

Another problem for the anthropometric model is that it only takes the face shape into account and neglects the skin aging process which is predominant during adult aging. In other words, the craniofacial change is treated as the sole criterion in estimating the age while ignoring any facial texture information. So to the best of our knowledge, till now, no successful results using the anthropometric model are reported with age estimation through all age levels on a large database.

Figure 2.5: The eight distance ratios of facial feature points by Ramanathan and Chellappa [1]

2.2.2 Active Appearance Models (AAM)

The Active Appearance Models (AAM) was proposed by T. Cootes et al. [30] initially for formulating a model which can interpret facial images. To be specific, the model was designed for robustly describing a face's pose, identity, expression and so on. Actually the AAM model is based on the Active Shape Model (ASM) [31], which is a fast approach applying the principal component analysis to get the point distribution model. But the ASM has the disadvantages that it cannot incorporate all the gray-scale information of facial images and it is not robust when introducing new images. However, the AAM overcomes such defects. Basically, it matches an appearance statistical model containing both shape and texture information to a new facial image, which is suitable for the representation of face aging: the shape corresponds to the facial contour and texture corresponds to the skin changing. Consequently the AAM model is widely used in the facial aging problem. Later Lanitis et al. [32] proposed a particular aging function, $\mathbf{b} = \mathbf{f}^{-1}(age)$, in which *age* stands for people's physical age in the facial image and $\mathbf{b}$ stands for the vector of parameters which contain the learned AAM information from the face image. So this aging function actually maps between numerical values of the age and the corresponding parametric representations of facial images.

Some researchers [33] made comparisons between the AAM and anthropometric model based methods [29][5][1] and the results clearly show that the AAM outcompetes the anthropometric model methods for the following reasons: 1) the AAM model can estimate through all age levels rather than only young ages; 2) in the AAM model, both face shape and texture are considered as the representations rather than the mere information of facial geometry in the anthropometry based approaches.

2.2.3 AGing pattErn Subspace (AGES)

Geng et al. proposed the AGing pattErn Subspace (AGES) [6, 34], which can deal with the lack of aging databases with sufficient images and continuous ages. AGES builds an

Figure 2.6: The vectorization and schematic diagram of the aging pattern in AGES [6]. “m” stands for the missing facial images in the corresponding age.

aging pattern which includes a chronological-based facial images of one person. Naturally, since the aging databases only have a few facial images for the same person at different ages, this so-called “aging pattern” is highly incomplete. Just as Fig. 2.6 shows, for example, the aging pattern contains the age of 0-8, but only three facial images (with the age of 2, 5 and 8 respectively) in the database are available. In this case, AGES utilizes an EM-like approach to learn for the missing part of the representative subspace.

During the training process, firstly it adopts the principal component analysis to represent the aging subspace. When encountering the condition of insufficiency and discontinuity, the aforementioned EM-like learning approach will be conducted iteratively in order to minimize the error of reconstruction originated from the difference between the missing part and the available part. The terminal for this iteration is that the reconstruction error reaches the value which is small enough. Next for the stage of estimation, an appropriate corresponding aging pattern and related “place” (order) of that aging pattern will be matched to the newly-introduced test image. Specifically, the new test image will be traversed for each age label of the aging pattern and the age label with minimal reconstruction error is chosen. Then this age label is the estimated age.

Experimental results demonstrate that AGES got the Mean Absolute Error (MAE) of 6.77 years [6]. However one problem might exist that AGES requires the database containing multiple facial images of the same person with different ages, but there are some aging databases which contains only one facial image for a person, in which AGES cannot work. What's more, it is not an easy task to build a large aging database with multiple facial images for the same individual in different ages.

2.2.4 Age Manifold

Unlike the above-mentioned face aging representation with the pattern of aging subspace for each person [6][34], the age manifold [7] is a universal representation which can be commonly used for a large population. It has a broad prerequisite that the images of different individuals for every age label are allowed, so no further limitations are set to require multiple different ages for each person, which shows better flexibility in database collection and algorithm applications. In extreme cases, the age manifold can permit only one facial image at a single age for one person. In essence, the age manifold utilizes the manifold learning approaches to obtain the low-dimensional manifold of aging representations in the embedded subspace. Suppose the face image database is $A = \{\mathbf{x}_i\} \in \mathbb{R}^D, i = 1, 2, \dots, n$ with m persons in chronological order and the corresponding age label set (ground truth) is $G = \{g_i\}, i = 1, 2, \dots, n$. Next step is to obtain a low-dimensional manifold Γ in the A 's embedded subspace with the representation of $\Omega = \{\mathbf{y}_i\} \in \mathbb{R}^d, i = 1, 2, \dots, n$. According to the concept of manifold learning, $d \ll D$. In fact, after the step of manifold learning, a one-to-one mapping is made between the manifold space and the original image space. That is

$$\text{Original image space} \Rightarrow \text{Manifold subspace} : \Omega \stackrel{\Gamma}{=} Q(A, G) \quad (2.3)$$

Figure 2.7: Performance comparison of different linear manifold algorithms for facial age estimation [7]

where $Q(\cdot)$ is the project function. Generally, there are two ways of projecting the image information into the manifold space, linear or nonlinear. In order to avoid the relatively low generalization performance of nonlinear manifold approaches, the author [7] adopted linear methods: 1) Principal Component Analysis (PCA), 2) Locality Preserving Projections (LPP) [35], 3) Neighborhood Preserving Projections (NPP) [36], 4) Orthogonal Locality Preserving Projections (OLPP) [37]. Fig. 2.7 [7] shows the comparison of different manifold learning approaches in age estimation. The age data points from 93 to 0 are marked from red to blue. As can be seen, the PCA manifold shows no obvious manifold tendency since it is unsupervised. NPP achieves similar (no better) results to PCA. LLE yields to a ellipsoid-like shape with lower ages at periphery and high ages at the center. Both OLPP and LPP demonstrate a clearer manifold trend and thus out-compete other methods, which indicates that OLPP and LPP are more suitable for facial age estimation. Guo et al. [18][38] further proposed the Locally Adjusted Robust Regression (LARR) algorithm, which can get a better estimation performance and smaller MAE of 5.07 years on the public FG-NET database when combining the OLPP-based age manifold representation.

In [39], Chen et al. assumed that age ranking can be described by the low-dimensional-subspace ranking features. Driven by this assumption, they formulated facial age estimation problem using a subspace learning approach, in which the subspace jointly embeds the aging manifold and the implicit age ranks. Moreover, they realized an semi-supervised age estimation so that the unlabeled data's age rank can be approximated. Their reporting MAE is 4.56 years.

2.2.5 Appearance-based Models

Local Binary Patterns (LBP) [40] is a popular feature which is widely used for its capability of extracting texture information on faces. Later it is used in the application of automatic age classification [8], which roughly classifies an individual into the corresponding class with a 10-year-interval. In respect of realization, firstly the face images

10	%83,29	%84,24	%70,14	%68,96	%69,89	%65,99
20	%81,66	%88,5	%66,34	%65,38	%68,95	%63,86
30	%78,90	%76,58	%87,76	%88,87	%75,23	%75,78
40	%77,78	%74,33	%87,65	%89,01	%86,22	%89,22
50	%75,67	%76,32	%78,54	%87,75	%87,69	%89,69
60	%77,20	%73,34	%72,98	%75,42	%87,23	%91,32

Figure 2.8: Classification results of age classification using LBP [8]

are segmented into several subregions for LBP feature extractions, which would be later connected for a face vector as the face descriptor. When new face image is introduced, the system can extract the LBP histograms and use it for classification. The experiments are done with 6 age classes (from 10 to 60 with a 10-year-interval) using FERET image database [41] and the author's own database. The results are shown in Fig. 2.8. As reported, this method achieves the accuracy of 80%. Also, similar research was done by Yang and Ai with AdaBoost [42]. They used the FERET and CMU PIE database [43] as their training datasets and obtained the accuracy of 80% - 90% in age classification. It is worth mentioning that they also use their algorithms in two other applications with satisfying results: gender classification and ethnicity classification. Unfortunately, the above two algorithms only realize the "coarse" age estimation within a certain age range or even quite general description like "children", "oldness" and "adult" and cannot estimate an individual's exact age. Later, Gao and Ai [44] used the Gabor feature and fuzzy LDA method in face age classification. Their experiments show that Gabor feature outperforms LBP feature in the task of age estimation.

Since human can get age information apparently from the facial local evidence, Yan et al. [45] proposed Spatially Flexible Patch (SFP) as a new face feature descriptor. This feature descriptor includes the information of both position information and local appearance on the human face; also to a certain extent, it solves the issue of sample insufficiency. Modeled with Gaussian Mixture Model (GMM) and tested on the YAMAHA

aging database, their approach achieves the average MAE of 8.53 and 7.82 for females and males respectively.

Suo et al. [46] designed a facial representation model which contains a hierarchical graphical face distribution, namely high, middle and low resolution respectively. In this way, detailed parts of the face can be obtained from coarse to fine. What's more, the information of geometry, topology, photometry and configuration are embedded in extracted features with specific filters to different parameters. Their model is tested with different regression methods: Age group specific Linear Regression (ALR), Multi-Layer Perceptron (MLP), Support Vector Regression (SVR) and Logistic Regression. Finally they get the optimal MAE of 4.6754 on their own database and 5.974 on the FG-NET aging database respectively with MLP.

Guo et al. [47] then used the Biologically Inspired Features (BIF) [48][49] for facial age estimation. Basically the BIF feature is based on a series of Gabor filters and Guo et al. [47] revised it to use smaller-size Gabor filters and proposed a new operator for encoding the face subtlety for aging. Also, their approach can achieve robustness as well as the capability of age estimation for cross-race population. Basically it is a feedforward model (called "HMAX"), which is the primate visual object recognition pathway. To be more specific, it is consisted of alternating layers, namely complex (C) and simple (S) cell units with incremental complexity as layers progress from the primary visual cortex (V1) to inferior temporal cortex (IT) [47][49]. Then the first layer S1 is built by convolving a series of Gabor filters on the input facial image. The second layer C1 is created by choosing the max values on S1, which can have the advantage of robustness against scales changes and small shifts. Their experimental results demonstrate that the BIF feature with appropriate linear SVM and SVR methods can achieve the MAE of 4.77 years on the FG-NET database and MAEs of 3.47 and 3.91 years for males and females on the YGA database, respectively, which is an evidence to prove BIF's superiority in facial age estimation.

In [50], the authors extracted the demographic informative features using a boosting-based approach. Firstly, BIF face representation is computed, followed by feature selection which retains the feature discriminability and reduces the dimensionality. Then multi-class AdaBoost and multi-level decision tree are utilized to choose a subset of BIF features as selected demographic informative features. They also found that the quality of facial image would have an impact on demographic estimation and thus proposed a learning-based image quality assessment method. In this method, high-quality and low-quality instances are trained for the ensemble of binary SVM classifiers after resampling.

In [51], Lu et al. realized that the local discriminative features exhibits more robustness against lighting conditions than holistic features. So the local facial features were extracted from raw pixels of face patches into low-dimensional binary codes by learning several hashing functions. These binary codes with far-away age labels are projected far while those with near age labels are projected close. Then the facial feature representation can be constructed by pooling and encoding these binary codes as histogram features.

In [52], the facial features were extracted using the dense scale invariant feature transform (DSIFT). Specifically, DSIFT is obtained by calculating the SIFT descriptors of densely sampled features and these features are in the same orientation and scale. To do that, the image firstly needs to be separated into a series of regions of interests (ROIs) and each ROI can compute the corresponding DSIFT features, which are then combined into a whole feature vector. In this way, some redundant SIFT descriptors can be removed and the feature vector's dimensionality can be reduced.

In [53], the authors believed that the curvature information of a face is significant for the wrinkle as people age. So they extracted the extended curvature Gabor (ECG) features as the descriptor of facial features. What's more, they demonstrated that the ECG kernel is effective for capturing the information of curved wrinkles. In order to avoid high computation complexity, they further utilized a regression-based method for feature selection, by which a small number of discriminative facial features can be extracted.

2.3 Age Estimation Algorithms

After building the aging representations/models, age estimation algorithms are then developed to obtain the estimated age for given images. In fact, age estimation based on facial images is a special pattern recognition problem because of the particularity of human's age. On the one hand, from infancy to the old age, each age label can be regarded as one class. Consequently, age estimation is a classification problem. From the other point of view, age labels are a series of numerical values with the meaning of chronological orders. In this sense, age estimation can be treated as a regression problem. Then starting from these two aspects, some researchers also take advantages of both classification and regression, in which the hybrid method was proposed. In addition, a large portion of researchers develop dedicated techniques based on the characteristics of aging process.

2.3.1 Classification

In [54], Kanno et al. used artificial neural network for age group classification, which contains four classes. This work obtains an accuracy of 80% on the datasets of 110 young male facial images. In [33], Lanitis et al. evaluated the performance of automatic age estimation based on four classifiers, i.e. quadratic functions, shortest distance (nearest neighbor) classifier, supervised neural network (Multilayer Perceptron) and unsupervised neural network (Self-Organizing Map) [55]. They use AAM model and Principal Component Analysis (PCA) to construct the facial image representation. After that, both "single step" and hierarchical age estimators are designed. Particularly in the hierarchical age estimator, they firstly train global age classifier and then three age specific classifiers for different age ranges (i.e. 0-10 years, 11-20 years and 21-35 years) are trained as local age classifiers. In experimental part, a small dataset containing 400 facial images with the age range of 0-35 is used. The author conclude that hierarchical estimators achieve better estimation results compared to single step estimators; besides, for four different classifiers, quadratic function classifier can obtain the MAE of 5.04 (single step), which

is higher than SOM and MLP, but lower than the shortest distance classifier. What's more, a subset of 32 images are shown to 20 observers for testing human's ability in facial age estimation. The average error for these human observers is 3.64 years.

Ueki et al. [56] developed an aging database WIT-DB using the images of Japanese subjects. This database has 11-class age-groups, which are divided based on the actual age. Then they proposed a two-phased method: firstly 2DLDA and LDA are used to reduce the dimensionality as well as extract features; secondly, in the low-dimensional feature space, 11 Gaussian models for the corresponding age-groups are estimated using the EM algorithm. Their results show that for the 5-year range classification, the accuracy rate is about 43 percent for female and 50 percent for male; for the 10-year range, the accuracy rate is about 63 percent for female and 71 percent for male; for the 15-year range, the rate is about 73 percent for female and 80 percent for male. Besides, they put forward an interesting opinion that the classification accuracy for male is higher than that for female; in other words, using female images to classify age-groups tend to be more difficult. In [57], the spectrum data for facial images are firstly extracted by fast Fourier transform (FFT), which will be used in the genetic algorithm (GA) as feature selection. Then the supervised self-organizing map (SOM) utilizes the selected feature data to train and get the estimated age. Finally they get the age error margin of 6.8 years on selected features and 9.96 years on all features; at the same time, human's age perception error is 4.94 years according to their test. As a result, they conclude that 1) the estimation based on fewer data (excluding invalid or even noisy information) tends to be more accurate and 2) it is feasible to get a good estimation result which is almost similar to human's perception ability. Also, in [18], Guo et al. used support vector machine (SVM) as the classifier and they obtain the MAEs of 5.52 for male and 5.55 years for female on the UIUC-IFP-Y Aging Database. For FG-NET aging database, the MAE is 7.16 years.

2.3.2 Regression

Lanitis et al. [32] formulated the age estimation problem as aging functions, i.e., $age = f(\mathbf{b})$ ($\mathbf{b}$ is a parametric vector of 50 raw model dimensions), which project the facial images into a multi-dimensional model parameter space. They have explored three types of mathematical formulation for age functions, namely linear, quadratic and cubic. For each type, a genetic algorithm (GA) is used to calculate the corresponding weights and offsets. Zhou et al. [58] proposed the image based regression (IBR) for multiple-output regression; also the boosting method is employed to select features from redundant Haar-like features. The authors show that IBR can obtain 5.81 for MAE on the FG-NET database; at the same time, IBR can maintain a high running speed, which means its computational cost is quite low. Yan et al. [17] solved the age estimation problem based on nonlinear regression, which uses the training samples with uncertain nonnegative labels. The regression is formulated by solving a semidefinite programming (SDP) problem. The MAE they obtain is 5.78 years on average for the FG-NET database; for the Yamaha database, they get the MAE of 9.79 years and 10.36 years for female and male, respectively. The authors also demonstrate that their SDP-based algorithm is superior to the traditional quadratic regression and Multilayer Perceptron (MLP). However, SDP suffers from high computational cost; in other words, especially when the training dataset is huge, the training speed will be slow or even it is difficult to get the SDP solution. Fu et al. [7, 14] utilized manifold learning algorithms to learn a low-dimensional representation from the initial facial image space by Orthogonal Locality Preserving Projections (OLPP) and Conformal Embedding Analysis (CEA), which will be used as the extracted aging features. Then multiple linear regression including linear, quadratic and cubic is trained to estimate age values. The MAEs are about eight years and five to six years for OLPP and CEA, respectively. Furthermore, the author has found that quadratic function is more appropriate to be the regression function compared to the linear and

cubic functions, because the linear function might lead to under-fitting problem and cubic or higher order functions may cause over-fitting problem. Besides, for the methods of manifold embedding, they conclude that OLPP and CEA show their superiority because they can discriminate aging patterns better against dimension changes and error level compared to Principal Component Analysis (PCA) and Neighborhood Preserving Projections (NPP) so that they have a more unified and stable distribution. In [16], a bilinear ranking model was proposed for age estimation, which combine the bilinear regression and kernel trick. During this process, Expectation Maximization (EM) algorithm is used to obtain relevant parameters. The reported MAE for FG-NET is 5.33 years; for Yamaha database, the MAEs are 6.95 and 6.95 for female and male, respectively. [59] further improved the estimation performance by firstly encoding facial images as ensemble of coordinate patches, which are modeled by Gaussian Mixture Models (GMM); and then using patch-kernel for regression to measure the final age. In [46], Suo et al. compared the age estimation performances of different regression methods, including Logistic Regression (multi-class AdaBoost [60]), Multilayer Perceptron (MLP), Support Vector Regression (SVR) and Age group specific Linear Regression (ALR). The evaluation is performed on the FG-NET and their own database, respectively. Experimental results show that MLP achieves the best performance (with the MAEs of 5.974 and 4.6754 years on FG-NET and their DB, respectively). In [61], Ni et al. constructed an automatic and universal age estimator from web facial images. Firstly the facial images from websites along with age-related queries are obtained and preprocessed, including robust face detection, outlier removal with PCA and so on. Then robust multi-instance regression (RMIR) is proposed to train a universal age estimator on a huge self-collected database. During the testing phase, their method is evaluated on three databases and get the MAEs of 9.49, 7.42 and 8.60 on FG-NET, MORPH-1 and MORPH-2 databases, respectively, which is quite acceptable and robust in purely “wild” environment with various noises. [62] employed metric learning by means of semidefinite programming and

integrate with conventional regression method (such as k nearest neighbors (kNN)) to improve the age estimation performance to a large extent. Also, through a simple linear transformation, the low-level feature space is mapped into the high-level semantic space. As a result, the underlying semantic data structure can be found by the learned metric. [63] utilized the kernel partial least squares (KPLS) regression method for facial age estimation, which can simultaneously realize the reduction of feature dimension and learn the regression function. Such characteristics make this framework easier for real-time applications. This method can also be used to solve gender classification and ethnicity estimation problems. [39] adopted SVR to learn an aging function with their ranking features (relative age-related features) so that the estimated age can be obtained. In [53], Kim et al. used a random forest regression approach as the age estimator, which belongs to ensemble learning and is robust to outliers. Their method includes the idea of bootstrap aggregating and random feature selection so that the generalization error can be minimized. This method has high efficiency for huge feature data.

2.3.3 Hybrid of Classification and Regression

In the above analysis, age estimation can be formulated as classification or regression problem. Then naturally, a straightforward question comes: which one is superior? In [18], Guo et al. gave the answer. They choose the corresponding representatives for classifiers and regressors (i.e., SVM and SVR, respectively), which are tested on same databases. Then an interesting phenomenon happens: for the FG-NET database, SVR gets the MAE of 5.16 years, which is much better than SVM's 7.16 years; on the contrary, on the UIUC-IFP-Y aging database, SVR achieves worse results than SVM (7.47 versus 5.52 for male and 7.00 versus 5.55 for female). Concluding from the above results, the performance of classification and regression is dependent on specific facial image data.

Another opinion is to formulate age estimation using both regression and classification, fully combining both advantages. Driven by this idea, Guo et al. [18, 38] proposed

a locally adjusted robust regressor (LARR) to learn and predict human's ages. Firstly, in the initial stage SVR is used to train a robust regressor through all ages; then for the regression result, a local adjustment based on classification is conducted within bounded ranges, which uses a binary-tree-structured search for pairwise trained linear SVMs. Experimental results prove the validity of this algorithm: LARR obtains the MAEs of 5.25 and 5.30 for female and male of UIUC-IFP-Y aging database, respectively; for the FG-NET database, the MAE of LARR reaches 5.07 years. However, LARR still suffers from its defect: the local search range cannot be automatically determined and it has to be heuristically selected by human beings. For example, the authors try the local search ranges as a series of powers of two (4, 8, 16, 32, 64) and manually select the one with best performance instead of a data-driven way. Dealing with this issue, Guo et al. [64] proposed a sequential fusion method called PFA based on Bayesian decision rule. They firstly train a SVR as the regressor, which is then treated as an intermediate step for feeding a SVM (as the classifier). Finally the probabilistic outputs for SVM and SVR are fused by multiplication. By this means, automatic fusion of classification and regression (without relevant parameters' manual selection) can be realized; furthermore, the estimation performance is further improved to some extent: PFA achieves the MAEs of 5.11 and 5.12 for female and male in the UIUC-IFP-Y database, respectively; the MAE for the FG-NET database is 4.97 years.

In [65], the authors proposed a hierarchical framework for age estimation: firstly all facial images are segmented into several groups. Secondly, for each group, a separate Relevance Vector Machine (RVM) regression [66] is learned. Thirdly, multiple classifiers are used to categorize test facial images into one age group and the corresponding regressor is utilized to obtain the estimated ages. The MAE for this method is 6.2 years.

Han et al. [50] proposed a generic hierarchical architecture of between-group classification and then within-group regression. For the classification, a two-level binary decision tree are constructed after three binary SVM classifiers are trained. Then for

each group, four SVRs are separately trained to obtain an exact estimated age. What's worth mentioning is that the regression models are trained with overlapping age ranges so as to extenuate the error caused by the group misclassification of facial images that are near the group borders. Furthermore, this generic framework can also be applied to gender and race estimation.

In [67], the authors proposed a multi-stage age estimation framework called grouping estimation fusion (GEF), which consists of age grouping, age estimation within groups and decision fusion for final estimated age. In age grouping, the gradient orientation pyramid (GOP) is used to extract facial features and SVM is for the classification into age groups. For the second phase, within the group range, both global and local (mouth, eyes and nose) features are obtained by extracting LBP, HOG and BIF feature descriptors. Then SVR uses these combined features for predicting the exact age in the corresponding group. For the third phase, six efficient decision-fusion strategies are conducted for the output (estimated ages) of second phase.

2.3.4 Ordinal Ranking Based Methods

As is stated above, basically most of the traditional age estimation approaches formulate facial age estimation problem based on classification, regression or combination of the two. Suppose a dataset of N training samples is given, $\{(x_i, y_i) | i = 1, \dots, N\}$, in which x_i represents the i th face image and y_i represents the corresponding age label. In multi-class classification, every sample will be regarded as a single independent age label for training; as a result, a multi-classifier is obtained to estimate a person's age. But the trouble is, the age labels have no relationship with each other, i.e., each age label is only treated as a separate entity in the training process while in essence, human's age labels are sequential. So this kind of multi-classification method may omit some connotative information of the correlation among different age labels, which together compose the fine-ordered age set. For instance, two images with adjacent age labels for the same person will be more

similar than those with far-apart labels. In short, multi-class classification cannot take full advantage of the correlation among ordinal age labels. In contrast, the regression method aims to find the best mapping from raw images to the corresponding ages and get a function for age estimation. However, craniofacial and skin changes in different age levels would result in unstable random process in feature space, so the kernels used to assess the similarities among different ages could be drifted. In order to overcome the aforementioned defects of classification and regression approaches, ordinal hyperplanes ranker (OHRank) [68] based age estimation was proposed. As everyone knows, the aging process is diversified for different age levels. As an analogy, the aging process from 22 to 25 would have a different tendency compared to that from 62 to 65. So it is more credible to compare two age labels relative sequence (smaller or larger) than the differences among labels. In spite of all the above merits, OHRank only utilizes a single feature set as the feature representation model, so it fails to synthetically include all discriminative information from all available feature sets. More specifically, each feature set has its own advantages over others. For example, the anthropometry models [5] mathematically model the growing of people's head from babyhood to adulthood, so it reveals some information of the face's size and proportion; Active Appearance Models (AAM) [30] can represent both shape and texture information instead of only facial geometry; age manifold [7] learn the age tendency from different face images at every age label. So OHRank cannot integrate these discriminative information of different feature sets. Later Weng et al. [69] presented a multi-feature ordinal ranking (MFOR) method to utilize multiple feature sets simultaneously, so the feature information's discriminative power is further reinforced. Experiment results demonstrate that the performance is further improved. [70] learned the ordinal discriminant features, which retains both the geometrical information and ordinal information. Then it employed OHRank method to convert the whole age estimation problem into several binary classification subproblems. Also for the purpose of involving the information of unlabeled facial images into the

learning procedure, the authors extended their method to semi-supervised application by adding a regularization penalty parameter for each binary sub-classification. Li et al. [71] further proposed an ordinal distance metric learning approach to preserve both the data's ordinal relationship and the local geometry information for different facial images. In [51], Lu et al. proposed a cost-sensitive local binary feature learning (CS-LBFL) scheme, in which two kinds of local discriminative facial features are learned. Then the ordinal hyperplanes ranker is used as the age estimator. The authors further compared the performance of CS-LBFL using OHRank and the traditional regressor SVR; the results proved the superiority of ordinal ranking. [52] proposed a relative attribute SVM+ (raSVM+) classifier in each binary classification subproblem of ordinal hyperplanes ranking. raSVM+ take advantages of relative attributes as special-treated data and controls the outliers for the training sets. So it can minimize the possibility of producing classification errors. [72] proposed a cost-sensitive label ranking method for facial age estimation, in which the age labels are ranked in descending order according the predicted relevance to the given face. Then the matrix trace norm regularization is performed for an aggregation matrix of the linear prediction functions for different ages. Moreover, it was extended by the kernel method to get a better nonlinear generalization performance.

Chapter 3

Ordinal Ranking Based Approaches for Facial Age Estimation

3.1 Multi-Feature Extreme Ordinal Ranking Machine (MFEORM) for Facial Age Estimation

3.1.1 Introduction

As is stated in Chapter 2, a lot of facial age estimation algorithms have been put forward, some of which are able to obtain rather satisfying performance. However, most of these approaches largely rely on the complicated mathematical optimization solution. Take OHRank and MFOR as an example: these two ordinal-ranking-oriented methods achieve top performance so far but they are all constructed under SVM-based formulation, so the SVM parameters must be computed iteratively by working out complicated and time-consuming optimization problems. As a result, calculation complexity would be a heavy burden for improving efficiency.

Recognizing this point, a multi-feature extreme ordinal ranking machine (MFEORM) is proposed for facial age estimation. Basically, the proposed approaches can be divided into three stages: 1) representing features using certain feature extraction models, 2) processing the obtained feature sets and 3) applying certain algorithms to estimate age. On the first stage, multiple feature models are used parallelly to represent our facial image database. For the second stage, since it is more logical and reasonable to distinguish which

is older / younger between two facial images than to directly predict the age from images, the more reliable “larger or smaller than” information is used for one binary classification at each age. In this case, the abstract age estimation problem can be downgraded into K-1 binary classification subproblems where K represents the number of total age labels. For the third stage, an ultrafast Extreme Learning Machine (ELM) with kernel function is applied to get a series of classifiers. These classifiers are then integrated together according to a certain rule which will be illustrated in the following. Experiment results explicitly demonstrate that MFEORM is able to notably reduce the runtime (even up to nearly one hundred times faster) while achieving similar or even better estimation results against state-of-the-art methods.

The rest of Sec. 3.1 is organized as follows: firstly, the regression theory and the extreme learning machine will be briefed respectively in Sec. 3.1.2 and 3.1.3. Then the proposed method will be detailed in Sec. 3.1.4. After that, experiment results and remarks are reported in Sec. 3.1.5. Finally, concluding remarks are shown in Sec. 3.1.6.

3.1.2 Background on Regression Theory and Kernel Ridge Regression

Firstly, consider a generic regression problem for linear model. Given an instance $\mathbf{x} = (x_1; x_2; \dots; x_d)$ with d attributes, where x_i is the value in the i^{th} attribute. The linear model tries to learn a prediction function with the linear combination of the attributes, namely

$$f(\mathbf{x}) = w_1x_1 + w_2x_2 + \dots + w_dx_d + b \quad (3.1)$$

In vector form, it can be written as

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b \quad (3.2)$$

in which $\mathbf{w} = (w_1; w_2; \dots; w_d)$.

Given the datasets $D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_m, y_m)\}$, in which $\mathbf{x}_i = (x_{i1}; x_{i2}; \dots; x_{id})$ and $y_i \in R$. The multivariate linear regression is to learn $f(\mathbf{x}_i) = \mathbf{w}^T \mathbf{x} + b$ which satisfies $f(\mathbf{x}_i) \simeq y_i$. To facilitate the discussion, we have the following representations: $\hat{\mathbf{w}} =$

$$(\mathbf{w}; b) = (w_1, w_2, \dots, w_d, b)^T, \mathbf{y} = (y_1; y_2; \dots; y_m) = (y_1, y_2, \dots, y_m)^T, \mathbf{X} = \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_m^T \end{bmatrix} \text{ and}$$

$$\hat{\mathbf{X}} = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1d} & 1 \\ x_{21} & x_{22} & \dots & x_{2d} & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{m1} & x_{m2} & \dots & x_{md} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1^T & 1 \\ \mathbf{x}_2^T & 1 \\ \vdots & \vdots \\ \mathbf{x}_m^T & 1 \end{bmatrix}. \text{ Then the problem can be formulated}$$

using the ordinary least square method as

$$\min_{\hat{\mathbf{w}}} (\mathbf{y} - \hat{\mathbf{X}}\hat{\mathbf{w}})^T (\mathbf{y} - \hat{\mathbf{X}}\hat{\mathbf{w}}) \quad (3.3)$$

Let $E_{\hat{\mathbf{w}}} = (\mathbf{y} - \hat{\mathbf{X}}\hat{\mathbf{w}})^T (\mathbf{y} - \hat{\mathbf{X}}\hat{\mathbf{w}})$. Taking the derivative of $E_{\hat{\mathbf{w}}}$ with respect to $\hat{\mathbf{w}}$ and let it equal 0, we can get the closed-form solution

$$\hat{\mathbf{w}}^* = (\hat{\mathbf{X}}^T \hat{\mathbf{X}})^{-1} \hat{\mathbf{X}}^T \mathbf{y} \quad (3.4)$$

However, in applications, sometimes $\hat{\mathbf{X}}^T \hat{\mathbf{X}}$ is not a full-rank matrix or strong correlation among column vectors exists. In this case, $\hat{\mathbf{X}}^T \hat{\mathbf{X}}$ is a singular matrix or close to a singular matrix, which leads to the ill-posed problem and the solution's instability.

To solve this issue and transform the ill-posed problem to well-posed one, the L2-norm regularization term is introduced and added to the loss function, which is called ridge regression [73]. By doing this, the estimation is more numerical stable and not susceptible to the independent variables' collinearity and data's small changes. Thus, it can achieve higher accuracy. Mathematically, the original formulation (loss function) of the ridge regression problem is: $\min_{\hat{\mathbf{w}}} C(\mathbf{y} - \hat{\mathbf{X}}\hat{\mathbf{w}})^T (\mathbf{y} - \hat{\mathbf{X}}\hat{\mathbf{w}}) + \mathbf{w}^T \mathbf{w}$. Note that when including an intercept term b in the regression, we usually leave this coefficient unpenalized. Otherwise we could add some constant amount c to the vector $\mathbf{y}$, and this would not result in the same solution. So here the regularization term should be $\mathbf{w}^T \mathbf{w}$ instead of $\hat{\mathbf{w}}^T \hat{\mathbf{w}}$. Also

notice that if we do the centralization for $\mathbf{X}$, then the intercept estimate ends up just being $\hat{b} = \bar{y}$. So for convenience, we usually assume that $\mathbf{y}$ and $\mathbf{X}$ have been centralized and thus the formulation of ridge regression doesn't include the intercept. That is

$$\min_{\mathbf{w}} C(\mathbf{y} - \mathbf{X}\mathbf{w})^T(\mathbf{y} - \mathbf{X}\mathbf{w}) + \mathbf{w}^T\mathbf{w} \quad (3.5)$$

where $\mathbf{w}^T\mathbf{w}$ is the L2 regularization and C is a user-defined parameter which controls the size of the coefficients and amount of regularization. Let $E_{\mathbf{w}} = C(\mathbf{y} - \mathbf{X}\mathbf{w})^T(\mathbf{y} - \mathbf{X}\mathbf{w}) + \mathbf{w}^T\mathbf{w}$. Taking the derivative of $E_{\mathbf{w}}$ with respect to $\mathbf{w}$, we can obtain

$$\frac{\partial E_{\mathbf{w}}}{\partial \mathbf{w}} = -2C\mathbf{X}^T\mathbf{y} + 2C\mathbf{X}^T\mathbf{X}\mathbf{w} + 2\mathbf{w} = 2C(\mathbf{X}^T\mathbf{X} + \frac{\mathbf{I}}{C})\mathbf{w} - 2C\mathbf{X}^T\mathbf{y} \quad (3.6)$$

Let $\frac{\partial E_{\mathbf{w}}}{\partial \mathbf{w}} = 0$, we can get

$$\mathbf{w} = (\mathbf{X}^T\mathbf{X} + \frac{\mathbf{I}}{C})^{-1}\mathbf{X}^T\mathbf{y} \quad (3.7)$$

Notice that

$$(\mathbf{X}^T\mathbf{X} + \frac{\mathbf{I}}{C})\mathbf{X}^T = \mathbf{X}^T\mathbf{X}\mathbf{X}^T + \frac{\mathbf{I}}{C}\mathbf{X}^T = \mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \frac{\mathbf{I}}{C}) \quad (3.8)$$

Multiplying each part of the above equation by $(\mathbf{X}^T\mathbf{X} + \frac{\mathbf{I}}{C})^{-1}\mathbf{y}$ at the right and $(\mathbf{X}^T\mathbf{X} + \frac{\mathbf{I}}{C})^{-1}$ at the left, we have

$$\mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \frac{\mathbf{I}}{C})^{-1}\mathbf{y} = (\mathbf{X}^T\mathbf{X} + \frac{\mathbf{I}}{C})^{-1}\mathbf{X}^T\mathbf{y} \quad (3.9)$$

By substituting Equ. 3.7 into Equ. 3.9, we can get

$$\mathbf{w} = \mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \frac{\mathbf{I}}{C})^{-1}\mathbf{y} \quad (3.10)$$

For a new instance $(\mathbf{x}', y')$, its prediction is

$$\hat{y}' = \mathbf{x}'^T\mathbf{w} = \mathbf{x}'^T\mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \frac{\mathbf{I}}{C})^{-1}\mathbf{y} \quad (3.11)$$

Define

$$K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T\mathbf{x}_j \quad (3.12)$$

Then Equ. 3.11 can be represented as

$$\hat{y}' = \mathbf{x}'^T \mathbf{w} = \begin{bmatrix} K(\mathbf{x}', \mathbf{x}_1) \\ K(\mathbf{x}', \mathbf{x}_2) \\ \vdots \\ K(\mathbf{x}', \mathbf{x}_m) \end{bmatrix}^T (\mathbf{K} + \frac{\mathbf{I}}{C})^{-1} \mathbf{y} \quad (3.13)$$

where $\mathbf{K} = \begin{bmatrix} K(\mathbf{x}_1, \mathbf{x}_1) & K(\mathbf{x}_1, \mathbf{x}_2) & \cdots & K(\mathbf{x}_1, \mathbf{x}_m) \\ K(\mathbf{x}_2, \mathbf{x}_1) & K(\mathbf{x}_2, \mathbf{x}_2) & \cdots & K(\mathbf{x}_2, \mathbf{x}_m) \\ \vdots & \vdots & \ddots & \vdots \\ K(\mathbf{x}_m, \mathbf{x}_1) & K(\mathbf{x}_m, \mathbf{x}_2) & \cdots & K(\mathbf{x}_m, \mathbf{x}_m) \end{bmatrix}$. Furthermore, we can use the

kernel method to map the input into a higher dimensional kernel space, in which our data can be linearly separable and thus the performance can be improved. So a mapping $\mathbf{x} \rightarrow h(\mathbf{x})$ can be defined from the input space to a high dimensional space. Consequently Equ. 3.12 can be generalized as

$$K(\mathbf{x}_i, \mathbf{x}_j) = h(\mathbf{x}_i)^T h(\mathbf{x}_j) \quad (3.14)$$

To sum up, based on Equ. 3.12 and 3.14, kernel ridge regression (KRR) [73] can be summarized as

$$\hat{y} = \begin{bmatrix} K(\mathbf{x}, \mathbf{x}_1) \\ K(\mathbf{x}, \mathbf{x}_2) \\ \vdots \\ K(\mathbf{x}, \mathbf{x}_m) \end{bmatrix}^T (\mathbf{K} + \frac{\mathbf{I}}{C})^{-1} \mathbf{y} \quad (3.15)$$

with $K(\mathbf{x}_i, \mathbf{x}_j) = h(\mathbf{x}_i)^T h(\mathbf{x}_j)$ and $\mathbf{K} = \begin{bmatrix} K(\mathbf{x}_1, \mathbf{x}_1) & K(\mathbf{x}_1, \mathbf{x}_2) & \cdots & K(\mathbf{x}_1, \mathbf{x}_m) \\ K(\mathbf{x}_2, \mathbf{x}_1) & K(\mathbf{x}_2, \mathbf{x}_2) & \cdots & K(\mathbf{x}_2, \mathbf{x}_m) \\ \vdots & \vdots & \ddots & \vdots \\ K(\mathbf{x}_m, \mathbf{x}_1) & K(\mathbf{x}_m, \mathbf{x}_2) & \cdots & K(\mathbf{x}_m, \mathbf{x}_m) \end{bmatrix}$.

For example, if RBF kernel is used, then $K(\mathbf{a}, \mathbf{b}) = \exp(-\gamma \|\mathbf{a} - \mathbf{b}\|^2)$.

3.1.3 Randomized Single Hidden-layer Feedforward Neural Networks

Conventional learning techniques like support vector machines (SVMs) and neural networks have been suffering from: 1) slow training and learning speed, 2) human involvement and 3) unsatisfactory generalization performance. However in recent years, randomized single hidden-layer feedforward neural networks [74][75][73][76], which draws

Figure 3.1: A typical example of single-layer feedforward neural network(SLFN)

more and more attention, conquers these drawbacks and gets a satisfying performance to a certain extent. Among these, extreme learning machine (ELM) [77][78][79] is used in this thesis. Primarily, ELM is based on generalized single-hidden layer feedforward neural networks (SLFNs). ELM has the following advantages:

- 1) In ELM, all hidden layer parameters of SLFNs don't need to be tuned and do not rely on training samples. They only need to be randomly generated and reduce human intervention.
- 2) ELM has much faster speed and more superior generalization performance.

Let's start from the structure of SLFNs. Figure 3.1 shows a typical SLFN's construction. Generally speaking, a SLFN can be described as

$$f(\mathbf{x}) = \sum_{i=1}^L \mu_i p_i(\mathbf{x}) = \sum_{i=1}^L \mu_i P(\mathbf{a}_i, \mathbf{b}_i, \mathbf{x}), \mu_i \in \mathbf{R}, \mathbf{x} \in \mathbf{R}^w \quad (3.16)$$

where μ_i represents the output weight connecting the i^{th} neuron and the output nodes, $p_i(\mathbf{x})$ is the output function $P(\mathbf{a}_i, b_i, \mathbf{x})$ of the i^{th} neuron, $\mathbf{a}_i$ is the weight vector linking the i^{th} node and the input nodes, b_i is the bias of the i^{th} node. Specially, we have

$$p_i(\mathbf{x}) = P(\mathbf{a}_i, b_i, \mathbf{x}) = p(\mathbf{a}_i \cdot \mathbf{x} + b_i) \text{ for additive nodes} \quad (3.17)$$

$$p_i(\mathbf{x}) = P(\mathbf{a}_i, b_i, \mathbf{x}) = p(b_i \|\mathbf{x} - \mathbf{a}_i\|) \text{ for RBF nodes} \quad (3.18)$$

Many researchers have built the theoretical foundation [80][81][82] that an SLFN is able to learn N arbitrary distinguishing samples with zero error provided this SLFN has any bounded non-linear activation function and N hidden neurons at most. More precisely, suppose the training samples $(\mathbf{x}_i, y_i)$ are given, where $\mathbf{x}_i \in \mathbf{R}^w, y_i \in \mathbf{R}$. Also, let

$$\hat{y}_j = \sum_{i=1}^Q \mu_i p_i(\mathbf{x}_j) = \sum_{i=1}^Q \mu_i P(\mathbf{a}_i, b_i, \mathbf{x}_j), \quad j = 1, \dots, N \quad (3.19)$$

Then the above mentioned “zero error” means

$$\sum_{j=1}^Q \|\hat{y}_j - y_j\| = 0 \quad (3.20)$$

In other words, a combination of μ_i and $(\mathbf{a}_i, b_i)$ can be found such that

$$\sum_{i=1}^Q \mu_i P(\mathbf{a}_i, b_i, \mathbf{x}_j) = y_j, \quad j = 1, \dots, N \quad (3.21)$$

To make it concise, Eq. 3.21 is formulated as

$$\mathbf{U}\mu = \mathbf{y} \quad (3.22)$$

where $\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}_{N \times 1}$,

$\mathbf{U} = \begin{bmatrix} \mathbf{u}(\mathbf{x}_1)^T \\ \vdots \\ \mathbf{u}(\mathbf{x}_N)^T \end{bmatrix}_{N \times Q} = \begin{bmatrix} P(\mathbf{a}_1, b_1, \mathbf{x}_1) & \cdots & P(\mathbf{a}_Q, b_Q, \mathbf{x}_1) \\ \vdots & \cdots & \vdots \\ P(\mathbf{a}_1, b_1, \mathbf{x}_N) & \cdots & P(\mathbf{a}_Q, b_Q, \mathbf{x}_N) \end{bmatrix}_{N \times Q}$ (hidden layer feature mapping) and $\mu = \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_Q \end{bmatrix}_{Q \times 1}$ (output weight vectors).

In accordance with the above traditional theories of neural network, only when all the hidden layer parameters (e.g. μ_i and $(\mathbf{a}_i, b_i)$) are adjustable can SLFNs work as the universal approximators. In order to minimize the workload of tuning these parameters, researchers proposed incremental approaches in which parameters of existing hidden neurons remain unchanged while those of newly added neurons will be tuned and then fixed. In all the above methods, the hidden layer parameters need to be adjusted for greater than or equal to once. However, this situation will not happen in extreme learning machine. For ELM, all the hidden layer parameters do not need to be iteratively tuned; instead, they are randomly selected and independent from training samples. In theory, Huang et al. [83] has proved that for the purpose of letting SLFNs serve as universal approximators, the hidden layer parameters can be randomly chosen and the output weight vectors connecting the output layer and hidden layer can be analytically determined. In this case, for additive nodes, activation functions can be arbitrary bounded non-constant continuous piecewise functions $g: R \rightarrow R$; for RBF nodes activation functions can be any arbitrary integrable continuous piecewise functions $g: R \rightarrow R$ and $\int_R g(x)dx \neq 0$. Furthermore, Huang et al. [77] demonstrated that as long as activation functions are infinitely differentiable, the hidden layer parameters can be randomly selected. After selection, the hidden neuron parameters $(\mathbf{a}_i, b_i)$ will keep unchanged. Up to now, the sole unknown parameters are output weights μ_i connecting hidden layer and output layer. So the least square method can be used to calculate the output weight μ_i .

In accordance with the theory of feedforward neural network [84], ELM can be mathematically formulated by:

Minimize: $\|\mathbf{U}\mu - \mathbf{y}\|$ (training error) and

minimize: $\|\mu\|$ (norm of weights)

Therefore, training such an SLFN is the same as seeking out a least-squares solution of this $\mathbf{U}\mu = \mathbf{y}$ linear system. Suppose this optimal solution is $\hat{\mu}$, so we have

$$\|\mathbf{U}\hat{\mu} - \mathbf{y}\| = \min_{\mu} \|\mathbf{U}\mu - \mathbf{y}\| \quad (3.23)$$

According to the mathematical theory of matrices [85], the smallest norm least-squares solution for the above system is $\hat{\mu} = \mathbf{U}^\dagger \mathbf{y}$, where $\mathbf{U}^\dagger$ represents the Moore-Penrose generalized inverse of $\mathbf{U}$. In short, ELM can be summarized by the following steps:

- 1) Randomly selecting the hidden layer parameters, i.e., the input weight vector $\mathbf{a}_i$ and the hidden layer biases b_i , where $i = 1, \dots, L$ (number of hidden neurons).
- 2) Calculating the hidden layer output, namely the matrix $\mathbf{U}$.
- 3) Calculating the output weight, namely the vector μ : $\mu = \mathbf{U}^\dagger \mathbf{y}$.

When calculating the Moore-Penrose pseudoinverse $\mathbf{U}^\dagger$ of a matrix $\mathbf{U}$, the orthogonal projection approach can be used with efficiency which can be described as:

- 1) When $\mathbf{U}^T \mathbf{U}$ is nonsingular, $\mathbf{U}^\dagger = (\mathbf{U}^T \mathbf{U})^{-1} \mathbf{U}^T$;
- 2) When $\mathbf{U} \mathbf{U}^T$ is nonsingular, $\mathbf{U}^\dagger = \mathbf{U}^T (\mathbf{U} \mathbf{U}^T)^{-1}$.

Furthermore, in order to obtain more stable solution and better generalization performance, [79] proposed the constrained-optimization-based ELM with the following formulation:

$$\begin{aligned} \min : \quad & \frac{1}{2} \|\mu\|^2 + C \frac{1}{2} \sum_{i=1}^N \|\xi_i\|^2 \\ \text{s.t.} : \quad & \mathbf{u}(\mathbf{x}_i) \mu = y_i - \xi_i, \quad i = 1, \dots, N \end{aligned} \quad (3.24)$$

Following the ridge regression theory, the solution to the above problem can be obtained: $\mu = (\frac{\mathbf{I}}{C} + \mathbf{U}^T \mathbf{U})^{-1} \mathbf{U}^T \mathbf{y}$ (when $\mathbf{U}^T \mathbf{U}$ is nonsingular) or $\mu = \mathbf{U}^T (\frac{\mathbf{I}}{C} + \mathbf{U} \mathbf{U}^T)^{-1} \mathbf{y}$ (when $\mathbf{U} \mathbf{U}^T$ is nonsingular). Since $\mu = \mathbf{U}^\dagger \mathbf{y}$, the solution is like adding a positive value to the diagonal of $\mathbf{U} \mathbf{U}^T$ or $\mathbf{U}^T \mathbf{U}$ during the calculation of $\mathbf{U}^\dagger$. According to and the output function $\hat{y} = \mathbf{u}(\mathbf{x}) \mu$, we have:

$$\hat{y} = \mathbf{u}(\mathbf{x}) \mu = \mathbf{u}(\mathbf{x}) \mathbf{U}^T (\frac{\mathbf{I}}{C} + \mathbf{U} \mathbf{U}^T)^{-1} \mathbf{y} \quad (3.25)$$

with the corresponding $\mu = \mathbf{U}^T(\frac{\mathbf{I}}{C} + \mathbf{U}\mathbf{U}^T)^{-1}\mathbf{y}$ or

$$\hat{y} = \mathbf{u}(\mathbf{x})\mu = \mathbf{u}(\mathbf{x})(\frac{\mathbf{I}}{C} + \mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\mathbf{y} \quad (3.26)$$

with the corresponding $\mu = (\frac{\mathbf{I}}{C} + \mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\mathbf{y}$.

Following the theory of kernel ridge regression, the kernel-based ELM is also available, in which kernel matrix is:

$$K(x_i, x_j) = h(x_i)^T h(x_j) \quad (3.27)$$

So using kernel-based method, the output function will become

$$\hat{y} = \begin{bmatrix} K(\mathbf{x}, \mathbf{x}_1) \\ \vdots \\ K(\mathbf{x}, \mathbf{x}_N) \end{bmatrix}^T (\frac{\mathbf{I}}{C} + \mathbf{K})^{-1}\mathbf{y} \quad (3.28)$$

Eq. 3.28 indicates that the kernel function $K(\mathbf{a}, \mathbf{b})$ (e.g. for RBF kernel, $K(\mathbf{a}, \mathbf{b}) = \exp(-\gamma\|\mathbf{a} - \mathbf{b}\|^2)$) can be directly calculated without knowing specific hidden layer feature mapping $\mathbf{u}(\mathbf{x})$ and the number of hidden nodes Q .

3.1.4 Proposed Method — Multi-feature Extreme Ordinal Ranking Machine (MFEORM)

Basically for the proposed MFEORM, facial age estimation is regarded as an ordinal-ranking-oriented issue. More specifically, first the database of the i th feature models can be divided as follows:

$$M_k^{i(+)} = \{(x_j^i, +1) | y_j > k\}, 1 \leq k \leq K - 1 \quad (3.29)$$

$$M_k^{i(-)} = \{(x_j^i, -1) | y_j \leq k\}, 1 \leq k \leq K - 1 \quad (3.30)$$

where K is the number of total age labels of the database, j is the instance number and y_j is the age labels. Then the problem can be separated into $K-1$ classification subproblems.

For each subproblem k , the purpose is to find whether the face image is older or younger than k . So each subproblem is equivalent to a basic binary classification problem. In each binary classification problem, it is assigned that $z_k[j] = -1$ if $x_j^i \in M_k^{i(-)}$ and $z_k[j] = +1$ if $x_j^i \in M_k^{i(+)}$. Next the aforementioned rule is applied for all i feature models and combined together as our total training sets. Then RBF-kernel-based ELM can be used to train each of k binary classifiers. After traversing k from 1 to $K-1$, $K-1$ binary classifiers are obtained, which imply the ordinal relationships among all age labels. We call these $K-1$ classifiers the “weak classifiers”. In the end, these “weak classifiers” can be comprehensively integrated to form the final result according to our preference-collection rule.

In age estimation, the most popular and most commonly used performance measurement is the mean absolute error (MAE), which can be described by

$$MAE = \sum_{j=1}^N |y_j^* - y_j| / N \quad (3.31)$$

where y_j^* is the estimated age, y_j is authentic age and N is the number of test images. The reason why MAE is the suitable for performance measurement is summarized as follows. As everyone knows, the objective of age estimation is to estimate a person’s age. So when measuring its performance, intuitively we will think of a method to compare the difference between the ground truth (actual/chronological age) and the estimated age. The bigger the difference is, the worse the performance would be. Then how to mathematically determine such a “difference”? Given that the estimated age may be larger or smaller than the ground truth, this difference should be an absolute value (or error) to reflect the deviation degree. Also because the measurement is based on the whole datasets, so the mean computation is needed for those absolute values. The mean computation is just MAE. In a nutshell, MAE actually depicts the mean estimation deviation from the ground truth for the whole datasets, so naturally it is appropriate for performance measurement of facial age estimation.

The steps of the proposed MFEORM algorithm are as follows:

- 1) For each target age k ($1 \leq k \leq K - 1$),
 - a) For each feature model, separate the raw training data into $M_k^{i(+)}$ and $M_k^{i(-)}$.
 - b) Integrate all i feature models' information together as the total training sets.
 - c) Using the kernel-based extreme learning algorithm to train each binary classifier (the so-called "weak classifiers") and get a decision function $f_k(\mathbf{x})$ according to equation (13).
- 2) Construct a preference-collection rule from all subproblems:

$$Estimated\ age = 1 + \sum_1^{K-1} \llbracket f_k(\mathbf{x}) > 0 \rrbracket \quad (3.32)$$

where $\llbracket \cdot \rrbracket$ is equal to 1 if the inner part holds and otherwise equals 0. For example, suppose all the binary classifiers have been trained based on the proposed algorithm and now one face image $\mathbf{P}_1$ whose authentic age is 25 is used to test. So $\mathbf{P}_1$ is substituted into $f_k(\mathbf{x})$ for all $1 \leq k \leq K - 1$ and get all the $f_k(\mathbf{P}_1)$. Finally we find that $f_k(\mathbf{P}_1) > 0$ holds for $1 \leq k \leq 26$, so the final estimated age is 27.

3.1.5 Experiments

3.1.5.1 Data Sets

A series of experiments on the popular benchmark aging database FG-NET [86] were conducted. FG-NET has 1,002 grayscale or color facial images of 82 people, which includes comprehensive poses, expressions and lighting environments, just as Figure 3.2 shows. All people's age range is from 0 to 69, whose age range distribution is shown in Table 3.1. In order to realize uniform processing, all facial images in FG-NET were converted to grayscale, aligned and normalized. Finally histogram equalization was conducted in order to decrease the illumination influence.

Figure 3.2: Facial image examples of one subject in sequential order of different age values in the FG-NET database

Table 3.1: FG-NET’s age level distribution

Range of Age	FG-NET	
	#img.	%
0 - 9	371	37.03
10 - 19	339	33.83
20 - 29	144	14.37
30 - 39	79	7.88
40 - 49	46	4.59
50 - 59	15	1.50
60 - 69	8	0.80
Total	1002	100

3.1.5.2 Experiment Settings

Basically, three feature models for information extraction were used from FG-NET raw images: Active Appearance Model (AAM) [30], local binary patterns (LBP) and Bio-inspired feature(BIF) [47], which would be combined together for a total dataset. Active Appearance Models (AAM) can represent both shape and texture information instead of only facial geometry, which is popularly selected by other age estimation methods. LBP is also a widely-used feature for texture classification in computer vision. BIF was selected because of its high age estimation accuracy. The information extracted from the above three feature models can complement each other; together they were combined as a total dataset. For AAM features, the feature dimension was set to retain 95% of variability. For BIF features, the number of bands was set at 8 (16 scales totally) with 4 orientations each. Also, to reduce the entire feature space, principal component analysis (PCA) was used to reduce the dimension. More specifically, all the three feature models would be reduced to 100 dimensions, respectively. Specially, the AAM model includes both shape and texture information, so these two sub-properties would be reduced to 50 dimensions respectively (in total 100 dimensions). For extreme learning machine, RBF-kernel-based ELM was used. Also, leave-one-person-out (LOPO), a popular test procedure, was utilized for the test strategy, which was suggested in [38][6][87][16][88][17]. In terms of the accuracy,

Table 3.2: Consumed time of the compared MFEORM and MFOR

Method	Runtime(s)	MAE(Mean Absolute Error)
MFEORM [Ours]	320.6	4.50
MFOR	27437.0	4.58

MFEORM’s experimental results were compared with MFOR [69], OHRank [68], WAS [6], AGES [6], RUN1 [16], RUN2 [17], Multi-Task Warped Gaussian Process (MTWGP) [87] and SVM. For the time consumed, because the accuracy of other algorithms is lower than MFOR so we only need to compare the consumed-time results with MFOR. Also, in order to find out the inner properties of MFEORM, different combinations of MFEORM’s inner parameters were tried to analyze its inner characteristics. Furthermore, the number of feature dimensions’ impact on the final estimation result will be shown and that the multi-feature (three-feature dataset) outcompetes two-feature and single-feature dataset based on the proposed MFEORM will be verified in the following Sec. 3.1.5.3.

3.1.5.3 Experiment Results and Analysis

The biggest advantage of our algorithm is the extreme speed. So firstly the consumed time of MFEORM was tested and compared with MFOR.

Table 3.2 shows the comparison of MFEORM and MFOR in respect of the runtime and accuracy(MAE). As can be explicitly seen, MFEORM can shorten the runtime by a significant amount while maintaining high accuracy. Because previous facial age estimation solutions based on multi-feature ordinal ranking usually formulate their models as complex constrained-optimization problems with inequality constraints, so solving these convex optimization problems need huge quantities of time. But for our algorithm, the input weights and hidden layer parameters are randomly generated and analytically determined, so the hidden-layer parameters need not be tuned like SVM. As a result, the proposed method can be much faster than the previous multi-feature ordinal ranking methods.

Table 3.3: MAEs of compared MFEORM and other popular age estimation algorithms on FG-NET database

Method	MAE
MFEORM [Ours]	4.50
MFOR	4.58
OHRank	4.60
RUN1 [16]	5.78
RUN2 [17]	5.33
AGES [6]	6.82
WAS [6]	8.06
MTWGP [87]	4.83
SVM	7.25

Table 3.4: Different regularization coefficient with fixed kernel parameter in MFEORM

Regularization Coefficient C	MAE	Runtime(s)
10	4.630	310.3
40	4.546	312.3
50	4.546	310.7
100	4.556	311.1
250	4.549	310.0
400	4.545	288.9
500	4.540	310.3
600	4.541	289.7
1000	4.545	309.8
10000	4.549	313.1

Table 3.3 displays the MAE results of different algorithms derived from the FG-NET database, which demonstrates that our algorithm achieve a similar or even better accuracy compared to other popular facial age estimation algorithms.

To take it one step further, next the inner properties of MFEORM are explored. Based on the aforementioned description, there exists two uncertain parameters, regularization coefficient C and the kernel parameter γ . For these two parameters, experiments are conducted to explore their influences towards experiment results.

Firstly, different regularization coefficient C but same kernel parameter γ which equals 50 were assigned respectively. Table 3.4 shows the tuning results. Finally regularization coefficient C was set at 500.

Table 3.5: Different kernel parameter with fixed regularization coefficient in MFEORM

Kernel Parameter	MAE	Runtime(s)
10	4.628	297.3
100	4.603	292.2
500	4.692	296.8
1000	4.661	295.4
2500	4.626	297.4
5000	4.695	296.3
10000	4.876	297.2

Table 3.6: Comparison of different number of feature models

Feature Models		MAE	Runtime(s)
Single-feature	AAM	4.83	247.2
	LBP	7.56	269.7
	BIF	5.92	245.4
Two-feature	AAM + BIF	4.74	276.9
	LBP + BIF	5.56	271.7
	AAM + LBP	4.59	272.7
Multi-feature (three-feature)	AAM + LBP + BIF	4.50	320.6

Next, different kernel parameter γ with same regularization coefficient C at 500 were assigned respectively. Table 3.5 shows the coarse tuning results. Then γ was fine-tuned and set at 31 with the optimal MAE of 4.50 and runtime of 320.6 seconds.

Remark: The estimation accuracy and runtime of our MFEORM is insensitive to its two inner parameters, namely the kernel parameter γ and the regularization coefficient C .

The influence of different number of feature models on the final results was further explored (Regularization coefficient C is set at 500 and kernel parameter γ is 31 uniformly). As Table 3.6 indicates, conclusion can be made that for the three feature models used here, the more feature models, the better results. Particularly, in the single-feature test, AAM shows its superior estimation performance in MFEORM, which is better than LBP and BIF features, so we conjecture that this is because AAM gets the feature from

Table 3.7: Comparison of different dimensions' impact

Dimensions (after PCA dimension-reduction)	MAE	Runtime(s)
20 (Shape 10 + Texture 10)	5.59	232.5
40 (20 + 20)	4.86	235.7
60 (30 + 30)	4.85	240.2
100 (50 + 50)	4.83	247.2
120 (60 + 60)	4.79	250.3
140 (70 + 70)	4.78	260.2
160 (80 + 80)	4.85	264.4
200 (100+100)	4.88	278.0

both face shape and texture information separately. So next AAM feature may as well be used to analyze the influence of number of dimensions on the final results.

Table 3.7 displays the tests which were made to explore the influence of number of dimensions (after PCA dimension-reduction) on the final results (Regularization coefficient C is set at 500 and kernel parameter is 31 uniformly). Generally speaking, the estimation accuracy and runtime of our MFEORM are insensitive to the number of dimensions, which means that there's no need to waste a large amount of time on tuning the inner parameters of MFEORM. However, from a nuanced prospective, when the dimensions are too small, the dataset will suffer from insufficient information from feature models, so the deviation will become larger. In contrast, when the dimensions are oversized, the dataset will include not only main components but also some interference information, which can also lead a relatively large deviation.

3.1.6 Concluding Remarks

MFEORM combines multi-feature space, age's natural characteristics of ordinal information and extreme learning machine's rapid learning rate, achieving similar or even better performances with much less time compared to state-of-the-art methods. Experimental results demonstrate for the used three feature models, the more feature models, the better performance. Further experiments from internal and external aspects have been performed respectively so that the following properties and advantages of MFEORM are

Figure 3.3: Flow chart of ordinal hyperplane ranking for facial age estimation

obtained: the performance of MFEORM including estimation accuracy and runtime is insensitive to its two inner parameters and the number of dimensions (after dimension reduction) so there's no need to consume much time on choosing different inner parameters of MFEORM and the number of dataset dimensions.

3.2 Advanced Ordinal Ranking for Facial Age Estimation

3.2.1 Introduction

Just as stated in the foregoing contents, for ordinal hyperplane ranking [68] (for easy understanding, just using basic expressions and ignoring further improvement, such as cost sensitivities and multi-feature), the age label y_i is treated as an order of rank, $y_i \in \{1, \dots, K\}$, where K is the total number of age labels. Then for every target age label $k = 1, \dots, K - 1$, divide the whole dataset into two parts M_k^+ and M_k^- , where

$$M_k^+ = \{(x_i, +1) | y_i > k\} \quad (3.33)$$

$$M_k^- = \{(x_i, -1) | y_i \leq k\} \quad (3.34)$$

Next, as Fig. 3.3 illustrates, for each k , solve the question “is the face older than k ?” and use the above two separated subsets to train a binary classifier using Support Vector Machine (SVM). So in total $k-1$ binary classifiers and decision functions f_k are obtained, in which $f_k = 1$ represents “this image is older than k ” and $f_k = 0$ stands for “not older

Figure 3.4: Ideal and different deviated conditions for the ranking rule of ordinal ranking than k ". Such a series of classification imply the sequential relationships between age labels. Then a ranking rule is constructed by collecting the preference information from all subproblems:

$$Estimated\ age = d(x) = 1 + \sum_{k=1}^{K-1} \mathbb{I}[f_k(x) = 1] \quad (3.35)$$

where $\mathbb{I}[\cdot]$ is equal to 1 if the inner part holds and 0 otherwise.

As can be apparently seen, this ranking rule is reasonable and it reflects the ordinal nature of age. The desired target age label shape should maintains the classification result of 1 at the beginning of target age values; then it drops to 0 at a certain age label

and remains constant with the increase of target age label, which is shown in Fig. 3.4(a). However, the rule is too ideal because sometimes the classifiers may encounter accidental deviations (misclassification). Our experiments find different types of deviated conditions, as Figs. 3.4(b)-(d) demonstrates. Inspired by these defects, two new approaches are proposed to overcome these disadvantages, through which accidental classification deviations can be effectively offset and thus estimation performance is improved.

3.2.2 Proposed Advanced Ordinal Ranking in facial age estimation

Just as Figs. 3.4(b)-(d) show, three kinds of inappropriate target age label shapes were found by extensive beforehand experimental results when using existing ranking rule of ordinal hyperplane ranking. In the following, these three deviations will be discussed respectively.

- i) Swing (Fig. 3.4(b)). This condition happens most, which means the classification results swing frequently between 0 and 1 in a small range of target age labels. Precisely, it's better to call it an "unstable range" rather than the "classification error" because the exact drop point of target age labels cannot be confirmed. In dealing with this condition, further experiments using other mathematical manipulations (e.g., weighted average) were conducted and only parallel or even worse results were obtained. So for this condition, the ranking criterion remains Eq. 3.35.
- ii) Wrong drop and wrong rise (Figs. 3.4(c)-(d)). Obviously, these two conditions are authentic misclassification (error) at a certain target age label. So revisions can be conducted to eliminate such "defective point" and smooth these sudden wrong drop & rise.

As a result, the steps for our first proposed Advanced Ordinal Ranking (AOR1) with new ranking rule are as follows:

- 1) Scan through the whole target age label shape and detect whether the shape has the above three deviated conditions or their combination. If yes, revise them according to Criterion i) and ii).
- 2) Calculate the estimated age using Eq. 3.35.

To take it one step further, notice that classification results for each binary classifier subproblem are the cumulative factor, through which an accumulated age result of integer is obtained. But as analyzed above, each classification may cause unexpected misclassification problem. So the query is conducted: “Is there any other more appropriate cumulative factors which can reduce the influence of those misclassifications to minimum, for example, importing decimals as the estimated age?” Consequently, driven by this idea, the classification accuracy for each binary classification subproblem is taken into account. The new cumulative factor can be formulated as $accuracy(k) * f_k(x)$, where $accuracy(k)$ represents the classification accuracy of the k^{th} binary classifier and $f_k(x)$ is the classification result of the x^{th} facial image in the k^{th} binary classifier. Based on this criterion, the rule of our second proposed Advanced Ordinal Ranking for age estimation (AOR2) is:

$$d(x) = 1 + \sum_{k=1}^{K-1} \frac{1}{2} [accuracy(k) * f_k(x) + 1] \quad (3.36)$$

where $f_k(x) \in \{-1, 1\}$.

In this case, for each target age label, a decimal cumulative factor is obtained, which implies both the belonged class and the relative possibility of this belonged class; also, the impact of potential misclassification can be avoided.

For the binary classifier in each subproblem of ordinal ranking, the Extreme Learning Machine (ELM) [77][78][79] is chosen with its RBF kernel for achieving similar or even better classification performance with much less time consumed compared to most of the

existing classifiers. The output function for the k th binary classification subproblem is

$$f_k(\mathbf{x}) = \left[\begin{array}{c} K(\mathbf{x}, \mathbf{x}_1) \\ \vdots \\ K(\mathbf{x}, \mathbf{x}_N) \end{array} \right]^T \left(\frac{\mathbf{I}}{C} + \mathbf{K} \right)^{-1} \mathbf{y} \quad (3.37)$$

where $K(\mathbf{a}, \mathbf{b})$ is the kernel function (e.g. for RBF kernel, $K(\mathbf{a}, \mathbf{b}) = \exp(-\gamma \|\mathbf{a} - \mathbf{b}\|^2)$), $\mathbf{K}$ is the kernel matrix, $\mathbf{y}$ is the vector of ground truth age values, C is the regularization coefficient as the ELM's inner constant parameter, $\mathbf{I}$ is the identity matrix.

3.2.3 Experiments

3.2.3.1 Datasets and Settings

The experiments were conducted on the popular benchmark aging database FG-NET [86]. To process the database uniformly, all image instances in FG-NET were pre-processed by conversion to grayscale, alignment, normalization and histogram equalization. For the stage of feature extraction, the procedures were the same as those in Sec. 3.1.

In age estimation, the most popular performance measurement is the mean absolute error (MAE), which can be described by

$$MAE = \sum_{j=1}^N |y_j^* - y_j| / N \quad (3.38)$$

where y_j^* is the estimated age, y_j is authentic age and N is the number of test images. Consequently, the smaller the MAE is, the better age estimation performance will be.

The second measurement is Cumulative Score (CS) [6], which can be defined as

$$CS(L) = (M_{e < L} / M_t) \times 100\% \quad (3.39)$$

where $M_{e < L}$ is the number of test facial images with the absolute error e smaller than the error level L .

In terms of the accuracy (including MAE and CS) and runtime, the experimental results of our two proposed methods were compared with Rank-RS [39], ECG [53], CS-LBFL [51], MFOR [69], OHRank [68], WAS [6], AGES [6], RUN1 [16], RUN2 [17], Multi-Task Warped Gaussian Process (MTWGP) [87] and SVM.

Table 3.8: MAEs and runtime of the compared ordinal-ranking-based age estimation algorithms

Method	MAE(Mean Absolute Error)	Runtime(s)
AOR2 [Ours]	4.41	297.7
AOR1 [Ours]	4.49	289.6
MFOR	4.58	27437.0
OHRank	4.60	3040.5

Table 3.9: MAEs of compared popular age estimation algorithms

Method	MAE
AOR2 [Ours]	4.41
AOR1 [Ours]	4.49
CS-LBFL	4.43
ECG	4.46
Rank-RS	4.56
MFOR	4.58
OHRank	4.60
RUN1	5.78
RUN2	5.33
AGES	6.82
WAS	8.06
MTWGP	4.83
SVM	7.25

3.2.3.2 Experimental results

Table 3.8 shows the comparisons between the two algorithms and other popular ordinal-ranking-based algorithms in respect of MAE and runtime. As can be explicitly seen, both AOR1 and AOR2 get topmost efficacy in performance and high efficiency. Smaller MAEs prove the validity of our proposed methods and advantages over the existing ordinal-ranking-based age estimation algorithms; less runtime reflects our algorithms' higher efficiency with the power of ELM as a high-speed and accurate classifier compared to other existing ordinal ranking based age estimation algorithms.

Table 3.9 displays the MAE results of different algorithms, which demonstrates that the performance of our algorithm ranks top compared to other popular facial age esti-

mation approaches.

Fig. 3.5 demonstrates the compared CS curves for different error level L . When CS value is fixed, the algorithm with smaller L is superior. On the other hand, for a fixed L , the method with taller CS value will be more competitive. From this figure, for almost all error level L , our proposed algorithm yields highest precision among compared method, which is a further evidence that the proposed method outperforms the existing mainstream age estimation algorithms, especially those based on ordinal ranking.

3.2.4 Concluding Remarks

In Sec. 3.2, two methods AOR1 and AOR2 are proposed with new ranking rules in the ordinal-ranking-based facial age estimation. AOR1 uses certain criteria to remove the accidental classification deviations (misclassification, such as conditions of swing, wrong drop & rise); AOR2 utilizes the product of the classification result and relevant classifier accuracy in each ordinal ranking subproblem (binary classification) as the cumulative factor when calculating the estimated age. Moreover, multiple feature models and the Extreme Learning Machine (ELM) are applied for further improvement of accuracy and efficiency. The experimental results clearly demonstrate that the proposed methods outperform other existing methods in respect of estimation accuracy and consumed time, especially those ordinal-ranking-based ones.

Figure 3.5: Cumulative Scores (CS) of compared age estimation algorithms

Chapter 4

Strategic Decision-making Learning from Label Distributions for Facial Age Estimation

4.1 Introduction

In spite of all the aforementioned merits, traditional age estimation algorithms fail to consider that age labels have certain relationship with each other to different degrees. More specifically, because aging is a slow and gradually varied process, the adjacent labels around a certain age label X will necessarily have connection with label X and thus can inevitably describe the characteristics of label X to different degrees. So in [16], Yan et al. proposed a solution to age ranking problem based on the training samples with uncertain age labels: a small range (within one year) was set as the uncertain labels for a given age label. In [89], the *label sensitive* concept was proposed for better exploiting the information of ordinal relationship among age labels. In their training phase, for a given age label, samples belonging to its neighbouring age labels are also involved; in other words, the weight of each sample in computing the quantities of a specific age label was assigned according to label similarity. However, these two approaches only treat one facial image instance with a single age label (not multiple age labels); also as above mentioned, almost all age estimation algorithms suffer from insufficient training datasets. Inspired by these defects, Geng et al. [90][91] proposed the Label Distribution Learning

(LDL) for facial age estimation. This method takes full advantages of the similarities between the authentic age and its adjacent ages instead of regarding each age label as one isolated entity; that is to say, this method better reveals the nature of human’s aging process than other existing age estimation algorithms.

In detail, label distribution learning [90][91] starts from the observation and intuitive common sense that the faces at neighbouring ages tend to have much similarity. An extreme case is that a person’s face would look like almost the same in the last day of his or her 20-year-old and the first day of 21-year-old. In other words, adjacent ages can make a great contribution to the chronological (authentic) age. Actually in the real world, people are also more accustomed to judge a person’s age by ”about 30-year-old” or ”around 40-50” rather than directly telling the exact age number. So LDL allocates each facial image an ”age label distribution with multi-label” instead of the ”chronological age as single-label”. Based on this idea, in order to represent the degree that those adjacent age labels describe this facial image, LDL firstly introduces the concept of *description degree*. Specifically, suppose we have a facial image $\mathbf{x}$, then the *description degree* $d_{\mathbf{x}}^y \in [0, 1]$ (y represents a certain age label in the whole age range) is the level at which the age y depicts this instance. Also, the description degree must satisfy the restricted condition $\sum_y d_{\mathbf{x}}^y = 1$. Figure 4.1 demonstrates three different types of label distribution for six-class labels. Type (a) shows the most traditionally frequently-used and simplest case: single label. In the figure, the instance is allocated only one label y_4 and thus y_4 ’s description degree $d_{\mathbf{x}}^{y_4} = 1$ while the description degree for y_1, y_2, y_3, y_5, y_6 is 0, which means the sole label y_4 is able to totally describe this instance with other labels contributing nothing. Fig. 4.1b is another case called multiple label, in which multiple labels (two, three or more) would have even description intensities. As can be seen, y_3 and y_6 is allocated evenly to describe the instance and each takes up 0.5 (50%) as their description degree, accompanied by other remaining labels contributing nothing with the corresponding description degrees 0. A more general case is in Fig. 4.1c: each label is allocated a description degree $d_{\mathbf{x}}^y \in [0, 1]$ with $\sum_y d_{\mathbf{x}}^y = 1$.

(a) Single Label

(b) Multiple Label

(c) General Label Distribution

Figure 4.1: Three different types of label distribution

In general, label distribution is more flexible to represent the ambiguity. However, the learning algorithms from label distribution [90][91] suffer from their simple decision-making criteria which only blindly choose the age label with maximum description degree as the final predicted age and fail to take the characteristics of aging process into account. Actually in practice, when we finish the step of learning from label distribution and head into the last step of decision-making, it is far from being rare that many age labels would have very similar description degrees. So in this case, rashly assigning the one with highest description degree would neglect all the other age labels with close description degrees, which may consequently enlarge the final estimation deviation and degrade the algorithm's overall performance. In this chapter, a series of strategic learning algorithms are proposed for decision-making to effectively solve this problem in the application of facial age estimation.

4.2 Label Distribution Learning and Its Decision-Making Criterion

Figure 4.2 shows the example of three different types of multiple labels for age representation, in which Fig. 4.2(a) and Fig. 4.2(b) are two primary age label distributions, namely Gaussian-like distribution and triangle distribution; Fig. 4.2(c) indicates the multiple label with same description degrees. In the first two distributions, the description degree of the chronological age (authentic age) is the highest; for other age labels on both sides of the chronological age, description degrees decrease symmetrically to the same extent. Particularly, the condition of Fig. 4.2(a) is not called “Gaussian distribution” but “Gaussian-like distribution”, because Gaussian distribution is a continuous function for the independent variable traversing the set of all real numbers. However, the age label is a series of discrete integers with a limited domain. So in the application of age estimation, we only use the shape of Gaussian distribution and discretize the previous “probability density” to constitute description degrees. In detail, firstly calculate $pdf(y)$

(a) Gaussian-like distribution (with seven-class labels)

(b) Triangle distribution (with bottom length of six)

(c) Multiple label with same description degrees (with seven-class labels)

Figure 4.2: Two primary age label distributions (a), (b) and multiple age labels with same description degrees (c) for the chronological age a

for all possible age label y ($pdf(\cdot)$ stands for probability density function) and then do the normalization $d_{\mathbf{x}}^y = pdf(y) / \sum_y pdf(y)$ so that $\sum_y d_{\mathbf{x}}^y = 1$ is guaranteed. Note that when generating the label distribution, the description degrees of all age labels are greater than 0 for Gaussian-like distribution (Fig. 4.2(a)), which means all age labels are involved and contribute to the label distribution; but in the condition of Fig. 4.2(b)-(c), only part of age labels contributes to the label distribution (the description degrees of other age labels are 0).

Generally, Label Distribution Learning (LDL) utilizes the methods in statistics to learn a conditional probability mass function from the ready-made label distribution and get the corresponding description degrees [90][91]. In detail, the description degree $d_{\mathbf{x}}^y$ can be mathematically seen as $d_{\mathbf{x}}^y = P(y|\mathbf{x}) \in [0, 1]$, which means that for an instance $\mathbf{x}$, the description degree of the label y equals the conditional probability of y given $\mathbf{x}$. Next, suppose the input space is denoted by $\aleph = \aleph^w$ and the label set $\Psi = \{y_1, y_2, y_3, \dots, y_t\}$ (t is the total number of labels) which contains all involving labels. Then given a training set with n instances $\aleph = \{(\mathbf{x}_1, D_1), (\mathbf{x}_2, D_2), \dots, (\mathbf{x}_n, D_n)\}$, where $\mathbf{x}_i \in \aleph$ is the i^{th} instance and $D_i = \{d_{\mathbf{x}_i}^{y_1}, d_{\mathbf{x}_i}^{y_2}, \dots, d_{\mathbf{x}_i}^{y_t}\}$ is the label distribution for the i^{th} instance, the objective is to learn the approximated conditional probability mass function $p(y|\mathbf{x})$ from the training set $\aleph$, in which $\mathbf{x} \in \aleph$ and $y \in \Psi$.

In order to solve the above-mentioned question, the parameter vector $\boldsymbol{\theta}$ needs to be introduced as $p(y|\mathbf{x}; \boldsymbol{\theta})$. Then the problem becomes: to find out a suitable $\boldsymbol{\theta}$ that can generate a label distribution approximating D_i given $\mathbf{x}_i$. Here Kullback-Leibler (KL) divergence is used as the measurement of two distributions' similarity, which can be represented by:

$$D_{KL} = \sum_i \sum_j \left(d_{\mathbf{x}_i}^{y_j} \ln \frac{d_{\mathbf{x}_i}^{y_j}}{p(y_j|\mathbf{x}_i; \boldsymbol{\theta})} \right). \quad (4.1)$$

Thus, the optimal solution $\boldsymbol{\theta}_{opt}$ for the parameter vector $\boldsymbol{\theta}$ should be obtained by mini-

mizing the KL divergence, namely

$$\begin{aligned}\theta_{opt} &= \arg \min_{\theta} \sum_i \sum_j \left(d_{\mathbf{x}_i}^{y_j} \ln \frac{d_{\mathbf{x}_i}^{y_j}}{p(y_j|\mathbf{x}_i; \theta)} \right) \\ &= \arg \max_{\theta} \sum_i \sum_j \left(d_{\mathbf{x}_i}^{y_j} \ln p(y_j|\mathbf{x}_i; \theta) \right).\end{aligned}\quad (4.2)$$

Then $p(y|\mathbf{x}; \theta)$ can be formulated by maximum entropy model [92] as

$$p(y|\mathbf{x}; \theta) = \frac{1}{\sum_y \exp(\sum_k \theta_{y,k} \tau_k(\mathbf{x}))} \exp\left(\sum_k \theta_{y,k} \tau_k(\mathbf{x})\right), \quad (4.3)$$

where $\tau_k(\mathbf{x})$ represents the k^{th} feature in $\mathbf{x}$ and $\theta_{y,k}$ is one element in the model parameter vector θ . From Eq. 4.2 and Eq. 4.3, the objective function $\Omega(\theta)$ can be derived as

$$\Omega(\theta) = \left(\sum_{i,j} d_{\mathbf{x}_i}^{y_j} \sum_k \theta_{y_j,k} \tau_k(\mathbf{x}_i) \right) - \left(\sum_i \ln \sum_j \exp\left(\sum_k \theta_{y_j,k} \tau_k(\mathbf{x}_i)\right) \right). \quad (4.4)$$

IIS-LLD and BFGS-LLD [90][91] are the main algorithms in dealing with the above optimization problem. In the following, BFGS-LLD is used as the example to illustrate the working process. Our target is to maximize Eq. 4.4, which can be converted into minimizing $\Gamma(\theta) = -\Omega(\theta)$. The second order Taylor expansion of $\Gamma(\theta)$ at $\theta = \theta^{(q)}$ is

$$\begin{aligned}\Gamma(\theta^{(q+1)}) &\approx \Gamma(\theta^{(q)}) + \nabla \Gamma(\theta^{(q)}) (\theta^{(q+1)} - \theta^{(q)}) + \frac{1}{2} (\theta^{(q+1)} - \theta^{(q)})^T \nabla^2 \Gamma(\theta^{(q)}) \\ &\quad (\theta^{(q+1)} - \theta^{(q)})\end{aligned}\quad (4.5)$$

where $\nabla \Gamma(\theta^{(q)})$ and $\nabla^2 \Gamma(\theta^{(q)})$ are the gradient and Hessian matrix of $\Gamma(\theta)$ (denoted as $\mathbf{H}(\theta^{(q)})$) at $\theta^{(q)}$ respectively; $(\theta^{(q+1)} - \theta^{(q)})$ is the update step (denoted as Φ_q). So Eq. 4.5 becomes

$$\Gamma(\theta^{(q+1)}) \approx \Gamma(\theta^{(q)}) + \nabla \Gamma(\theta^{(q)}) \Phi_q + \frac{1}{2} \Phi_q^T \mathbf{H}(\theta^{(q)}) \Phi_q \quad (4.6)$$

However, the above-mentioned method needs to repeatedly calculate the inverse Hessian matrix, which may suffer from high complexity (for both calculation and memory space) and thus low efficiency, especially when the feature dimension is high. Moreover, in many

cases the required Hessian matrix can not keep positive definite, so that would result in the failure of implementation. To solve this problem, the approximation matrix $\mathbf{B}$ of Hessian matrix $\mathbf{H}$ is introduced with the constraint of the following secant condition

$$\nabla\Gamma(\boldsymbol{\theta}^{(q+1)}) - \nabla\Gamma(\boldsymbol{\theta}^{(q)}) = \mathbf{B}(\boldsymbol{\theta}^{(q+1)}) \cdot \boldsymbol{\Phi}_q \quad (4.7)$$

Let $\mathbf{u}_q = \nabla\Gamma(\boldsymbol{\theta}^{(q+1)}) - \nabla\Gamma(\boldsymbol{\theta}^{(q)})$, then Eq. 4.7 can be reduced to

$$\mathbf{u}_q = \mathbf{B}(\boldsymbol{\theta}^{(q+1)}) \cdot \boldsymbol{\Phi}_q \quad (4.8)$$

In BFGS method, $\mathbf{B}(\boldsymbol{\theta}^{(q)})$ is updated as

$$\mathbf{B}^{-1}(\boldsymbol{\theta}^{(q+1)}) = \left(\mathbf{I} - \frac{\boldsymbol{\Phi}_q \mathbf{u}_q^T}{\mathbf{u}_q^T \boldsymbol{\Phi}_q} \right) \mathbf{B}^{-1}(\boldsymbol{\theta}^{(q)}) \left(\mathbf{I} - \frac{\mathbf{u}_q \boldsymbol{\Phi}_q^T}{\mathbf{u}_q^T \boldsymbol{\Phi}_q} \right) + \frac{\boldsymbol{\Phi}_q \boldsymbol{\Phi}_q^T}{\mathbf{u}_q^T \boldsymbol{\Phi}_q} \quad (4.9)$$

For line search, the searching direction is

$$\mathbf{d}_q = -\mathbf{B}^{-1}(\boldsymbol{\theta}^{(q)}) \cdot \nabla\Gamma(\boldsymbol{\theta}^{(q)}) \quad (4.10)$$

In this condition, $\boldsymbol{\Phi}_q = \lambda_q \mathbf{d}_q$ where λ_q is the step length. Note that λ_q should be obtained in line search satisfying the strong Wolfe conditions:

$$\Gamma(\boldsymbol{\theta}^{(q)} + \lambda_q \mathbf{d}_q) \leq \Gamma(\boldsymbol{\theta}^{(q)}) + c_1 \lambda_q \nabla\Gamma(\boldsymbol{\theta}^{(q)})^T \mathbf{d}_q \quad (4.11)$$

$$\left| \nabla\Gamma(\boldsymbol{\theta}^{(q)} + \lambda_q \mathbf{d}_q)^T \mathbf{d}_q \right| \leq c_2 \left| \nabla\Gamma(\boldsymbol{\theta}^{(q)})^T \mathbf{d}_q \right| \quad (4.12)$$

where $0 < c_1 < c_2 < 1$.

Particularly for age estimation, the elements of $\nabla\Gamma(\boldsymbol{\theta})$ is calculated respectively by

$$\frac{\partial\Gamma(\boldsymbol{\theta})}{\partial\theta_{y_j,k}} = \sum_i \frac{\exp\left(\sum_k \theta_{y_j,k} \tau_k(\mathbf{x}_i)\right) \tau_k(\mathbf{x}_i)}{\sum_j \exp\left(\sum_k \theta_{y_j,k} \tau_k(\mathbf{x}_i)\right)} - \sum_i d_{\mathbf{x}_i}^{y_j} \tau_k(\mathbf{x}_i) \quad (4.13)$$

After $p(y|\mathbf{x}; \boldsymbol{\theta})$ is learned from the training set, the label distribution of any new instance $\mathbf{x}'$ is $p(y|\mathbf{x}'; \boldsymbol{\theta})$. Then the final predicted age is obtained by the following decision-making criterion:

$$y_{final} = \arg \max_y p(y|\mathbf{x}'; \boldsymbol{\theta}), \quad (4.14)$$

which can be explained as: to choose the age label with the maximum description degree in the calculated label distribution of this new instance.

4.3 Strategic Decision-making Label Distribution Learning (SDM-LDL) for Facial Age Estimation

So far, three main algorithms based on label distribution learning have been proposed, i.e., IIS-LLD, BFGS-LLD and CPNN [90][91]. However, these algorithms seem to put more emphasis on dealing with pure mathematical problems (optimization and parameter tuning) for obtaining the label distribution output $p(y|\mathbf{x};\boldsymbol{\theta})$; on the other hand, they only pick the age label with the highest description degree and neglect the distribution of other labels with similar description degrees which may also contribute much to decision making. In other words, they fail to design more appropriate and complex decision-making criteria specialized for the application of facial age estimation.

Fundamentally, the reason why the decision-making criterion of original LDL is deficient can be explained as follows: the obtained distribution is not symmetrical along two sides of the maximum description degree. In other words, the obtained age label distribution suffers from distributing unevenly alongside the center of the maximum description degree; specifically, there exists the possibility when the neighboring age labels with relatively high description degrees are located more intensively on one side than the other side. For this condition, the decision-making rule should lean to the abovementioned “more intensive” range/side. That is, more age labels on the “intense” side should be involved in and contribute to the final estimation process than the ones on the “sparse” side (the other side). Consequently, if we still simply pick the highest description degree, then the estimated age will have larger deviations and all of the neighboring high description degrees only second to the maximum value will become meaningless. Inspired by this defect, a series of strategic decision-making algorithms for label distribution learning (SDM-LDL) are proposed for age estimation.

Now suppose that $p(y|\mathbf{x};\boldsymbol{\theta})$ has already been learned from the training set using IIS-LLD, BFGS-LLD or CPNN. Then the label distribution of a new instance $\mathbf{x}'$ can be calculated by $p(y|\mathbf{x}';\boldsymbol{\theta})$ for all y (age labels). In order to obviously compare the differences

between these algorithms, both newly proposed algorithms with different decision-making strategies and the original LDL decision-making rule without any strategy are listed below.

Original Decision-making Rule without Strategy Scan through all age labels y and search for the maximum $p(y|\mathbf{x}'; \boldsymbol{\theta})$, then the predicted age would be chosen as $y_p = \arg \max_y p(y|\mathbf{x}'; \boldsymbol{\theta})$, namely to directly select the age label with the maximum description degree. It is worth noting that in this method, only one age label gets involved in determining the final predicted age.

Newly proposed Strategic Decision-making Algorithm (SDM-LDL) with Strategy 1 As mentioned above, the initially obtained age label distribution does not distribute evenly along two sides of the maximum description degree, so merely selecting the age label with highest description degree will lead to large deviation from the ground truth and suboptimal estimation performance. Consequently, one natural and straightforward idea is to choose multiple age labels with higher description degrees as the “age label base” and the final result (age) can be estimated as the mean value for this “base”. Then here comes another question: how to determine the number of the chosen age labels? In Strategy 1, this value (hereinafter referred to as N) is manually selected. The detailed procedure can be summarized as follows. Rank all description degrees in descending order and extract the top N description degrees and their corresponding age labels $\{y_{s_1}, y_{s_2}, \dots, y_{s_N}\}$. Then add these age labels up and obtain the mean value as the final age. (Note that here N is a positive integer in the range 2-10 which is pre-chosen by us and the influence of different values for N would be further shown and compared in the experiment section.) So in this strategy, the predicted age could be mathematically expressed as

$$y_p = \frac{1}{N}(y_{s_1} + y_{s_2} + \dots + y_{s_N}). \quad (4.15)$$

This strategy involves N age labels in determining the final predicted age.

SDM-LDL with Strategy 2 In Strategy 1, the final estimated age is determined as the mean value of those selected top N age labels. In essence, this action evenly (equally) considers all the N labels which have different description degrees; in other words, the description degree information for these N labels is not utilized. Then how to make use of both the abovementioned description degree information and the top N age labels? An effective solution is to calculate the weighted sum, in which the weights are obtained by normalizing the corresponding description degrees of these pre-chosen N labels. The concrete steps are as follows. Rank all description degrees in descending order and extract the top N description degrees and their corresponding age labels $\{y_{s_1}, y_{s_2}, \dots, y_{s_N}\}$. Then for these N degrees, do the normalization and get the normalized weights. Finally, accumulate the product of the weights and their corresponding age labels. Note that here N is also pre-chosen and tested within the range 2-10 as in Strategy 1 in order to see which value in this range would get the best performance and whether the value of N would have a regular impact for the final estimation result. In this strategy, the predicted age could be formulated as

$$y_p = \sum_{y_s=y_{s_1}}^{y_{s_N}} y_s * \frac{p(y_s|\mathbf{x}'; \boldsymbol{\theta})}{\sum_{y_s=y_{s_1}}^{y_{s_N}} p(y_s|\mathbf{x}'; \boldsymbol{\theta})}. \quad (4.16)$$

This strategy involves N age labels and their corresponding description degrees in determining the final predicted age.

SDM-LDL with Strategy 3 In Strategy 1 and 2, the number of age labels is manually chosen. Whether this value is properly selected or not will have a direct impact on age estimation performance. So another question comes to the mind: can we find an appropriate adaptive value for the number of chosen age labels so that Strategy 1 can be autonomously conducted without human's interference? Driven by this question, we focus on the differences between every two adjacent description degrees in descending order: to a great extent, the largest difference is an indicator to distinguish the age labels

with higher description degrees from those with lower description degrees. The whole process can be described as follows. Rank all description degrees in descending order $\{y_{s_1}, y_{s_2}, \dots, y_{s_t}\}$ (t is the total number of age labels), calculate the differences between adjacent description degrees and obtain the set $\{di_1, di_2, \dots, di_{t-1}\}$, where $di_1 = y_{s_1} - y_{s_2}$, $di_2 = y_{s_2} - y_{s_3}$, ..., $di_{t-1} = y_{s_{t-1}} - y_{s_t}$. Then find the maximum di (denoted by di_G ; in other words, the sequence number of this maximum di is denoted as G) and calculate the mean value from the top G description degrees in descending order, namely $\{y_{s_1}, y_{s_2}, \dots, y_{s_G}\}$. Consequently, the predicted age could be described as

$$y_p = \frac{1}{G}(y_{s_1} + y_{s_2} + \dots + y_{s_G}). \quad (4.17)$$

In this strategy, G age labels directly contribute to the final determination of the predicted age.

SDM-LDL with Strategy 4 Inspired by Strategy 2 and 3, we are motivated to combine the advantages of these two methods. Firstly the proper G is obtained autonomously, then the normalization and weighted sum are conducted to use both the age label and the corresponding description degree information. The specific procedure is summarized as follows. Rank all description degrees in descending order $\{y_{s_1}, y_{s_2}, \dots, y_{s_t}\}$ (t is the total number of age labels), calculate the differences between adjacent description degrees and obtain the set $\{di_1, di_2, \dots, di_{t-1}\}$, where $di_1 = y_{s_1} - y_{s_2}$, $di_2 = y_{s_2} - y_{s_3}$, ..., $di_{t-1} = y_{s_{t-1}} - y_{s_t}$. Then find the maximum di (denoted by di_G); for these G degrees, do the normalization and get the normalized weights. Finally, accumulate the product of the weights and their corresponding age labels. Consequently, the predicted age could be described as

$$y_p = \sum_{y_s=y_{s_1}}^{y_{s_G}} y_s * \frac{p(y_s|\mathbf{x}';\boldsymbol{\theta})}{\sum_{y_s=y_{s_1}}^{y_{s_G}} p(y_s|\mathbf{x}';\boldsymbol{\theta})}. \quad (4.18)$$

In this strategy, G age labels and their corresponding description degrees directly contribute to the final determination of the predicted age.

It is worth mentioning that Strategy 3 and 4 utilize successive differences to seek out a specific boundary distinguishing between the age labels with great contribution and small contribution, so that the age labels with great contribution would be adopted for the final predicted age and those with small contribution would be discarded.

SDM-LDL with Strategy 5 Strategy 1-4 adopt only partial age labels so that the complete description degrees are not fully exploited (only part of description degrees involved). So this strategy is designed to take advantage of all age labels and all the description degrees: multiply age labels by their corresponding description degrees respectively and adopt the cumulative value of these product as the final result. Then the predicted age would be calculated by

$$y_p = \sum_{y=y_1}^{y_t} y * p(y|\mathbf{x}'; \boldsymbol{\theta}). \quad (4.19)$$

Table 4.1: An example of the obtained age label distribution for a new facial image x' (with the authentic age of 16)

Age Label	0	1	2	...	12	13	14	15	16
Description degree	0.0010	0.0017	0.0026	...	0.0480	0.0570	0.0642	0.0738	0.0898
Age Label	17	18	19	20	21	22	...	68	69
Description degree	0.0909	0.0913	0.0786	0.0667	0.0540	0.0389	...	1.0667e-05	2.7528e-05

Table 4.2: Based on Table 4.1, the age label distribution with descending-order description degrees and successive differences di

Age Label	18	17	16	19	15	20	14	13	21	12	22	...
Description degree (descending order)	0.0913	0.0909	0.0898	0.0786	0.0738	0.0667	0.0642	0.0570	0.0540	0.0480	0.0389	...
Difference (di)	4e-04	0.0011	0.0112	0.0048	0.0071	0.0025	0.0072	0.0030	0.0060	0.0091	...	...

For a more distinct demonstration, an example is here for further illustration, as is shown in Table 4.1, 4.2 and Fig. 4.3. Suppose $p(y|\mathbf{x}; \boldsymbol{\theta})$ has already been learned from

the training set using IIS-LLD, BFGS-LLD or CPNN. Now when a new facial image $\mathbf{x}'$ comes in, the corresponding age label distribution can be obtained as in Table 4.1 and Fig. 4.3a. Then on the basis of Table 4.1, all these description degrees are ranked in descending order and also calculate the successive differences di one by one, which is shown in Table 4.2 and Fig. 4.3(b)-(c). Next, different types of age are listed as follows:

Table 4.3: The comparison of deviations from authentic age based on the example of Table 4.1

	Deviation from authentic age (absolute value)
Original LDL	2.0000
SDM-LDL (Str 1)	0.5000
SDM-LDL (Str 2)	0.6732
SDM-LDL (Str 3)	1.0000
SDM-LDL (Str 4)	1.0055
SDM-LDL (Str 5)	0.2586

Table 4.3 shows the comparison of deviation from the authentic age **16** using the proposed SDM-LDL with all strategies as well as the original LDL. As can be seen from the table, all the SDM-LDL achieve smaller deviation compared to the original LDL without any decision-making strategy. Although it is only a possible example of the obtained age label distribution, this illustration vividly shows the deficiency of original LDL’s decision-making criterion due to the obtained distribution’s asymmetry along two sides of the maximum description degree. On the contrary, the proposed SDM-LDL with various strategies is specially designed to suit the inner characteristics of aging and also make up for this drawback, which takes full advantage of the whole age label distribution. In other words, SDM-LDL offsets the deficiency and deviations caused by original LDL’s decision-making rule to the greatest extent. As for the above example, the relatively high description degrees occur more frequently on the left side of the maximum description degree (**18** with the description degree of 0.0913) than on the right side (15, 16, 17 on the left side versus 19 on the right side). So the predicted age should lean to the direction of “smaller than 18” rather than 18, as obtained from the five strategies of SDM-LDL.

(a)

(b)

(c)

Figure 4.3: Histogram illustrations for the example of Table 4.1 and 4.2: (a) corresponds to Table 4.1; (b) and (c) correspond to Table 4.2

4.4 Experiments

4.4.1 Experimental Environment Settings

The aging database which our experiments were conducted on is the most popular facial aging benchmark: FG-NET [86]. For uniform processing, all facial instances in FG-NET were converted to grayscale, aligned and normalized. Finally histogram equalization is conducted in order to decrease the illumination influence.

In order to increase the accuracy of the final predicted age, three feature models for information extraction were used from FG-NET raw images: Active Appearance Model (AAM) [30], local binary patterns (LBP) and Bio-inspired feature(BIF) [47], which would be combined together for a total dataset. Active Appearance Models (AAM) can represent both shape and texture information instead of only facial geometry, which is also popularly selected by other age estimation methods. LBP is also a widely-used feature for texture classification in computer vision. BIF was selected because of its high age estimation accuracy. The information extracted from the above three feature models can complement each other; together they were combined as a total dataset. For AAM features, the feature dimension was set to retain 95% of variability. For BIF features, the number of bands was set at 8 (16 scales totally) with 4 orientations each. Also, to reduce the entire feature space, principal component analysis (PCA) was used to reduce the dimension. More specifically, all the three feature models would be reduced to 100 dimensions, respectively. Particularly, the AAM model includes both shape and texture information, so these two sub-properties would be reduced to 50 dimensions respectively (in total 100 dimensions). Also, leave-one-person-out (LOPO), a popular test procedure, was utilized for the test strategy, which was suggested in [38][6][16][88][17].

4.4.2 Experimental Results

Like Chapter 3, the mean absolute error (MAE) is also used as the performance measurement. Just as mentioned above, when initially generating the age label distribution

for a given chronological age, there are three different ways, i.e., Gaussian-like distribution, triangle distribution and multi-label distribution with equal description degrees (Fig. 4.2(a)-(c)). These three conditions have their respective features and thus the final predicted ages based on them are different from each other. Consequently, these three conditions will be firstly analyzed separately and then compared with each other to obtain the overall conclusion. Note that our proposed algorithm is applicable to all existing LDL methods (IIS-LLD, BFGS-LLD, CPNN); however, for all the following LDL-based experiments (including original LDL and the proposed SDM-LLD), only BFGS-LLD method is utilized in computing $p(y|\mathbf{x}; \boldsymbol{\theta})$ to maintain consistency.

Table 4.4: MAEs of the compared original LDL algorithm without decision-making strategy and the proposed SDM-LLD algorithms with different strategies on the condition that age label distribution is generated as Gaussian-like (σ from 1 to 10; for Strategy 1 and 2, best results are shown in the table along with the value of N then; for every σ , the optimal MAEs are marked in bold)

σ	1	2	3	4	5
MAE(Ori)	6.261	5.868	5.573	5.462	5.431
MAE(Str 1)	5.471 (N=8)	5.251(N=9)	5.219(N=9)	5.159(N=9)	5.190(N=9)
MAE(Str 2)	5.588(N=10)	5.246(N=10)	5.180(N=10)	5.127(N=10)	5.139(N=10)
MAE(Str 3)	6.084	5.753	5.478	5.290	5.225
MAE(Str 4)	6.088	5.760	5.473	5.280	5.226
MAE(Str 5)	5.513	5.080	4.938	4.920	5.023
σ	6	7	8	9	10
MAE(Ori)	5.321	5.366	5.322	5.320	5.330
MAE(Str 1)	5.186(N=9)	5.192(N=9)	5.172(N=5)	5.173(N=7)	5.192(N=7)
MAE(Str 2)	5.153 (N=9)	5.166(N=9)	5.171(N=5)	5.168(N=7)	5.187 (N=7)
MAE(Str 3)	5.181	5.112	5.125	5.184	5.321
MAE(Str 4)	5.161	5.098	5.089	5.130	5.252
MAE(Str 5)	5.157	5.344	5.550	5.828	6.126

Figure 4.4: The line chart (variation tendency) of MAEs with respect to different standard deviations for the original LDL algorithm and the proposed SDM-LDL algorithms with different strategies when age label distribution is generated as Gaussian-like

Gaussian-like distribution First of all, when generating the Gaussian-like age label distribution, the controlled variable is the standard deviation σ . So in the following experiment of this part, when realizing every strategy mentioned above, an integer range 1-10 will be assigned to the standard deviation and comparison will be made for the final predicted age with different standard deviations. Note that for the standard deviation σ , the integer range 1-10 is quite a broad range because for $\sigma = 5, 6, 7$ and even greater, the age label distribution tends to become “flatter and flatter”, which means description degrees of different age labels would get closer to the description degree of the actual age and the description degrees’ disparities among different age labels become less and less obvious. However, the standard deviation is given a great range in order to more clearly display the rule and tendency of the standard deviation’s impact on the overall estimation performance.

Table 4.4 shows the comprehensive comparisons including different strategies used in the proposed SDM-LDL algorithms and different standard deviations σ when age label distribution is generated as Gaussian-like. In the vertical direction, for every value of σ , almost all the SDM-LDL algorithms with different strategies obtain smaller MAE than the original LDL algorithm, which demonstrate the validity and superiority of the proposed algorithms with all strategies. Besides, when σ varies, different strategies show their respective advantages. For example, when $\sigma = 1$, Strategy 1 gets the best performance; when $\sigma = 6, 10$, Strategy 2 gets the optimal results; when $\sigma = 7, 8, 9$, Strategy 4 gets the smallest MAEs and when $\sigma = 2, 3, 4, 5$, Strategy 5 outperforms all other strategies. Horizontally, when σ increases from 1 to 10, MAEs of all LDL-based algorithms (including original LDL and SDM-LDL) exhibit the general tendency of first decreasing and then increasing. This indicates that there exists an optimal value (or a small range) of σ to suit different algorithms and for different algorithms, such an optimal value varies. For instance, if using SDM-LDL with Strategy 5, then the optimal value for σ is 4; however in the utilization of SDM-LDL with Strategy 3, the optimal value for σ is 7.

More intuitively from Figure 4.4, most parts of the fold lines of Strategy 1-5 fall below that of the original LDL algorithm and this means our proposed algorithms get superior accuracy compared to the original one. Also in general, almost all fold lines in Fig. 4.4 follow the tendency of first going down and then going up. Moreover, the fold line located in the bottom is changing with the variation of σ , indicating the best strategy varies for different standard deviations. Noticing that for Strategy 5, when σ gets bigger, the MAE firstly drops quickly and then rises dramatically, so the appropriate σ value for Strategy 5 can be selected from 2,3,4,5. In fact, the reason why Strategy 5 fold line drastically climbs afterwards is obvious: when σ becomes very large and extends a certain range, the description degrees for all age labels will get very close. In this case, when all the products of age labels and their corresponding description degrees are accumulated according to Strategy 5, the final obtained age would approach the median of the whole age label range, which, as a result, makes the deviation bigger and bigger. (Just imagine in extreme cases when σ approaches to infinity so that the description degrees of all age labels are equivalent, then if using Strategy 5, the calculated age result will be the median value in the age label range.) In particular, for the most commonly used standard deviation σ ranging from 1 to 5, the fold line of the original LDL falls steeply and tends to be volatile, which means the original LDL is not stable enough for typical values of standard deviation. Also, in this range of σ , the original LDL doesn't reach the optimal MAE. In contrast, the proposed SDM-LDL with Strategy 1,2,5 all reach the lowest value of MAE for $\sigma \in [1, 5]$ with big advantages over the "Original" fold line as well as more stability. Besides, chances are greater that one is going to use the LDL-based algorithms in facial age estimation and doesn't have so much time to conduct a series of trials seeking for the optimal standard deviation: he or she only chooses from the most commonly used value 1-5. In this case, our proposed SDM-LDL with Strategy 1,2,5 are more likely to obtain optimal results or the results approaching the optimal MAE. What's more, the SDM-LDL with Strategy 1 and 2 shows stability with their more stable and flatter fold

Table 4.5: The impacts of different values of N (2-10) on MAE using the proposed SDM-LDL algorithms with Strategy 1 and 2 when age label distribution is generated as Gaussian-like (σ from 1 to 10; for every σ , the optimal MAEs are marked in bold)

σ		1	2	3	4	5	6	7	8	9	10
MAE(Str 1)	N=2	5.978	5.725	5.500	5.384	5.414	5.286	5.339	5.264	5.279	5.278
	N=3	5.741	5.592	5.397	5.327	5.358	5.275	5.303	5.221	5.237	5.226
	N=4	5.569	5.528	5.362	5.301	5.339	5.258	5.306	5.205	5.222	5.229
	N=5	5.508	5.401	5.335	5.252	5.263	5.233	5.248	5.172	5.193	5.201
	N=6	5.484	5.362	5.286	5.226	5.278	5.206	5.246	5.196	5.196	5.208
	N=7	5.511	5.307	5.233	5.195	5.219	5.189	5.209	5.181	5.173	5.192
	N=8	5.471	5.285	5.245	5.200	5.224	5.199	5.211	5.213	5.188	5.254
	N=9	5.531	5.251	5.219	5.159	5.190	5.186	5.192	5.207	5.189	5.223
	N=10	5.627	5.263	5.241	5.196	5.205	5.241	5.245	5.260	5.226	5.305
MAE(Str 2)	N=2	6.025	5.741	5.501	5.385	5.413	5.287	5.339	5.265	5.280	5.278
	N=3	5.870	5.623	5.408	5.328	5.359	5.274	5.303	5.222	5.238	5.228
	N=4	5.782	5.568	5.373	5.300	5.337	5.257	5.303	5.203	5.221	5.229
	N=5	5.724	5.473	5.343	5.256	5.266	5.233	5.247	5.171	5.191	5.200
	N=6	5.689	5.422	5.296	5.223	5.270	5.199	5.238	5.191	5.191	5.204
	N=7	5.647	5.367	5.251	5.194	5.215	5.181	5.201	5.174	5.168	5.187
	N=8	5.621	5.323	5.224	5.180	5.200	5.175	5.192	5.195	5.175	5.240
	N=9	5.599	5.284	5.203	5.143	5.161	5.153	5.166	5.185	5.171	5.208
	N=10	5.588	5.246	5.180	5.127	5.139	5.170	5.189	5.216	5.193	5.273

lines as evidence: throughout the range $\sigma \in [1, 10]$, the MAE of Strategy 1 and 2 always remains a relatively low value with no sharp fluctuations. Note that for SDM-LDL with Strategy 1 and 2, only best results are presented in the table along with the value of N at that time, so one natural question comes: when N traverses from 2 to 10, what the variation trend of MAE will be? Or how the value of N influences the estimation performance?

Table 4.5 shows the results of different N's (from 2 to 10) impacts on MAEs using the proposed SDM-LDL algorithms with Strategy 1 and 2. As can be seen, for both Strategy 1 and 2, the MAEs generally tend to be smaller when N is bigger, no matter what value of σ is. For example, for Strategy 2, when N takes the value of relatively big integers, like 7, 9 and 10, the majority of results are optimal whatever σ is. Therefore, normally taking the value greater than 5 for N will get better performance.

Table 4.6: MAEs of the compared original LDL algorithm without decision-making strategy and the proposed SDM-LDL algorithms with different strategies on the condition that age label distribution is generated as triangle (bottom length from 4 to 30; for S-strategy 1 and 2, best results are shown in the table along with the value of N then; for every bottom length, the optimal MAEs are marked in bold)

Bottom length	4	6	8	10	12	14	16
MAE(Ori)	6.310	6.144	5.994	5.856	5.623	5.574	5.556
MAE(Str 1)	5.536 (N=7)	5.398 (N=7)	5.338(N=8)	5.266(N=7)	5.199(N=9)	5.175(N=10)	5.206(N=10)
MAE(Str 2)	5.903(N=10)	5.617(N=10)	5.397(N=10)	5.252(N=10)	5.191(N=10)	5.156(N=10)	5.159(N=10)
MAE(Str 3)	6.247	6.038	5.887	5.178	5.551	5.440	5.438
MAE(Str 4)	6.242	6.041	5.890	5.723	5.549	5.439	5.436
MAE(Str 5)	5.890	5.582	5.318	5.127	5.016	4.953	4.928
Bottom length	18	20	22	24	26	28	30
MAE(Ori)	5.486	5.366	5.458	5.492	5.564	5.491	5.479
MAE(Str 1)	5.188(N=9)	5.170(N=9)	5.187(N=9)	5.228(N=7)	5.219(N=9)	5.250(N=9)	5.244(N=7)
MAE(Str 2)	5.155(N=10)	5.148(N=9)	5.153(N=10)	5.169(N=10)	5.196(N=10)	5.217(N=9)	5.239(N=9)
MAE(Str 3)	5.360	5.234	5.292	5.312	5.295	5.277	5.234
MAE(Str 4)	5.353	5.229	5.284	5.301	5.290	5.259	5.231
MAE(Str 5)	4.925	4.938	4.966	5.012	5.081	5.156	5.208

Triangle distribution Just as Fig. 4.2(b) shows, unlike Gaussian-like distribution, triangle distribution only takes advantage of partial age labels which are located on both sides near the chronological age; in other words, it only allocates the description degree to the “neighboring” age labels of the authentic age while description degrees of other age labels keep 0. Furthermore, the description degree reaches to the peak value at the chronological age, of which it drops linearly and symmetrically on both sides.

When generating the triangle age label distribution, the controlled variable is the bottom length. Figure 4.5 demonstrates three examples of different bottom length, i.e., 4, 6 and 8. As can be seen from this figure, when the bottom length becomes greater, the description degree of the chronological age is smaller and the differences between the chronological age and other neighboring age labels are smaller. Intuitively from the “shape”, the triangle becomes flatter with the increase of the bottom length (from Fig. 4.5(a) to (b)-(c)). In experiment, the bottom length is given a broad value range of $\{2, 4, 6, \dots, 30\}$ to examine its impact and variation tendency for the estimated age.

Table 4.6 shows the MAE results with different bottom length (from 4 to 30) using the original LDL and the proposed SDM-LDL algorithms with all strategies. As can be seen

(a) Bottom length of 4

(b) Bottom length of 6

(c) Bottom length of 8

Figure 4.5: Different bottom length when generating the triangle age label distribution for the chronological age a

clearly, when firstly generating the age label distribution as triangle, the proposed SDM-LDL algorithm with Strategy 5 almost outperforms all the other strategies, especially when the bottom length is not very small; also, similar to the conclusion from Gaussian-like age label distribution, vertically for every value of the bottom length, MAEs of all the proposed strategies are smaller without exception, indicating the validity and superiority of the proposed SDM-LDL in the case of triangle distribution. On the other hand, Figure 4.6 can also be used as a corroboration of this conclusion: for almost all the range of the bottom length, the fold line of Str 5 remains in the lowest except from 4 to 8, where it only lags behind the fold line of Str 1. Moreover, the proposed SDM-LDL algorithms with all strategies are superior to the original LDL algorithm with all fold lines falling below the original line. As for the triangle age label distribution, when the bottom length becomes greater, the MAE also has the trend of firstly decreasing sharply and then increasing slightly (or becoming stable). Moreover, the SDM-LDL algorithms achieve better and more stable results than the original LDL, especially for Strategy 1 and 2.

Figure 4.6: The line chart (variation tendency) of MAEs with respect to different bottom length for the original LDL algorithm and the proposed SDM-LDL algorithms with different strategies when age label distribution is generated as triangle

Table 4.7: The impacts of different values of N (2-10) on MAE using the proposed SDM-LDL algorithms with Strategy 1 and 2 when age label distribution is generated as triangle (bottom length from 4 to 30; for every value of bottom length, the optimal MAEs are marked in bold)

Bottom length		4	6	8	10	12	14	16	18	20	22	24	26	28	30
MAE(Str 1)	N=2	5.861	5.802	5.833	5.640	5.549	5.486	5.477	5.453	5.373	5.445	5.438	5.444	5.414	5.436
	N=3	5.656	5.646	5.692	5.524	5.456	5.402	5.402	5.385	5.358	5.395	5.398	5.409	5.364	5.403
	N=4	5.640	5.578	5.523	5.449	5.421	5.309	5.384	5.374	5.326	5.375	5.351	5.382	5.365	5.369
	N=5	5.553	5.463	5.444	5.338	5.324	5.302	5.300	5.293	5.247	5.289	5.276	5.304	5.324	5.326
	N=6	5.542	5.423	5.388	5.323	5.260	5.269	5.309	5.261	5.258	5.250	5.245	5.305	5.314	5.313
	N=7	5.536	5.398	5.341	5.266	5.217	5.229	5.258	5.248	5.207	5.199	5.228	5.250	5.261	5.244
	N=8	5.673	5.438	5.338	5.276	5.221	5.208	5.234	5.227	5.201	5.209	5.230	5.258	5.258	5.259
	N=9	5.774	5.437	5.344	5.275	5.199	5.181	5.237	5.188	5.170	5.187	5.234	5.219	5.250	5.268
	N=10	5.845	5.485	5.386	5.307	5.231	5.175	5.206	5.211	5.222	5.215	5.235	5.259	5.314	5.300
MAE(Str 2)	N=2	6.071	5.888	5.838	5.663	5.554	5.485	5.483	5.451	5.371	5.443	5.439	5.444	5.416	5.434
	N=3	5.984	5.793	5.729	5.567	5.481	5.417	5.414	5.388	5.357	5.396	5.398	5.410	5.367	5.402
	N=4	5.964	5.743	5.634	5.496	5.456	5.339	5.391	5.374	5.319	5.375	5.353	5.383	5.366	5.368
	N=5	5.941	5.706	5.562	5.431	5.384	5.332	5.321	5.301	5.244	5.296	5.283	5.312	5.327	5.327
	N=6	5.929	5.673	5.496	5.370	5.330	5.294	5.313	5.265	5.246	5.251	5.244	5.301	5.308	5.309
	N=7	5.919	5.648	5.457	5.334	5.289	5.263	5.270	5.250	5.207	5.202	5.223	5.247	5.256	5.242
	N=8	5.912	5.640	5.438	5.309	5.252	5.229	5.229	5.218	5.184	5.194	5.207	5.236	5.237	5.241
	N=9	5.908	5.629	5.416	5.280	5.221	5.193	5.208	5.176	5.148	5.166	5.199	5.199	5.217	5.239
	N=10	5.903	5.617	5.397	5.252	5.191	5.156	5.159	5.155	5.151	5.153	5.169	5.196	5.242	5.243

Table 4.7 shows the variation trend of MAE with N from 2 to 10 using Strategy 1 and 2 of the proposed SDM-LDL on the condition that age label distribution is generated as triangle. For both Strategy 1 and 2, small values of N (2-6) yield inferior performance against bigger values (7-10). Particularly for Strategy 2, whatever the bottom length is, the best performance always happens when N equals to 9 and 10. As a conclusion, for Strategy 1 and 2, if the age label distribution is initially generated as the triangle style, then it's better to allocate N a relatively big value (7-10) for better estimation performance, especially for Strategy 2, where N should be given the value of 9 and 10.

Multi-label distribution with equal description degrees As Fig. 4.2(c) shows, in this situation, description degrees are evenly distributed to the chronological age and the adjacent age labels. So when initially generating age label distribution in this style, the controlled variable is the number of age labels. Figure 4.7 shows the example of different number of age labels: if the number of age labels is 5 and 7, then the description degrees for all involved age labels are $1/5$ and $1/7$ respectively. In other words, when the number of age labels enlarges, the description degrees for all involved age labels will reduce

(a) Number of age labels: 5

(b) Number of age labels: 7

(c) Number of age labels: 9

Figure 4.7: Different number of age labels when generating age label distribution as multi-label with equal description degrees for the chronological age a

Table 4.8: MAEs of the compared original LDL algorithm without decision-making strategy and the proposed SDM-LDL algorithms with different strategies on the condition that age label distribution is generated as multi-label with equal description degrees (number of age labels from 3,5,7,... to 15; for Strategy 1 and 2, best results are shown in the table along with the value of N then; for every number of age labels, the optimal MAEs are marked in bold)

Number of labels	3	5	7	9	11	13	15
MAE(Ori)	6.312	5.998	5.830	5.700	5.698	5.732	5.690
MAE(Str 1)	5.541 (N=5)	5.343 (N=8)	5.219 (N=7)	5.175(N=8)	5.169(N=7)	5.137(N=9)	5.139(N=9)
MAE(Str 2)	5.979(N=10)	5.638(N=10)	5.312(N=10)	5.219(N=10)	5.145(N=10)	5.127(N=10)	5.132(N=10)
MAE(Str 3)	6.242	5.952	5.742	5.645	5.540	5.537	5.530
MAE(Str 4)	6.245	5.957	5.737	5.650	5.542	5.548	5.528
MAE(Str 5)	5.974	5.626	5.263	5.143	5.017	5.002	5.009

accordingly. In experiment, to explore its influence for the estimation performance, the number of age labels is allocated as 3, 5, 7, ..., 15.

Table 4.8 demonstrates the estimation performance of all compared LDL-based algorithms with different number of age labels (from 3 to 15) when initially generating age label distribution as multi-label with equal description degrees. Again, all the proposed methods outcompete the existing original LDL algorithm. Besides, when the number of age labels is small (3-7), Strategy 1 obtains the best results; however, as the number of age labels gets bigger (9-15), Strategy 5 shows advantages over the others in performance. Figure 4.8 also supports this argument: when the number of age labels are small (3-7), the lowest fold line is Strategy 1; when the coordinates of the horizontal axes are larger, the lowest line becomes Strategy 5. In the end, the fold lines of all methods tend to be stable. Also, Strategy 1 shows more stability with the evidence of remaining the relatively low MAE for different number of age labels.

Figure 4.8: The line chart (variation tendency) of MAEs with respect to different number of age labels for the original LDL and the proposed SDM-LDL with different strategies when age label distribution is generated as multi-label with equal description degrees

Table 4.9: MAEs of the compared three different patterns in generating age label distribution (the controlled variables in each pattern are traversed within the whole given range and best results are reported; for every row in the table, the optimal MAEs are marked in bold)

	Gaussian-like	Triangle	Multi-label with equal description degrees
MAE(Ori)	5.320	5.366	5.69
MAE(Str 1)	5.159	5.170	5.137
MAE(Str 2)	5.127	5.148	5.127
MAE(Str 3)	5.112	5.178	5.530
MAE(Str 4)	5.089	5.229	5.528
MAE(Str 5)	4.920	4.925	5.002

As to different N’s impacts on the final estimated age for Strategy 1 and 2, the conclusion is the same as in “Triangle distribution” part. So the detailed experimental data are omitted for simplicity.

Overall comparison of the proposed SDM-LDL algorithms and other popular algorithms The preceding parts detailedly discuss the respective estimation results when age label distribution is initially generated by three different patterns, namely Gaussian-like, triangle and multi-label with equal description degrees. In the following, these three patterns are compared as a whole to see which one is the best choice when generating the age label distribution.

As can be apparently seen from Table 4.9, for almost all LDL-based methods, including original LDL and the proposed SDM-LDL with different strategies, generating Gaussian-like age label distribution yields highest precision and thus achieve best performance, followed by triangle distribution; the worst choice is using multi-label distribution with equal description degrees.

Table 4.10 demonstrates the performance of proposed SDM-LDL algorithm with different strategies compared with other existing popular facial age estimation algorithms [93][94][95][96][65][62][97][38][6][98][99][47] and the conventional single classification methods SVM and kNN. From the table, our proposed SDM-LDL with all strategies

outperforms the original LDL; when compared with other existing popular methods, SDM-LDL can also achieve relatively good or even superior performance, which proves SDM-LDL’s validity and advantages.

Table 4.10: MAEs of different facial age estimation algorithms

Method	MAE
SDM-LDL(Str 1)	5.137
SDM-LDL(Str 2)	5.127
SDM-LDL(Str 3)	5.112
SDM-LDL(Str 4)	5.089
SDM-LDL(Str 5)	4.920
Original LDL	5.32
Hierarchical Framework [93]	4.97
LBP Kernel Density Estimate [94]	5.09
Local radon Features [95]	6.18
Cumulative Attribute SVR [98]	4.67
Grassmann Manifold [96]	5.89
Hierarchical Model [99]	4.89
Shape-based age estimation [65]	6.2
Regression using a learned distance metric [62]	5.04
Bio-inspired Features [47]	4.77
Synchronized Submanifold Embedding [97]	5.21
Manifold Learning and Locally Adjusted Robust Regressor [38]	5.07
Facial Aging Patterns (AGES) [6]	6.77
SVM	7.25
kNN	8.24

4.5 Concluding Remarks

This chapter proposes a novel strategic decision-making algorithm with a series of strategies for label distribution learning in facial age estimation. All strategies are specially designed to suit the characteristics of aging problem. Comprehensive experiments for three different kinds of age label distribution (Gaussian-like, triangle and multi-label with equal description degrees) prove the validity, superiority and stability of the pro-

posed SDM-LDL algorithms against the original LDL and other existing facial age estimation algorithms. Also, the respective advantages and properties for each strategy in SDM-LDL are summarized. Further experiments find out the performance's variation tendency within each kind of age label distribution so that the most suitable value (or range) of those uncertain variables are obtained.

Chapter 5

Conclusions and Future Research

5.1 Conclusions and Discussions

This thesis focuses on studying the problem of human’s facial age estimation. Several approaches for improving the age estimation performance are introduced and the detailed experiments and analysis are conducted for evaluating the validity and superiority of the proposed approaches.

First, traditional methods are almost originated from solving complicated mathematical optimization problems and thus consume huge quantities of time in the training process. To refrain from such algorithm complexity while maintaining a high estimation accuracy, a multi-feature extreme ordinal ranking machine (MFEORM) is proposed for facial age estimation, which combines multi-feature space, age’s natural characteristics of ordinal information and ELM’s rapid learning speed. Also, the Extreme Learning Machine (ELM) is first introduced into the field of facial age estimation. Experimental results clearly demonstrate that the proposed approach can sharply reduce the runtime (even up to nearly one hundred times faster) while achieving comparable or better estimation performances than the state-of-the-art approaches. The inner properties of MFEORM are further explored with more advantages.

Second, ordinal-ranking-oriented age estimation algorithms achieve top performance in facial age estimation. However, further experiments show that this approach suffers

from its ideal ranking rule, which sometimes would cause unnecessary estimation deviations and degrade the performance. Consequently, two approaches are proposed with new ranking rules which minimize the accidental deviations of binary classifiers and tactfully combine the accuracy and obtained label in each binary classification substep for ranking criteria. Also, ELM is used as the binary classifier, taking full advantages of its high learning speed and accuracy. Experimental results from public datasets are presented to demonstrate that the proposed algorithms reduce the mean absolute error and improve age estimation performance while reducing the runtime significantly.

Third, label distribution learning is among the state-of-the-art methodologies in facial age estimation. It takes the age of each facial image instance as a label distribution with a series of age labels rather than the single chronological age label which is commonly used. However, this methodology is deficient in its simple decision-making criterion: the final predicted age is only selected at the one with maximum description degree. In many cases, different age labels may have very similar description degrees. Consequently, blindly deciding the estimated age by virtue of the highest description degree would miss or neglect other valuable age labels which may contribute much to the final predicted age. Driven by this point, a strategic decision-making label distribution learning algorithm (SDM-LDL) is proposed with a series of strategies specialized for different types of age label distribution. Experimental results from the most popular aging face database FG-NET show the superiority and validity of all the proposed strategic decision-making learning algorithms over the existing label distribution learning and other single-label learning algorithms for facial age estimation. The inner properties of SDM-LDL are further explored with more advantages.

Despite the above merits of this thesis, we should recognize the limitations of the proposed approaches. At present, it is still a difficult problem of facial age estimation to estimate the age of a person who uses cosmetics or had plastic surgery. Because for these two conditions, the facial skin texture - a facial age estimation system mainly

relies on - has been changed. To the best of my knowledge, till now there is still no publication that can solve this problem. Even relevant publications are very few and those papers' conclusion is only to confirm that the use of cosmetics has influence on facial age estimation algorithms, which agrees with our common expectation. Actually several potential directions can be explored to decrease the age estimation deviations caused by cosmetics or plastic surgery. 1) Makeup detection methods can be firstly utilized to determine whether a facial image is with makeup or not. After that, two different classifiers are respectively to be trained for the corresponding class. Since the facial features (texture, color and even shape) are different between the facial image with and without makeup, the dedicated two classifiers can potentially learn different correlations between facial features and age. In this way, the age estimation deviation can be mitigated. 2) Deep Neural Network (DNN) has the powerful ability to learn from a more abstract level with the layers' increasing. So DNN with more layers can potentially represent the subtle variance with or without makeup/plastic surgery and can thus improve the performance when makeup/plastic surgery exists. 3) Up to now, the facial aging databases with makeup or plastic surgery are still not available. So we can collect a new database with the instances of makeup or plastic surgery faces. This new database will help expedite not only our work (as training and testing sets), but also the computer vision community (as benchmark) to improve the age estimation performance in the presence of makeup or plastic surgery.

In fact, there exist some other human face databases that can be used for facial age estimation, like MORPH (Craniofacial Longitudinal Morphological Face Database), FRGC (Face Recognition Grand Challenge dataset), GROUPS database and so on. The reason why we choose FG-NET database can be summarized into the following points. Firstly, FG-NET is the most popular dataset in the field of facial age estimation. That means almost all facial age estimation approaches will report estimation results for FG-NET; in other words, FG-NET is a universal benchmark for convenient comparisons in facial age

estimation. For other databases, however, many approaches don't have estimation results so sometimes it's rather difficult to compare based on those databases. Particularly for FRGC, to the best of my knowledge, only very few publications on age estimation have reported the experimental results in this database. Secondly, MORPH has its own deficiency. For example, MORPH 1 (the old version) contains only 515 photographs, which is much less than FG-NET; MORPH 2 (the new version and frequently-used MORPH) contains few facial images with the age under 20 and even no image which is younger than 15: this will lead to the deviation in the training and testing phase. Another example is GROUPS database. Its defect is that its age annotations are grouped into several age intervals instead of single age labels, which is not appropriate for training accurate age estimation labels. I conjecture these are also the reasons why this article [Panis et al (2015). Overview of research on facial ageing using the FG-NET ageing database. IET Biometrics] involves FG-NET rather than other databases.

Furthermore, almost all the existing public facial aging databases share a common defect: the number and proportion of Asians is very small. As we know, for a machine learning based task, the composition of the training set will have a big impact on the overall performance. Consequently, in the real application scenarios of Asian countries, facial age estimation systems trained by these databases will have a larger deviation than those used in non-Asian countries. So a potential method to improve this point is to collect a facial aging dataset with an appreciable amount of Asians. Also, this dataset should include all ages and the number of people in each age should be evenly distributed; for each person in the dataset, mugshots from more ages are wanted. I believe that once completed, this dataset can avoid the existing databases' disadvantages and achieve higher and more stable estimation performance.

5.2 Future Works

So far, several human facial age estimation methods and frameworks has been presented. However, further potential improvements can still be explored and some problems remain

open, which are summarized in the following points:

- In recent years, several other neural network architectures which are similar to the extreme learning machine have been reported, such as the feedforward neural networks with random weights, Random Vector Functional Link Networks (RVFL) and so on. Among them, some works [100][101][102] have achieved quite satisfying performance. So next, these methods are to be explored and applied to Chapter 3's framework for comparing the overall age estimation performance.
- At present almost all popular aging databases mainly contain facial images with the neutral expression. However, in many practical application scenes, age estimation must be done with different expressions, like happy, angry, sad and so on. In this case, how to estimate the individual's age with high accuracy remains a problem. What's more, if multiple facial expressions are taken into account, new databases from the psychology society with the ground truth of both age and expression should be further discovered (because there are few adequate databases in the computer vision field). In fact, a limited number of related papers have been published [103][104], but almost none can get satisfying results; more specifically, the MAEs and accuracy that they obtain are undesirable (normally around 10 years). So next I am going to explore how to decrease the MAE and improve the estimation performance in the scenarios with multi-expression facial images. Based on empirical data, the "MAE within 5 years" is a relatively acceptable result.
- As is known to all, deep learning (DL) is now sweeping almost all machine learning fields and applications because of its novel network structure and superior performance, in which the deep convolutional neural network (DCNN) has been utilized in the areas of speech analysis and image recognition. Recent related works in these two years mainly formulate their models on existing popular DCNN architectures like VGG-16, which is later fine-tuned on their respective datasets to get

the outputs of deep neural network. However, as can be seen from the tendency of the annual ImageNet Large Scale Visual Recognition Competition (ILSVRC), the chance of using a single DCNN framework to get the optimal performance is becoming less and less while the ensemble methods can achieve better results. So in the future, I am going to explore the ensemble/cascaded framework in the application of facial age estimation. Several possible directions can be: 1) to train a cascaded DCNN framework firstly for coarsely classifying a person into corresponding age interval and secondly for regression to get the exact age label in that interval; 2) to train multiple DCNNs with different data distributions and use ensemble methods to achieve the estimated age.

- Since this year, apparent age estimation has become a more and more popular branch of age estimation, where each facial image is recognized and labelled by multiple persons. In this case, ground truth of the age is the mean value and the standard deviation represents the uncertainty. In other words, apparent age estimation focuses more on the face itself regardless of the chronological age (authentic age) that one facial image stands for. Apparently, in the real world, it has a lot of potential applications. Besides, because of the booming ChaLearn LAP competition on Apparent Age Estimation [105], a public dataset is made using more than 7000 facial images with apparent age annotations, which provides an appropriate benchmark for apparent age estimation. So one of prospective research paths can be emphasized in this field, figuring out the pros & cons of top winners' algorithms in the ChaLearn LAP AAE competition and developing our method to further improve the performance of apparent age estimation.

Appendix A

Salient Object Detection with Spatiotemporal Background Prior for Video

A.1 Introduction

In normal conditions, the research of facial age estimation is conducted in a relatively mild environment: almost all instances (images) in the existing databases only include one human face without the disturbance of other backgrounds or complex scenes. So it is easy to directly do the pre-processing for them, such as cropping, face alignment, normalization and so on. But now if we consider real-world application scenarios, especially those requiring real-time processing, the facial images that a camera captures will contain various background information, which can be seen as the serious noise for the age estimation problem. In this case, the total data's dimensionality or size will be very huge, which is difficult to realize real-time. So in these practical applications, how to firstly detect the facial region without huge data dimensionality involved remains a problem. Salient object detection provides a feasible idea for the pre-processing of raw data and can act as a powerful tool for dimension reduction of image data. It can be a prior to be fed into the face detection system. In this way, before the image data enter into the "face detection" step, they have already been gotten rid of background information and only contain the salient object part. According to empirical knowledge, salient

objects only account for a small proportion of an image, most of which are occupied by the background. So after salient object detection, the raw image data will remain a small part and the dimensionality will be sharply reduced for subsequent processing.

Driven by this idea, we are led to the direction of salient object detection. Salient object detection has been a very popular topic in recent years, which is evolved from visual attention research for images and videos. Different from traditional saliency research which refers to the prediction of eye fixations, salient object detection aims to find visual objects by measuring the dissimilarity of each region based on the fact that there exists statistic divergence between foreground region and background region. In application, salient object detection can be utilized in plenty of fields, such as image segmentation, image localization, image compression and so on. Particularly, this appendix focuses on salient object detection.

Many methodologies have been proposed to detect objects in an image or video, such as [106][107][108][109][110][111][112]. Instead of the traditional center-surrounding weight mechanism, more and more methods employ the concept “background prior” [113], which can be regarded as a set of cues or templates for the background and shows a higher performance against conventional salient object detection methodologies. At present, most of the published papers treat the image boundary as background priors based on the empirical conclusion that for most images, the salient object will not appear on the boundary. However, this is not always the case: in certain circumstances, the image boundary is vulnerable for background priors. As can be seen in Fig. A.1(a), red regions of the first image represent the image boundary as background priors. The second and third images are saliency maps generated respectively by two state-of-the-art salient object detection approaches using image boundary as background priors. From these two images, we can see that the salient object cannot be completely detected and also the background is mistakenly regarded as components of the salient object. However, in Fig. A.1(b), we manually select a few patches from the background as background priors, which highlights uniform detection results from the background.

Figure A.1: What are good background priors?

So what can be called good background priors? As Fig. A.1 shows, good background priors possess the following two characteristics:

- (i) *As accurate as possible*. In other words, background priors should contain minimum or even no contents of the foreground.
- (ii) *As sufficient as possible*. That means background priors ought to include as much background as we can.

The most relevant field to our topic is video object segmentation via background subtraction. Different from background subtraction, background priors don't need to cover all background region.

Seen from the above analysis, the existing methods still have certain space for further improvement because of their respective defects. In this appendix, according to the aforementioned two criteria, a new salient object detection framework for videos is proposed combining both spatial and temporal background priors. Specifically, since the background part is rigid for most natural videos, the temporal background priors are obtained based on the background homography between frames. The homography is estimated on point correspondences by means of SIFT flow [114] in two frames. The spatial background priors are achieved utilizing the algorithm proposed in [115], which are combined with abovementioned temporal background priors to generate the spatiotemporal background priors. For saliency value computation, a specially-designed motion-based graph is introduced to highlight uniform and accurate salient objects. In summary, the contributions of this appendix are as follows.

- (i) Particularly for videos, besides image boundary, the temporal information (relation and difference among adjacent frames in chronological order) is also a significant hint, which needs to be considered especially for those videos that include complex scenes. Inspired by this idea, the temporal background priors are proposed as the complement of original definition “background prior” (in essence a more appropriate name should be “spatial background priors”). Combining these two concepts, the “universal” spatiotemporal background priors is proposed to take full advantage of video information, by which an efficient and reliable salient object detection method for videos is built.
- (ii) When producing final saliency maps, more and more graph-based models were proposed in recent years, in which graph construction is vitally important. However, they only take into account the spatial information of an image or video. This is not sufficient for describing the relationship among different nodes of the whole graph. In this work, both an appearance graph and a novel motion graph are

constructed, which simultaneously considers spatial and temporal relationship of different homogeneous objects.

- (iii) The proposed framework is novel and is first introduced into the field of salient object detection for videos (video saliency).

In the field of salient object detection for images, Liu et al. [106] proposed a salient object detection method based on conditional random field (CRF) to fuse both local and global features. But it generates the bounding box indicating the location of the salient object, failing to give precise segmentation results. Achanta et al. [107] proposed a simple and efficient method for salient region detection in frequency domain, utilizing low-level features. Cheng et al. [108] proposed a regional contrast based method for saliency extraction, in which they incorporate color histogram cue, spatial cue and center bias. Both [107] and [108] aim to obtain fine-boundary salient objects. Later, in [116], Jiang et al. claimed an iterative energy minimization model to get the shape prior and saliency map. Wei et al. [113] used the geodesic distance to measure the difference between one superpixel and background priors as the saliency value of this superpixel. Yang et al. [117] employed the manifold ranking technique to obtain saliency values by computing the dissimilarity of background priors. For more complex scenes of images, Yan et al. [118] built a multi-scale propagation model that covers the shortage of single scale based methods. Note that [113, 116–118] are still based on region segmentation and adopt more features and contrast metrics compared to [108]. Recently, image boundary as background priors becomes a popular cue to find salient regions [117][113][119][115][120][121], that is because salient objects always locate at or near the center of images. It can extract salient objects with precise contours compared to traditional center bias mechanism. Jiang et al. [119] proposed an absorbing Markov chain model for saliency detection, in which the time that each patch transits to image boundary is regarded as the measurement of saliency degree. In [115], Zhu et al. overcame the deficiency of image boundary as background

priors by providing the definition of boundary connectivity. It is particularly suitable when a proportion of salient objects is located at the border of an image. In [120], Cellular Automata was introduced for saliency detection to refine the background-based map which indicates the differences between every region and image boundary.

Different from salient object detection for images, temporal information and more context exist in the field of video saliency detection. Itti and Baldi [122] substantially defined surprise, namely visual attention, from the perspective of Bayesian framework. Their approach can be applied in different spatiotemporal scales, modalities and levels of abstraction. In [123], a visual attention model on spatiotemporal domain was proposed, in which color histograms contrast and motion contrast are computed respectively as spatial and temporal cues. Guo et al. [124] suggested a phase spectrum based saliency detection algorithm for images and videos. Using their method, salient areas can be calculated by Fourier transform. Kim et al. [125] combined the features of edge, color orientations and temporal gradients to compute the spatiotemporal saliency map. Rudoy et al. [126] used a sparse candidate set containing three types of candidates (static, motion and semantic candidates, respectively) to generate the saliency map. Fang et al. [127] proposed a video saliency detection model in compressed domain, where four kinds of features are extracted from DCT coefficients and motion vectors of a video. It is worth noting that the aforementioned methods mainly focus on the prediction of eye fixation. Rahtu et al. [109] proposed a saliency-guided segmentation approach for images and videos. In their method, some low-level features are used to obtain saliency maps and conditional random field (CRF) model is built to get corresponding segmentation results. Zhong et al. [128] proposed a method for temporal saliency based on dynamic consistency of optical flow, which is then combined with a classical spatial saliency method to create a spatio-temporal attention model. Luo and Yuan [129] simultaneously detected and track salient object, in which a path of moving object can be found according to saliency maps. Wang et al. [130] proposed a framework of video object segmentation based on video

saliency detection. For video saliency detection part, edge and motion information is employed to obtain the initial objects. Then the final saliency map is obtained by using the coherence between frames.

Unsupervised video object segmentation is another topic that directly obtains pixel-level objects and is most related to video object saliency. Zhang [131] proposed a layered directed acyclic graph (DAG) based video segmentation model using object proposal as inputs. Brox [110] put forward a video segmentation method using long term point trajectories from the perspective of optical flow. In [132], key-segment was introduced for clustering in order to predict salient objects in videos. Li et al. [133] presented a long-term appearance model by iterative update to identify the video object in future frames. Papazoglou [134] first obtained initial foreground using optical flow and then utilize four constraints to label the final video objects.

In recent years, co-saliency is an extension of saliency and used to discover the common saliency on multiple images, but the distinction from video saliency is that the images may come from different scenes. Co-saliency can be seen as a combination of similar objects identification and object detection tasks, which was first put forward in [135]. Later Li et al. [136] proposed a co-saliency model combining single-image and multi-image saliency maps. For multi-image saliency, hierarchical representation of image pairs is utilized to build a co-multilayer graph. Fu et al. [137] devised three cues, namely contrast, spatial and corresponding, to measure the saliency of each region. Among them, the corresponding cue creates connectivities of multiple images. In [138], a hierarchical segmentation based co-saliency model was presented, integrating intra-saliency within images and global similarity from other images. Zhang's [139] co-saliency framework used a Bayesian framework to fuse intra-image contrast and intra-group consistency.

A.2 Our approach

Fig. A.2 shows the overview of the proposed salient object detection method with spatiotemporal background priors for videos. Initially, the pre-processing step is conducted

Figure A.2: Framework of the proposed approach

to get superpixels as the basic unit of our approach and SIFT flow [114]. Next, to estimate background priors, first the SIFT flow of all frames is used to obtain temporal background priors; spatial background priors are further achieved utilizing the algorithm proposed in [115]. The above two types of background priors are combined to generate the spatiotemporal background priors. In saliency map generation, with these spatiotemporal background priors, the information of color and motion is used to compute appearance saliency maps and motion saliency maps respectively. These two types of saliency maps are then fused to produce final saliency maps.

A.2.1 Temporal background priors estimation

A.2.1.1 Initial temporal background priors estimation using SIFT flow and background homography

SIFT flow is a relatively new method to align an image to its nearest neighbors in a large image corpus containing complex scenes [114]. It can match densely sampled, pixel-wise SIFT features between two images with the preservation of spatial discontinuities. Compared with traditional optical flow methods, SIFT flow does not need the assumptions of brightness constancy and piecewise smoothness of the pixel displacement field. It is robust to sudden illumination variation and scene changes. So SIFT flow is used to obtain the point correspondence between two frames. With the help of SIFT flow, the motion trend for each point can be estimated. Mathematically, suppose a video $V = \{F^1, F^2, \dots, F^N\}$ is given, where F^t is the t^{th} frame of the video and N is the total

number of frames. For each pixel $p_i^j \in F^j$ where i is the pixel index, its correspondence $p_i^k \in F^k$ can be found by using SIFT flow between F^j and F^k .

Next, the homography for background will be estimated based on the assumption that the background part is rigid for most natural videos. Essentially, the background homography actually reflects the motion tendency of the majority of points in the scene from one frame to another adjacent frame. In detail, suppose the coordinate of p_i^j is (x_i^j, y_i^j) and that of its correspondence p_i^k is (x_i^k, y_i^k) . The background homography $H_{j,k}$ is a 3×3 matrix for approximating the planar transformation between two adjacent frames F_j and F_k , which can be expressed by

$$\begin{bmatrix} \hat{x}_i^k \\ \hat{y}_i^k \\ t \end{bmatrix} = H_{j,k} \begin{bmatrix} x_i^j \\ y_i^j \\ 1 \end{bmatrix} \quad (\text{A.1})$$

where $\hat{p}_i^k = [\hat{x}_i^k, \hat{y}_i^k, t]^T$ is the projection of p_i^j in the form of homogeneous coordinates. In order to obtain the appropriate $H_{j,k}$ that can represent the transformation of background, a criterion is provided that minimizes the error between $\hat{p}_i^k$ and p_i^k for all pixels. This process can be formulated as

$$ERR(H_{j,k}) = \sum_i^M \|p_i^k - \hat{p}_i^k\|_2 \quad (\text{A.2})$$

where M is the number of pixels in a frame. Here RANSAC algorithm is adopted on point correspondences. Specifically, every time four pairs of point correspondences are randomly selected and $H_{j,k}$ and $ERR(H_{j,k})$ are estimated. The above process is repeated for one hundred times to find the optimal $H_{j,k}$ that has minimal $ERR(H_{j,k})$. Then the estimating error of p_i^j between F_j and F_k would be

$$err_{j,k}(p_i^j) = \begin{bmatrix} x_i^k \\ y_i^k \\ 1 \end{bmatrix} - \begin{bmatrix} \hat{x}_i^j/t \\ \hat{y}_i^j/t \\ 1 \end{bmatrix} \quad (\text{A.3})$$

where t is a scale factor; $\hat{x}_i^j/t$ and $\hat{y}_i^j/t$ are the estimated coordinates of point correspondence for p_i^j . A threshold denoted as $thres_1$ is further set for 2-norm of this error. The

empirical knowledge shows that for videos, foreground and background have opposite motion tracks and that the background region is normally much larger than the foreground region. So if the aforementioned error $err_{j,k}(p_i^j)$ is less than the threshold, it means the motion tendency of point p_i^j agrees with that of the background and p_i^j can thus be regarded as the initial temporal background prior. In the same way, if the error is bigger than the threshold, p_i^j tends to have different movement trajectory from the background and is therefore more likely to be part of the foreground. This process can be formulated as

$$\underbrace{BP_{T(initial)}(p_i^j)}_{\text{pixel level}} = \begin{cases} 0, & \text{if } \|err_{j,k}(p_i^j)\|_2 \geq thres_1 \\ 1, & \text{if } \|err_{j,k}(p_i^j)\|_2 < thres_1 \end{cases} \quad (\text{A.4})$$

where $BP_{T(initial)}(p_i^j)$ represents the initial status of the temporal background prior for the i^{th} pixel in the j^{th} frame. $thres_1$ is set to 1, which constrains the disparity between the estimated coordinate and the real coordinate within the distance of one-pixel unit so as to get more accurate initial temporal background priors.

A.2.1.2 Temporal background priors after multiple-pair constraint and consistency propagation

In Sec. A.1, two standards of good background priors are mentioned. So for this part, two steps are designed for refinement of temporal background priors according to these two criteria respectively.

Multiple-pair constraint The purpose of this measure is to improve the accuracy of temporal background priors. In 3.1.1, the initial temporal background priors are obtained comparing one frame and its next frame (i.e. two frames). However, as is known to all, the difference between two adjacent frames is small or even tiny, which may cause the failure of extracting the precise background priors. One example can be used to more specifically illustrate this point. Imagine the video scene where a girl (foreground) is running on the grass (background): from the first frame to the second frame, only the girl's legs move

and the trunk part of her body remains static; till the third and fourth frames we can see the trunk move. In the above case, if we still compare one frame and its next (one pair), the girl's trunk might be treated as the initial temporal background priors, which actually is a part of foreground. Consequently, the constraint of multiple frame pairs is conducted eliminating the situation that a portion of foreground is mistakenly classified as the temporal background prior and thus making temporal background priors more accurate. In detail, the whole procedure in 3.1.1 is conducted between the j^{th} frame and its subsequent $2^{l-1}th$ ($l = 1, 2, 3, \dots, L$) frames respectively (totally for L times) to get the relevant initial temporal background priors $BP_{T(initial)}^{j,j+2^{l-1}}$. In order to integrate multiple comparison results for all frames under the multiple-pair constraint, let the intermediate status for temporal background priors be

$$BP_{T(mp)}(p_i^j) = \begin{cases} 1, if \frac{1}{L} \sum_{l=1}^L BP_{T(initial)}^{j,j+2^{l-1}}(p_i^j) = 1 \\ \frac{L-1}{L}, if 0 < \frac{1}{L} \sum_{l=1}^L BP_{T(initial)}^{j,j+2^{l-1}}(p_i^j) < 1 \\ 0, if \frac{1}{L} \sum_{l=1}^L BP_{T(initial)}^{j,j+2^{l-1}}(p_i^j) = 0 \end{cases} \quad (A.5)$$

Then the objective cost function $\mathfrak{S}_a$ is designed to obtain the accurate temporal background priors, which is expressed as

$$\mathfrak{S}_a = \sum_j \sum_i \|BP_T(p_i^j) - BP_{T(mp)}(p_i^j)\|_2 \quad (A.6)$$

where $BP_T(p_i^j) \in \{0, 1\}$ and $BP_T(p_i^j)$ represents the variable to be further optimized indicating the final status of temporal background priors.

Consistency propagation In general, traditional methods would more or less suffer from losing some parts of the background. Driven by this problem, consistency propagation is proposed to explore and excavate more potential temporal background priors based on the motion consistency between frames. This step corresponds to the above “sufficient” criterion. The main idea of consistency propagation is as follows: if one point

in the current frame is classified as the temporal background prior, then the correspondences of this point in several adjacent frames (both forward and backward) should also be labeled as temporal background priors. In this way, with the consistency between the corresponding points of adjacent frames, the temporal background priors are further propagated. To sum up, this step can be formulated as

$$\begin{aligned} \mathfrak{S}_s = & \underbrace{\sum_{j=1}^{N-1} \sum_i \|BP_T(p_i^j) - BP_T(p_i^{j+1})\|_2}_{\text{forward}} + \\ & \underbrace{\sum_{j=N}^2 \sum_i \|BP_T(p_i^j) - BP_T(p_i^{j-1})\|_2}_{\text{backward}} \end{aligned} \quad (\text{A.7})$$

where p_i^{j+1} and p_i^{j-1} are the corresponding points of p_i^j in the forward frame (the $j+1^{th}$ frame) and backward frame (the $j-1^{th}$ frame), respectively.

Combining “multiple-pair constraint” and “consistency propagation”, the overall temporal background priors can be obtained solving the following optimization problem:

$$\begin{aligned} \min : \mathfrak{S} &= \mathfrak{S}_a + \mathfrak{S}_s \\ &= \underbrace{\sum_j \sum_i \|BP_T(p_i^j) - BP_{T(mp)}(p_i^j)\|_2}_{\substack{\text{intra-frame} \\ \text{inter-frame}}} + \\ & \quad \underbrace{\sum_{j=1}^{N-1} \sum_i \|BP_T(p_i^j) - BP_T(p_i^{j+1})\|_2 + \sum_{j=N}^2 \sum_i \|BP_T(p_i^j) - BP_T(p_i^{j-1})\|_2}_{\text{temporal consistency}} \end{aligned} \quad (\text{A.8})$$

where the $BP_T(\bullet)$ is in pixel level. Then SLIC method [140] is utilized to convert the pixel-level $BP_T(\bullet)$ into superpixel-level temporal background priors, which are denoted by $BP_T^*(sp)$ (sp represents the superpixel).

A.2.2 Spatial background priors estimation

Traditionally, many saliency detection methods use image boundary as the background priors. However, in some cases image boundary is not always effective. In this section,

a technique is specially designed using superpixel-level boundary connectivity [115] to extract the potential background priors in spatial domain.

A brief introduction of boundary connectivity is given below, which is denoted by $BndCon(\bullet)$. It is actually a conception for describing the degree to which a region is connected to image boundaries. The specific definition of boundary connectivity can be given in [115] as

$$BndCon(sp) = \frac{Len_{bnd}(sp)}{\sqrt{Area(sp)}} \quad (A.9)$$

where $Area(sp)$ approximately measures the number of superpixels in the region where sp is located; $Len_{bnd}(sp)$ approximately measures the number of superpixels that belongs to the boundary in the region where sp is located. When $BndCon(sp)$ is larger, the proportion of boundary superpixels will be higher in the region where sp is located. In other words, the greater value of boundary connectivity means this region is more likely to be a part of background; similarly, smaller boundary connectivity corresponds to the foreground. Due to the limitation of space, the specific computation of $Area(sp)$ and $Len_{bnd}(sp)$ can be referred in [115].

Here a threshold $thres_2$ is set to obtain the spatial background priors for the j^{th} frame, which can be defined as

$$BP_S^j(sp) = \begin{cases} 1, & \text{if } BndCon(sp) \geq thres_2 \\ 0, & \text{if } BndCon(sp) < thres_2 \end{cases} \quad (A.10)$$

where sp represents a superpixel in the current frame. Based on the illustration of [115] and our experience, the boundary connectivity values for the superpixels in salient objects tend to be less than 1; on the contrary, those values for the superpixels in background regions have the tendency to be greater than 1. So here $thres_2$ is set as 1.

A.2.3 Fusion of spatiotemporal background priors

From Sec. A.2.1 and A.2.2, the temporal and spatial background priors have already been obtained respectively. In this section, the above two aspects are fused to form the

spatiotemporal background priors for the j^{th} frame the under the unit of superpixel, which is expressed as

$$BP_{ST}^j = BP_T^j \cup BP_S^j \quad (\text{A.11})$$

where BP_T^j represents the set of those superpixels in the j^{th} frame with $BP_T^*(sp) = 1$ and similarly, BP_S^j is the set of superpixels with $BP_S^j(sp) = 1$.

A.2.4 Graph-based saliency detection

As is known, spatial and temporal information are two kinds of foremost information for video analysis. However, to the best of our knowledge, these two aspects haven't been used simultaneously in graph-based salient object detection. Traditionally, it is only the spatial information that is extracted to generate the graph for the computation of saliency values. In this section, driven by this point a graph-based saliency method is proposed to make the best of both spatial and temporal information on superpixel level, which is especially suitable for salient object detection in videos. The whole process of saliency computation is shown in Fig. A.3 using one frame instance.

Figure A.3: One frame instance to show the computation of saliency values based on the obtained spatiotemporal background priors from Sec. A.2.3: (a) the input frame in a video; (b) extracting superpixels from (a); (c) and (d) are the graphs constructed by appearance and motion information respectively, in which black dots represent background prior superpixels; (e) is the appearance saliency map based on geodesic distances; after setting a threshold, the appearance-based binary salient region (ABSR) as query can be obtained like (f); by using manifold ranking, motion saliency map (g) can be obtained from (d) and (f); final saliency map (h) is generated as the fusion of (e) and (g).

A.2.4.1 Appearance saliency

In this part, appearance saliency is computed by constructing a graph, which measures the geodesic distance to background priors. In detail, firstly an undirected weighted graph $G = \{V, \varepsilon\}$ is created, where V represents the set of all nodes in this graph and each node corresponds to one superpixel. Besides, there exists an edge between every two adjacent nodes and the set of all edges are denoted by ε . Each edge has a weight, which is defined as the color difference between two nodes and this difference is measured in Euclidean distance. It is worth mentioning that the color of each node is represented by the mean value of all pixels in its corresponding superpixel in R, G, B channels, respectively. The geodesic distance refers to the accumulated edge weights along the shortest path between two nodes in a graph model. It can measure the similarity between two nodes in the graph model of an image. With the above declaration, the appearance saliency for each superpixel in the j^{th} frame is the geodesic distance between this superpixel and its nearest spatiotemporal background priors in the j^{th} frame. It can be denoted as

$$\begin{aligned}
 Sal_{app}^j(sp) &= geo(sp, BP_{ST}^j) \\
 &= \min_{B \in BP_{ST}^j} \min_{P_1=sp, P_2, \dots, P_n=B} \sum_{i=1}^{n-1} weight(P_i, P_{i+1}) \\
 &s.t. (P_i, P_{i+1}) \in \varepsilon
 \end{aligned} \tag{A.12}$$

where sp represents each superpixel and $j = 1, 2, 3, \dots, N$; $weight(P_i, P_{i+1})$ is the weight of the edge between P_i and its adjacent node P_{i+1} ; $\{P_1, P_2, \dots, P_n\}$ is the node set of the path between sp and B (B is one node in BP_{ST}^j).

A.2.4.2 Motion saliency

During recent years, several graph-based saliency detection methods [113][117] have been proposed. However, these traditional methods are based on two hypotheses. On the one hand, they assume that the appearance of different parts of the salient object should

be resemblant. However, in real cases, the appearance of every portion of the salient object may vary considerably. As a result, saliency values for the salient object are likely to become differentiated (cannot be guaranteed uniform). On the other hand, they assume that the appearance of the salient object differs sharply from background priors. Nevertheless, the appearance similarity may exist in some of the joint regions between salient objects and the background, which results in the failure of highlighting these parts of the salient object. For further vivid interpretation, in Fig. A.3(e), the girl’s hair cannot be completely detected because both the girl’s hair and the neighboring background are in the dark region and the appearances of these two don’t have distinct difference. That means when only “appearance information” is used, the girl’s head and hair is not salient; in other words, the salient object is not completely detected. Consequently, it is likely that for traditional methods, the foreground fails to be fully detected as the salient object owing to the limitation of merely using “appearance information”. Thus, traditional methods can only achieve suboptimal performance.

More generally speaking, the sole “appearance feature” is not the best choice for extracting homographies. Because there exists the possibility that for one object, as a homography, the appearance of its each part might be different. This situation will lead to the incapability of obtaining accurate and uniform saliency maps.

From a different perspective, actually different parts of one homography (object) shares similar motion, even if the appearance for every part of this object is different. Inspired by this point, the video’s temporal (motion) information is used as the other crucial hint for videos to build motion saliency maps. It can complement the deficiency of appearance information in videos and complete the manifestation of saliency maps. The general idea is to utilize inter-frame motion information to measure the motion similarity between each superpixel and pre-chosen appearance-based salient regions, by which the motion saliency map is produced. As is known to all, in videos the motion tendency for different parts of the salient object is similar. So the motion similarity can help us to

seek for the potential missing regions of the salient object, making the detected salient object more accurate and integrated.

In detail, firstly the appearance saliency maps are extracted for all frames from Sec. A.2.4.1 and the mean saliency values for each map are computed. These mean saliency values are then used as the thresholds (denoted as $thres_3$) to get binary appearance-based salient regions of the corresponding frames. Actually $thres_3$ is an adaptive threshold which serves to extract the superpixels of salient objects as queries. This process can be expressed as

$$thres_3^j = \frac{1}{N_{sp}} \sum_{sp} Sal_{app}^j \quad (A.13)$$

where N_{sp} is the total number of superpixels and $j = 1, 2, 3, \dots, N$. Then for each frame, scan through all superpixels to pick out the relevant appearance-based binary salient region (denoted as $ABSR(\bullet)$) and record the index number of superpixels in this region, which is denoted as

$$ABSR(j) = \{sp | Sal_{app}^j(sp) > thres_3^j\} \quad (A.14)$$

Meanwhile, another undirected weighted graph of motion $G' = \{V, \varepsilon_{mot}\}$ is constructed which is similar to the graph in Sec. A.2.4.1 in terms of both graph structure and nodes. The difference between that graph and this motion graph is that the weights of each edge in this motion graph has distinct connotation: it represents the motion difference of two adjacent superpixels. Specifically, suppose we have two adjacent superpixels sp_1 and sp_2 in the j^{th} frame, the motion difference (weight of the edge) between sp_1 and sp_2 is defined as

$$w_{mot}^j(sp_1, sp_2) = \left\| \frac{1}{n_{sp_1}} \sum err_{j,j+1}(p_1) - \frac{1}{n_{sp_2}} \sum err_{j,j+1}(p_2) \right\|_2 \quad (A.15)$$

where p_1 and p_2 is a pixel in sp_1 and sp_2 respectively; n_{sp_1} and n_{sp_2} is the number of pixels in sp_1 and sp_2 respectively.

Till this step, the required motion graph has been obtained. Next, the foreground query (FQ) step in the manifold ranking based method [117] is used to seek out those superpixels which have similar motion as the appearance-based salient region. From this step, motion saliency value is obtained. The motion saliency for the j^{th} frame can then be expressed as

$$Sal_{mot}^j(sp) = FQ(sp, ABSR(j)) \quad (A.16)$$

A.2.4.3 Fusion of Appearance and Motion Saliency

Now both appearance and motion saliency values (maps) have been acquired from Sec. A.2.4.1 and A.2.4.2, respectively. Naturally, the next step is integrating these two saliency values as a fusion of overall saliency maps. It aims to combine the advantages of both sides, complement each other and make the saliency more robust and optimized. In mathematical expression, the saliency value for the superpixel sp in the j^{th} frame can be shown as

$$Sal^j(sp) = norm(Sal_{app}^j(sp)) + norm(Sal_{mot}^j(sp)) \quad (A.17)$$

where $norm(\bullet)$ represents the unity-based normalization process into $[0, 1]$ for the current j^{th} frame. Specifically, the reason why the final saliency value is the “sum” of appearance and motion saliency is actually twofold. On one hand, appearance and motion information can offset each other, all together making the salient object more complete; on the other hand, this kind of “addition” can highlight the salient object while fading background regions.

A.3 Experiments and Analysis

In this section, the proposed approach are compared and evaluated for effectiveness and robustness against several state-of-the-art saliency methods [111, 112, 117, 124, 130, 137].

Among them, [117] is designed for image saliency detection; [111, 112, 124, 130, 137] aim at video saliency detection. Using the source codes that corresponding authors provide, we obtain the saliency maps of above methods respectively. Also during experiments, different datasets are used, including both popular datasets with simple scenes and difficult datasets with complex scenes. A series of comprehensive comparisons and demonstration both quantitatively and visually prove the validity and superiority of the proposed method.

A.3.1 Datasets with simple scenes: SegTrack and SegTrack v2

SegTrack [141] and SegTrack v2 [133] are the most popular benchmark with relatively simple scenes in the fields of video saliency, tracking, segmentation and so on. SegTrack is a video segmentation dataset with full pixel-level annotations on one or multiple objects at each frame within each video. It contains six videos with pixel-wise ground truth, namely “Birdfall”, “Cheetah”, “Girl”, “Monkeydog”, “Parachute” and “Penguin”. “Birdfall” was shot by stationary camera and other remaining videos were taken using moving cameras. Besides, each frame of a video in SegTrack corresponds to one binary mask as the ground truth. SegTrack v2 can be seen as an updated version of SegTrack. Different from SegTrack, SegTrack v2 consists of 14 videos under more challenging circumstances, like appearance change, motion blur, occlusion, complex deformation, interacting objects, slow motion and so on. Note that in SegTrack, the ground truth for multiple salient objects is incomplete and incorrect: some only contain one object as the ground truth, like “Monkeydog”, “Penguin” and “Cheetah”. So for the above videos in SegTrack, we use the ground truth from SegTrack v2 for evaluation, which is complete and correct. It is worth mentioning that many researchers skip “Penguin” video and only use the remaining five videos for test because of the incomplete ground truth and also complexity. Here all videos are used to keep the objectivity and impartiality of experimental results for a complete dataset.

Figure A.4: Quantitative comparisons between the proposed method and other six existing approaches using SegTrack and SegTrack v2 datasets as the benchmark of simple scenes: (a) and (c) are precision-recall curves for SegTrack and SegTrack v2 respectively by setting the thresholds from 0 to 255 for obtained saliency maps; (b) and (d) are mean absolute errors (MAE) for SegTrack and SegTrack v2 respectively.

To evaluate the all-round performances and disparities between our method and other existing ones, the experimental results are based on both quantitative criteria and visualization effects. For the former, two popular metrics are used, including precision-recall (PR) curve and mean absolute error (MAE). Specifically in PR curve, “precision” refers to the percentage of salient pixels which are allocated correctly in the obtained saliency maps; “recall” represents the percentage of detected salient pixels. For plotting the curve, obtained saliency maps are binarized with a series of thresholds from 0 to 255. Then 256 pairs of precision-recall combinations are generated and the curve can be drawn. From the implication of PR curve’s definition, the ideal condition is that both precision and recall are equal to 1, so the curve closer to (1, 1) (top right corner) corresponds to better performance.

Another assessment criterion is the mean absolute error (MAE) [142], which aims at a more balanced comparison between the binary ground truth GT and the continuous saliency map S for all frame pixels. It can be defined as

$$MAE = \frac{1}{N_p} \sum_{i=1}^{N_p} |GT(i) - S(i)| \quad (\text{A.18})$$

where N_p is the number of frame pixels and i is the pixel index. As can be seen from the above expression, MAE depicts the degree of approximation between the obtained continuous saliency map (normalized to the scale of $[0, 1]$) and binary ground truth. Smaller MAE means smaller dissimilarity and better performance.

Fig. A.4 shows the resulting quantitative comparisons between the proposed methods and other six approaches when SegTrack and SegTrack v2 are used to evaluate the performance of simple scenes. As can be seen in Fig. A.4(a)(c), the PR curve describes the degree to which saliency maps highlight salient objects (regions). Higher PR curve which is closer to (1, 1) corresponds to better performance. For the most range, our method ranks the highest, which shows that the proposed algorithm outweighs other listed methods to a great extent. It is noteworthy that the proposed method achieves the

Figure A.5: Visualization effects for the compared methods using SegTrack and SegTrack v2 as the benchmark of simple scenes.

best precision rate up to 0.83 in SegTrack. For SegTrack v2, although more videos with more challenging circumstances are included, this number still remains a high (or even higher) level which reaches up to 0.9. Besides, when recall=0.6, precision still remains at a high level, which is above 0.7 and 0.8 respectively in SegTrack and SegTrack v2. This indicates our saliency maps are more precise than other existing methods while maintaining a high-level response to salient regions. More intuitively from Fig. A.4(b)(d) in terms of MAE, our method improves the performance by 22% and 30% respectively in SegTrack and SegTrack v2 when compared to the smallest value of other compared methods. This indicates the validity and superiority of the proposed method in the case of simple scenes. For further analysis, other existing methods mostly use the information from image/frame boundary. It may result in inaccuracy when salient objects are located near the boundary region. Consequently, these methods are likely to lose some portions of salient objects. In contrast, the success of the proposed method stems from that 1) effective and accurate background priors are used to distinguish the salient object from the video; 2) the saliency maps of the proposed method are more uniform. This point is attributed to the utilization of motion saliency. It makes the portions of similar motions share corresponding similar saliency values, which lets the salient object more uniform.

Fig. A.5 provides the comparisons of visualization effects for different methods. As can be clearly seen, the proposed method achieves optimal performance, which is nearest to the ground truth when compared with other six methods. The defects of these methods can be summarized into several categories: 1) the detected result is incomplete and only parts of the salient object are detected; 2) parts of the background are wrongly detected as components of the salient objects; 3) the detected salient object contains too much noise; 4) salient objects cannot be detected. However, in contrast, our results show two important characteristics of detected salient objects: 1) *Accuracy*. In comparison to other traditional approaches where the image boundary is used as background priors, the proposed method takes full advantage of information from both spatial and temporal

domain to obtain the better and more robust background priors. 2) *Integrity*. The information of motion saliency is used, which makes all parts of the salient object be detected as a homography.

A.3.2 Datasets with complex scenes: Freiburg-Berkeley Motion Segmentation Dataset

Freiburg-Berkeley Motion Segmentation Dataset (FBMS-59) is a popular and more challenging dataset in the field of moving object segmentation and video saliency. It is a large benchmark with 59 heterogeneous video sequences, many of which contains complex scenes. So it is an appropriate dataset for evaluating the performance of different salient object detection methods in tough and complicated situations. Note that FBMS-59 is a rather big dataset including both training sets (29 videos) and test sets (30 videos); besides, the ground truth images are provided only for a small proportion of frames instead of all frames. So for efficiency and handy processing, we select its test set with the corresponding given ground truth for experiment.

Same methods of comparison and assessment criteria are utilized as those in Sec. A.3.1. The experimental results are shown in Fig. A.6. For Fig. A.6(a), in most ranges, the PR curve of the proposed method is the highest and closest to (1,1); for Fig. A.6(b), our MAE is the lowest and obtain a 25% improvement against the second lowest method. These two quantitative comparisons show that even when video scenes become more complex, the proposed method can still achieve the best performance in all compared approaches. These results further prove that our method is accurate and robust against different scenarios.

Fig. A.7 demonstrates some typical kinds of complex scenes. As can be seen from Fig. A.7(a), both the white bus and the red car are salient objects and our method can simultaneously detect these two; however, for remaining compared methods, the best result still fails to detect thoroughly. (b) is from a video in which a dog is running past a

(a)

(b)

Figure A.6: Quantitative comparisons of PR curve and MAE between the proposed method and other six existing approaches using FBMS-59 datasets as the benchmark of complex scenes.

Figure A.7: Some typical examples of complex scenes and the corresponding visualization effects using the compared methods: (a) multiple salient objects; (b) partial occlusion; (c) complete occlusion; (d) complex background; (e) color similarity between salient object and background; (f) light disturbance.

horse. This frame is just shot when partial body of the dog is overlapped by the horse’s leg and it causes partial occlusion. Our method can precisely detect the dog while others can’t. (c) is cut from a video where a rabbit is circuiting a elliptical machine and at this time the rabbit’s whole body is entirely occluded by the machine. The visualization result of our method honestly reflect the truth: it cannot detect the salient object; meanwhile, others detect “false” salient regions. For (d), the background information is very complex: tanglesome weeds, shabby factory with broken windows and doors. Under such tough circumstances, our method can still detect all regions of the whole tractor and even the loader in the tractor is clear. However, other compared ones detect more than that due to the interference of the complex background: a portion of the factory is wrongly deemed as the salient object. In (e), the salient object (lion) and the background share similar color in some regions, which is difficult to distinguish the foreground from the background and causes the failure of integrated detection. Nevertheless, since the proposed method utilizes motion saliency, it can make up for the deficiency when only appearance information is used. Thus, the proposed method can obtain an exact and intact detection. As for (f), an old man is repairing something on the ladder. Because the light filters through the slits around the man, other methods detect these illuminated regions as more salient and neglect the “authentic” salient object. By contrast, our method can correctly represent the salient object with the existence of light disturbance. To sum up, all the aforementioned extreme cases in a more complex scene illustrate that the proposed method is capable of resisting various disturbance and maintaining the accurate and robust performance in a video containing complex scenes.

A.4 Concluding Remarks

In this appendix, a novel graph-based salient object detection approach for videos has been proposed using spatiotemporal background priors. Firstly the temporal background

priors are obtained based on the background homography between frames, which is estimated by applying RANSAC on point correspondences utilizing SIFT flows of multiple-pair frames. Then, the spatial background priors are combined with abovementioned temporal background priors to generate the spatiotemporal background priors. Finally, by respectively constructing the appearance graph and motion graph, the saliency map for each frame is obtained after measuring the difference from the spatiotemporal background priors. Compared with other state-of-the-art methods, the proposed method achieves highest performance on different challenging datasets of both simple and complex scenes, which exhibits higher robustness and accuracy.

Author's Publications

Journal papers:

- (i) (*: equal contributions) Tao Xi*, **Wei Zhao***, Han Wang and Weisi Lin, “Salient Object Detection with Spatiotemporal Background Priors for Video”, *IEEE Transactions on Image Processing*, accepted. DOI: 10.1109/TIP.2016.2631900.
- (ii) **Wei Zhao** and Han Wang, “Strategic Decision-Making Learning from Label Distributions: An Approach for Facial Age Estimation”, *Sensors*, 16(7):994, 2016.
- (iii) **Wei Zhao** and Han Wang, “Facial Age Estimation Based on Advanced Ordinal Ranking”, *Electronics Letters*, 51(12): 903-904, 2015.
- (iv) **Wei Zhao**, Han Wang and Guang-Bin Huang, “Multifeature Extreme Ordinal Ranking Machine for Facial Age Estimation”, *Mathematical Problems in Engineering*, vol. 2015, Article ID 840840, 2015.

References

- [1] N. Ramanathan and R. Chellappa, “Modeling age progression in young faces,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 387–394, 2006.
- [2] J. T. Todd, S. M. Leonard, and R. E. Shaw, “The perception of human growth,” *Scientific Am.*, vol. 242, no. 2, pp. 106–114, 1980.
- [3] M. Gonzalez-Ulloa and E. Flores, “Senility of the face: basic study to understand its causes and effects,” *Plastic and Reconstructive Surgery*, vol. 36, pp. 239–246, 1965.
- [4] A. Stone, “The aging process of the face & techniques of rejuvenation,” Available at http://www.aaronstonemd.com/Facial_Aging_Rejuvenation.shtm, 2010.
- [5] Y. Kwon and N. Lobo, “Age classification from facial images,” *Computer Vision and Image Understanding*, vol. 74, no. 1, pp. 1–21, 1999.
- [6] X. Geng, Z. Zhou, and K. Smith-Miles, “Automatic age estimation based on facial aging patterns,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 29, no. 12, pp. 2234–2240, 2007.
- [7] Y. Fu, Y. Xu, and T. Huang, “Estimating human ages by manifold analysis of face pictures and regression on aging features,” in *IEEE Conf. on Multimedia and Expo*, pp. 1383–1386, 2007.

REFERENCES

- [8] A. Gunay and V. V. Nabiyev, “Automatic age classification with lbp,” in *Int’l Symp. Computer and Information Sciences*, pp. 1–4, 2008.
- [9] A. Gallagher and T. Chen, “Estimating age, gender, and identity using first name priors,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2008.
- [10] A. Gallagher and T. Chen, “Understanding images of groups of people,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2009.
- [11] A. K. Jain, S. C. Dass, and K. Nandakumar, “Soft biometric traits for personal recognition systems,” in *Int’l Conf. Biometric Authentication*, pp. 731–738, 2004.
- [12] J. E. Kloeppel, “Step right up, let the computer look at your face and tell you your age,” Available at <http://news.illinois.edu/news/08/0923age.html>.
- [13] “Electronic customer relationship management (ecrm),” Available at <http://en.wikipedia.org/wiki/ECRM>.
- [14] Y. Fu and T. Huang, “Human age estimation with regression on discriminative aging manifold,” *IEEE Trans. on Multimedia*, vol. 10, no. 4, pp. 578–584, 2008.
- [15] N. Ramanathan and R. Chellappa, “Face verification across age progression,” *IEEE Trans. on Image Processing*, vol. 15, no. 11, pp. 3349–3361, 2006.
- [16] S. Yan, H. Wang, T. Huang, and Q. Yang, “Ranking with uncertain labels,” in *IEEE Conf. on Multimedia and Expo*, pp. 96–99, 2007.
- [17] S. Yan, H. Wang, X. Tang, and T. Huang, “Learning auto-structured regressor from uncertain nonnegative labels,” in *Int’l Conf. on Computer Vision*, pp. 1–8, 2007.
- [18] G. Guo, Y. Fu, T. Huang, and C. Dyer, “Locally adjusted robust regression for human age estimation,” in *IEEE Workshop on Applications of Computer Vision*, pp. 1–6, 2008.

REFERENCES

- [19] A. M. Albert, K. Ricanek, and E. Pattersonb, “A review of the literature on the aging adult skull and face: implications for forensic science research and applications,” *Forensic Science Int'l*, vol. 172, no. 1, pp. 1–9, 2007.
- [20] “Aging of the face,” Available at <http://www.face-and-emotion.com/dataface/facets/aging.jsp>, 2010.
- [21] G. Rhodes and T. Tremewan, “Averageness, exaggeration, and facial attractiveness,” *Psychological science*, vol. 7, no. 2, pp. 105–110, 1996.
- [22] K. A. Deffenbacher, T. Vetter, J. Johanson, and A. J. O’Toole, “Facial aging, attractiveness, and distinctiveness,” *Perception*, vol. 27, pp. 1233–1243, 1998.
- [23] J. H. Langlois and L. A. Roggman, “Attractive faces are only average,” *Psychological science*, vol. 1, no. 2, pp. 115–121, 1990.
- [24] “Beautycheck - causes and consequences of human facial attractiveness (summary),” Available at http://www.uni-regensburg.de/Fakultaeten/phil_Fak_II/Psychologie/Psy_II/beautycheck/english/zusammen/zusammen1.htm.
- [25] A. J. O’Toole, T. Vetter, H. Volz, and E. M. Salter, “Three-dimensional caricatures of human heads: distinctiveness and the perception of facial age,” *Perception*, vol. 26, pp. 719–732, 1997.
- [26] A. J. O’Toole, T. Price, T. Vetter, J. C. Bartlett, and V. Blanz, “3d shape and 2d surface textures of human faces: the rold of ‘averages’ in attractiveness and age,” *Image and vision computing*, vol. 18, pp. 9–19, 1999.
- [27] T. R. Alley, *Social and Applied Aspects of Perceiving Faces*. Laurence Erlbaum Assoc., 1988.

- [28] L. Farkas, *Anthropometry of the Head and Face*. Raven Press, 1994.
- [29] Y. Kwon and N. Lobo, “Age classification from facial images,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 762–767, 1994.
- [30] T. F. Cootes, G. J. Edwards, and C. J. Taylor, “Active appearance models,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 681–685, 2001.
- [31] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, “Active shape models- their training and application,” *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, 1995.
- [32] A. Lanitis, C. J. Taylor, and T. F. Cootes, “Toward automatic simulation of aging effects on face images,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 442–455, 2002.
- [33] A. Lanitis, C. Draganova, and C. Christodoulou, “Comparing different classifiers for automatic age estimation,” *IEEE Trans. on SMC-B*, vol. 34, no. 1, pp. 621–628, 2004.
- [34] X. Geng, Z. Zhou, Y. Zhang, G. Li, and H. Dai, “Learning from facial aging patterns for automatic age estimation,” in *ACM Conf. Multimedia*, pp. 307–316, 2006.
- [35] X. F. He and P. Niyogi, “Locality preserving projections,” in *NIPS*, pp. 585–591, 2003.
- [36] Y. W. Pang, L. Zhang, Z. K. Liu, N. H. Yu, and H. Q. Li, “Neighborhood preserving projections (npp): A novel linear dimension reduction method,” in *Int’l Conf. on Intelligent Computing*, pp. 117–125, 2005.

REFERENCES

- [37] D. Cai, X. F. He, J. W. Han, and H. J. Zhang, “Orthogonal laplacianfaces for face recognition,” *IEEE Trans. on Image Processing*, vol. 15, no. 11, pp. 3608–3614, 2006.
- [38] G. Guo, Y. Fu, C. Dyer, and T. Huang, “Image-based human age estimation by manifold learning and locally adjusted robust regression,” *IEEE Trans. on Image Processing*, vol. 17, no. 7, pp. 1178–1188, 2008.
- [39] Y.-L. Chen and C.-T. Hsu, “Subspace learning for facial age estimation via pairwise age ranking,” *IEEE Trans. on Information Forensics and Security*, vol. 8, no. 12, pp. 2164–2176, 2013.
- [40] T. Ahonen, A. Hadid, and M. Pietikainen, “Face description with local binary patterns: application to face recognition,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037–2041, 2006.
- [41] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, “The feret evaluation methodology for face recognition algorithms,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1090–1104, 2000.
- [42] Z. Yang and H. Ai, “Demographic classification with local binary patterns,” in *Int’l Conf. on Biometrics*, pp. 464–473, 2007.
- [43] T. Sim, S. Baker, and M. Bsat, “The cmu pose, illumination and expression (pie) database,” in *IEEE Conf. on Auto. Facial and Gesture Recog.*, pp. 53–58, 2002.
- [44] F. Gao and H. Ai, “Face age classification on consumer images with gabor feature and fuzzy lda method,” in *Int’l Conf. Advances in Biometrics*, pp. 132–141, 2009.
- [45] S. Yan, M. Liu, and T. S. Huang, “Extracting age information from local spatially flexible patches,” in *IEEE Conf. Acoustics, Speech, and Signal Processing*, pp. 737–740, 2008.

REFERENCES

- [46] J. Suo, T. Wu, S. Zhu, S. Shan, X. Chen, and W. Gao, “Design sparse features for age estimation using hierarchical face model,” in *IEEE Conf. Automatic Face and Gesture Recognition*, pp. 1–6, 2008.
- [47] G. Guo, G. Mu, Y. Fu, and T. Huang, “Human age estimation using bio-inspired features,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 112–119, 2009.
- [48] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio, “Robust object recognition with cortex-like mechanisms,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 29, no. 3, pp. 411–426, 2007.
- [49] M. Riesenhuber and T. Poggio, “Hierarchical models of object recognition in cortex,” *Nature Neuroscience*, vol. 2, no. 11, pp. 1019–1025, 1999.
- [50] H. Han, C. Otto, X. Liu, and A. K. Jain, “Demographic estimation from face images: human vs. machine performance,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 37, no. 6, pp. 1148–1161, 2015.
- [51] J. Lu, V. E. Liong, and J. Zhou, “Cost-sensitive local binary feature learning for facial age estimation,” *IEEE Trans. on Image Processing*, vol. 24, no. 12, pp. 5356–5368, 2015.
- [52] S. Wang, D. Tao, and J. Yang, “Relative attribute svm+ learning for age estimation,” *IEEE Trans. on Cybernetics*, vol. 46, no. 3, pp. 827–839, 2016.
- [53] J. Kim, D. Han, S. Sohn, and J. Kim, “Facial age estimation via extended curvature gabor filter,” in *IEEE Conf. on Image Processing*, 2015.
- [54] T. Kanno, M. Akiba, Y. Teramachi, H. Nagahashi, and T. Agui, “Classification of age group based on facial images of young males by using neural networks,” *IEICE Trans. Information and Systems*, vol. E84-D, no. 8, pp. 1094–1101, 2001.

REFERENCES

- [55] T. Kohonen, *Self-Organizing Maps, 3rd ed.* Berlin/Heidelberg, Germany: Springer-Verlag, 2001.
- [56] K. Ueki, T. Hayashida, and T. Kobayashi, “Subspace-based age-group classification using facial images under various lighting conditions,” in *IEEE conf. on Automatic Face and Gesture Recognition*, pp. 43–48, 2006.
- [57] H. Fukai, H. Takimoto, Y. Mitsukura, and M. Fukumi, “Apparent age estimation system based on age perception,” in *Proc. SICE Ann. Conf.*, pp. 2808–2812, 2007.
- [58] S. K. Zhou, B. Georgescu, X. S. Zhou, and D. Comaniciu, “Image based regression using boosting method,” in *IEEE Conf. on Computer Vision*, pp. 541–548, 2005.
- [59] S. Yan, X. Zhou, M. Liu, M. Hasegawa-Johnson, and T. S. Huang, “Regression from patch-kernel,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.
- [60] P. Viola and M. Jones, “Robust real-time object detection,” *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.
- [61] B. Ni, Z. Song, and S. Yan, “Web image mining towards universal age estimator,” in *ACM Int’l Conf. on Multimedia*, pp. 85–94, 2009.
- [62] B. Xiao, X. Yang, Y. Xu, and H. Zha, “Learning distance metric for regression by semidefinite programming with application to human age estimation,” in *ACM Int’l Conf. on Multimedia*, pp. 451–460, 2009.
- [63] G. Guo and G. Mu, “Simultaneous dimensionality reduction and human age estimation via kernel partial least squares regression,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2011.

REFERENCES

- [64] G. Guo, Y. Fu, T. S. Huang, and C. Dyer, “A probabilistic fusion approach to human age prediction,” in *IEEE Conf. on Computer Vision and Pattern Recognition Workshops*, pp. 1–6, 2008.
- [65] P. Thukral, K. Mitra, and R. Chellappa, “A hierarchical approach for human age estimation,” in *IEEE Int’l Conf. on Acoustics, Speech and Signal Processing*, pp. 1529–1532, 2012.
- [66] M. E. Tipping, “Sparse bayesian learning and the relevance vector machine,” *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.
- [67] K.-H. Liu, S. Yan, and C. C. J. Kuo, “Age estimation via grouping and decision fusion,” *IEEE Trans. on Information Forensics and Security*, vol. 10, no. 11, pp. 2408–2423, 2015.
- [68] K. Chang, C. Chen, and Y. Hung, “Ordinal hyperplanes ranker with cost sensitivities for age estimation,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 585–592, 2011.
- [69] R. Weng, J. Lu, G. Yang, and Y. Tan, “Multi-feature ordinal ranking for facial age estimation,” in *IEEE Int’l Conference and Workshops on Automatic Face and Gesture Recognition*, pp. 1–6, 2013.
- [70] C. Li, Q. Liu, W. Dong, X. Zhu, J. Liu, and H. Lu, “Human age estimation based on locality and ordinal information,” *IEEE Trans. on Cybernetics*, vol. 45, no. 11, pp. 2522–2534, 2015.
- [71] C. Li, Q. Liu, J. Liu, and H. Lu, “Ordinal distance metric learning for image ranking,” *IEEE Trans. on Neural Networks and Learning Systems*, vol. 26, no. 7, pp. 1551–1559, 2015.

REFERENCES

- [72] S. Feng, C. Lang, J. Feng, T. Wang, and J. Luo, “Human facial age estimation by cost-sensitive label ranking and trace norm regularization,” *IEEE Trans. on Multimedia*, vol. 19, no. 1, pp. 136–148, 2017.
- [73] C. Saunders, A. Gammerman, and V. Vovk, “Ridge regression learning algorithm in dual variables,” in *Int. Conf. on Machine Learning*, pp. 515–521, 1998.
- [74] Y.-H. Pao, G.-H. Park, and D. J. Sobajic, “Learning and generalization characteristics of the random vector functional-link net,” *Neurocomputing*, vol. 6, no. 2, pp. 163–180, 1994.
- [75] W. F. Schmidt, M. A. Kraaijveld, and R. Duin, “Feedforward neural networks with random weights,” in *ICPR*, 1992.
- [76] S. An, W. Liu, and S. Venkatesh, “Face recognition using kernel ridge regression,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1–7, 2007.
- [77] G. Huang, Q. Zhu, and C. Siew, “Extreme learning machine: Theory and applications,” *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, 2006.
- [78] G. Huang and C. Siew, “Extreme learning machine with randomly assigned rbf kernels,” *Internatoinal Journal of Information Technology*, vol. 11, no. 1, pp. 16–24, 2005.
- [79] G. Huang, H. Zhou, X. Ding, and R. Zhang, “Extreme learning machine for regression and multiclass classification,” *IEEE Trans. on SMC-B*, vol. 42, no. 2, pp. 513–529, 2012.
- [80] S. Huang and Y. Huang, “Bounds on the number of hidden neurons in multilayer perceptrons,” *IEEE Trans. on Neural Networks*, vol. 2, no. 1, pp. 47–55, 1991.

- [81] M. Sartori and P. Antsaklis, “A simple method to derive bounds on the size and to train multilayer neural networks,” *IEEE Trans. on Neural Networks*, vol. 2, no. 4, pp. 467–471, 1988.
- [82] G. Huang and H. Babri, “Upper bounds on the number of hidden neurons in feed-forward networks with arbitrary bounded nonlinear activation functions,” *IEEE Trans. on Neural Networks*, vol. 9, no. 1, pp. 224–229, 1998.
- [83] G. Huang, L. Chen, and C. Siew, “Universal approximation using incremental constructive feedforward networks with random hidden nodes,” *IEEE Trans. on Neural Networks*, vol. 17, no. 4, pp. 879–892, 2006.
- [84] P. Bartlett, “The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network,” *IEEE Trans. on Information Theory*, vol. 44, no. 2, pp. 525–536, 1998.
- [85] D. Serre, *Matrices: Theory and Applications*. New York: Springer, 2002.
- [86] “The fg-net aging database.” Available at <http://www.fgnet.rsunit.com/>.
- [87] Y. Zhang and D. Yeung, “Multi-task warped gaussian process for personalized age estimation,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 2622–2629, 2010.
- [88] P. Yang, L. Zhong, and D. Metaxas, “Ranking model for facial age estimation,” in *Int’l Conf. on Pattern Recognition*, pp. 3404–3407, 2010.
- [89] W.-L. Chao, J.-Z. Liu, and J.-J. Ding, “Facial age estimation based on label-sensitive learning and age-oriented regression,” *Pattern Recognition*, vol. 46, no. 3, pp. 628–641, 2013.

REFERENCES

- [90] X. Geng, C. Yin, and Z.-H. Zhou, “Facial age estimation by learning from label distributions,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, no. 10, pp. 2401–2412, 2013.
- [91] X. Geng and R. Ji, “Label distribution learning,” in *International Conference on Data Mining Workshops (ICDMW)*, pp. 377–383, 2013.
- [92] A. L. Berger, S. D. Pietra, and V. J. D. Pietra, “A maximum entropy approach to natural language processing,” *Computational Linguistics*, vol. 22, no. 1, pp. 39–71, 1996.
- [93] Y. Liang, X. Wang, L. Zhang, and Z. Wang, “A hierarchical framework for facial age estimation,” *Mathematical Problems in Engineering*, 2014.
- [94] J. Ylioinas, A. Hadid, X. Hong, and M. Pietikainen, “Age estimation using local binary pattern kernel density estimate,” in *Image Analysis and Processing*, pp. 141–150, 2013.
- [95] A. Gunay and V. V. Nabiyev, “Age estimation based on local radon features of facial images,” in *Computer and Information Sciences III*, pp. 183–190, 2013.
- [96] T. Wu, P. Turaga, and R. Chellappa, “Age estimation and face verification across aging using landmarks,” *IEEE Trans. on Information Forensics and Security*, vol. 7, no. 6, pp. 1780–1788, 2012.
- [97] S. Yan, H. Wang, Y. Fu, J. Yan, X. Tang, and T. S. Huang, “Synchronized sub-manifold embedding for person-independent pose estimation and beyond,” *IEEE Trans. on Image Processing*, vol. 18, no. 1, pp. 202–210, 2009.
- [98] K. Chen, S. Gong, T. Xiang, and C. C. Loy, “Cumulative attribute space for age and crowd density estimation,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 2467–2474, 2013.

REFERENCES

- [99] L. Zhang, X. Wang, Y. Liang, and L. Xie, “A new method for age estimation from facial images by hierarchical model,” in *Int’ Conf. on Innovative Computing and Cloud Computing*, pp. 88–88, 2013.
- [100] L. Zhang and P. N. Suganthan, “A survey of randomized algorithms for training neural networks,” *Information Sciences*, vol. 364–365, pp. 146–155, 2016.
- [101] L. Zhang and P. N. Suganthan, “A comprehensive evaluation of random vector functional link networks,” *Information Sciences*, vol. 367–368, pp. 1094–1105, 2016.
- [102] L. Zhang and P. N. Suganthan, “Visual tracking with convolutional random vector functional link network,” *IEEE Trans. on Cybernetics*, vol. 47, no. 10, pp. 3243–3253, 2016.
- [103] G. Guo and X. Wang, “A study on human age estimation under facial expression changes,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 2547–2553, 2012.
- [104] C. Zhang and G. Guo, “Age estimation with expression changes using multiple aging subspaces,” in *IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems*, pp. 1–6, 2013.
- [105] S. Escalera, J. Fabian, P. Pardo, X. Baro, J. Gonzalez, J. Escalante, D. Misevic, U. Steiner, and I. Guyon, “Chalearn looking at people 2015: apparent age and cultural event recognition datasets and results,” in *IEEE Int’l Conf. on Computer Vision Workshops*, pp. 243–251, 2015.
- [106] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H.-Y. Shum, “Learning to detect a salient object,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 33, no. 2, pp. 353–367, 2011.

- [107] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, “Frequency-tuned salient region detection,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1597–1604, 2009.
- [108] M. Cheng, G. Zhang, N. J. Mitra, X. Huang, and S. Hu, “Global contrast based salient region detection,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 409–416, 2011.
- [109] E. Rahtu, J. Kannala, M. Salo, and J. Heikkila, “Segmenting salient objects from images and videos,” in *European Conf. on Computer Vision*, pp. 366–379, 2010.
- [110] T. Brox and J. Malik, “Object segmentation by long term analysis of point trajectories,” in *European Conf. on Computer Vision*, pp. 282–295, 2010.
- [111] H. J. Seo and P. Milanfar, “Static and space-time visual saliency detection by self-resemblance,” *Journal of Vision*, vol. 9, no. 12, pp. 1–27, 2009.
- [112] Y. Fang, Z. Wang, W. Lin, and Z. Fang, “Video saliency incorporating spatiotemporal cues and uncertainty weighting,” *IEEE Trans. on Image Processing*, vol. 23, no. 9, pp. 3910–3921, 2014.
- [113] Y. Wei, F. Wen, W. Zhu, and J. Sun, “Geodesic saliency using background priors,” in *European Conf. on Computer Vision*, pp. 29–42, 2012.
- [114] C. Liu, J. Yuen, and A. Torralba, “Sift flow: dense correspondence across scenes and its applications,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 353–367, 2010.
- [115] W. Zhu, S. Liang, Y. Wei, and J. Sun, “Saliency optimization from robust background detection,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 2814–2821, 2014.

REFERENCES

- [116] H. Jiang, J. Wang, Z. Yuan, T. Liu, and N. Zheng, “Automatic salient object segmentation based on context and shape prior,” in *British Machine Vision Conference*, pp. 110.1–110.12, 2011.
- [117] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, “Saliency detection via graph-based manifold ranking,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 3166–3173, 2013.
- [118] Q. Yan, L. Xu, J. Shi, and J. Jia, “Hierarchical saliency detection,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1155–1162, 2013.
- [119] B. Jiang, L. Zhang, H. Lu, C. Yang, and M.-H. Yang, “Saliency detection via absorbing markov chain,” in *IEEE Conf. on Computer Vision*, pp. 1665–1672, 2013.
- [120] Y. Qin, H. Lu, Y. Xu, and H. Wang, “Saliency detection via cellular automata,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 110–119, 2015.
- [121] T. Xi, Y. Fang, W. Lin, and Y. Zhang, “Improved salient object detection based on background priors,” in *Pacific-Rim Conference on Multimedia*, pp. 411–420, 2015.
- [122] L. Itti and P. Baldi, “Bayesian surprise attracts human attention,” in *Neural Information Processing Systems*, pp. 547–554, 2006.
- [123] Y. Zhai and M. Shah, “Visual attention detection in video sequences using spatiotemporal cues,” in *ACM Intl. Conf. on Multimedia*, pp. 815–824, 2006.
- [124] C. Guo, Q. Ma, and L. Zhang, “Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.

- [125] W. Kim, C. Jung, and C. Kim, “Spatiotemporal saliency detection and its applications in static and dynamic scenes,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 21, no. 4, pp. 446–456, 2011.
- [126] D. Rudoy, D. B. Goldman, E. Shechtman, and L. Zelnik-Manor, “Learning video saliency from human gaze using candidate selection,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1147–1154, 2013.
- [127] Y. Fang, W. Lin, Z. Chen, C.-M. Tsai, and C.-W. Lin, “A video saliency detection model in compressed domain,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 24, no. 1, pp. 27–38, 2014.
- [128] S. Zhong, Y. Liu, F. Ren, J. Zhang, and T. Ren, “Video saliency detection via dynamic consistent spatio-temporal attention modelling,” in *AAAI Conf. on Artificial Intelligence*, pp. 1063–1069, 2013.
- [129] Y. Luo and J. Yuan, “Salient object detection in videos by optimal spatio-temporal path discovery,” in *ACM Intl. Conf. on Multimedia*, pp. 509–512, 2013.
- [130] W. Wang, J. Shen, and F. Porikli, “Saliency-aware geodesic video object segmentation,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 3395–3402, 2015.
- [131] D. Zhang, O. Javed, and M. Shah, “Video object segmentation through spatially accurate and temporally dense extraction of primary object regions,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 628–635, 2013.
- [132] Y. J. Lee, J. Kim, and K. Grauman, “Key-segments for video object segmentation,” in *IEEE Conf. on Computer Vision*, pp. 1995–2002, 2011.
- [133] F. Li, T. Kim, A. Humayun, D. Tsai, and J. M. Rehg, “Video segmentation by tracking many figure-ground segments,” in *IEEE Conf. on Computer Vision*, pp. 2192–2199, 2013.

REFERENCES

- [134] A. Papazoglou and V. Ferrari, “Video segmentation by tracking many figure-ground segments,” in *IEEE Conf. on Computer Vision*, pp. 1777–1784, 2013.
- [135] D. E. Jacobs, D. B. Goldman, and E. Shechtman, “Cosaliency: where people look when comparing images,” in *ACM Symposium on User Interface Software and Technology*, pp. 219–228, 2010.
- [136] H. Li and K. N. Ngan, “A co-saliency model of image pairs,” *IEEE Trans. on Image Processing*, vol. 20, no. 12, pp. 3365–3375, 2011.
- [137] H. Fu, X. Cao, and Z. Tu, “Cluster-based co-saliency detection,” *IEEE Trans. on Image Processing*, vol. 22, no. 10, pp. 3766–3778, 2013.
- [138] Z. Liu, W. Zou, L. Li, L. Shen, and O. L. Meur, “Co-saliency detection based on hierarchical segmentation,” *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 88–92, 2014.
- [139] D. Zhang, J. Han, C. Li, and J. Wang, “Co-saliency detection via looking deep and wide,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 2994–3002, 2015.
- [140] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, “Slic superpixels compared to state-of-the-art superpixel methods,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.
- [141] D. Tsai, M. Flagg, and J. M. Rehg, “Motion coherent tracking with multi-label mrf optimization,” in *British Machine Vision Conference*, pp. 1–11, 2010.
- [142] F. Perazzi, P. Krahenbuhl, Y. Pritch, and A. Hornung, “Saliency filters: Contrast based filtering for salient region detection,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 733–740, 2012.