Thesis for the degree of Doctor of Philosophy

Salient Region Detection Methods with Application to Traffic Sign Recognition from Street View Images

KEREN FU

CHALMERS

Signal Processing Group
Department of Signals and Systems
Chalmers University of Technology

Göteborg 2016

Salient Region Detection Methods with Application to Traffic Sign Recognition from Street View Images
KEREN FU
ISBN 978-91-7597-493-4

Doktorsavhandlingar vid Chalmers Tekniska Högskola
Ny serie nr 4174
ISSN 0346-718X

Signal Processing Group
Department of Signals and Systems
Chalmers University of Technology
SE-412 96 Göteborg, Sweden
Telephone: + 46 (0)31-7721000

This thesis has been prepared using LaTeX.

Printed by Chalmers Reproservice,
Göteborg, Sweden 2016.

To my parents

Abstract

In the computer vision community, saliency detection refers to modeling the selective mechanism in human visual attentions. Outputs of saliency detection algorithms are called saliency maps, which represent conspicuousness levels of different scene areas. Since saliency detection is an effective way to estimate regions of interest that may be attractive to human eyes, numerous applications range from object recognition, image compression, to content-based image editing and image retrieval. This thesis focuses on salient region detection, which aims at detecting and segmenting holistic salient objects from natural images. Despite of many existing models/algorithms and rapid progress in this field over the past decade, improving the detection performance in complex and unconstrained scenarios remains challenging. This thesis proposes five innovative methods for salient region detection. Each method is designed to solve some issues in the existing models. The main contributions of this thesis include: 1) A novel method that induces saliency maps through eigenvectors of the normalized graph cut for better visual clustering of objects and background. It leads to more accurate saliency estimation. 2) A novel data-driven method for salient region detection based on continuous conditional random field (C-CRF). It provides an optimal way to integrate various unary saliency features with pairwise cues. 3) A robust graph-based diffusion method, referred to as manifold-preserving diffusion (MPD). Based on two assumptions on manifold—smoothness and local reconstruction, the method preserves the manifold used in the saliency diffusion. 4) A superpixel-based method that effectively computes color contrast and color distribution attributes of images in a unified manner. 5) A new geodesic propagation method that is used to optimize coarse salient regions for rendering visual coherence. In addition, driven by applications, this thesis also addresses traffic sign recognition (TSR) problem from street view images. As a new application linking between saliency detection and TSR, salient region detection of traffic signs is investigated in order to enhance the sign classification performance.

Keywords: Salient region detection, normalized cut, continuous conditional random field, manifold, adaptive graph edge weights, saliency propagation, geodesics, color contrast, color distribution, traffic sign recognition

Acknowledgement

First of all, I would like to express my sincere gratitude to my supervisors for their time and patience, which make my thesis work possible. I wish to thank Prof. Irene Yu-Hua Gu at Chalmers and Prof. Jie Yang at Shanghai Jiao Tong University in China for their support and constructive suggestions. Special thanks to Prof. Irene Yu-Hua Gu, who has helped me a lot with my project and research articles during my stay in Sweden. Her enthusiasm and well-knit style on research has impressed me a lot and taught me how to do research scientifically.

I am grateful to Chalmers University of Technology, Sweden and Shanghai Jiao Tong University, China for providing me the chance and scholarship to conduct this research and finish my double-degree study. I would like to thank my colleagues in the signal processing group and friends at Chalmers S2 for their help at different time and aspects during my study in Sweden. Specially, I would like to express my great appreciation to our group leader Prof. Tomas Mckelvey for accepting me in the signal processing group. I would like to thank my colleagues Yixiao Yun and Mohammad Alipoor for stimulating discussions on research. I owe many thanks to Ann-Christine Lindbom, Agneta Kinnander and Natasha Adler for their help and kind support to me.

Further, I would like to acknowledge Volvo Cars AB for the funding support to me. Sincere thanks go to Dr. Anders Ödblom and Feng Liu from Volvo Cars AB for their fruitful discussions and collaboration on the traffic sign project, which makes me aware of how the scientific research could be combined with industrial needs.

Finally, I feel so grateful to my dear parents for supporting and encouraging me in my entire PhD life. I wish to share this happy moment of graduation with them. None of this can become possible without their love, care, and smile.

Keren Fu
Göteborg, October 2016

List of Publications

This thesis is based on the following appended publications

Paper 1: **Keren Fu**, Chen Gong, Irene Yu-Hua Gu, Jie Yang, "Normalized Cut-based Saliency Detection by Adaptive Multi-Level Region Merging," IEEE Transactions on Image Processing, 24(12): 5671-5683, 2015.

Paper 2: **Keren Fu**, Irene Yu-Hua Gu, Jie Yang, "Saliency Detection by Fully Learning A Continuous Conditional Random Field," *Submitted to journal*.

Paper 3: **Keren Fu**, Irene Yu-Hua Gu, Chen Gong, Jie Yang, "Robust manifold-preserving diffusion-based saliency detection by adaptive weight construction," Neurocomputing, 175: 336-347, 2016.

Paper 4: **Keren Fu**, Chen Gong, Jie Yang, Yue Zhou, Irene Yu-Hua Gu, "Superpixel based Color Contrast and Color Distribution Driven Salient Object Detection," Signal Processing: Image Communication, 28(10): 1448-1463, 2013.

Paper 5: **Keren Fu**, Chen Gong, Irene Yu-Hua Gu, Jie Yang, "Geodesic Saliency Propagation for Image Salient Region Detection," In proc. of 20th IEEE *International Conference on Image Processing (ICIP)*, 2013.

Paper 6: **Keren Fu**, Irene Y.H. Gu, Anders Ödblom, "Traffic Sign Recognition using Salient Region Region Features: A Novel Learning-based Coarse-to-Fine Scheme," In proc. of IEEE *Intelligent Vehicles Symposium (IVS)*, 2015.

Paper 7: **Keren Fu**, Irene Y.H. Gu, Anders Ödblom, Feng Liu, "Geodesic Distance Transform-based Salient Region Segmentation for Automatic Traffic Sign Recognition," In proc. of IEEE *Intelligent Vehicles Symposium (IVS)*, 2016.

Other publications by the author

8. Keren Fu, Chen Gong, Jie Yang, Yue Zhou, "Salient Object Detection via Color Contrast and Color Distribution," In proc. of 11th *Asian Conference on Computer Vision (ACCV)*, 2012.

9. Keren Fu, Chen Gong, Yixiao Yun, Yijun Li, Irene Yu-Hua Gu, Jie Yang, Jingyi Yu, "Adaptive Multi-Level Region Merging for Salient Object Detection," In proc. of *British Machine Vision Conference (BMVC)*, 2014.

10. Keren Fu, Irene Yu-Hua Gu, Yixiao Yun, Chen Gong, Jie Yang, "Graph Construction for Salient Object Detection in Videos," In proc. of 22nd *International Conference on Pattern Recognition (ICPR)*, 2014.

11. Keren Fu, Chen Gong, Irene Yu-Hua Gu, Jie Yang, Xiangjian He, "Spectral Salient Object Detection," In proc. of IEEE *International Conference on Multimedia & Expo (ICME)*, 2014.

12. Keren Fu, Kai Xie, Chen Gong, Irene Yu-Hua Gu, Jie Yang, "Effective Small Dim Target Detection by Local Connectedness Constraint," In proc. of IEEE *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2014.

13. Keren Fu, Chen Gong, Irene Y.H. Gu, Jie Yang, Pengfei Shi, "Salient Object Detection using Normalized Cut and Geodesics," In proc. of IEEE *Int'l Conference on Image Processing (ICIP)*, 2015.

14. Keren Fu, Irene Yu-Hua Gu, "Recognition of Chinese Traffic Signs from Street Views," *Technical Report (No. R004/2015)*, Chalmers University of Technology, 2015.

15. Keren Fu, Irene Y.H. Gu, Jie Yang, "Learning full-range affinity for diffusion-based saliency detection," In proc. of IEEE *Int'l Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016.

16. Keren Fu, Chen Gong, Yu Qiao, Jie Yang, Irene Yu-Hua Gu, "One-Class Support Vector Machine-Assisted Robust Tracking," *Journal of Electronic Imaging*, 22(2), 2013.

17. Chenjie Ge, **Keren Fu**, Fanghui Liu, Li Bai, Jie Yang, "Co-saliency detection via inter and intra saliency propagation," Signal Processing: Image Communication, 44: 69-83, 2016.

18. Yijun Li, **Keren Fu**, Zhi Liu, Jie Yang, "Efficient Saliency-Model-Guided Visual Co-Saliency Detection," IEEE Signal Processing Letters, 28(5):588-592, 2015.

19. Chen Gong, **Keren Fu**, Qiang Wu, Enmei Tu, Jie Yang, "Semi-supervised classification with pairwise constraints," Neurocomputing, 139: 130-137, 2014.

20. Chen Gong, **Keren Fu**, Artur Loza, Qiang Wu, Jia Liu, Jie Yang, "PageRank Tracker: From ranking to tracking," IEEE Transactions on Cybernetics (TCYB), 44(6): 882-893, 2014.

21. Kai Xie, **Keren Fu**, Tao Zhou, Junhao Zhang, Jie Yang, Qiang Wu, "Small target detection based on accumulated center-surround difference measure," Infrared Physics & Technology, 67: 229-236, 2014.

22. Lei Zhou, **Keren Fu**, Yijun Li, Yu Qiao, Xianjian He, Jie Yang, "Bayesian Salient Object Detection Based on Saliency Driven Clustering," Signal processing: Image Communication, 29(3): 434-447, 2014.

23. Chen Gong, **Keren Fu**, Lei Zhou, Jie Yang, Xiangjian He, "Scalable semi-supervised classification via Neumann series," Neural Processing Letters, 42(1): 187-197, 2014.

24. Chen Gong, **Keren Fu**, Enmei Tu, Jie Yang, Xiangjian He, "Robust object tracking using linear neighborhood propagation," Journal of Electronic Imaging, 2013.

25. Yijun Li, **Keren Fu**, Lei Zhou, Yu Qiao, Jie Yang, "Saliency Detection via Foreground Rendering and Background Exclusion," In IEEE *Int'l Conference on Image Processing (ICIP)*, 2014.

26. Yijun Li, **Keren Fu**, Lei Zhou, Yu Qiao, Jie Yang, Bai Li, "Saliency Detection based on Extended Boundary Prior with Foci of Attention," In IEEE *Int'l Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014.

27. Kai Xie, **Keren Fu**, Tao Zhou, Jie Yang, Qiang Wu, Xiangjian He, "Small Target Detection Using an Optimization-based Filter," In IEEE *Int'l Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

28. Chen Gong, Dacheng Tao, **Keren Fu**, Jie Yang, "FLAP: Fick's Law Assisted Propagation for Semi-supervised Learning," IEEE Transactions on Neural Networks and Learning Systems (TNNLS), 26(9): 2148-2162, 2015.

29. Chen Gong, Dacheng Tao, **Keren Fu**, Jie Yang, "ReLISH: Reliable Label Inference via Smoothness Hypothesis," In *AAAI Conference on Artificial Intelligence (AAAI)*, 2014.

30. Chen Gong, Dacheng Tao, **Keren Fu**, Jie Yang, "Signed Laplacian Embedding for Supervised Dimension Reduction," In *AAAI Conference on*

Artificial Intelligence (AAAI), 2014.

31. Chen Gong, Tongliang Liu, Dacheng Tao, **Keren Fu**, Enmei Tu, Jie Yang, "Deformed graph Laplacian for semi-supervised learning," IEEE Transactions on Neural Networks and Learning Systems (TNNLS), 26(10): 2261-2274, 2015.

32. Tao Zhou, Xiangjian He, Kai Xie, **Keren Fu**, Junhao Zhang, Jie Yang, "Robust visual tracking via efficient manifold ranking with low-dimensional compressive features," Pattern Recognition 48(8): 2459-2473, 2015.

33. Tao Zhou, Xiangjian He, Kai Xie, **Keren Fu**, Junhao Zhang, Jie Yang, "Visual Tracking via Graph-based Efficient Manifold Ranking with Low-Dimensional Compressive Features," In *Int'l Conference on Multimedia and Expo (ICME)*, 2014.

34. Chen Gong, Dacheng Tao, Wei Liu, S.J. Maybank, Meng Fang, **Keren Fu**, Jie Yang, "Saliency Propagation from Simple to Difficult," In IEEE *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Contents

Acronyms

PCA:	Principal Component Analysis
ML:	Maximized Likelihood
GMM:	Gaussian Mixture Model
CRF:	Conditional Random Field
MSRA:	Microsoft Research Asia
Ncut:	Normalized graph cut
ROI:	Region of Interest
DOG:	Difference of Gaussians
TSR:	Traffic Sign Recognition
ADAS:	Advanced Driver Assistance System
C-CRF:	Continuous Conditional Random Field
IR:	Inhibition of Return
FOA:	Focus of Attention
WTA:	Winner Take All
kNN:	k-Nearest Neighbourhood
SLIC:	Simple Linear Iterative Clustering
SVM:	Support Vector Machine
CNN:	Convolutional Neural Network
FRST:	Fast Radial Symmetrical Transform
HOG:	Histogram of Oriented Gradient
LDA:	Linear Discriminant Analysis
MAE:	Mean Absolute Error

PR:	Precision-Recall
OCA:	Overall Classification Accuracy
MPD:	Manifold-Preserving Diffusion
MPDS:	Manifold-Preserving Diffusion-based Saliency
SGD:	Signed Geodesic Distance
SGT:	Signed Geodesic Transform
GP:	Geodesic Propagation

Part I

Introductory chapters

Chapter 1

Introduction

1.1 Saliency detection

Images appearing on televisions, mobile devices, as well as computer screens enrich our daily life. However, processing such vast amount of visual information in images in short time is a computational resource-demanded task. Usually, information differs in importance. Some is crucial and grabs person's attention, while other is less important. Thereby, an automatic mechanism that answers what information might be necessary to pick up for further processing is practically useful. A feasible way to realize the above is lighted by modeling the selective mechanism of human visual attention, or called visual saliency. According to studies of neurobiology and cognitive psychology [1, 2], human eyes are capable of instinctively focusing on a certain subset of visual information and capture with different priorities for further processing in the brain. Such a mechanism, called visual saliency, derives from the long evolution process of human beings and guarantees humans the ability to understand complex visual scenarios in very short time. Besides, it alleviates the need to process the otherwise vast incoming visual data. Such a mechanism has been investigated by multiple disciplines such as cognitive psychology [1–3], neuroscience [4], and computer vision [5]. Researchers categorize visual saliency processes into two types [5], namely bottom-up and top-down. These two types of saliency have the following features, respectively:

- Fast, unconscious, data driven, low-level feature driven
- Slow, task driven, knowledge driven, semantic feature driven

In the computer vision community, modeling visual saliency on images is referred to as *saliency detection*, which aims at detecting salient image parts that easily attract the human attention. These parts are often called

"salient regions", "important regions", or "regions of interest". Although attention processes of human rely on bottom-up influences [6–8] and top-down influences [9–11] as aforementioned, saliency detection in this thesis mainly considers bottom-up factors, namely influences from low-level features. Such type of saliency is data and stimulus driven. This means without specifying any prior tasks or knowledge, bottom-up saliency detection can be applied to detect generic salient regions. In the past decade, this type of saliency detection has become a hot research field in computer vision community [5].

The results of saliency detection are called *saliency maps*, where the pixel-wise intensity indicates the degree of being salient (Figure 1.2). Since such results indicate potential regions of interest (ROI), they have been applied to many computer vision applications, e.g., object detection and recognition [10, 12–14], image and video compression [15], video summarization [16], content-based image editing [17–23] and image retrieval [24–26]. It is worth noting that in many applications above, saliency detection makes it possible for smart and autonomous image processing without any human interaction. Figure 1.1 shows some example applications. There is no doubt that saliency detection with more accurate results will improve these applications, and therefore it is worthy studying in depth.

Aiming at different goals and tasks, saliency models are categorized into *eye fixation modeling* and *salient region detection*[1] [27]. Most of the early saliency models [6, 14, 15, 28] belong to the former, aiming at predicting where human look in a scene. Their basis dates back to the "feature integration theory" [1] proposed by cognitive psychologists Triesman and Gelade, stating what kinds of visual features are important and how they are combined to direct human attention. Neurobiologists Koch and Ullman [2] first propose a feed-forward model to combine these features and introduce the concept of a saliency map, i.e., a topographic map that represents conspicuousness of scene locations. A winner-take-all neural network is introduced in their work to select the most salient locations and an inhibition of return mechanism is employed to simulate eye shift. The first complete implementation of Koch and Ullman's model [2] is proposed by Itti *et al.* [6]. As one of the pioneer work, Itti *et al.* [6] propose a "center-surround" operator as local feature contrast in color, intensity, and orientation on an image pyramid. Such a center-surround operator characterizes the stimuli fed to visual neural cells and is realized througth Difference of Gaussians (DOG).

Although eye fixation prediction is the origin of saliency detection and has gained a lot of progress since then, these methods have a typical shortcoming which limits their performance in many applications. This drawback

[1] Eye fixation modeling and salient region detection are two research directions of saliency detection. They aim at different tasks. However, when they individually appear in certain context or applications, for simplicity both of them might be called saliency detection instead.

Figure 1.1: Sample applications of saliency detection. Images from the corresponding references are reproduced.

is that they tend to generate selectively sparse saliency maps (see Figure 1.2 for an illustrative example). When using them to detect salient objects in relatively large sizes, such models often highlight merely edges, corners of the objects instead the whole. Driven by emerging computer vision applications [12, 13, 15–20, 24, 25] as well as the development of modern display devices, a new branch of saliency detection called *salient region/object detection*[2] [27] as aforementioned has emerged. *In contrast to predicting eye fixations, the goal of salient region detection is to detect and segment entire salient objects in a scene.* The output is a saliency map where the pixel intensity represents the probability of belonging to salient objects. From this definition, one can see this problem in its essence is a figure/ground segmentation problem, and the goal is to only segment the salient foreground objects from image background. Note that this problem differs from the traditional image segmentation which aims at partitioning an image into perceptually coherent regions.

Compared to fixation prediction models, salient region detection methods often compute center-surround contrast [30] or global contrast [7] based on image regions. Hierarchical image segmentation [29, 31] is favored as a multi-scale operation instead of the commonly used image pyramid in eye fixation prediction [6]. Figure 1.2 shows a visual comparison between Itti's model [6] and a state-of-the-art salient region detection model [29]. It is worthy noting that in recent years the research wave of salient region detection has already exceeded that of eye-fixation prediction. According to the survey of Borji *et al.* [32], in 2013 year's top conferences on computer vision including: CVPR (IEEE Conference on Computer Vision and Pattern Recognition), ICCV (IEEE International Conference on Computer Vision), ECCV(European Conference on Computer Vision), the number of papers on salient region/object detection is about five times of those on eye-fixation prediction. This thesis addresses salient region detection in natural images. Despite many previous methods/models on salient region detection exist, improving the performance in complex scenarios yet remains challenging. One of the fundamental challenges is how to emphasize entire salient objects uniformly and suppress irrelevant background from unrestricted complex scenes. In order to improve the detection accuracy, this thesis proposes several novel techniques driven by different motivations to solve certain shortcomings in existing models.

[2] In this research field, terms "salient region detection" and "salient object detection" are often used interchangeably and they refer to the same task. In some parts of this thesis, they are used interchangeably as well.

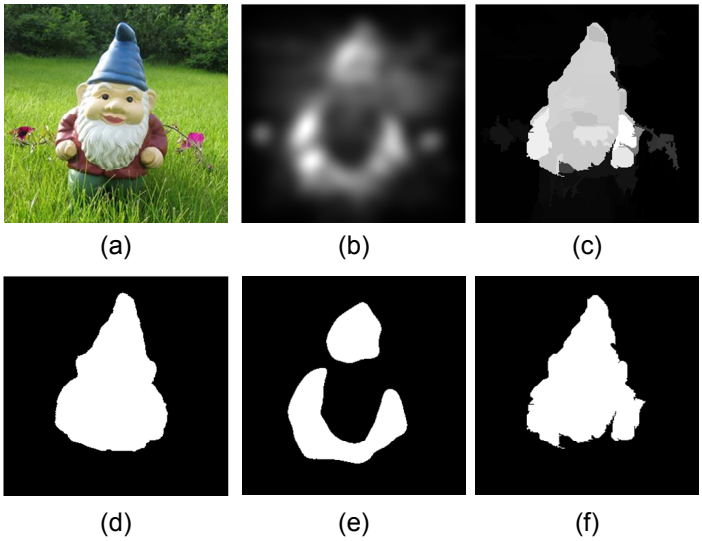

Figure 1.2: An example of visual comparison between Itti's model [6] and a state-of-the-art salient region detection model DRFI [29]. From left to right, top to bottom: (a) A sample image from a benchmark dataset MSRA-1000 [8] for salient region/object detection. (b) The saliency map generated by Itti's model [6]. (c) The saliency map generated by DRFI [29]. (d) The annotated ground truth provided by the dataset. (e) Simple adaptive thresholding [8] on the saliency map (b). (f) Simple adaptive thresholding on the saliency map (c). This example is supposed to show the visual difference between eye fixation prediction models and salient region detection models.

1.2 Traffic sign recognition

Due to recently renewed efforts in vehicle safety and autonomous driving, computer vision-based traffic sign recognition (TSR) has drawn increasing interest lately from academic researchers and industries. It aims at detecting and locating the traffic signs automatically in the captured images/videos, and then classifying these signs. Numerous applications of TSR are listed below:

1) **Advanced driver assistance systems (ADAS)** [33]: Many traffic accidents have happened because the drivers ignored or neglected the traffic signs. TSR is able to remind drivers about the signs appeared on the road and assist them to make the judgement. Hence, traffic accidents can be reduced. TSR is a crucial part of ADAS. Nowadays, many well-known automobile manufacturers such as BMW, Benz, Volvo, Tesla have already

added TSR modules into some of their products.

2) **Intelligent autonomous driving** [34]: Autonomous driving technology is supported by the intelligent computer control systems inside a car to make the car be able to drive itself without human operation. Through the perception of road situation around the car by multiple sensors, information about the road situation, the position of the car, as well as the obstacles can be simultaneously acquired. According to the information, the systems control the speed and turning of the car, making the car drive safely and reliably to the destination. To enable the autonomous car aware of the situation on the road as well as the corresponding traffic rules, TSR is one crucial function that must be included. A famous application instance is Google's self-driving car project[3].

3) **Road maintenance** [35]: Traffic signs are usually placed on the road side or hung above the road. They are infrastructures of the road and require regular maintenance in order to check whether there has been any loss, damage, or malicious occlusion. The traditional way for checking is to rely on human operators to check along the road, or watching the recorded video sequences to confirm the occurrence and status of signs. This no doubt is a tedious and time-consuming task. The TSR can assist or even replace human operators to complete this mission.

4) **Sign inventory** [35]: TSR can help collect images of signs appearing along a road and at the same time link to global positioning system (GPS) to mark the geographical coordinates of these signs, leading to automatic annotation on the map. This finally results in a sign inventory map. Such a map is useful in navigation as well as geographic information systems (GIS).

The challenges in TSR are three-fold and are summarized as follows:

- Degeneration of sign appearances due to light variations, view angle changes, image compression, size changes, occlusion, motion blur, and many more.
- Background noise/clutter whose appearances look similar to those of signs.
- Similarity across some sign classes.

The work related to this field investigated in this thesis is driven by a Volvo project that aims at recognizing Chinese road signs from street view images. Since China has very large population and hence crowded traffic, it uses a variety of signs and complicated combination to guide traffic activities. As a result, it makes sign recognition on Chinese traffic signs an interesting yet challenging task. Though there exist some public traffic sign datasets (e.g., German sign dataset [36, 37], Belgium sign dataset [38]),

[3]https://www.google.com/selfdrivingcar/

Figure 1.3: A street view image (resolution 960*640), where two signs are annotated. The color of their bounding boxes indicate their category and the bottom-left text in the blue background indicates the classes they belong to. Sign #1 means speed limit 120 and sign #2 means speed limit 100.

there is no any public Chinese sign dataset. Besides, Chinese traffic signs are different from the European ones. Thereby, images used for research purpose are collected from a public web source—street view images. Street view images are captured by multiple cameras mounted on the top of a moving car. The car performs street view shooting each time after it moves forward for a certain distance (e.g., 10m). Images captured in different orientations are stitched to generate a 360-degree full street view scene. Therefore, street view images are real-world images which show road scenes in the perspective of a moving vehicle on the road. Likewise, the traffic signs contained in street view images present various distortion and degeneration of signs in real-world scenarios. Using street view images to study TSR problems is hence reasonable and fits to the applications in ADAS and autonomous driving. Google street view[4] is the most commonly used online street view service that covers up to 114 countries in the world, whereas Tencent street view[5] provides street images in major cities of China. Figure 1.3 shows an example of street view images collected from Tencent during our project, where the traffic signs are later manually annotated.

[4]http://www.google.com/streetview/
[5]http://map.qq.com/

In real-world applications, a complete TSR system needs to comprise two modules [35], namely sign detection (locating signs) and sign classification (determining types of signs). Since in practice these two modules are cascaded, the detection results of the first module will heavily influence the performance of subsequent classification. Different from some existing works which only focus on either sign detection or sign classification [39–44], this thesis exploits a complete detection and classification system to tackle Chinese sign recognition problem. In the proposed system, salient region detection is applied to obtain reliable feature extraction from detected signs, leading to enhanced sign classification.

1.3 Scope and addressed problems

1.3.1 Scope

This thesis focuses on robust salient region/object detection in natural images. The aim is to generate high quality saliency maps that emphasize holistic salient objects meanwhile suppress irrelevant background. The other branch namely eye fixation modeling [6, 14, 15, 28] is not investigated. As aforementioned, salient region/object detection aims at detecting and segmenting entire salient objects [27, 32] and can better benefit emerging applications [12, 13, 15–20, 24, 25]. Besides, the proposed saliency models focus on bottom-up salient region detection. The top-down factors [9–11] are not intentionally considered. As aforementioned, the bottom-up saliency detection does not rely on any prior-dependant tasks. Comparing to the top-down one, its applications are more universal.

1.3.2 Addressed problems

This thesis addresses the problems of salient region detection in natural images and proposes five innovative methods. Then we propose two methods to apply salient region detection to traffic sign recognition (TSR). More specifically, our methods are focused on different sub-problems:

- **Group salient objects and background for saliency detection:** Existing methods are often in favor of over-segmented regions upon which saliency levels are computed. Unfortunately, such local regions are less effective on capturing objects holistically and therefore are less emphasis on entire objects in the saliency maps. Additionally, noises are easily introduced by over-segmentation. As a result, existing methods often fail to detect an entire object in complex background. By investigating an optimized clustering/grouping process, more accurate saliency estimation could be achieved.

- **Learn a complete conditional random field (CRF) for saliency detection:** Previously, a CRF is often employed for salient object detection. Unfortunately, when utilizing a CRF with parameterized unary and pairwise energy potentials, existing methods only adopt manually designed parameters, or alternatively learn the parameters for the unary potentials. As such, the feature integration ability of CRF cannot be fully exploited. By learning a complete CRF, namely learning parameters for both types of potentials, we are able to learn to integrate more information and thus expect better detection results.
- **Apply graph-based diffusion to salient region detection:** In diffusion-based saliency detection, an image is first modeled by a graph. Next, a diffusion process is formulated to propagate the saliency information from nodes to nodes. While the diffusion performance heavily relies on the graph edge weights representing the similarity degree between nodes, existing works often define such weights with manually tuned Gaussian bandwidth parameters and fix them for all images. Since different images often have different properties such as colors, contrast, textures, using fixed bandwidth cannot reach an optimal for individual images. This thesis addresses the adaptive construction of graph weights in each image and aims at a new diffusion method for saliency detection.
- **Formulate the computation of color contrast and color distribution in a unified manner:** Color contrast and color distribution are two widely employed hypotheses for discovering salient objects, but there lacks an efficient and unified scheme to compute them. By integrating complementary contrast and distribution information, we are able to achieve a baseline method derived from color attributes.
- **Transfer geodesic distance to saliency detection:** Geodesic distance is often applied to images to measure the connectivity between image parts, and is widely used in interactive foreground segmentation. This thesis addresses how to transfer geodesic distance to saliency detection for rendering a coherent visual saliency map.
- **Traffic sign recognition from street view images using saliency-enhanced methods:** Since salient regions are supposed to be more informative, this thesis further attempts to employ salient regions for enhanced sign classification.

1.4 Outline of this thesis

The thesis is divided into two parts. The first part briefly describes the background and the proposed work. The second part includes publications

resulted from this thesis work. The first part of the thesis is organized as follows: Chapter 2 reviews related work and theories. Chapter 3 gives a thorough overview of existing work on salient region detection and traffic sign recognition. Summary of this thesis work is described in Chapter 4. Finally, conclusion and insights are drawn in Chapter 5.

Chapter 2

Background Theories and Methods

This chapter reviews the basic background theories where this thesis work is built upon, including: Itti's model for eye fixation prediction; global regional contrast for salient region detection; graph-related theory; and superpixel segmentation by SLIC (Simple Linear Iterative Clustering). Noting that although this thesis focuses on salient region detection, salient region detection is one branch in saliency detection and the saliency detection originates from Itti's model.

2.1 Itti's saliency model

In 1998, Itti *et al.* [6] proposed a visual attention system inspired by the behavior and the neuronal architecture of an early primate visual system [2]. It is the earliest saliency model implemented by using computer vision and image processing techniques. Itti's model is related to the so-called "feature integration theory" [1], explaining human visual search strategies. In Itti's model, visual input is first decomposed into a set of topographic feature maps. Different spatial locations then compete for saliency within each map, such that only locations which locally stand out from their surround can persist. All feature maps feed in a purely bottom-up manner, into a master saliency map which topographically codes for local conspicuity over the entire visual scene. In primates, such a map is believed to be located in the posterior parietal cortex as well as in the various visual maps in the pulvinar nuclei of the thalamus. The general architecture of Itti's model is shown in Figure 2.1. Below, we briefly review essential details of the model, as shown by the three stages in Figure 2.1.

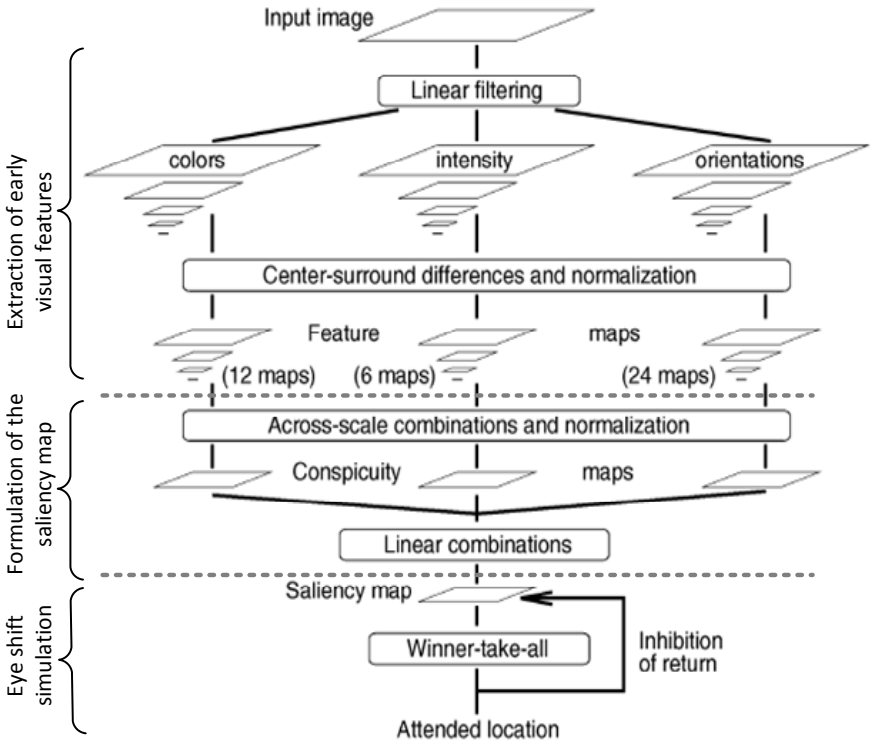

Figure 2.1: The general architecture of Itti's model (the picture is taken from [6] and is reproduced).

2.1.1 Extraction of early visual features

Denote the red, green, and blue channels of an input image as r, g, b. The intensity channel I is computed as $(r + g + b)/3$. Meanwhile, four color-enhanced channels R, G, B, Y are defined in Itti's model:

$$R = r - (g + b)/2 \tag{2.1}$$
$$G = g - (r + b)/2 \tag{2.2}$$
$$B = b - (r + g)/2 \tag{2.3}$$
$$Y = (r + g)/2 - |r - g|/2 - b \tag{2.4}$$

where negative values of R, G, B, Y are set to zeros. Then, Gaussian pyramids are created upon these channels as $I(\sigma), R(\sigma), G(\sigma), B(\sigma), Y(\sigma)$, where $\sigma \in [0, 1, ..., 8]$ is the scale parameter. In addition to intensity and color features, orientation features are considered as well. Let $O(\sigma, \theta)$ denote the Gabor pyramid obtained from I, where $\sigma \in [0, 1, ..., 8]$ indicates the

scale and $\theta \in \{0, \pi/4, \pi/2, 3\pi/4\}$ indicates the orientation. After all primitive features are prepared at hand, Itti's model defines a "center-surround" operation which characterizes the stimuli strength of different feature types fed to visual neural cells. Such a center-surround operation is realized through the difference of Gaussians, denoted by notation "$\ominus$". Note that the across-scale difference "$\ominus$" between different Gaussian scales is calculated by interpolation to the finer scale and then point-by-point subtraction. The center-surround difference upon intensity feature is calculated as:

$$I(c, s) = |I(c) \ominus I(s)| \tag{2.5}$$

where $c \in \{2, 3, 4\}$ corresponds to the "center", namely a pixel at a fine scale, whereas $s = c + \delta, \delta \in \{3, 4\}$, corresponds to the "surround", namely a pixel at a coarse scale. Similarly, the center-surround difference upon color features is calculated by considering the "color double-opponent" system (including red/green and yellow/blue) in human primary visual cortex:

$$RG(c, s) = |(R(c) - G(c)) \ominus (G(s) - R(s))| \tag{2.6}$$
$$BY(c, s) = |(B(c) - Y(c)) \ominus (Y(s) - B(s))| \tag{2.7}$$

Likewise, the center-surround difference upon orientation features is computed for different orientation θ as:

$$O(c, s, \theta) = |O(c, \theta) \ominus O(s, \theta)| \tag{2.8}$$

In total, 42 feature maps (see the first stage in Figure 2.1) are computed: 6 for intensity, 12 for colors, and 24 for orientations.

2.1.2 Formulation of saliency map

In the absence of top-down supervision, Itti's model adopts a map normalization operator $\mathcal{N}(\cdot)$, which globally promotes maps where a small number of strong peaks of activity (conspicuous locations) are present. Meanwhile $\mathcal{N}(\cdot)$ globally suppresses maps that contain numerous comparable peak responses. $\mathcal{N}(\cdot)$ consists of the following two steps: (a) Large amplitude differences are eliminated by normalizing the map values to a fixed range $[0, 1, ..., M]$, where M is the global maximum of the map; (b) Multiply the map by $(M - \bar{m})^2$, where $\bar{m}$ is the average of all its other local maxima. The underlying biological motivation for $\mathcal{N}(\cdot)$ is lateral cortical inhibition mechanism, in which neighboring similar features inhibit each other via specific, anatomically defined connections.

Feature maps are combined into three "conspicuity maps" (Figure 2.1), $\bar{I}$ for intensity, $\bar{C}$ for colors, and $\bar{O}$ for orientation, at the scale 4 of saliency map. They are obtained through across-scale summation, denoted by

notation "$\bigoplus$" which consists of reduction of each map to the scale 4 and point-by-point addition. The computation of $\bar{I}$, $\bar{C}$ and $\bar{O}$ is as follows:

$$\bar{I} = \bigoplus_{c=2}^{4} \bigoplus_{s=c+3}^{c+4} \mathcal{N}(I(c,s)) \tag{2.9}$$

$$\bar{C} = \bigoplus_{c=2}^{4} \bigoplus_{s=c+3}^{c+4} [\mathcal{N}(RG(c,s)) + \mathcal{N}(BY(c,s))] \tag{2.10}$$

$$\bar{O} = \sum_{\theta \in \{0,\pi/4,\pi/2,3\pi/4\}} \mathcal{N}[\bigoplus_{c=2}^{4} \bigoplus_{s=c+3}^{c+4} \mathcal{N}(O(c,s,\theta))] \tag{2.11}$$

The final saliency map S is obtained by normalization and a linear combination (Figure 2.1) of these three conspicuity maps:

$$S = \frac{1}{3}(\mathcal{N}(\bar{I}) + \mathcal{N}(\bar{C}) + \mathcal{N}(\bar{O})) \tag{2.12}$$

2.1.3 Simulation of eye shift

The inhibition-of-return (IR) process in Figure 2.1 is part of Itti's model that is employed to simulate human eye shift in free viewing. It detects the most salient location and directs focus of attention (FOA) towards it. After that, IR in a short time suppresses this salient location in the saliency map and its neighborhood in a small radius (equal to the radius of FOA), such that FOA is autonomously directed to the next most salient location. The suppression is achieved by setting saliency map values to 0. The following iteration will find the most salient point in a different location. This iterative process stops when the maximum of the saliency map reaches below a certain threshold. Computationally, the IR performs a similar process of selecting the global and local maxima. Since there are no any top-down attentional components modeled, the radius of FOA in Itti's model is fixed to one sixth of the smaller of the input image width or height [6]. In practice, IR is realized by a biologically plausible 2D "winner-take-all" (WTA) neural network [2] at the scale 4. The time constants, conductances, and firing thresholds of the simulated neurons are chosen so that the FOA jumps from one salient location to the next in approximately 30–70 ms, and that an attended area is inhibited for approximately 500–900 ms. The difference in the relative magnitude of these delays proves sufficient to ensure thorough scanning of the image and prevents cycling through only a limited number of locations.

2.2 Global regional contrast for salient region detection

The work of Cheng *et al.* [7] is one of the earliest to introduce regional contrast based salient region detection, which simultaneously evaluates global contrast differences and spatial weighted coherence scores. The motivation behind such contrast analysis is that human cortical cells may be hard-wired to preferentially respond to high contrast stimulus in their receptive fields. This is somewhat related to and coincides with the "center-surround" feature contrast hypotheses in Itti's model [6]. However, compared to Itti's model which typically measures local contrast, the method proposed by Cheng *et al.* [7] is global and use regions from image segmentation to capture non-local contrast information, and hence is more suitable for detecting a large-scale salient object from its surroundings. Below this thesis briefly reviews the regional contrast (RC) algorithm proposed by Cheng *et al.* [7].

In RC, an input image is firstly pre-segmented into non-overlapping regions using some image segmentation algorithm. Then a regional saliency map is computed, where saliency of a region is measured by the global color contrast between this target region and all other regions in the image. Suppose an input image is pre-segmented into N regions denoted as $\{r_i\}_{i=1}^{N}$. In RC, the global regional contrast saliency for a region r_i is computed as:

$$S(r_i) = \sum_{j=1}^{N} w(r_j) D_r(r_i, r_j) \tag{2.13}$$

where $D_r(r_i, r_j)$ is related to the appearance contrast between two regions. In RC, Cheng *et al.* adopt the color histograms of regions, which are extracted from quantized color space. Therefore, $D_r(r_i, r_j)$ is computed as:

$$D_r(r_i, r_j) = \sum_{l=1}^{n_i} \sum_{m=1}^{n_j} f(c_{i,l}) f(c_{j,m}) D(c_{i,l}, c_{j,m}) \tag{2.14}$$

where $f(c_{k,l})$ is the probability (in the normalized color histogram) of the l-th color $c_{k,l}$ among all n_k colors in the k-th region $r_k, k \in \{i, j\}$, and $D(c_{i,l}, c_{j,m})$ is the Euclidean distance between two colors in the color space. $w(r_j)$ is the weight for region r_j during contrast computation and is defined by considering both spatial influence and region size:

$$w(r_j) = \exp(-D_s(r_i, r_j)/\sigma_s^2)|r_j| \tag{2.15}$$

where $D_s(r_i, r_j)$ is the spatial distance between region r_i and r_j, and σ_s controls the strength of spatial weighting. Large values of σ_s reduce the effect of spatial weighting, so that contrast to farther regions would contribute more to the saliency of the current region. By letting $\sigma_s \to \infty$,

equal weights for all regions are resulted. $|r_j|$ is the size of region r_j to emphasize color contrast to bigger regions.

2.3 Graph-related theory

2.3.1 Fundamentals

Graph theory studies the graphs, which are mathematical structures for modeling relations between objects. A graph in this context consists of vertices (also called nodes or points) and edges (also called arcs or lines). A graph may be undirected, meaning that there is no distinction of directions of edges, or directed, meaning an edge points from one vertex to another. Since graphs involved in this thesis work are undirected, the introduction below will be focused on undirected graphs.

Let a graph G be denoted by $G = (V, E)$, where V is the set of vertices and E is the set of edges. Specially, E can be expressed as 2-element subsets of V. For example, an edge $e_{ij} = \langle v_i, v_j \rangle$ corresponds to the connection between two vertices v_i and v_j, and for an undirected graph, $\langle v_i, v_j \rangle$ and $\langle v_j, v_i \rangle$ are the same. In a graph, an edge is usually rendered with some weight, which is called edge weight as shown in Figure 2.2. The edge weights of graph should reflect certain relation between two vertices, usually either distinction or similarity. Note that "similarity", in some context, is also called "affinity". Below, we describe the definitions of adjacency matrix, edge weight matrix, degree matrix, and Laplacian matrix in the graph theory, which are frequently used:

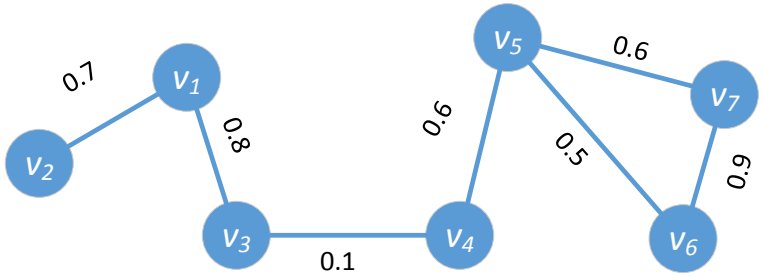

Figure 2.2: An undirected graph with 7 vertices ($v_1 \sim v_7$). The values beside edges are corresponding edge weights.

Adjacency matrix: An adjacency matrix is a square matrix used to represent the connections of a graph. The rows and columns are both indexed by vertices of the graph whereas the elements of the matrix indicate whether pairs of vertices are adjacent or not in the graph. The adjacency matrix is a binary matrix ($\{0, 1\}$ valued) often with 0 on its diagonal. If the graph is

undirected, the adjacency matrix is symmetric. For instance, the adjacency matrix of the graph in Figure 2.2, denoted as $\mathbf{A}$, is shown below:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix} \tag{2.16}$$

Edge weight matrix: Edge weight matrix is very similar to the adjacency matrix, however, its elements indicate edge weights of the corresponding edges. The elements equal to 0 if no edges exist between pairs of vertices. One can simply formulate an edge weight matrix by replacing the 1 entries in an adjacency matrix with the corresponding edge weights of the graph[1]. For example, the edge weight matrix of the graph in Figure 2.2, denoted as $\mathbf{W}$, is shown below:

$$\mathbf{W} = \begin{bmatrix} 0 & 0.7 & 0.8 & 0 & 0 & 0 & 0 \\ 0.7 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.8 & 0 & 0 & 0.1 & 0 & 0 & 0 \\ 0 & 0 & 0.1 & 0 & 0.6 & 0 & 0 \\ 0 & 0 & 0 & 0.6 & 0 & 0.5 & 0.6 \\ 0 & 0 & 0 & 0 & 0.5 & 0 & 0.9 \\ 0 & 0 & 0 & 0 & 0.6 & 0.9 & 0 \end{bmatrix} \tag{2.17}$$

Degree matrix: The degree matrix is a diagonal matrix where its ith diagonal entry indicates the degree of v_i. Note in different situations, the degree matrix of the adjacency matrix or the degree matrix of the edge weight matrix may be used. Since in this thesis, the term "degree matrix" usually refers to the degree matrix of edge weight matrix, therefore below the definition of the degree matrix is based on edge weight matrix. Based on such an edge weight matrix $\mathbf{W}$, the degree of the ith vertex v_i is defined as $d_{ii} = \sum_j w_{ij}$, where w_{ij} is the entry of $\mathbf{W}$. For example, the degree matrix of the graph in Figure 2.2, denoted as $\mathbf{D}$, is shown below:

[1] In some literatures, the edge weight matrix $\mathbf{W}$ of a graph is still called adjacency matrix. In those context, adjacency matrix and edge weight matrix indeed refer to the same thing, namely the edge weight matrix $\mathbf{W}$ described in this section. However, to be precise and distinguishable, they refer to different things in this section.

$$\mathbf{D} = \begin{bmatrix} 1.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.7 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.9 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.7 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.7 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1.5 \end{bmatrix} \tag{2.18}$$

Laplacian matrix: The Laplacian matrix, also called graph Laplacian, is defined as $\mathbf{L} = \mathbf{D} - \mathbf{W}$. This matrix acts as a very important role in many sub-fields of graph theory such as random walk, graph-based semi-supervised learning, spectral graph theory. The Laplacian matrix of the graph in Figure 2.2, denoted as $\mathbf{L}$, is shown below:

$$\mathbf{L} = \begin{bmatrix} 1.5 & -0.7 & -0.8 & 0 & 0 & 0 & 0 \\ -0.7 & 0.7 & 0 & 0 & 0 & 0 & 0 \\ -0.8 & 0 & 0.9 & -0.1 & 0 & 0 & 0 \\ 0 & 0 & -0.1 & 0.7 & -0.6 & 0 & 0 \\ 0 & 0 & 0 & -0.6 & 1.7 & -0.5 & -0.6 \\ 0 & 0 & 0 & 0 & -0.5 & 1.4 & -0.9 \\ 0 & 0 & 0 & 0 & -0.6 & -0.9 & 1.5 \end{bmatrix} \tag{2.19}$$

The Laplacian matrix has many interesting properties:

1. For every vector $\mathbf{f} \in \mathbb{R}^n$, $\mathbf{f}^T\mathbf{L}\mathbf{f} = \frac{1}{2}\sum_{i,j=1}^{n} w_{ij}(f_i - f_j)^2$, where f_i is the ith component of $\mathbf{f}$ and n is the dimension of $\mathbf{W}$ and $\mathbf{L}$.
2. For an undirected graph whose edge weights are non-negative, $\mathbf{L}$ is positive semi-definite.
3. Along with item 2, the smallest eigenvalue of $\mathbf{L}$ is 0. The corresponding eigenvector is the constant one vector.
4. Along with item 2, $\mathbf{L}$ has n non-negative, real-valued eigenvalues.
5. Along with item 2, the multiplicity of the eigenvalue 0 of $\mathbf{L}$ equals the number of connected components in the graph.

From the examples above, one can see that there are two essential questions to answer when building a graph. The one is whether there exists an edge between two vertices. The other is how to define the weight of an edge. Below we describe four ways commonly adopted for computing graph edge weights.

Binary 0/1 weight: If there exists an edge, then the corresponding edge weight is 1. In this case, the edge weight matrix $\mathbf{W}$ degrades to the adjacency matrix $\mathbf{A}$.

Feature distance: Suppose that a graph node v_i associates with a feature vector, denoted as $\mathbf{f}_i$. Then the edge weight can be computed as $w_{ij} = ||\mathbf{f}_i - \mathbf{f}_j||$, which typically measures the distinction between two nodes.
Gaussian similarity: In the situations where graph edge weights need to reflect similarity (or affinity) between vertices, the most common weights are induced by Gaussian kernel function:

$$w_{ij} = \exp(-\frac{||\mathbf{f}_i - \mathbf{f}_j||^2}{2\sigma^2}) \tag{2.20}$$

where σ is the kernel width. The closer the two nodes are in feature space, the larger the corresponding w_{ij} will be. The w_{ij} generated in this way ranges from 0 to 1. Noting that a graph edge weight matrix $\mathbf{W}$ where entries represent the similarity degree (such as those induced by Gaussian kernel) is often called *affinity matrix*. The corresponding graph is then called "similarity graph" or "affinity graph".
Locally linear embedding: Assume each graph node v_i can be optimally reconstructed using a linear combination of its graph neighbors $v_j \in \mathcal{N}(v_i)$ (where $\mathcal{N}(\cdot)$ denotes the neighborhood) in feature space, then the weights of edges connecting to v_i can be obtained automatically by solving the following optimization problem [45]:

$$\min_{w_{ij}} ||\mathbf{f}_i - \sum_{j|v_j \in \mathcal{N}(v_i)} w_{ij}\mathbf{f}_j||^2 \tag{2.21}$$

$$s.t. \quad \sum_j w_{ij} = 1, \quad w_{ij} \geq 0 \tag{2.22}$$

Besides the above four common ways of computing edge weights, other means for graph construction exist. However, how to build up graph edges meanwhile choosing an appropriate way for edge weight computation depends on specific applications.

2.3.2 Geodesic distance

Geodesic distance is originally defined as the length of the shortest path (called geodesic) between any pair of points on a nonlinear manifold. Figure 2.3 gives an example, where a 2-D manifold is embedded in a 3-D space $\mathbb{R}^3$. In the graph theory, the geodesic distance is usually considered between two vertices/nodes as the distance metric on a graph, which corresponds to the length of the shortest graph path. Below geodesic distance and its definition on images [46] are reviewed since they are relevant to this thesis work. The geodesic distance and its transform can be applied to image segmentation, edge-preserving filtering, denoising, stitching, and colorization [46]. Let $I(x)$ be an image: $\Psi \to \mathbb{R}^d$ ($d = 3$ for a color image while $d = 1$ for an intensity image), whose support $\Psi \subset \mathbb{R}^2$ is assumed to be continuous for the time

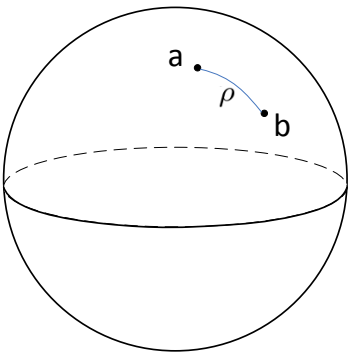

Figure 2.3: Example of a 2-D manifold (sphere) embedded in a 3-D space $\mathbb{R}^3$. a and b are manifold points. The geodesic ρ is the shortest curve between a and b on the manifold, and its length is called geodesic distance.

being. Given two points $a, b \in \Psi$, the geodesic distance between them on the image is defined as:

$$d_{geo}(a,b) = \inf_{\Gamma \in \mathcal{P}_{a,b}} \int_0^{l(\Gamma)} \sqrt{1 + \gamma^2 (\nabla I(s) \cdot \Gamma'(s))^2} ds \tag{2.23}$$

where $\mathcal{P}_{a,b}$ is the set of all possible differentiable paths between a and b. The spatial derivative $\Gamma'(s) = \partial\Gamma(s)/\partial(s)$ is the unit vector tangent to the direction of the path with arc length s, and $\nabla I(s)$ is the corresponding gradient vector. The dot-product $\nabla I(s) \cdot \Gamma'(s)$ computes the image gradient magnitude along the tangent. The geodesic factor γ weighs the contribution between the gradient and the spatial distance. When $\gamma = 0$, the integration will turn to the length of path, i.e., $l(\Gamma)$, and $d_{geo}(a,b)$ degenerates to Euclidean distance. Figure 2.4 gives an example to help understand the definition (2.23). Note that the definition (2.23) is an adaption of the generic definition of geodesic distance to images, since an image may be treated as an intrinsic manifold. Given a "seed" region Ω, one can define the *geodesic distance transform* D of pixel x as its minimum geodesic distance from Ω:

$$D(x;\Omega) = \min_{\{x' \in \Omega\}} d_{geo}(x, x') \tag{2.24}$$

In (2.23) the support $\Psi \subset \mathbb{R}^2$ is assumed continuous. However, in practice an image is pixel lattice and thereby has pixels with discrete coordinates. From (2.23), we can derive a discrete approximation for image lattice:

$$d_{geo}(a,b) = \min_{\Gamma \in \mathcal{P}_{a,b}} \sum_{p_k \in \Gamma} \sqrt{1 + \gamma^2 (\nabla I(p_k, p_{k+1}))^2} D_s(p_k, p_{k+1}) \tag{2.25}$$

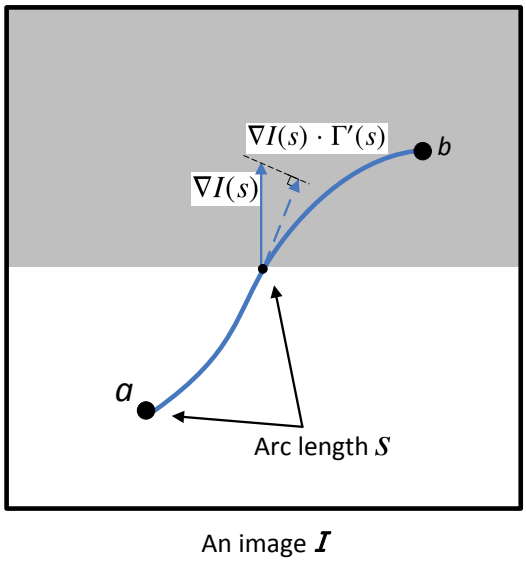

Figure 2.4: Example on a simple synthetic image to help understand definition (2.23). The synthetic image is gray-level and has its upper half to be gray (low intensity) and the lower half to be white (high intensity). The blue curve connecting two point a and b in the image indicates a path $\Gamma \in \mathcal{P}_{a,b}$. The solid blue arrow means the gradient vector $\nabla I(s)$ at the current point where as the dash blue arrow reveals the corresponding tangent direction along the path.

where p_k is the kth point on the discrete path Γ, $\nabla I(p_k, p_{k+1})$ is the gradient magnitude between p_k, p_{k+1}, and $D_s(p_k, p_{k+1})$ is the spatial distance between p_k, p_{k+1}. When further ignoring the influence of spatial distance but considering only the gradient, namely $\gamma \to \infty$, the following variation can be obtained:

$$d_{geo}(a,b) = \min_{\Gamma \in \mathcal{P}_{a,b}} \sum_{p_k \in \Gamma} \nabla I(p_k, p_{k+1}) D_s(p_k, p_{k+1}) \tag{2.26}$$

The computation of (2.26) (and also (2.25)) can be transferred to another perspective—computing the geodesic distance on a graph, where each point p_k of the lattice is a vertex while $\nabla I(p_k, p_{k+1}) D_s(p_k, p_{k+1})$ between two adjacent lattice points is the edge. Hence, calculating (2.26) is equivalent to finding the shortest path on a graph, which rightly coincides with the geodesic distance on the graph as aforementioned. Typically Floyd's/Dijkstra's algorithms are applied, but there also exist speedup computation techniques like raster scanning [47] for 4-adjacency/8-adjacency

neighborhood on image lattice. Since the geodesic distance on an image can describe the connectivity degree between two image locations, whereas objects usually present connectivity properties, superpixel-based geodesic distance is employed in this thesis for salient region detection and traffic sign recognition.

2.3.3 Normalized graph cut (Ncut)

Graph cut is a class of methods for finding a partition of a similarity graph, such that edges between different groups have low weights (indicating low similarities) and meanwhile edges within a group have high weights (indicating high similarities). It has a similar spirit to the aim of clustering: points within a same cluster are similar to each other while points in different clusters are dissimilar from each other. This section reviews a well-established graph cut method—normalized graph cut (Ncut) proposed by Shi *et al.* [48], which is relevant to this thesis work. Since the Ncut and spectral clustering are tightly related, Ncut is first reviewed and its connection to spectral clustering is then described.

From the partition point of view, a simplest example to start with is the min-cut example. Suppose a similarity graph $G = (V, E)$. Let $\mathbf{W}$ be the edge weight matrix of the graph, $\mathbf{D}$ be the degree matrix, and $\mathbf{L} = \mathbf{D} - \mathbf{W}$ be the Laplacian matrix. A cut is a partition of the graph vertices into two disjoint subsets A and B, where such cut can be specified by a series of edges. The corresponding cut cost is defined as $cut(A, B) := \sum_{v_i \in A, v_j \in B} w_{ij}$. Generally, for a given number k ($k \geq 2$) of subsets, the min-cut chooses a partition $\{A_1, ..., A_k\}$ that minimizes:

$$cut(A_1, ..., A_k) = \sum_{i=1}^{k} cut(A_i, \bar{A}_i) \tag{2.27}$$

where $\bar{A}$ denotes the complement of A. In particular for $k = 2$, min-cut is to search for a partition $\{A, B\}$ $(B = \bar{A})$ where the $cut(A, B)$ is minimized. This problem is relatively easy and can be solved efficiently. Unfortunately, in practice the min-cut often does not give satisfactory partition, since it often separates one (or few) individual vertex from the rest of the graph [48]. This is not satisfying to achieve in clustering because clusters should be *reasonably large* groups of points. An effective objective function to overcome this issue is by normalizing the cut values using cluster sizes, leading to the normalized cut (Ncut). Ncut is originally proposed by Shi *et al.* [48] aimed at minimizing:

$$Ncut(A_1, ..., A_k) = \sum_{i=1}^{k} \frac{cut(A_i, \bar{A}_i)}{assoc(A_i, V)} \tag{2.28}$$

where $assoc(A_i, V) := \sum_{v_l \in A_i, v_m \in V} w_{lm}$ is a measure of set size, i.e., the larger the cardinality $|A_i|$ is, the higher $assoc(A_i, V)$ will be. By defining a hard indicating vector for each A_i and relaxing the hard constraints (for details please refer to [48, 49]), the continuous indicating vectors for Ncut can be derived from the first k eigenvectors corresponding to the smallest eigenvalues of $\mathbf{D}^{-1}\mathbf{L}$, or the first k generalized eigenvectors[2] of:

$$(\mathbf{D} - \mathbf{W})\mathbf{u} = \lambda \mathbf{D}\mathbf{u} \tag{2.29}$$

where $\mathbf{u}$ and λ denote the eigenvector and eigenvalue. The solution of 2-way Ncut ($k = 2$ case) is given by its second smallest eigenvector.

Since the continuous indicating vectors for multi-cluster Ncut are derived from the first k generalized eigenvectors of system $(\mathbf{D} - \mathbf{W})\mathbf{u} = \lambda \mathbf{D}\mathbf{u}$ and contain valuable cluster information, k-means clustering can be applied to these eigenvectors to obtain labels corresponding to clusters, leading to the so-called spectral clustering (Algorithm 1).

Algorithm 1 Spectral Clustering

Require: Constructed similarity graph described by $\mathbf{W}$, cluster number k.
Ensure: Clusters $A_1, ..., A_k$ with $A_i = \{j | \mathbf{y}_j \in C_i\}$, where $\mathbf{y}_j$ and C_i are defined below.

1: Compute the degree matrix $\mathbf{D}$ and graph Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{W}$.
2: Compute the first k generalized eigenvectors $\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_k$ of the generalized eigen-problem $\mathbf{L}\mathbf{u} = \lambda \mathbf{D}\mathbf{u}$.
3: Let $\mathbf{U} \in \mathbb{R}^{n \times k}$ be the matrix containing the vectors $\mathbf{u}_1, ..., \mathbf{u}_k$ as columns.
4: For $i = 1, 2, ..., n$, let $(\mathbf{y}_i \in \mathbb{R}^k)$ be the vector corresponding to the i-th row of $\mathbf{U}$.
5: Cluster the points $(\mathbf{y}_i)_{i=1,...,n}$ in $\mathbb{R}^k$ with the k-means algorithm into clusters $C_1, ..., C_k$ and output clusters $A_1, ..., A_k$ with $A_i = \{j | \mathbf{y}_j \in C_i\}$.

2.3.4 Conditional random field (CRF)

Conditional random field (CRF) is originally proposed by Lafferty *et al.* [50] for labeling sequence data. Its definition is given according to [50]: *"Suppose X is a set of random variables over data sequences to be labeled, and Y is a set of random variables over corresponding label sequences. $G = (V, E)$ defines a graph constructed from the data sequences such that $Y = (Y_v)_{v \in V}$ is indexed by the vertices of G. (X, Y) formulates a conditional random field in case, when conditioned on X, each Y_v obeys the Markov property*

[2] Hereafter in this section, we use "the first k eigenvectors" instead of "the first k eigenvectors corresponding to the smallest eigenvalues" for simplicity.

with respect to the graph: $p(Y_v|X, Y_w, w \neq v) = p(Y_v|X, Y_w, w \sim v)$*, where* $w \sim v$ *means* w *and* v *are neighbors in* G*."* Such Markov property infers that the probability of Y_v is conditioned on both data sequence X and also labels, i.e., Y_w of neighboring vertices. CRF has become a popular class of statistical modelling methods in pattern recognition and machine learning, where they are used for *structured prediction*. In computer vision, CRFs are often used for semantic segmentation [51–53]. The below introduction for CRF is mainly about image labeling task, which is relevant to this thesis work.

For the image labeling task, given an image $\mathbf{x}$, the conditional probability distribution of a label configuration $\mathbf{y}$ (vector form) on the CRF is defined as:

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \exp\{-\mathbb{E}(\mathbf{y}, \mathbf{x})\} \tag{2.30}$$

where $\mathbb{E}(\mathbf{y}, \mathbf{x})$ is the energy function and $Z(\mathbf{x})$ is the partition function (i.e., normalization function) which sums over all possible label state. In a discrete case, the partition function is written as $Z(\mathbf{x}) = \sum_{\mathbf{y}} \exp\{-\mathbb{E}(\mathbf{y}, \mathbf{x})\}$, whereas in a continuous case, the partition function is written as $Z(\mathbf{x}) = \int_{\mathbf{y}} \exp\{-\mathbb{E}(\mathbf{y}, \mathbf{x})\} \mathrm{d}\mathbf{y}$. The energy function can be expressed as unary terms plus pairwise terms as:

$$\mathbb{E}(\mathbf{y}, \mathbf{x}) = \sum_i \underbrace{U_\alpha(y_i, \mathbf{x})}_{\text{Unary term}} + \sum_{i,j,i\sim j} \underbrace{P_\beta(y_i, y_j, \mathbf{x})}_{\text{Pairwise term}} \tag{2.31}$$

where y_i is the ith element of the label vector $\mathbf{y}$, vector α contains the parameters for unary potentials, and vector β contains the parameters for pairwise potentials. As introduced before, a CRF is often coupled with the definition of an undirected graph $G(V, E)$, where V is the set of graph nodes and E is the set of graph edges. The label assigned to each graph node $v_i \in V$ is denoted by y_i. In (2.31), the notation "$i \sim j$" means v_i and v_j are graph neighbors. The unary term U_α represents the dependency between a label and the image $\mathbf{x}$ at a specific node, whereas the pairwise term P_β encourages neighboring graph nodes to take similar labels (i.e., enforces labeling consistency). A general graphic model of CRF for image labeling task is given in Figure 2.5 (a), where a white vertex represents a label and the gray vertex represents the entire image. It can be seen that compared to an ordinary prediction model (Figure 2.5 (b)) where labels are predicted independently, a CRF (Figure 2.5 (a)) takes context into account.

Assuming that the parameters (α, β) of a CRF are given or estimated by learning, the optimal labeling vector $\mathbf{y}$ is often inferred by maximizing (2.30), or equivalently minimizing the negative logarithm of (2.30) as:

$$-\log p(\mathbf{y}|\mathbf{x}) = \mathbb{E}(\mathbf{y}, \mathbf{x}) + \log Z(\mathbf{x}) \tag{2.32}$$

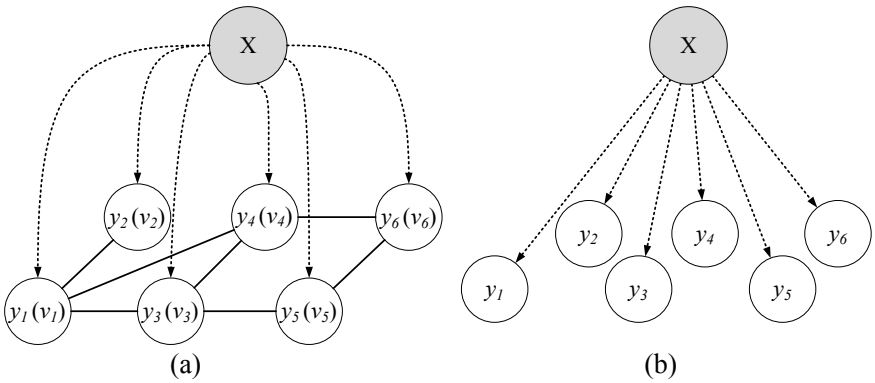

Figure 2.5: (a): A general graphic model of CRF for image labeling task. A white vertex (v_i) represents a label (y_i) and the gray vertex represents the entire image ($\mathbf{x}$). The dash arrows indicate the unary dependencies (conditions) while the solid lines indicate the pairwise relations associating with a graph, where the spirit of structured prediction can be seen. (b): An ordinary prediction model.

Since in the inference stage $\log Z(\mathbf{x})$ is a constant with respect to $\mathbf{y}$, one can choose to directly minimize the energy function $\mathbb{E}(\mathbf{y}, \mathbf{x})$. However, in many discrete cases, the exact inference of CRF is NP hard. This is because computing $Z(\mathbf{x})$ is usually intractable since it is summed over the exponentially possible assignments to $\mathbf{y}$. Several algorithms exist to obtain approximate solutions for such inference, including loopy belief propagation, mean field inference, linear programming relaxations and so on. For further introduction on parameter learning and inference of CRF, see [54] and [55].

2.3.5 Graph-based semi-supervised learning

Unlike ordinary supervised learning which only leverages labeled data, semi-supervised learning makes use of both labeled data (typically a small amount of labeled data) together with a large amount of unlabeled data for training. It has tremendous practical value. In many tasks, there is a paucity of labeled data and the labels may be difficult to obtain because they require large human annotation effort, special devices, or expensive and slow experiments. Figure 2.6 shows a straightforward comparison of semi-supervised learning to traditional supervised learning. Due to the use of unlabeled data, semi-supervised learning is capable of discovering latent data distribution. Specifically, graph-based semi-supervised learning is about how to induce the labels of unlabeled data from the graph structure when given only the labels of some labeled data.

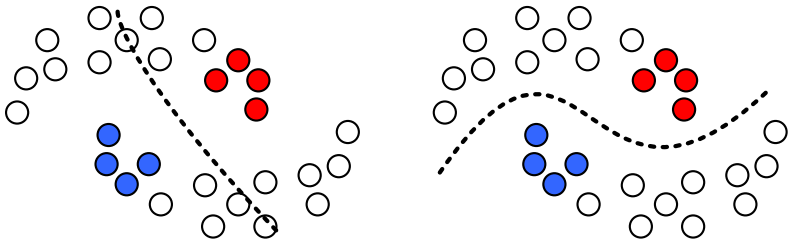

Figure 2.6: A comparison between supervised learning (left) and semi-supervised learning (right). Blue and red dots represent the labeled data in two classes whereas the white dots represent unlabeled data. The dash curve indicate the generated classification boundary.

Graph-based semi-supervised learning starts by constructing a graph from the training data. Given training data $\{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$ and $\{(\mathbf{x}_i)\}_{i=l+1}^{l+u}$, where $\{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$ are labeled data, $\{(\mathbf{x}_i)\}_{i=l+1}^{l+u}$ are unlabeled data, and $\mathbf{x}_i, y_i$ are the corresponding data vector and label, respectively. The vertices of the graph comprise labeled and unlabeled instances $\{(\mathbf{x}_i)\}_{i=1}^{l} \cup \{(\mathbf{x}_i)\}_{i=l+1}^{l+u}$, and the goal is to infer the unknown labels, i.e., $\{(y_i)\}_{i=l+1}^{l+u}$. This is made possible by graph edges that connect labeled vertices to unlabeled vertices. In graph-based semi-supervised learning, the graph edges should represent the *similarities* of the instances, and the idea is that if the edge weight between two vertices is large, then their labels are expected to be the same. Since in Section 2.3.1 we have already introduced the common ways for computing edge weights, below we further introduce some common ways in machine learning field to construct graph structure from a pool of given instances.

By treating each instance as a vertex in the graph, the following four ways are usually adopted for graph construction:

- **Fully connected graph**: Every pair of instances is connected by an edge.
- **ε-neighborhood graph**: If two instances satisfy $||\mathbf{x}_i - \mathbf{x}_j|| < \varepsilon$, then there will be an edge connecting them. ε is a threshold pre-determined.
- **k-nearest neighbourhood graph (kNN graph)**: Each instance defines its k-nearest neighbor instances in Euclidean distance. If one instance $\mathbf{x}_i$ is in the k-neighborhood of another instance $\mathbf{x}_j$, or $\mathbf{x}_j$ is in the k-neighborhood of $\mathbf{x}_i$, then they are connected by an edge. This means of constructing graph is very popular in machine learning field.

- **Mutual kNN graph**: Only when $\mathbf{x}_i$ is in the k-neighborhood of $\mathbf{x}_j$ and meanwhile $\mathbf{x}_j$ is in the k-neighborhood of $\mathbf{x}_i$, they are connected by an edge. Note the difference of the mutual kNN graph compared to the kNN graph, which makes the mutual kNN graph much sparser than the kNN graph.

The graph-based semi-supervised learning can be formulated as learning a label function f from the graph, which conducts the mapping $f(\mathbf{x}_i) \rightarrow y_i$. Below we briefly review three different graph-based semi-supervised learning algorithms. For simplicity, the case of binary labels $y_i \in \{-1, 1\}$ is considered. Similarly as before, we use notation $\mathbf{W}$ for the edge weight matrix and $\mathbf{L}$ for the Laplacian matrix of the graph.

Min-cut: In this approach, the positive labeled instances are treated as "source" vertices, as if some fluid is flowing out of them and through the edges. Similarly, the negative labeled instances are "sink" vertices, where the fluid would disappear. Min-cut can be used to find a partition of the graph into two sets, under the constraint that one set contains all the "source" vertices and the other contains all the "sink" vertices. Once the graph is split, the vertices connecting to the sources are labeled positive, and those to the sinks are labeled negative. Mathematically, the min-cut is to find a binary label function $f(\mathbf{x}) \in \{-1, 1\}$ by solving the below energy minimization:

$$\min_{f(\mathbf{x})|f(\mathbf{x})\in\{-1,1\}} \sum_{i,j|f(\mathbf{x}_i)\neq f(\mathbf{x}_j)} w_{ij} \quad , \text{ s.t. } \{f(\mathbf{x}_i) = y_i\}_{i=1}^{l} \tag{2.33}$$

The above min-cut problem can be treated as an integer programming problem because f is constrained to produce discrete values -1 or 1. Efficient polynomial-time algorithms exist to solve such a min-cut problem.

Harmonic function: In this approach, the discrete constraint for f is first relaxed to $\mathbb{R}$. By introducing a pairwise energy term, the f is found by solving the below energy minimization problem:

$$\min_{f(\mathbf{x})|f(\mathbf{x})\in\mathbb{R}} \sum_{i,j=1}^{l+u} w_{ij}(f(\mathbf{x}_i) - f(\mathbf{x}_j))^2 \quad , \text{ s.t. } \{f(\mathbf{x}_i) = y_i\}_{i=1}^{l} \tag{2.34}$$

The above pairwise energy enforces similar instances that are characterized by large w_{ij} to take similar labels, meanwhile the labels for labeled instances should definitely fit their original labels. The drawback of the relaxation is that in the solution, $f(\mathbf{x})$ is now a real value that does not directly correspond to a label. This can however be addressed by thresholding $f(\mathbf{x})$ at zero to produce discrete label predictions (i.e., if $f(\mathbf{x}_i) \geq 0$, predict $y_i = 1$, and

if $f(\mathbf{x}_i) < 0$, predict $y_i = 1$). It is worthy noting that since the regularizer $\sum_{i,j=1}^{l+u} w_{ij}(f(x_i) - f(x_j))^2 = 2\mathbf{f}^T\mathbf{L}\mathbf{f}$ where $\mathbf{f} = [f(x_1), f(x_2), ..., f(x_{l+u})]^T$, there exists closed-form solution for (2.34) by using Lagrange multipliers with matrix algebra.

Manifold regularization: Both min-cut and harmonic function fix $\{f(\mathbf{x}_i) = y_i\}_{i=1}^{l}$ for labeled instances. However, when some of the labels are wrong, one may prefer $f(\mathbf{x})$ to be able to occasionally disagree with the original labels. Manifold regularization is such an approach which further relaxes the constraint $\{f(\mathbf{x}_i) = y_i\}_{i=1}^{l}$ by adding an extra loss to the energy function:

$$\min_{f(\mathbf{x})|f(\mathbf{x})\in\mathbb{R}} \sum_{i,j=1}^{l+u} w_{ij}(f(\mathbf{x}_i) - f(\mathbf{x}_j))^2 + \lambda \sum_{i=1}^{l}(f(\mathbf{x}_i) - y_i)^2 \tag{2.35}$$

where $\lambda > 0$ is a parameter controlling the strength of fitness to the original labels. Also, there exist efficient algorithms to find the optimal $f(\mathbf{x})$. Note that besides the typical formulation of pairwise and loss term introduced in (2.35), there are many variations.

Remarks: In all, the graph-based semi-supervised learning assumption is that the labels should vary "smoothly" on the graph. That is, if two instances are connected by a strong edge, their labels tend to be the same. In this sense, it shares a common spirit with CRF mentioned before, since the CRF also enforces the label consistency over the graph. The main difference is that the CRF is a probabilistic framework for labeling.

2.4 Superpixel segmentation algorithm SLIC

Superpixel segmentation partitions image pixels into perceptually homogenous atomic regions, usually called "superpixels". An underlying concept of superpixels, which distinguishes them from common image regions, is that superpixels are generated from image over-segmentation, namely when more segments are generated than what may define whole object regions. In this sense, superpixels are highly over-segmented regions. By treating superpixels as processing units, it allows to replace the rigid structure constituted by image pixels. Abstracting images into superpixels offers the following advantages: 1) Eliminating unnecessary noise and details; 2) Capturing image redundancy and providing a convenient primitive for feature extraction and other subsequent processing; 3) Greatly reducing the computational complexity. Thanks to the above, superpixels have become key building blocks of many computer vision applications including image and object segmentation, object recognition, depth estimation, as well as saliency detection.

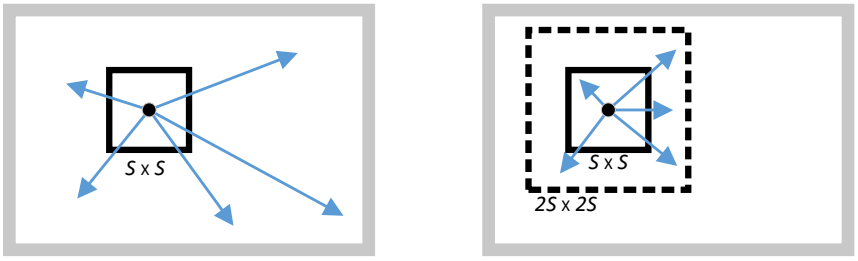

Figure 2.7: Difference of element (i.e., pixel) assignment between k-means and SLIC. In k-means (left), the assignment of a pixel (black dot) considers all cluster centers. However, in SLIC (right), the assignment of a pixel considers only cluster centers in a $2S \times 2S$ region (dash rectangle), which drastically reduces the cluster centers that need to consider for each pixel.

There exist many superpixel generation algorithms (see a survey in [56]). Below we briefly review an algorithm called Simple Linear Iterative Clustering (SLIC) proposed by Achanta *et al.* [56]. SLIC is extensively employed by existing work on saliency detection (e.g., [57] [58] [59–62]), and is often used by the methods proposed in this thesis as a pre-processing step.

SLIC is an adaption of k-means clustering. Similar to the cluster number in k-means algorithm, a crucial parameter in SLIC is k. It is the desired number of approximately equally-sized superpixels. For color images in the CIELab color space, the clustering procedure begins with an initialization step where k initial cluster centers $C_i = [l_i, a_i, b_i, x_i, y_i]^T$ are sampled on a regular grid. Here l_i, a_i, b_i, x_i, y_i respectively denote L, a, b color components in CIELab color space and spatial coordinates in the image plane. The grid interval is set as $S = \sqrt{N/k}$, where N represents the number of total pixels in an image. Each cluster center is later associated with a superpixel. To avoid the center being initialized on an edge or on a noisy pixel, Achanta *et al.* further suggest to switch a cluster center to the location corresponding to the lowest gradient in a 3×3 neighborhood.

The iterative clustering steps of SLIC are similar to those in the k-means, however, with a special modification on element assignment. Since in SLIC the expected spatial range of a superpixel is a region of approximate size $S \times S$, in each iteration an image pixel is only assigned to its nearest cluster center in a local spatial range $2S \times 2S$ around the pixel. An illustration for this is in Figure 2.7. The distance measure D which specifies the nearest neighbor is defined by considering both color and spatial distance as:

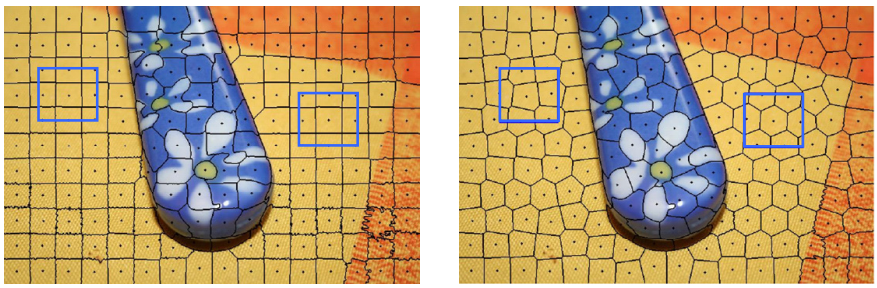

Figure 2.8: An example evolution of SLIC iteration process. Blue rectangles visualize the local $2S \times 2S$ range. The boundaries of superpixels are shown in black, whereas the locations of cluster centers are represented by the small black dots.

$$d_{lab} = \sqrt{(l_i - l_j)^2 + (a_i - a_j)^2 + (b_i - b_j)^2} \tag{2.36}$$

$$d_{xy} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}; \tag{2.37}$$

$$D = \sqrt{d_{lab}^2 + \frac{m^2}{S^2} d_{xy}^2} \tag{2.38}$$

where m is a parameter specifying the relative importance of spatial distance. When m is large, spatial proximity is more important and the resulting superpixels are more compact. When m is small, the resulting superpixels adhere more tightly to image boundaries but have less regular shape and size. After all pixels in the image are associated to the corresponding cluster centers. The cluster centers will be updated by computing the averages of the elements in individual clusters. This is the same as the updating process in k-means. In the next iteration, image pixels are associated with new clusters in a local range as aforementioned. The whole iteration process converges when the global residual error is below a threshold.

The time complexity of SLIC is $O(N)$, where N denotes the total number of pixels in the image. This means the time complexity of SLIC is independent from the superpixel number k. This is because a pixel only needs to be compared with the cluster centers in a local area $2S \times 2S$ on the image plane. A pixel falls into the local neighborhood of no more than eight cluster centers (Figure 2.8). Therefore, the convergence of SLIC can be achieved in 10 iterations [56]. This makes SLIC much efficient than k-means whose practical time complexity is $O(NkI)$, where I is the number of iterations needed for convergence.

To summarize, SLIC algorithm generates superpixels that align well to object boundaries, and is also fast to compute, memory-efficient, and simple

Figure 2.9: Sample results of segmenting images into superpixels of (approximate) size 64, 256, and 1024 pixels by using SLIC. Boundaries of superpixels are visualized in black. The superpixels are compact, uniform in size, and adhere well to region boundaries. This picture is taken from [56].

to use. It has very few parameters to tune, only including the superpixel number k and the weight of spatial distance m. Since the size of the superpixels can be estimated by N/k, some implementations of SLIC instead provide interface of superpixel size and m. For the quantitative results of comparing SLIC to state-of-the-art superpixel methods, interested readers are referred to [56]. Figure 2.9 shows several sample results from [56], where different numbers of SLIC superpixels are used for a same image.

Chapter 3

Overview of Related Work

This chapter reviews existing techniques and models on salient region/object detection and traffic sign recognition. Since this thesis mainly focuses on salient region detection, literatures on eye fixation prediction are not reviewed. For details on eye fixation prediction models, interested readers are referred to a comprehensive survey on visual attention modeling [5].

3.1 Salient region detection

We roughly subdivide existing methods of salient region/object detection into four categories: heuristic color contrast-based methods; learning-based methods; segmentation-assisted approaches; and graph-based saliency modeling. Methods beyond these four categories fall into the fifth category.

3.1.1 Heuristic color contrast-based methods

Methods of this category model the saliency using local or global color statistics. The underlying assumption is that salient objects "pop up" from their surroundings due to their unique color appearances. Thus they are supposed to present high color contrast to the rest image parts. Many methods for computing such contrast-based saliency have been proposed since 2006. Zhai *et al.* [63] introduce pixel-level saliency calculation based on histograms that only model luminance channel of an image. They define the saliency level of a pixel as its luminance contrast to all pixels in the image. Such global contrast computation is then converted into histogram analysis for efficiency. Achanta *et al.* [8] provide a frequency-tuned saliency estimation by calculating the feature distance between the low-pass filtered result of an image and the average color. This operation of [8] is equivalent to combining center-surround differences [6] of all bandwidth to detect

objects in different scales. Their method, to some extent, alleviates the bandwidth limitation of early Itti's model [6] so that not only edges and corners, but also the entire objects are highlighted. Goferman *et al.* [20] propose context-aware saliency detection, which combines local and global features to estimate patch saliency in multi-scales. To consider both local and global features, they compute saliency of a certain patch as its contrast to the nearest patches in the feature space. However, their method still tends to highlight edges meanwhile attenuate the inner parts of an object. Motivated by the work of Zhai *et al.* [63] and to better take advantage of color information, Cheng *et al.* [7] extend contrast saliency computation to color histograms. To reduce the bin number of color histograms, a color quantization technique is employed. Besides, they propose a saliency measure based on regional color contrast. Jiang *et al.* [30] also use regional contrast to define saliency. Instead, they only consider context information from neighboring regions. Perazzi *et al.* [64] propose "saliency filter", which formulates complete contrast and saliency estimation using high dimensional Gaussian filters. Wang *et al.* [65] compute pixel-wise image saliency by aggregating complementary appearance contrast measures with spatial priors. When computing the saliency of a pixel, they adopt shape-adaptive observation region to extract local information associated with the pixel. This is deemed more robust than using a whole image patch centered round the pixel. A more recent method [66] computes contrast-based saliency as dissimilarity/similarity to carefully selected background/foreground seeds. Most of the above contrast-based saliency are straightforward to compute, though the performance is often less satisfactory on images with complex background.

3.1.2 Learning-based methods

The concept of "learning to detect" in saliency detection originates from [22, 67]. The idea behind is to automatically discover feature integration rules from training data instead of manually designed rules. Judd *et al.* [22] propose to learn a saliency model from eye-tracking data, where low, middle and high-level image features are integrated by a linear SVM. Their work is, however, on eye-fixation prediction. Alex *et al.* [68] learn to score sampled windows from a given image, where the Bayesian theory is applied for cue integration. The posterior of the Bayesian model constitutes the final objectness score of a window. Khuwuthyakorn *et al.* [69] learn to integrate pixel-wise saliency features by a mixture of linear SVMs. Mehrani *et al.* [70] use confidence scores from a boosting classifier to formulate a saliency map, which is then fed to graph cut for figure-ground segmentation. Jiang *et al.* [29] propose to extract abundant discriminative features from image regions. A random forest regressor is trained to map regional features to final saliency scores.

Some methods on saliency detection are based on learning conditional random field (CRF). Learning is conducted first to obtain optimal parameters and then inference is applied on user input images to achieve final saliency maps. Representative works include: Liu *et al.* [67] detect and segment salient objects by aggregating pixel saliency cues in a CRF. The linear weights for those cues are learned under the maximized likelihood (ML) criteria by tree-reweighted belief propagation. Mai *et al.* [71] propose a saliency aggregation approach. Their method aggregates saliency maps output by existing saliency detection models using a CRF. Weights for aggregation are learned in a data-driven way from most similar images retrieved from a pre-defined dataset. Lu *et al.* [72] learn optimal combination of seeds for graph-based diffusion by maximizing figure-ground segregation, where the employed graph diffusion is tightly related to continuous CRF (C-CRF), and their method boils down to learning the linear parameters of unary terms of the C-CRF.

Besides, some recent data-driven techniques [73–75] consider deep learning for saliency detection. Due to the deep architecture of convolutional neural networks (CNNs) which is to able to fit highly non-linear models, impressive performance has been obtained. Since machine learning is able to discover latent and complicated feature integration rules from annotated data, learning-based methods can achieve good performance in complex scenarios attributed to the learning. However, high computation cost is needed for this type of methods due to feature extraction and learning, as comparing to the conventional color contrast-based methods.

3.1.3 Segmentation-assisted methods

Methods in this category aim at generating good segmentation, usually in hierarchy or multi-scale, to facilitate saliency computation. According to the figure-ground perceptual organization of human eyes described in Gestalt laws [76, 77], the region on the convex side of a curved boundary tend to be perceived as object (i.e., figure). Motivated by this, Lu *et al.* [78] exploit the concavity context in a scene and detect concave arcs from multi-scale segmentation. The detected arcs then contribute to an enhanced figure-ground segmentation phase. Yan *et al.* [31] propose a hierarchical saliency detection method that aims at eliminating the impact of small-scale patterns during saliency estimation. Their method merges regions according to a newly proposed scale metric which meets human perception. However, each region in a hierarchy is still evaluated by using local contrast and location prior. Cheng *et al.* [79] measure saliency by hierarchical soft abstraction. They form a 4-layer hierarchical structure including pixel layer, histogram layer, GMM layer and clustering layer with an index table to associate cross-layer relations efficiently. Saliency estimation using color contrast and distribution is conducted on the coarse layers and then propagated to the pixel

layer. Jiang *et al.* [59] segment an image into different salient regions by maximizing a submodular objective function. Then saliency of a region is modeled in terms of appearance and spatial location. In their method, how to assign over-segmented superpixels to the corresponding seeds is formulated as a "facility location problem" and is solved efficiently by finding a closed-form harmonic solution on a graph model. The above methods, benefiting from some optimized segmentation phases, could make entire objects emphasized and hence boost the final performance.

3.1.4 Graph-based saliency modeling

These methods represent each image by using a graph, where natures of salient objects such as high color contrast and compact color distribution are modeled. Gopalakrishnan *et al.* [80] perform random walks on graphs to find salient objects. The global pop-up and compactness properties of salient objects are modeled in random walks by the equilibrium access time. Wei *et al.* [81] propose to treat boundary parts of an image as the background. The patch saliency is defined as the shortest geodesic distance on a graph to image boundary. Zhu *et al.* [60] propose a saliency detection method based on robust background estimation from graph-based geodesic affinity.

Some methods propagate/diffuse the saliency energy from labeled seeds to the entire image through a graph. Yang *et al.* [57] propagate saliency via graph-based manifold ranking from four image borders separately. Four saliency maps generated are then multiplied to achieve the final one. Yang *et al.* [82] propose to use graph regularization on a convex-hull-based center prior map to achieve an edge-preserving final saliency map. Recently, Gong *et al.* [61] propose a new saliency propagation algorithm employing teaching-to-learn and learning-to-teach strategies to explicitly improve the propagation quality. Many graph-based diffusion models are related to the inference stage of continuous CRFs (as explained in Paper 2). Methods of this category can emphasize holistic objects and achieve good performance.

3.1.5 Other methods

Other notable work includes: Shen *et al.* [83] solve saliency detection as a low rank matrix recovery problem, where salient objects are represented by a sparse matrix (noise) and background is indicated by a low rank matrix. Though their idea is novel, such sparse and low rank assumption may hardly be satisfied in complex scenes, leading to unsatisfactory results. A Bayesian framework is adopted in [84]. Firstly, saliency points are applied to get a coarse location of the saliency region. Based on the rough region, a prior map is computed for the Bayesian model. The likelihood of the Bayesian model is defined according to the histograms in and outside the rough region. Margolin *et al.* [85] find the previously used patch distance may not

reliably characterize the saliency level of a patch in an image. From the idea of global pattern statistics, they perform principal component analysis (PCA) on a large number of collected patches from the image. The patch saliency is defined as L1-norm in PCA coordinates and then combined with color contrast saliency. Li *et al.* [62] propose to measure saliency by dense and sparse reconstruction errors, where the dictionaries for reconstruction are acquired from image boundary. More recently, Zhang *et al.* [86] perform saliency detection based on minimum barrier distance and show its robustness over the extensively used geodesic distance.

3.2 Traffic sign recognition

3.2.1 Traffic sign detection

State-of-the-art sign detection techniques can be divided into three categories: image segmentation followed by region analysis [41, 87–89], edge-based shape discovery [42–44, 90], sliding window detection approaches [40, 91, 92]. In addition, a forth category is also summarized in this thesis, namely saliency detection-based approaches [93–95].

Image Segmentation Followed by Region Analysis

In traffic sign detection, color segmentation is a classical and widely adopted technique. It is conducted by either segmenting images into non-overlapping regions [87], or extracting candidate regions from specific image color channels [41, 88, 89]. These regions are further filtered by color/shape analysis to pick up candidate regions of signs. Khan *et al.* [87] segment images by pixel clustering. After post-processing, potential sign regions are analyzed by principal shapes. Maldonado *et al.* [88] segment images by thresholding each color component in HSI color space. Potential sign regions are classified by linear SVMs to determine shapes. Gómez-Moreno *et al.* [89] benchmark a variety of color segmentation techniques in different color spaces for traffic sign recognition. To show which segmentation technique performs the best, their evaluation is done by fixing the subsequent detection and classification modules while only changing the segmentation method used. Since these methods described above rely heavily on image segmentation for generating complete sign regions, segmentation parameters need to be carefully tuned to avoid over- or under-segmentation of sign regions. For example, [87] uses k-means to pre-segment an image into 5 clusters. Though this method handles simple images, it suffers in cluttered or complex scenarios. To summarize, segmentation-based methods require images to have good color contrast and visual quality so that good segmentation can be performed. They are less robust to color distortion and luminance changes.

Edge-based Shape Discovery

In contrast to segmentation-based methods, this type of methods directly discover certain shapes of signs. Arlicot *et al.* [44] apply curve fitting to discover ellipse signs. Garrido *et al.* [90] use constrained Hough transform to find circles and lines in triangles. [42, 43] employ Fast Radial Symmetrical Transform (FRST, an edge-based voting scheme) proposed by Loy *et al.* [96] to discover circular and triangle signs. Since these methods need to examine edge points in images, they are robust to illumination changes but however less efficient due to a tremendous number of edge points. To improve the efficiency, Zhang *et al.* [41] combines color-based segmentation with FRST [96]. Another shortcoming of edge-based methods is that they are sensitive to shape distortion caused by view angle changes, since most shape detection methods assume signs to have regular polygon shapes or circular shapes.

Sliding Window Detection

A recent trend for sign detection is to use the sliding window strategy [40, 91, 92, 97, 98], a method widely used in category-specified object detection [99–102]. A sliding window scans an entire image exhaustively and a detector (usually a binary classifier) is used to determine whether or not the window contains any sign. In such a way, image segmentation is no longer required during sign detection. Bahlmann *et al.* [91] employ the Adaboost detector with Haar wavelet features for sign detection, which has been successfully applied to face detection [99]. Creusen *et al.* [40] use color-boosted histograms of oriented gradient (HOG) features with SVMs for sign detection. Mathias *et al.* [92] employ integral channel features [101] and Adaboost detectors. Wang *et al.* [97] propose a two-stage coarse-to-fine sliding window scheme. In the first stage, HOG with small sized windows and LDA are used for the efficiency. In the second stage, HOG with large sized windows and SVMs are used for the better accuracy. Møgelmose *et al.* [98] evaluate the detector proposed by Mathias *et al.* [92] on road signs in U.S. and find that sign detection still remains challenging for U.S. road scenes.

Saliency Detection-based Approaches

Traffic signs are markers placed along roads to inform drivers and communicate a wealth of information. In this case, they are designed to be easily realized at a glance, or in other word, to be salient. The idea of this category of methods is to use saliency detection to identify some pre-attentive regions of interest (ROI), where signs are contained. Therefore, searching space is narrowed down for subsequent detection. Most of related literatures are limited to applying saliency detection prior to sign detection. Kastner *et al.* [93] extract rectangular ROI from an attention map by region

growing and fusion. In each rectangular ROI, color segmentation and a cascade of weak classifiers are applied to identify signs. Won *et al.* [94] apply Itti's model [6] on augmented color channels and an edge channel to compute a saliency map. They show that top five candidate regions extracted from such a saliency map provide encouraging detection coverage of signs. Recently, Yuan *et al.* [95] apply graph-based manifold ranking saliency proposed in [57] to traffic sign detection. The computed saliency map that indicates salient regions is further processed by a multi-threshold segmentation algorithm. Generally speaking, this category of methods build up the connection between two research fields, namely saliency detection and traffic sign detection, by applying the former to the latter.

3.2.2 Traffic sign classification

A typical sign classification pipeline contains feature extraction, dimension reduction, and classification. Bahlmann *et al.* [91] employ naive Bayesian classifiers for road sign classification. Prior to the probabilistic modeling, a feature transformation is performed, using standard linear discriminant analysis (LDA). The features used in [91] are normalized gray-scale images of signs. Maldonado *et al.* [88] extract gray-scale images from every candidate blob as the features and use one-against-all SVMs for classification. No dimension reduction procedure is applied in between. Ruta *et al.* [103] present a class-specific discriminative feature selection scheme, where discriminative patches are pre-learned from training templates and then are used to calculate feature distance in a nearest neighbor classifier. Mathias *et al.* [92] test several features including the gray-scale image, the HOG, and the pyramid of HOG on different classifiers including nearest neighbor classifier, sparse representation-based classifier, and SVM. Several dimension reduction methods such as LDA and sparse representation based linear projection are also tested. In the work of Khan *et al.* [87], joint transmission correlation is used to match an input sign image to a training template in a joint power spectrum domain. Convolutional neural networks (CNNs) are used by Ciresan *et al.* [39], where the raw sign images are used directly as input to the networks. Their method is tested on the German sign classification dataset raised in [36] and achieves very encouraging performance.

Chapter 4

Summary of The Work in This Thesis

This chapter summarizes the thesis work on salient region detection and traffic sign recognition. About salient region detection, different proposed methods are driven by different motivations and target at different aspects. In traffic sign recognition, this thesis employs salient regions for improved sign classification, which is new in both saliency detection and traffic sign recognition communities.

4.1 Salient region detection methods

Attributed to different motivations, the five proposed methods (Method-1 to Method-5) contribute to different categories (Chapter 3), as shown in Figure 4.1. Below, each method and its main contributions are described.

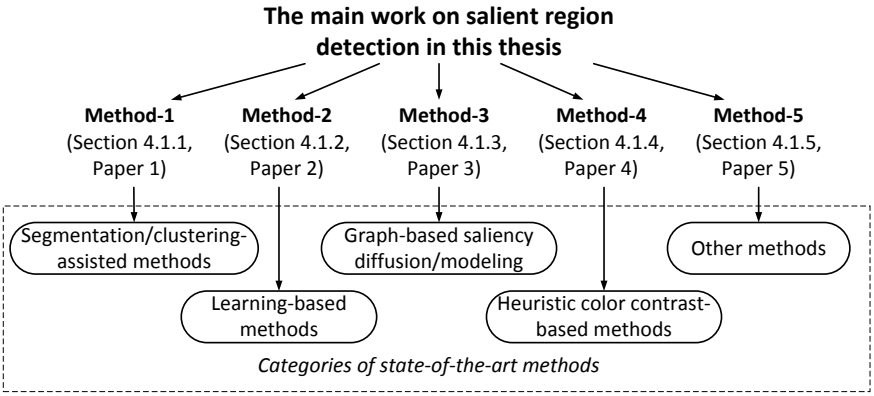

Figure 4.1: The summary of the thesis work on salient region/object detection.

4.1.1 Method-1: Normalized cut-based saliency detection by adaptive multi-level region merging

(Summary of Paper 1)

Problem addressed: This method addresses the grouping/clustering issue for salient objects and background, based on the intuition that better clustering of salient objects and background prior to saliency computation leads to more accurate saliency estimation.

Basic ideas: For better grouping objects and background so that more accurate saliency estimation can be conducted, Method-1 proposes the utilization of normalized graph cut (Ncut) for salient region detection. Since the Ncut normalizes graph cut cost as a fraction of the total edge connections to all graph nodes, it is a biased cut on fairly large set of graph vertices. Our intuition is to find good grouping of visual contents, usually large objects, meanwhile prevent grouping of small clusters that are usually noise. The Ncut rightly satisfies this demand. Additionally, the Ncut is a global criterion that partitions the graph in a non-parametric way, and is efficient to compute.

Big picture: The block diagram of Method-1 is shown in Figure 4.2, where we directly induce saliency maps via eigenvectors of the Ncut. We first implement the Ncut on a superpixel graph (vertices are superpixels) which captures both intrinsic color and edge information of image data. Starting from the superpixels, an adaptive multi-level region merging scheme is proposed to seek the cluster information from Ncut eigenvectors. Specifically, the adaptive multi-level region merging process operates on the reconstructed graph edge weights e_{ij}, which are reconstructed from $nvec$ eigenvectors $\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_{nvec}$ that correspond to $nvec$ smallest non-negative eigenvalues $\lambda_1, \lambda_2, ..., \lambda_{nvec}$:

$$e_{ij} = \sum_{k=1}^{nvec} \frac{1}{\sqrt{\lambda_k}} |\mathbf{v}_k(R_i^0) - \mathbf{v}_k(R_j^0)| \tag{4.1}$$

where λ_k is the corresponding eigenvalue and $\mathbf{v}_k(R_i^0)$ indicates the value in eigenvector $\mathbf{v}_k$ that corresponds to the superpixel R_i^0. Such reconstruction by integrating the differences between values of vertices on the eigenvectors indicates "inter-class distance" after Ncut [104]. During merging, the pairwise difference between regions at lth level is defined by criterion D:

$$D_{ij}^l = D(R_i^l, R_j^l) = \text{mean}_{v_k \in R_i^l, v_m \in R_j^l, e_{km} \in E}\{e_{km}\} \tag{4.2}$$

where R_i^l and R_j^l are two regions at lth level, "mean" is an averaging operator, and v_k, v_m are two vertices, namely superpixels in R_i^l and R_j^l satisfying

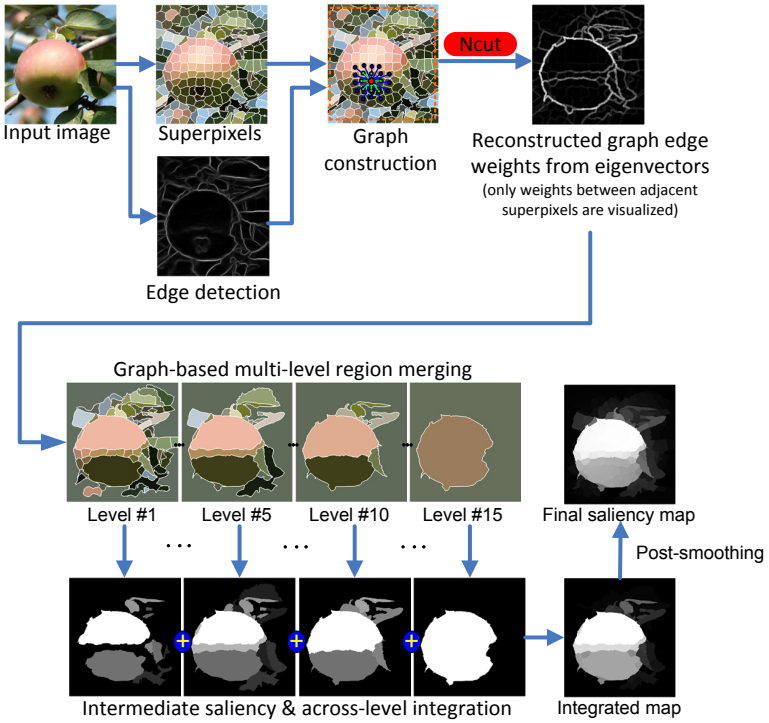

Figure 4.2: The block diagram of Method-1.

that there is graph connection between them. As the merging proceeds, the cluster information in Ncut vectors is gradually discovered and is turned into regions. At each level, three types of regional saliency measures (figure-ground contrast, center bias, and boundary cropping) are computed and then combined to generate intermediate saliency maps. The final saliency map is obtained after across-level integration and post-smoothing.

Main contributions:

- Apply the Ncut to salient region detection, and induce a saliency map by Ncut eigenvectors for better visual clustering.
- Embed saliency detection in an adaptive multi-level merging scheme to discover cluster information conveyed by Ncut eigenvectors.

Main results: Method-1 is tested and evaluated on four benchmark datasets including MSRA-1000, SOD, SED, CSSD and compared to 13 state-of-the-art methods. It is shown to result in uniform object enhancement and achieve state-of-the-art performance in terms of competent precision, recall and F-measure, meanwhile maintaining the lowest mean absolute error

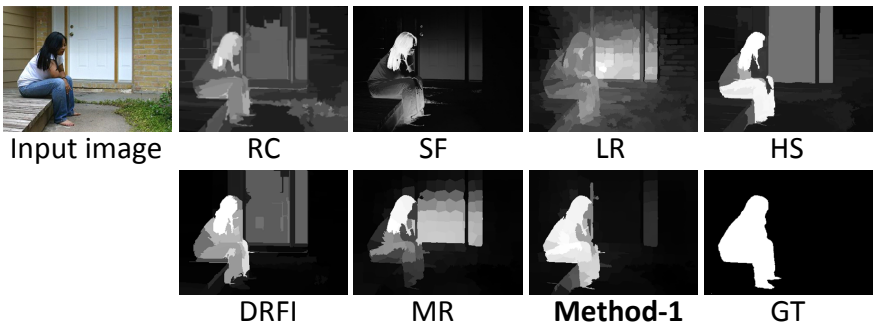

Figure 4.3: An example case where state-of-the-art methods: RC [7], SF [64], LR [83], HS [31], DRFI [29], MR [57] fail to detect the entire object. GT means the ground truth annotation.

(MAE). Figure 4.3 shows a visual example. Evaluations on MSRA-1000 (1000 images) [8] and CSSD (200 images) [31] are shown in Figure 4.4.

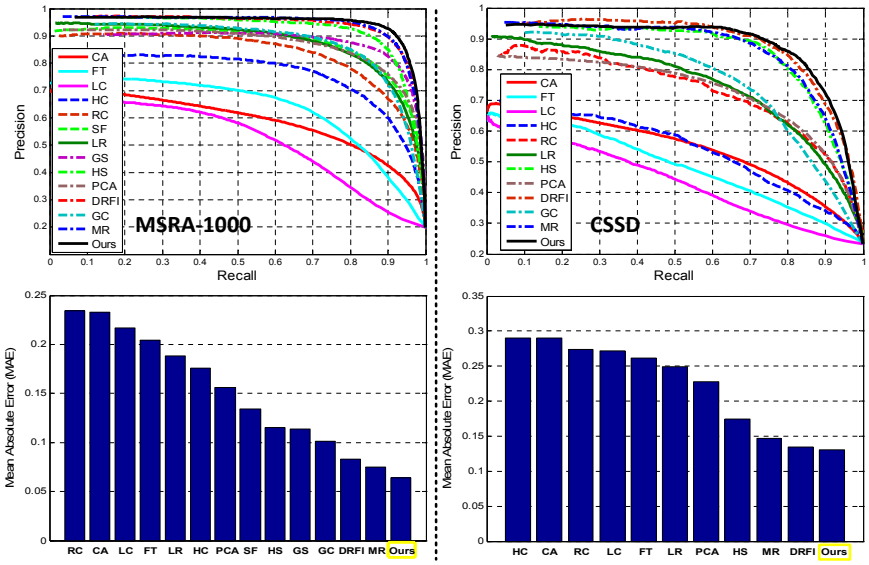

Figure 4.4: Quantitative comparison of Method-1 (Ours) to 13 state-of-the-art methods (CA [20], FT [8], LC [63], HC [7], RC [7], SF [64], LR [83], GS [81], HS [31], PCA [85], DRFI [29], GC [79], MR [57]) by precision-recall curves (1st row), and mean absolute error (MAE) (2nd row) on MSRA-1000 (1st column) and CSSD (2nd column) datasets. For more quantitative results, please refer to Paper 1.

4.1.2 Method-2: Saliency detection by fully learning a continuous conditional random field

(Summary of Paper 2)

Problem addressed: This method addresses the problem of constructing a continuous conditional random field (C-CRF) framework for salient region detection.

Basic ideas: In existing studies [57, 67, 71, 72], the power of CRF on feature integration has not been fully exploited. Method-2 proposes to fully learn a CRF, namely to learn both unary and pairwise parameters in order to exploit the power of CRF for feature integration. More specifically, Method-2 investigates a special CRF framework—*continuous CRF (C-CRF)* [105]. This is motivated by the idea that saliency detection is conventionally treated as a *continuous labeling problem*. Fully learning a C-CRF model allows us to capture more sophisticated interactions between image parts, leading to enhanced delineation between objects and background in the resulting saliency maps. To the best of the author's knowledge, applying the complete C-CRF learning and inference theories to saliency detection is the first time.

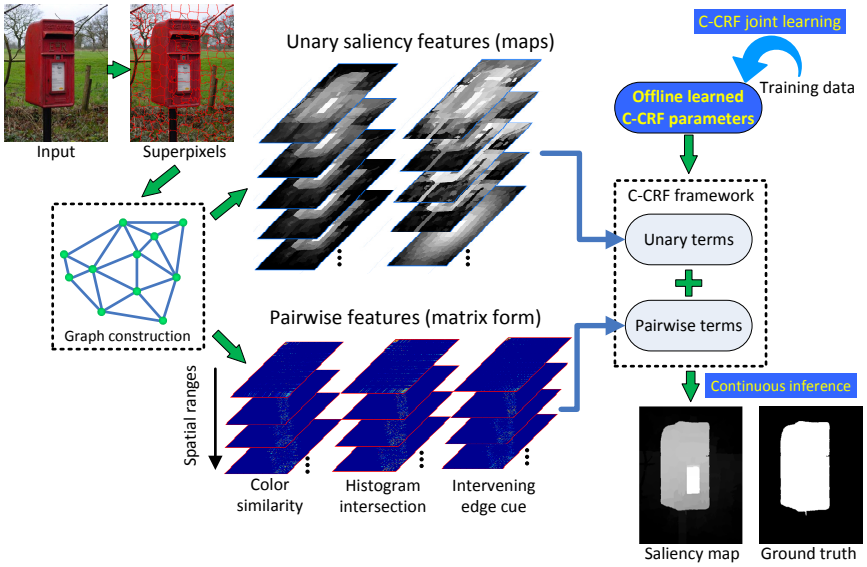

Figure 4.5: The big picture of Method-2.

Big picture: As shown in Figure 4.5, an input image is first over-segmented into superpixels and a superpixel graph is established to capture intrinsic image context. A C-CRF is defined in conjunction with this graph. Next, we extract various unary saliency features and pairwise cues, which are used to compose the unary and pairwise terms in the C-CRF energy function. By utilizing the off-line learned C-CRF parameters for both unary and pairwise potentials, the inference of the C-CRF corresponds to a final saliency map that is continuously valued.

The C-CRF energy function of Method-2 is formulated as:

$$\mathbb{E}(\mathbf{y}, \mathbf{x}) = \sum_{i=1}^{n} \sum_{k=1}^{d} \alpha_k (y_i - f_{i,k})^2 + \sum_{i,j,i\sim j} \frac{1}{2} \sum_{k=1}^{h} \beta_k S_{ij}^k (y_i - y_j)^2 \tag{4.3}$$

where α_k, $f_{i,k}$ are the kth components of unary parameter vector α and saliency feature vector $\mathbf{f}_i$ extracted from the ith superpixel. β_k is the kth components of pairwise parameter vector β and S_{ij}^k is the kth pairwise feature defined between superpixels. In Method-2, $\mathbf{f}_i$ is an initial description for the saliency level of superpixels, whereas S_{ij}^k is a positive affinity function that is large if superpixels are similar. Unary feature vector $\mathbf{f}_i$ comprises 11 features including connectivity-based features, contrast-based features, distribution heuristics, and clarity-based feature. These features are supposed to characterize general properties of salient objects. The pairwise features include color-based features and image edge-based features. Moreover, the pairwise features consider different spatial ranges (as shown in Figure 4.5) of graph connections by graph topology decomposition.

The C-CRF learning is formulated as follows: given N training images $\mathbf{x}^1, \mathbf{x}^2, ..., \mathbf{x}^N$ with their ground truth labels $\mathbf{y}^1, \mathbf{y}^2, ..., \mathbf{y}^N$, learn C-CRF parameters α and β. This is equivalent to minimizing the negative log-likelihood summed over all training images:

$$\begin{aligned} &\min_{\alpha,\beta} \sum_{i=1}^{N} \{-\log p(\mathbf{y}^i|\mathbf{x}^i) + \frac{\lambda_1}{2}||\alpha||_2^2 + \frac{\lambda_2}{2}||\beta||_2^2\} \\ &\text{s.t. } \alpha_k > 0, \quad \beta_k \geq 0 \end{aligned} \tag{4.4}$$

where λ_1 and λ_2 are regularization parameters. The above optimization problem can be solved by gradient descent.

Main contributions:

- Method-2 is the first to apply the complete C-CRF learning and inferring theory to saliency detection, leading to a data-driven way for feature integration.
- Method-2 differs from existing saliency models that have explicit/implict relation to CRF, evolving from partially learning unary terms to joint-

ly learning both unary plus pairwise terms, and from discrete to continuous field.

- A novel formulation of pairwise potentials for C-CRF defined on a superpixel graph is proposed. It is conducted by graph topology decomposition and enables learning pairwise parameters for different spatial ranges of graph connections. This alleviates the manual effort of tuning spatial connections of a graph.

- Extensive tests and comparisons show that Method-2 outperforms a range of state-of-the-art methods. Furthermore, integrating several best-performing state-of-the-art methods through a C-CRF further pushes the performance to a new high level.

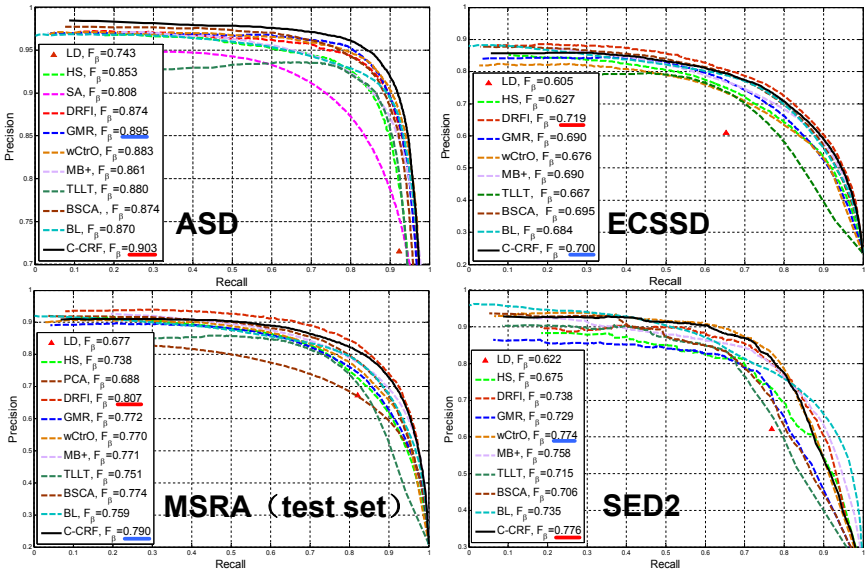

Figure 4.6: Quantitative comparisons (precision-recall curves and F_β scores) of Method-2 (C-CRF) to the state-of-the-art methods on 4 benchmark datasets. The best and the second best F_β are underlined by red and blue, respectively.

Main results: The power of C-CRF on integrating various unary and pairwise features has been tested and evaluated on 6 benchmark datasets. The results on 4 datasets are shown in Figure 4.6, where Method-2 achieves good precision-recall curves and F-measure scores that outperform most state-of-the-art methods. Among the compared methods, LD [21], SA [23], and GMR [22] are existing CRF-related methods. Besides, integrating existing models (HS [31], DRFI [29], GMR [57], wCtrO [60], and MB+ [86]) by C-

CRF further achieves a marked promotion over individual models, as shown in Figure 4.7.

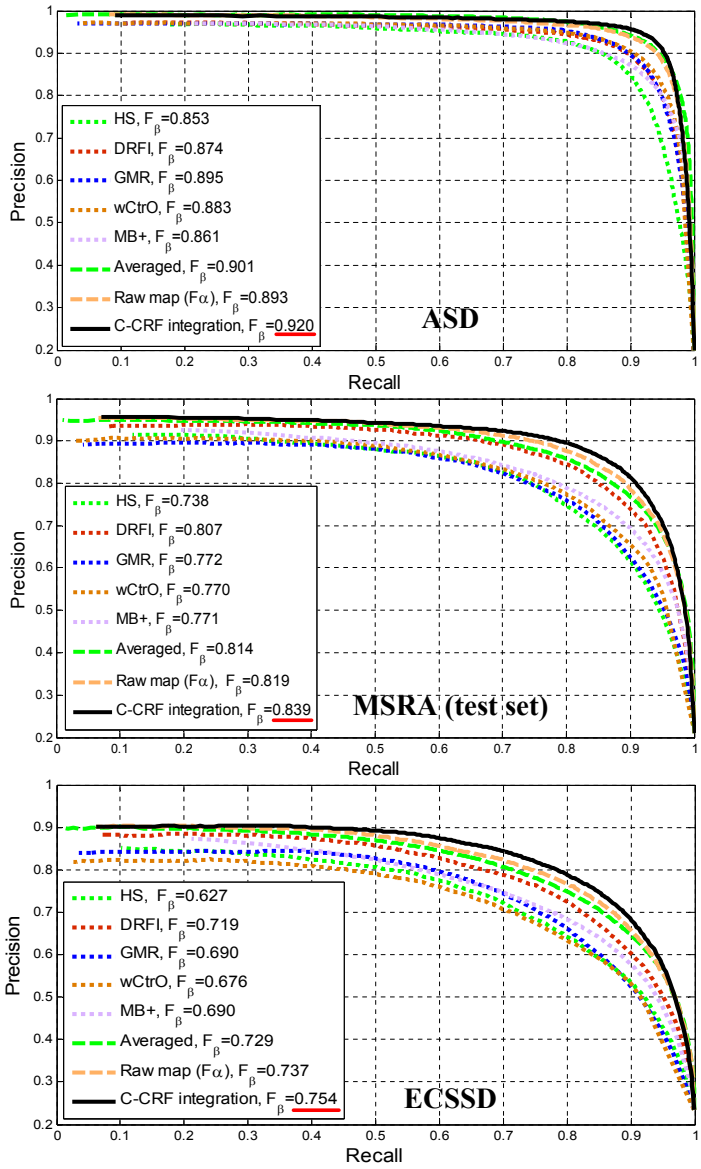

Figure 4.7: Integrating five state-of-the-art methods including HS [31], DRFI [29], GMR [57], wCtrO [60], and MB+ [86] by Method-2 (C-CRF Integration). The best F_β are underlined by red.

4.1.3 Method-3: Manifold-preserving diffusion-based saliency detection by adaptive weight construction

(Summary of Paper 3)

Problem addressed: This method addresses the problem of exploiting a diffusion-based detection scheme, where graph weights are autonomously constructed to adapt to different images.

Basic ideas: Existing methods often employ Gaussian similarity function with fixed bandwidth to construct graph edge weights, however cannot adapt to different images. The basic idea of Method-3 is to acquire the edge weights adaptively by minimizing local reconstruction errors on data manifold. Therefore, the obtained weights could better reflect the structural relationship between data points in a manifold perspective. The idea of using local reconstruction to solve the graph weights is inspired by LLE [106] and the similarity adaption technique proposed by Karasuyama *et al.* [107].

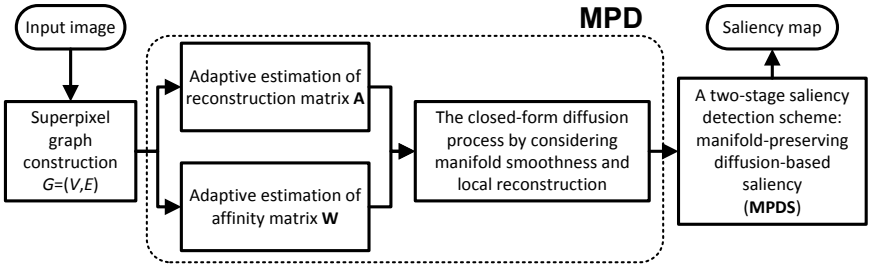

Figure 4.8: The big picture of Method-3. Note there are two contributed parts, MPD and MPDS, in Method-3.

Big picture: As shown in Figure 4.8, we first propose a novel diffusion scheme called *manifold-preserving diffusion* (MPD). MPD builds jointly upon two assumptions on data manifold, namely the *smoothness* and *local reconstruction*. Then we apply MPD to detect salient objects and propose a new detection scheme, referred to as manifold-preserving diffusion-based saliency (MPDS).

To be more specific, the MPD is conducted on a superpixel graph with n nodes, and the diffused result $\mathbf{s}$ is computed by minimizing the following energy function:

$$\arg\min_{\mathbf{s}} \quad \underbrace{\mu \sum_{i=1}^{n} k_i (s_i - y_i)^2}_{\text{weighted fitness term}} + \underbrace{\sum_{i=1}^{n} \sum_{j,j\sim i} \frac{1}{2} w_{ij} (s_i - s_j)^2}_{\text{manifold smoothness}} + \underbrace{\lambda \sum_{i=1}^{n} (s_i - \sum_{j,j\sim i} a_{ij} s_j)^2}_{\text{manifold reconstruction}} \tag{4.5}$$

where s_i, y_i are the ith elements of the diffused vector $\mathbf{s}$ and a pre-defined seed vector $\mathbf{y}$, w_{ij} encodes the similarity between vertices, a_{ij} encodes reconstruction contribution of v_j to v_i, $\lambda \geq 0$, $\mu > 0$ are balancing weights, and $k_i > 0$ is the weighting coefficient for the ith node. In (4.5), the smoothness terms reflect the conditional random field (CRF) property and enforce similar saliency on similar graph neighbors. Meanwhile, the reconstruction terms enforce the diffused value of a node to be linearly reconstructed by its graph neighbors. As w_{ij}, a_{ij} compose the entries of affinity matrix $\mathbf{W}$ and reconstruction matrix $\mathbf{A}$, Method-3 first estimates $\mathbf{W}$ and $\mathbf{A}$ *adaptively* (Figure 4.8) by minimizing local reconstruction errors in feature space [45, 106, 107]. The resulting adaptive weights enable adaption of diffusion to different images.

The block diagram of MPDS is shown in Figure 4.9. MPDS is an application of MPD and incorporates boundary prior, Harris convex hull, and foci convex hull for deriving different seeds for saliency diffusion.

Main contributions:

- Propose an effective graph-based diffusion method: manifold-preserving diffusion (MPD), that jointly exploits the assumptions of smoothness and local reconstruction on the manifold.
- Derive two types of graph edge weights by adaptively minimizing local reconstruction errors in feature space. Hence the method is more suitable to be applied on different images. This is different from previous work where the edge weights of graph are similarity functions parameterized by manually tuned parameter such as bandwidth.
- Propose a two-stage saliency detection scheme: manifold-preserving diffusion-based saliency (MPDS), that leverages MPD together with boundary prior, Harris convex hull, and foci convex hull. The proposed MPDS achieves better performance than 8 recently published methods on 5 benchmark datasets.

Main results: To validate MPD, we compare between MPD and a previous diffusion method called graph-based manifold ranking (GMR) [57]

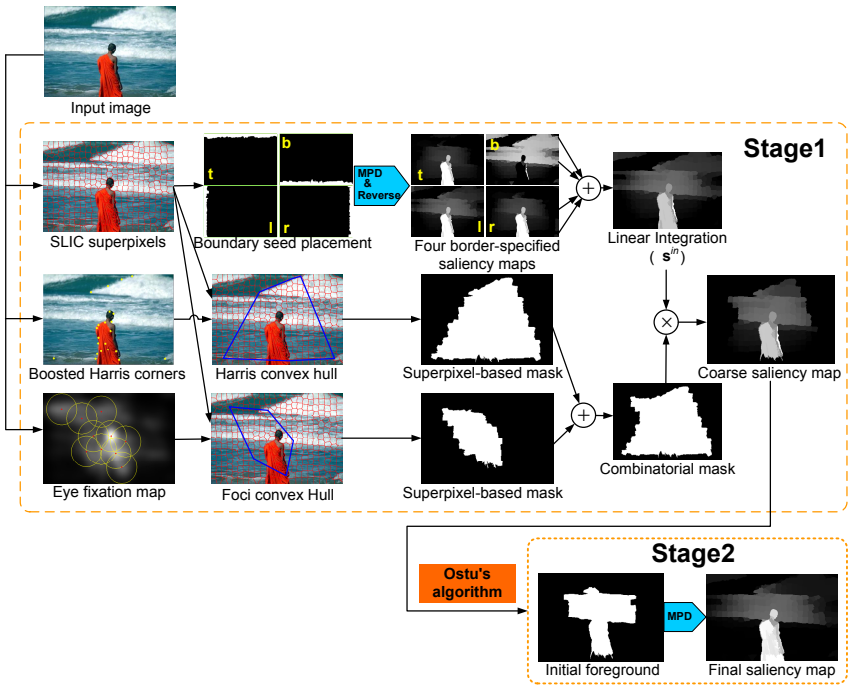

Figure 4.9: The block diagram of MPDS (Method-3). Stage 1: In the top pipeline, by specifying each image border as the background seeds, MPD (the blue block) is applied to perform diffusion for four times and generate four intermediate saliency maps. These four maps are then linearly integrated; In the middle pipeline, a Harris convex hull is generated from Harris corners to specify a region of interest; In the bottom pipeline, a foci convex hull is generated from foci centers [6] to specify a coarse saliency region. Stage 2: Ostu's algorithm is used to select a foreground mask from the coarse saliency map obtained after Stage 1. Next, MPD is applied again to achieve the final saliency map.

that is based on edge weights with fixed bandwidth. As shown in Figure 4.10, when diffusing on the same graph structures from the same seeds, MPD achieves better diffusion performance than GMR. Method-3 (MPDS) is compared with 8 state-of-the-art methods on 5 benchmark datasets. The results show that Method-3 is robust in terms of consistently achieving the highest weighted F-measure (F_β^w) and lowest mean absolute error (MAE), meanwhile maintaining comparable precision-recall curves. Salient objects in different background can be uniformly detected in the final saliency maps. Several visual comparisons are shown in Figure 4.11. It is worthy

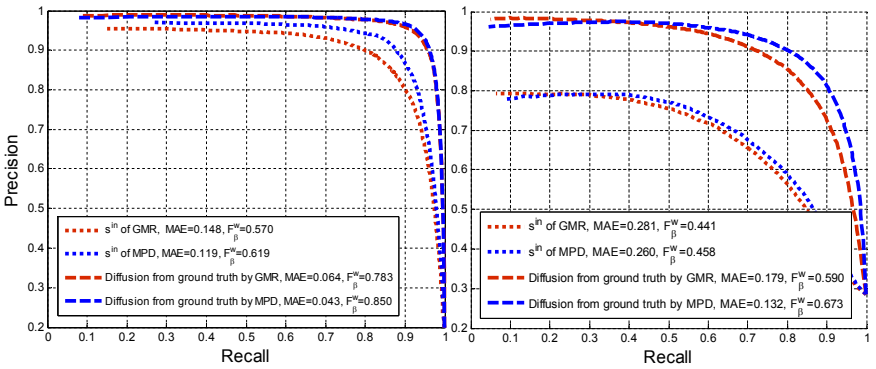

Figure 4.10: Diffusion evaluation of MPD and GMR [57] on MSRA-1000 (left) and ECSSD (right) datasets. The dot curves are the results by diffusion from background seeds (four image borders), whereas the dash curves are the results by diffusion from foreground seeds (the ground truth annotations). One can observe that the proposed MPD (blue curves) outperforms GMR (red curves).

noting that compared to GMR [57] (highlighted in the blue box) which is less adaptive to the cases where colors of objects and background are similar, Method-3 (highlighted in the red box) adapts better to different images and generates clearer object boundaries after diffusion.

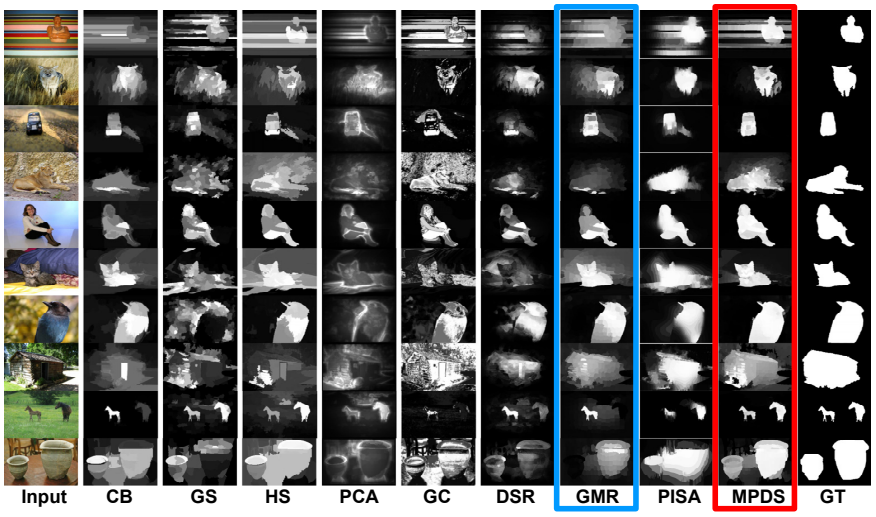

Figure 4.11: Visual comparisons of Method-3 (MPDS) to 8 state-of-the-art methods including CB [30], GS [81], HS [31], PCA [85], GC [79], DSR [62], GMR [57], PISA [65]. Our saliency maps show more consistency to the ground truth.

4.1.4 Method-4: Superpixel based color contrast and color distribution driven salient object detection

(Summary of Paper 4)

Problem addressed: This method addresses the problem of exploiting a unified computational scheme for integrating color contrast and color distribution cues.

Basic ideas: Previous work considers either color contrast [7, 8, 20, 63] or color distribution [108] for salient region detection, where the color information cannot be fully utilized, resulting in limited performance. We employ *superpixels* to compute these two color attributes efficiently, as superpixels are spatially compact atomic regions. In a unified superpixel-based manner, color contrast and distribution can be computed independently and then integrated. Method-4 exploits three hypotheses on salient objects:

> *Hypothesis-1.* A salient object often has strong color contrast to its surroundings. (*Color contrast*)
>
> *Hypothesis-2.* A salient object is often located close to the image center attributed to the habit of photographers. (*Color distribution*)
>
> *Hypothesis-3.* Colors of a salient object are more compactly distributed compared to the background. (*Color distribution*)

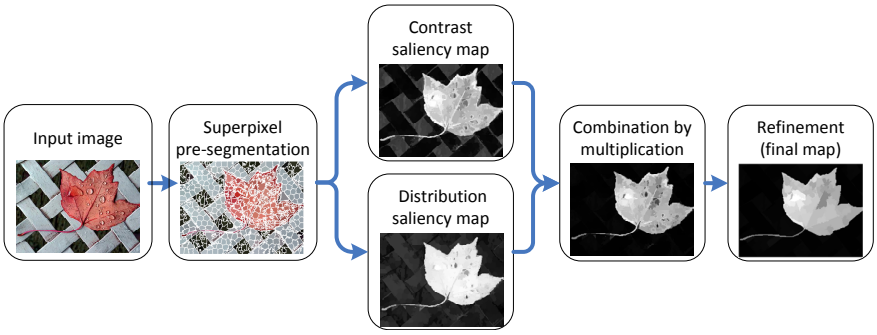

Figure 4.12: The block diagram of Method-4, including SLIC superpixel pre-segmentation (superpixel boundaries are visualized in white color superimposed on the original image), color contrast and color distribution computation, combination, and final refinement.

Big picture: Method-4 is a baseline approach by integrating color attributes to detect salient objects. Such attributes include color contrast and

color distribution. As shown in Figure 4.12, firstly the input image is over-segmented into spatially compact superpixels by using SLIC algorithm [56]. Then, two intermediate saliency maps, namely a color contrast map and a color distribution map, are computed. Considering *Hypothesis-1*, contrast-based saliency of a superpixel is computed by its color contrast to all superpixels in the image. The resulting saliency values are weighted by a distribution prior considering center bias (*Hypothesis-2*), and then refined by a saliency smoothing process so that superpixels with similar colors receive similar saliency. The outcome after this stage is the contrast saliency map as shown in Figure 4.12. Meanwhile considering *Hypothesis-3*, distribution-based saliency of a superpixel should be large when the color component it corresponds to has small spatial variance (namely compactly distributed) in the entire image. Method-4 models this hypothesis by subtracting the normalized color spatial variances from value 1.0. The outcome after this stage is the distribution saliency map as shown in Figure 4.12. Next, contrast saliency map and distribution map are superpixel-wisely multiplied. A refinement process by mean-shift segmentation is adopted to further eliminate noise and artifacts, and generate a coherent saliency map (Figure 4.12).

Main contributions:

- To utilize the three independent hypotheses, color contrast and color distribution saliency measures are formulated in a unified computational scheme based on superpixels. The obtained intermediate saliency maps are combined to achieve complementary performance. The effectiveness of incorporating color contrast with color distribution is validated qualitatively and quantitatively.
- Additional processes including global saliency smoothing and refinement by mean-shift segmentation are proposed, which enhance the final performance.

Main results: Tests and evaluation of Method-4 are done on benchmark datasets MSRA-1000 [8] and SOD [109]. The results are compared with 8 existing methods including CA [20], IT [6], SR [14], FT [8], LC [63], HC [110], RC [110], SF [64]. Method-4 is shown to perform well on background suppressing and uniform object enhancement (Figure 4.13). As shown in Figure 4.14 left, higher precision than existing methods is obtained from Method-4, where noticeable improvement can be observed. The complementary performance of incorporating color contrast with color distribution is validated on MSRA-1000 (Figure 4.14 right). The exploited distribution cues and saliency smoothing procedure are shown to be useful for boosting the detection performance. Additionally, we apply Method-4 to the application of *content-aware image resizing* (Figure 4.15) and shows its superiority over existing models RC [110] and SF [64].

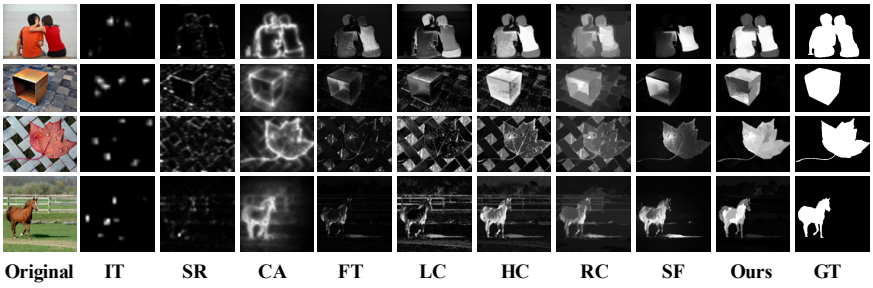

Figure 4.13: Qualitative comparisons of Method-4 (Ours) to 8 existing saliency methods.

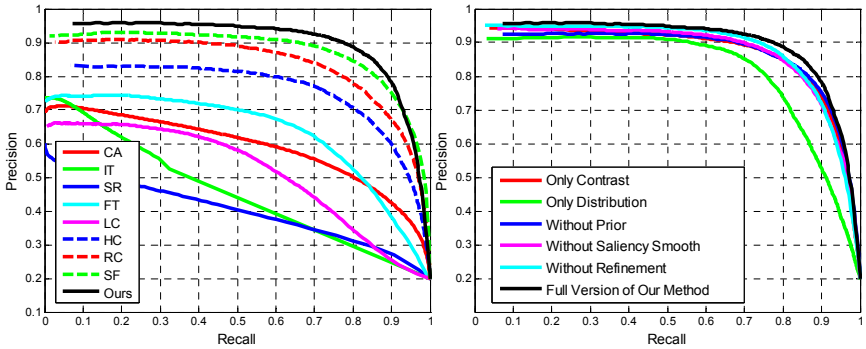

Figure 4.14: Quantitative evaluation on MSRA-1000. Left: Comparisons of precision-recall curves. Right: The impact of individual phases (ablation experiment).

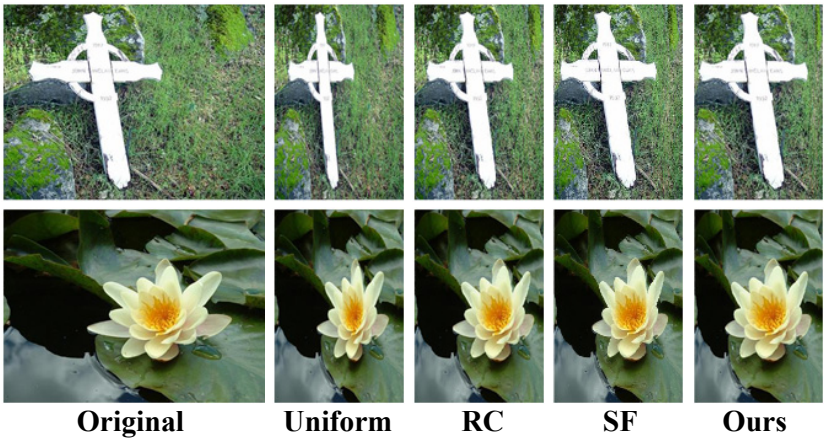

Figure 4.15: Apply RC, SF and Method-4 (Ours) to content-aware resizing, where the "Uniform" means uniform resizing.

4.1.5 Method-5: Geodesic saliency propagation for image salient region detection

(Summary of Paper 5)

Problem addressed: This method addresses employing geodesics to maintain visual coherence of saliency maps. The aim is to uniformly detect salient objects meanwhile suppress background.

Basic ideas: Objects and background usually show the property of connectivity, namely either objects and background often comprise connected regions. The basic idea of Method-5 is to utilize geodesic distance, which is a connectivity measure, to propagate saliency values and enhance salient objects from a set of coarse saliency maps. After propagation, connected regions should have coherent saliency.

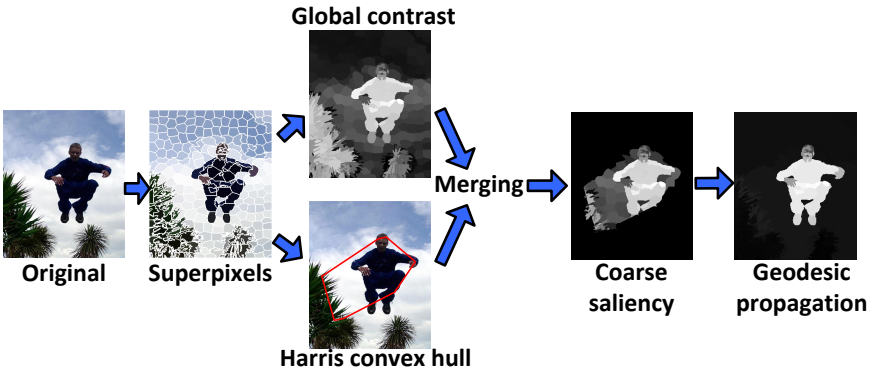

Figure 4.16: The block diagram Method-5. After the proposed geodesic propagation, the background clutter in the color contrast map is suppressed whereas a missing object part beyond the convex hull is recovered.

Big picture: As shown in Figure 4.16, similarly to other proposed methods, Method-5 computes saliency upon superpixels. An initial coarse saliency map is first formulated to detect potential salient regions, where Harris convex hull is adopted to exclude background clutter as much while the color contrast is employed to highlight salient objects from background. Next, the coarse saliency values of superpixels are propagated towards all supepixels. The extent of propagation between two superpixels is manipulated by geodesic distance:

$$S_i^{\text{propagation}} = \sum_j f_{j\to i} S_j^{\text{coarse}} \tag{4.6}$$

where $S_i^{\text{propagation}}$ is the transmitted saliency aggregated from superpixels R_j, and S_j^{coarse} is the corresponding saliency in the coarse saliency map. The saliency energy transmitted from a specific superpixel R_j to superpixel R_i is manipulated by term $f_{j\to i} S_j^{\text{coarse}}$, where $f_{j\to i}$ is the propagation intensity specified by:

$$f_{j\to i} = \frac{1}{\mathcal{N}} e^{-\beta d(R_i, R_j)} \tag{4.7}$$

where $d(R_i, R_j)$ is the geodesic distance between R_i and R_j, β is a parameter controlling the transmitting intensity, and $\mathcal{N} = \sum_j e^{-\beta d(R_i, R_j)}$ is the normalization factor. Due to such propagation in terms of connectivity, over-suppressed parts of a salient object can be recovered and falsely detected background can be suppressed (Figure 4.16). Method-5 differs from the geodesic saliency method in [81], as [81] defines the saliency of a patch as its shortest geodesic distance from a specified seed set (equivalent to geodesic distance transform). In our case, using geodesic transform is not robust to the noise in the coarse saliency map. Another propagation-related work is [108]. The main difference from [108] is that [108] conducts propagation through graph-based PageRank rather than geodesics used by Method-5.

Main contributions:

- A novel saliency propagation method based on geodesic distance is proposed and tested. Its effectiveness on saliency detection is validated upon initial coarse saliency maps derived from global color contrast and Harris convex hull.

Main results: Tests and comparisons are performed on a public dataset MSRA-1000 (1000 images) and compared with 9 existing methods. As shown in Figure 4.17, Method-5 achieves improved detection results compared to existing methods including CA [20], IT [6], SR [14], FT [8], LC [63], HC [110], RC [110], SF [64], and GS [81], where GS [81] also uses geodesic distance but is based on geodesic transform. Furthermore, the effectiveness of Method-5 is demonstrated by robust detection from convex hull and color contrast that are not accurate (Figure 4.18).

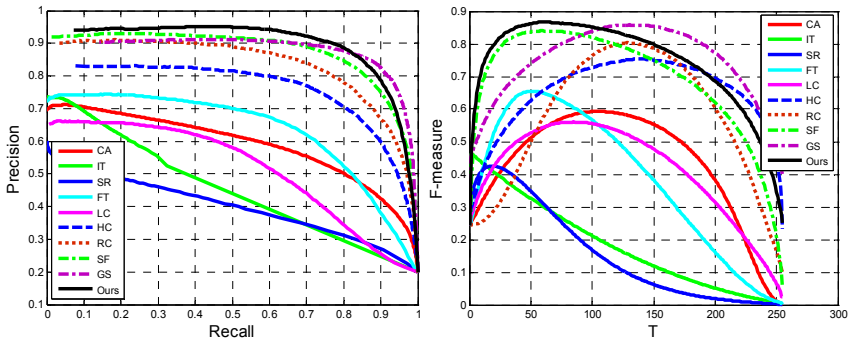

Figure 4.17: Performance comparisons of Method-5 (Ours) to 9 existing methods. Left: precision-recall curves; Right: F-measure curves by varying threshold T (x-axis).

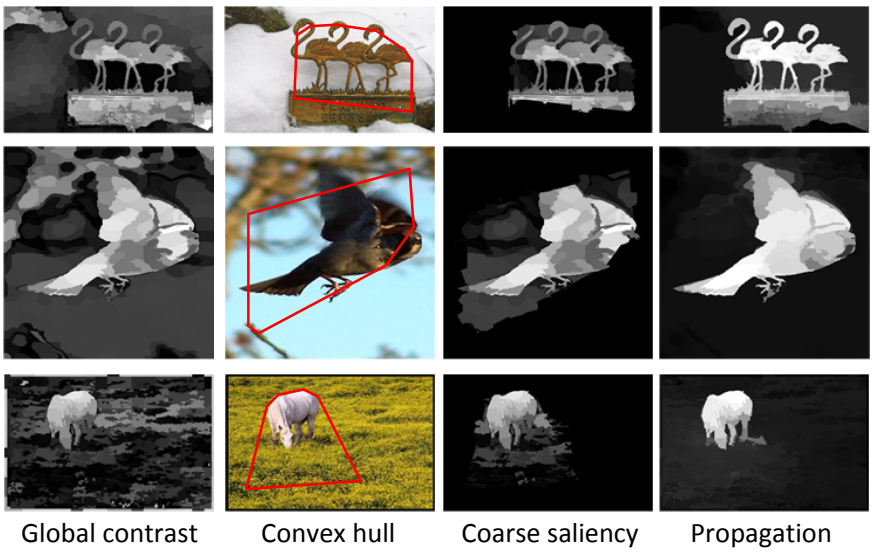

Figure 4.18: Three more examples of propagation, where the initial results from color contrast (1st column) and Harris convex hull (2nd column) are not accurate.

4.1.6 Comparison of the proposed saliency methods

(An extended work)

This section aims at benchmarking the proposed saliency methods (Method-1 to Method-5) under widely used metrics and shows their pros and cons.

Datasets

Four datasets were used for this benchmarking, including ASD [8] (also called MSRA-1000 interchangeably in some literatures and also the appended papers in this thesis), ECSSD [31], MSRA [67], and DUT-OMRON [57]. They are briefly summarized as follows:

ASD [8]: Contains 1000 images selected from the MSRA database [67]. Pixel-level ground truth is provided by [8]. In this dataset, each image usually contains one single object. This dataset is extensively used and very popular for evaluation of saliency methods, e.g., in [7, 31, 79, 85, 111].

ECSSD [31]: Constructed by [31], contains 1000 images extended from their early CSSD dataset [31] with diversified patterns in both foreground and background. Ground truth masks are produced by five subjects. The first edition of this dataset was realized soon after the paper [31] in 2013, but the ground truths were recently updated in April, 2015.

MSRA [67]: This dataset contains 5000 images most of which have an unambiguous salient object in each image. Originally released ground truths of this dataset in [67] are only bounding boxes. Lately the pixel-wise ground truth masks are provided along with the work in [29]. The most popular ASD [8] dataset takes 1000 images from this dataset. To avoid duplicated evaluation on both ASD and MSRA, images that belong to ASD are excluded from MSRA and the remaining 4000 images are used for our benchmarking.

DUT-OMRON [57]: Constructed by [57], contains 5,168 images manually selected from more than 140,000 images. Images of this dataset have one or more salient objects and relatively complex background. Three types of ground truths are available, i.e., bounding boxes, eye-tracking points, and pixel-wise masks. In our benchmarking, pixel-wise masks are used for the evaluation of salient region detection.

The above datasets were chosen based on the following reasons: 1) Being widely-used, 2) Having a large number of images ($\geq$ 1000), 3) Presenting different biases (e.g., number of salient objects, background clutter).

Models Compared

We list the compared methods in Table 4.1. We also include a recent method MB+ [86] into comparison to have a reference to the state-of-the-art performance. MB+ was presented in ICCV (International Conference on Computer Vision) 2015 , one of the top conferences in computer vision and

Table 4.1: Models compared in this section.

Name	Description
NCS (Method-1)	Normalized Cut Saliency
C-CRF (Method-2)	Continuous Conditional Random Field saliency
MPDS (Method-3)	Manifold-Preserving Diffusion-based Saliency
CD (Method-4)	Color Contrast and Distribution-based saliency
GP (Method-5)	Geodesic Propagation-based saliency
MB+ ([86])	Extended Minimum Barrier saliency

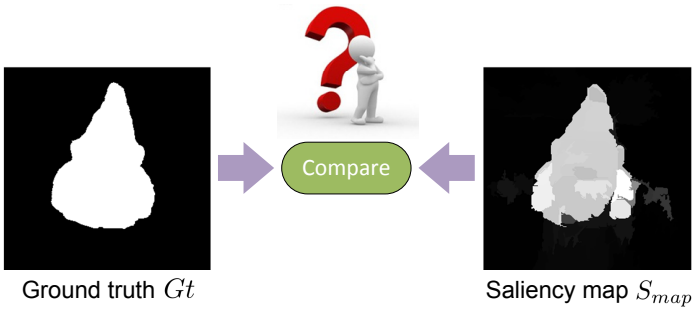

Figure 4.19: The evaluation problem for salient region/object detection is formulated by comparing between a binary ground truth map Gt and a continuous-valued saliency map S_{map}. The original image is shown in Figure 1.2.

image processing, as an oral presentation. The method applies minimum barrier distance (MBD) transform to saliency detection. A preliminary saliency map $\mathcal{B}$ is obtained by computing MBD of image pixels to image boundary. Then the map is combined with backgroundness cues and further enhanced by some post-processing. The enhanced map is denoted as $\mathcal{B}+$. Here abbreviation "MB+" stands for their extended version that generates the enhanced saliency map $\mathcal{B}+$. We use the public code of implementation that the authors provide.

Evaluation Metrics

Let a saliency map be S_{map} and the corresponding binary ground truth be Gt. Assume that prior to evaluation, S_{map} is normalized into the range [0,1] by $\frac{S_{map}-\min\{S_{map}\}}{\max\{S_{map}\}-\min\{S_{map}\}}$, whereas in the groundtruth Gt, pixels are either labeled as "salient" (with values 1), or "non-salient" (with values 0). The evaluation problem is to compare between a continuous-valued map S_{map} and a binary map Gt, as shown in Figure 4.19. Below we describe four metrics that are commonly adopted for evaluating a salient region detection

model. Some of these metrics are adopted in the appended papers in Part II of this thesis.

1) *Precision-Recall (PR)* [7, 8] is defined as:

$$\text{Precision}(T) = \frac{|M(T) \cap Gt|}{|M(T)|}, \quad \text{Recall}(T) = \frac{|M(T) \cap Gt|}{|Gt|} \tag{4.8}$$

where $M(T)$ is the binary mask map obtained by directly thresholding the saliency map S_{map} with a threshold T, and $|\cdot|$ is the total area of the mask(s) in the map. By varying T from 0 to 1, a precision-recall curve can be obtained.

2) *F-measure* (F_β) [7, 8] is a metric integrating precision and recall, which is defined as:

$$F_\beta = \frac{(1+\beta^2)\text{Precision} \cdot \text{Recall}}{\beta^2 \cdot \text{Precision} + \text{Recall}} \tag{4.9}$$

where β^2 is a non-negative harmonic weight between the precision and the recall. $\beta^2 = 0.3$ is usually set since the precision is often weighted more than the recall [8]. In order to get a single-valued F_β score instead of a curve, existing works usually first binarize S_{map} into a foreground mask map, leading to a single precision value and recall value. The most common way to do this is the adaptive thresholding suggested by Achanta *et al.* [8], where the adaptive threshold is defined as *two times of the mean value* of the saliency map.

3) *Mean Absolute Error (MAE)* [64, 79] is defined as:

$$MAE = \frac{1}{W \cdot H} \sum_{x=1}^{W} \sum_{y=1}^{H} |S_{map}(x,y) - Gt(x,y)| \tag{4.10}$$

where $S_{map}(x,y)$ and $Gt(x,y)$ correspond to the saliency value and ground truth value at pixel location (x,y), respectively. W and H are the width and height of S_{map}. It can be seen that MAE basically is an accumulation of pixel-wise errors, and the result provides an intuitive measure of the difference between S_{map} and Gt.

4) *Weighted F-measure* (F_β^w) is recently proposed by Margolin *et al.* [112]:

$$F_\beta^w = \frac{(1+\beta^2)\text{Precision}^w \times \text{Recall}^w}{\beta^2 \times \text{Precision}^w + \text{Recall}^w} \tag{4.11}$$

where Precision^w and Recall^w are the weighted precision and recall. The difference between (4.11) and (4.9) is that Precision^w and Recall^w in (4.11) can directly compare a non-binary map against a binary

ground truth without thresholding. Since the definition of Precision^w and Recall^w is a bit complex, here they are omitted. Interested readers are referred to [112] for more details.

Among the above four metrics, high precision-recall curves, high F_β, F_β^w, and low MAE indicate good saliency models. It is worth noting that the four metrics sometimes do not agree with each other. The same observation is reported in [27]. The reason is that they have concerned different aspects and properties of a model. In this sense, they could be instructive to selecting appropriate models for specific application requirements.

Performance

A. Quantitative comparison

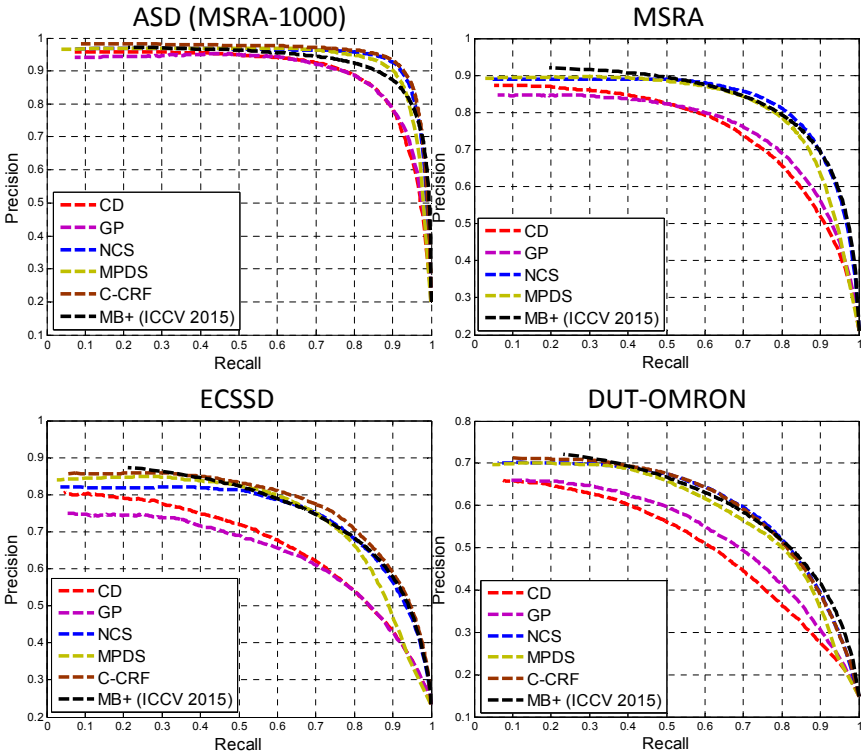

Figure 4.20: Quantitative comparison by precision-recall curves.

The quantitative comparison results of the six models aforementioned are shown in Figure 4.20 and Figure 4.21. Figure 4.20 shows the precision-recall curves. Comparison by F_β, MAE, and F_β^w are shown in Figure 4.21. Note

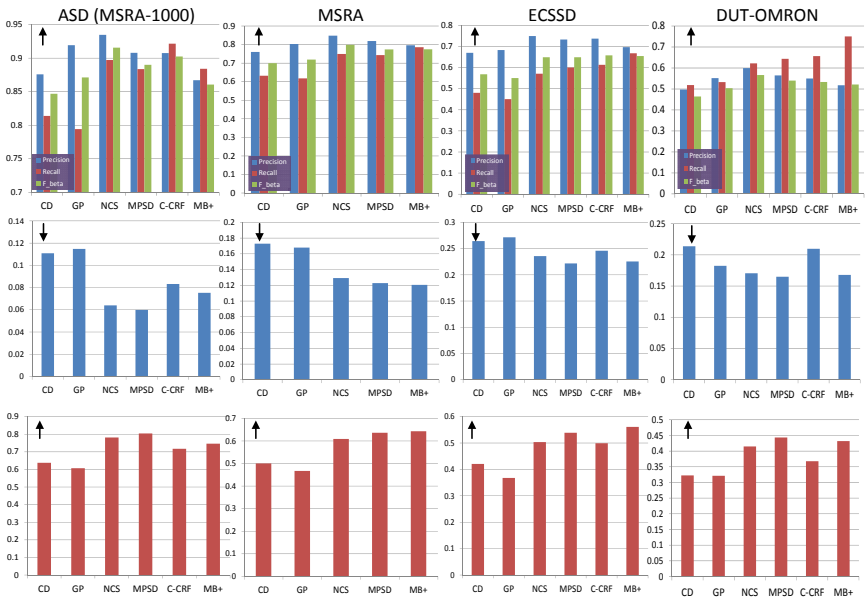

Figure 4.21: Quantitative comparison by F_β (the top row), MAE (the middle row), and F_β^w (the bottom row). The upper left markers ↑ in the figures mean higher criterion values indicate better performance, whereas the markers ↓ mean lower criterion values indicate better performance.

that since Method-2 (C-CRF) uses 3000 images from MSRA for training the continuous conditional random field, this method is not compared on the MSRA dataset. Regarding to the precision-recall curves, several observations can be found from Figure 4.20: *i*) The baseline models CD and GP that integrate the simple color contrast and color distribution or enhance the saliency maps by geodesic propagation consistently achieve inferior performance comparing to NCS, MPDS, and C-CRF. *ii*) The NCS, MPDS, C-CRF achieve state-of-the-art performance.

From Figure 4.21, one can observe that the metrics sometimes do not agree with each other. Regarding to the F_β, NCS is the best model on the three out of the four datasets, namely on ASD, MSRA, and DUT-OMRON. On ECSSD, C-CRF is the best, which performs slightly better than NCS, MPSD and MB+. In terms of the MAE criterion, MPDS consistently performs the best on ASD, ECSSD and DUT-OMRON. The state-of-the-art method MB+ is also good at MAE, since it is the best on MSRA and the second best on ECSSD and DUT-OMRON. Finally regarding to F_β^w, MPSD is the best on ASD and DUT-OMRON, whereas MB+ is the best on MSRA and ECCSD. However, the three metrics in Figure 4.21 agrees with

the precision-recall curves on the fact that CD and GP still rank bottom.

Figure 4.22: Qualitative comparison for the methods proposed in this thesis (CD, GP, NCS, MPDS, C-CRF) and a state-of-the-art salient region detection model MB+ [86] on four benchmark datasets. GT means the ground truth.

B. Qualitative comparison

Qualitative comparison results are shown in Figure 4.22, where three sample images from each image set are included. Several observations can be found from Figure 4.22. First, regarding to the visual quality, CD and GP are inferior to the other contenders. Though they could generate decent results on some simple images (1st, 2nd, 4th rows in Figure 4.22), they still

have limited capability on emphasizing entire objects in complex scenarios. CD may be easily distracted by some background clutter with high contrast (e.g., 7th row) whereas GP is prone to some small-scale high-contrast patterns (e.g., 4th, 11th, 12th rows). Second, varied performance of NCS, MPDS, C-CRF, and MB+ reveal there should be potential advantages for each method over the others. For example, only NCS and C-CRF succeed on highlighting the object meanwhile suppressing the background in the last row in Figure 4.22. The saliency maps of MPDS are perceptually the closet to the ground truth in the 1st, 8th, and 9th rows. MB+ often provides saliency maps with high foreground-background contrast, namely the saliency assignment is either very high or very low. There are very few regions with middle-level saliency in the resultant saliency maps. However, MB+ has some limitation on preserving clear object boundaries in cluttered scenes, such as the 6th-8th rows in Figure 4.22.

C. Efficiency and speed

The average running time on ASD dataset of the proposed methods is shown in Table 4.2. The running time was all acquired on an Intel i7-4720HQ 2.6GHz laptop with 8GB memory using non-optimized Matlab code. Regarding to the running time, CD and GP are two fastest methods due to their simplicity in saliency computation. The speed of NCS and C-CRF is close, since the former requires multi-level region merging and eigenvector solving whereas the latter needs to extract various unary and pairwise features. MPDS has the longest running time because it obtains adaptive graph weights on every image through optimization. Note the computation time of MPDS varies on different images, from 1 second to about 8 seconds.

Table 4.2: Average CPU time in seconds on ASD dataset. All methods are based on Matlab implementation without optimization. We only have compared the running time on Matlab, so MB+ whose released code is C++ is not considered here.

Methods	CD	GP	NCS	MPDS	C-CRF
Time(sec)	1.4	0.8	2.6	3.9	2.1

Discussion and Remarks

From the results obtained, the rankings of models based on the average rankings over all datasets are summarized in Table 4.3. From Table 4.3, we can conclude that: 1) In terms of precision-recall and F_β, the top two models are C-CRF and NCS. 2) Under F_β^w and MAE, MPDS and MB+ perform the best. 3) About the processing speed, GP and CD are the fastest.

As aforementioned, different metrics are instructive to different applica-

Table 4.3: Rankings of models under different evaluation metrics over all datasets. The ranking position of a method is the average of its ranking over all four datasets. The best two rankings in each row are highlighted in red and blue.

Methods	CD	GP	NCS	MPDS	C-CRF	MB+
Precision-recall	6	5	2	4	1	3
F_β	6	5	1	3	2	4
F_β^w	5	6	3	1	4	2
MAE	6	5	3	1	4	2
Time	2	1	4	5	3	null

tions. CD and GP may be good choices on simple images since they run fast and achieve fairly good results. They may also be suitable for some efficiency-demanded tasks. NCS and C-CRF achieve good precision-recall curves and F_β. They could be decent choices for tasks, e.g., the Saliency Cut in [7], that benefit from high precision under a pre-defined recall rate. High F_β means that the obtained binary masks by simple adaptive thresholding have good accuracy of fitting to entire objects. NCS and C-CRF hence are also good candidates for applications that require initial object masks as input but are limited to only efficient segmentation methods due to hardware/platform constraints. MPDS and MB+ achieve low MAE and high F_β^w, so their saliency maps present high contrast between foreground and background and are good approximation to the binary ground truth. Among the proposed methods, NCS is a good trade-off between precision-recall curve and F_β^w, since it ranks the second among the proposed methods on both precision-recall curve (following C-CRF) and F_β^w (following MPDS).

4.2 Traffic sign recognition methods

This section describes two proposed methods (Method-6 and Method-7) on traffic sign recognition. As shown in Figure 4.23, Method-6 introduces a complete traffic sign recognition framework based on coarse-to-fine learning, together with a sign salient region extraction method based on geodesic propagation. Inspired by the work in Method-6, Method-7 further proposes an improved method towards sign salient region extraction, which is based on signed geodesic transform. In the following we describe each method and its main contributions in detail.

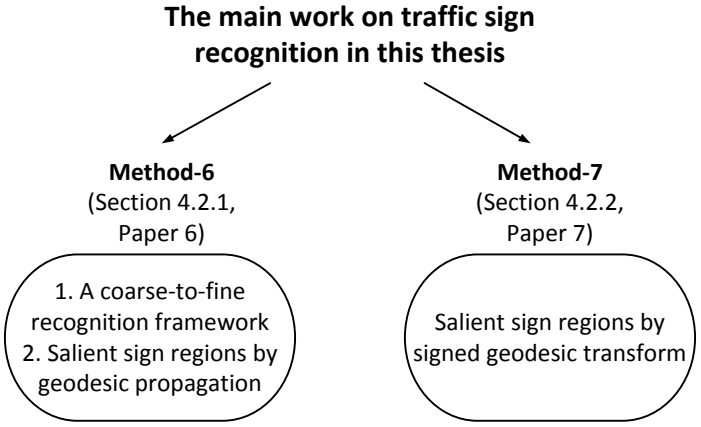

Figure 4.23: The summary of the thesis work on traffic sign recognition.

4.2.1 Method-6: A novel coarse-to-fine recognition scheme with salient region features

(Summary of Paper 6)

Problem addressed: This method addresses applying salient region detection to traffic sign recognition (TSR).

Basic ideas: Despite in different countries, traffic signs are usually divided into several categories. Each category has a certain type of meanings. For example, the category "prohibitory signs" describes some actions that are forbidden. Signs in the same category usually share common attributes such as shapes and colors. Motivated by this, Method-6 formulates sign detection as coarse classification between sign categories versus the background, whereas sign classification is formulated as fine classification of signs within

each category. In the classification stage, a conventional mean to extract features for classification is to consider an entire detection window [91, 92]. However, since the image within the window could contain structural background and a not-well-aligned sign, such feature extraction does not provide accurate characterization of a sign. To enhance classification, Method-6 investigates a scheme for segmenting salient sign regions. To achieve this, we modify Method-5 and use the sign detection window to replace the Harris convex hull that provides a coarse saliency region, and then employ geodesic propagation to obtain an enhanced sign region.

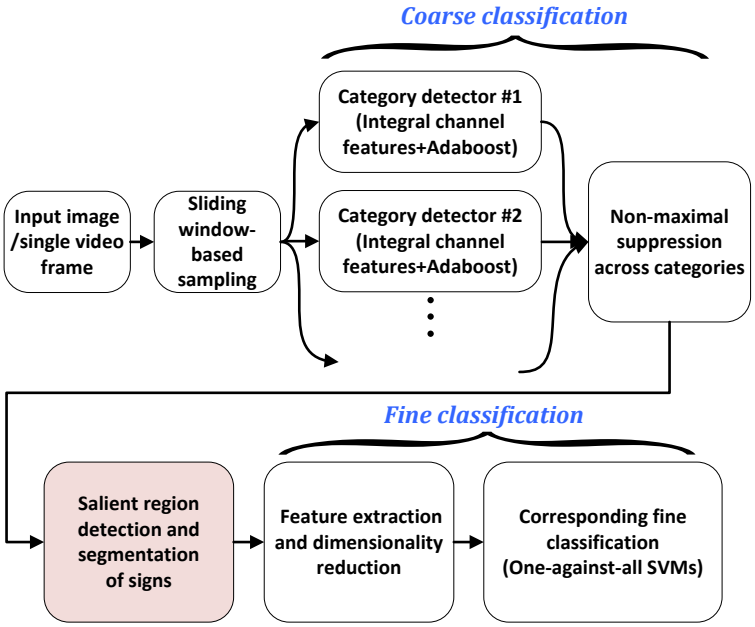

Figure 4.24: The block diagram of Method-6 (a coarse-to-fine learning scheme) for traffic sign recognition, where salient region detection (the pink module) is inserted as an intermediate module. In Method-6 there are two learning stages, corresponding to coarse classification and fine classification, respectively.

Big Picture: As shown in Figure 4.24, sign detection and classification are designed through learning in a coarse-to-fine manner. In the coarse stage, classification is performed between individual sign categories and background. In other words, this stage is equivalent to sign detection. Its

outcome is a coarse classification result indicating whether a window belongs to a sign category or background. As shown Figure 4.24, category detectors in Method-6 uses integral channel features [101, 113] and Adaboost classifiers. Next, a saliency-based method is employed to select an informative region of sign from a window. In the fine stage, features are extracted from segmented salient regions, and then classified by one-against-all support vector machines (SVMs). Figure 4.28 shows an example of salient region detection and segmentation.

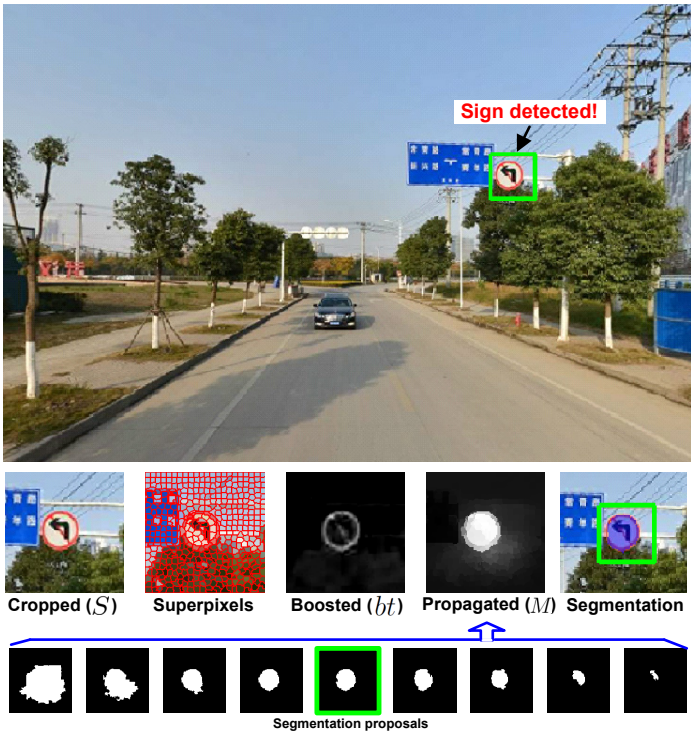

Figure 4.25: Detection and segmentation of a Chinese traffic sign (prohibitory sign) in a street view image. The second row visualizes some intermediate results. From left to right are: cropped local image (S), superpixel segmentation, boosted color channel (bt), propagated saliency map (M), and final segmentation (light blue mask superimposed on the sign). The last row shows some generated segmentation proposals, where the one selected automatically is outlined by green rectangle.

Main contributions:

- A learning-based TSR scheme (including both sign detection and classification) is proposed for street view images, through a coarse-to-fine process.
- A saliency-based feature extraction method is proposed through salient region detection and segmentation, where geodesic propagation is employed.
- The work that utilizes salient region detection for sign classification is new. It differs from previous work where saliency detection is only applied as a pre-processing step in traffic sign detection [93–95].

Table 4.4: Evaluation of sign detection. TP: True positive. FP: False positive. FN: False negative

Category	Training samples	Test images	TP	FP	FN	Precision	Recall
Prohibitory	1367	220	341	19	23	94.72%	93.68%
Warning	980	188	180	8	15	95.74%	92.31%

Figure 4.26: Sign examples of Chinese prohibitory and warning signs considered in Method-6.

Main results: Method-6 is tested on recognizing two Chinese sign categories (prohibitory signs and warning signs) from Tencent street view images (`http://map.qq.com/`). The collected street view images are split into training and testing sets, where each image contains at least one sign from the two categories. Training samples are then collected from the training set. In the experiments, sign detection and sign classification are evaluated separately. Table 4.4 shows the results from sign detection. Method-6 achieves over 90% precision and recall rate on both categories. To test sign classification, we further collected samples and manually categorized them into different classes. In the end, each sign class contains approximate 200 sign samples. For exact classes involved, see Figure 4.26. In each class, we randomly choose 80% samples for training, and the remaining 20% samples for testing. Such a process is repeated for 20 times. The resulting confusion matrices (averaged) are shown in Figure 4.28, where Method-6 achieves high true positive rate and relatively low false positive rate. Figure 4.27 shows some examples of simultaneous detection, segmentation, and classification.

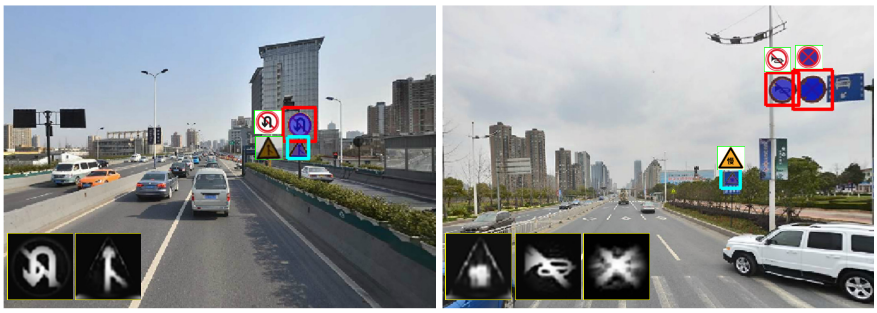

Figure 4.27: Recognizing multiple categories of signs. Prohibitory signs are detected as red boxes. Warning signs are detected as cyan boxes. Light blue masks superimposed on the signs are segmented salient regions. Extracted feature maps are shown on the bottom left. Artificial images beside the detected signs mean the classification results from Method-6.

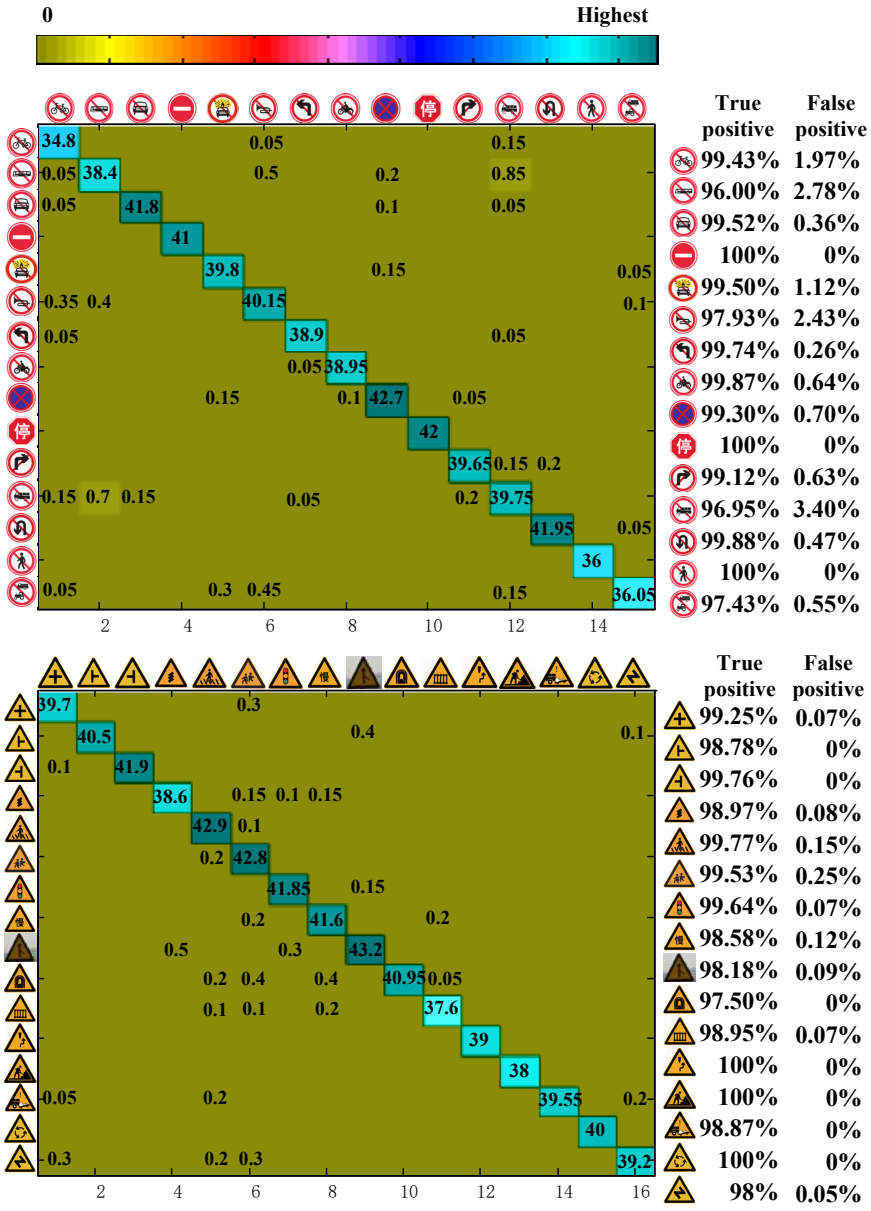

Figure 4.28: The classification confusion matrices (zero entries are not numbered). Top: From 15 classes in the prohibitory category; Bottom: From 16 classes in the warning category.

4.2.2 Method-7: Geodesic distance transform-based salient region segmentation for sign recognition

(Summary of Paper 7)

Problem addressed: The same as Method-6, this method addresses applying salient region detection to traffic sign recognition (TSR).

Basic ideas: Based on the observation that a sign in a detected window usually does not touch window boundary, Method-7 proposes to compute a saliency map through geodesic distance transform from window boundary. To concern the location prior and also make the detected salient region more distinguishable from the background, the signed geodesic transform (SGT) is employed.

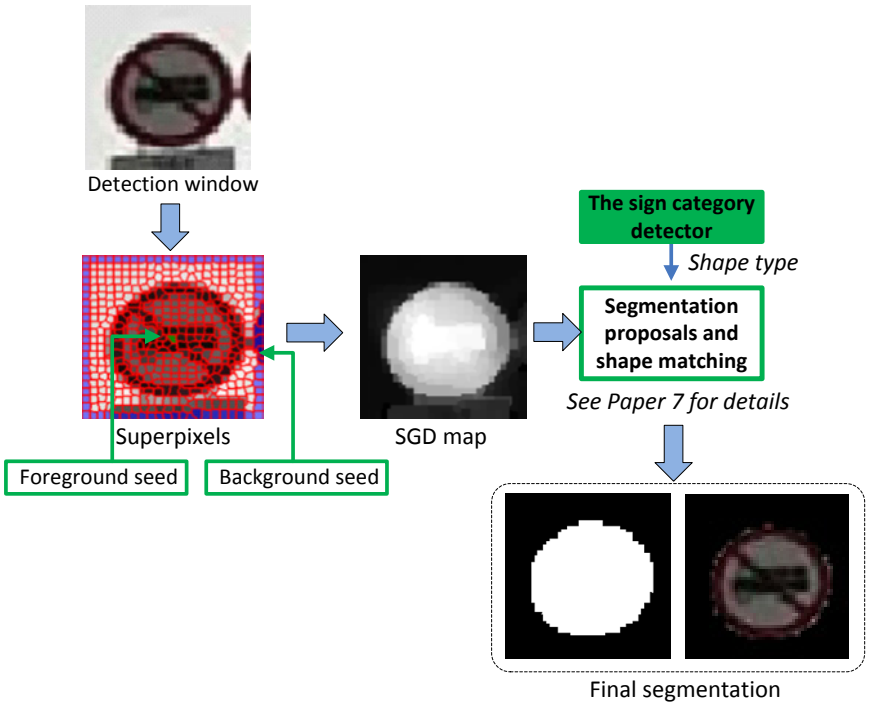

Figure 4.29: The block diagram of Method-7, which generates a salient region automatically from the sign detection window.

Big picture: As shown in Figure 4.29, first the image in the detection window is over-segmented into superpixels. Next, a superpixel-based signed geodesic distance map (SGD map) is obtained by applying signed geodesic

transform (SGT) regarding to specified foreground and background seeds. Finally, a sequence of processes (in the module of "segmentation proposals and shape matching" in Figure 4.29) are proposed to obtain a segmentation mask from the distance map. As results, Method-7 can automatically generate a salient sign region from the detection window.

Main contributions:

- We propose a SGT-based method that is employed after sliding window-based detection. It is able to automatically extract salient sign regions with different shapes.

- We propose an effective method that yields the final segmentation from the signed geodesic distance map. Different from generic object segmentation, our method incorporates shape constraints and achieves robust segmentation.

Table 4.5: Classification results (420 images, with/without Method-7).

	without Method-7	with Method-7
Correct classification (signs)	584	673
False classification (signs)	109	20
Classification accuracy	84.27%	97.11%

Main results: The experimental validation of Method-7 is based on the coarse-to-fine TSR system proposed in Method-6. Following the work of Method-6, we have further collected 11,683 street images from Tencent street view. 8,044 images are used for training purpose and 3,639 images are used for testing. To validate Method-7, we took 420 images from the testing set as a "segmentation validation set". The reason for not using the entire testing set is that the dataset was under construction and the manual annotations had not been finished. On this validation set, we focus on the performance on prohibitory signs, where the sign detector achieves 98.3% precision and 97.33% recall (higher than that reported in Method-6 attributed to more training data). A sign detection window is then classified into one of 30 sign classes (also increased compared to the results in Method-6). The experiment is conducted *with* and *removing* the segmentation module (namely Method-7) from the TSR system, so that the impact of Method-7 can be seen. As shown in Table 4.5, incorporating Method-7 drastically improves the classification accuracy from 84.27% to 97.11%. Several visual results are shown in Fig.4.30, where extracting features from

the whole windows leads to false classification (1st row in Fig.4.30) whereas incorporating Method-7 leads to correct classification (2nd row in Fig.4.30).

Figure 4.30: Effectiveness of the segmentation module (Method-7). 1st row: Erroneous classification without the saliency-based segmentation (Method-7). 2nd row: Correct classification by incorporating the saliency-based segmentation (Method-7). The segmentation masks are indicated by the light green color superimposed on the signs. The artificial pictures show the classification results from our TSR system.

Extended experiments and results

A. On extended Chinese dataset

Our formal Chinese dataset comprises 8,044 training images and 3,237 testing images. All these images are full-resolution street view images, not sign images. Training samples are extracted from the training set and then are manually categorized into different classes. In total, there are three sign categories with different number of classes, whose introduction is in Table 4.6. Details of classes are shown in Figure 4.31-Figure 4.33. Regarding to the 3,237 testing images, we manually annotate the ground truth for sign detection and classification. In Figure 4.34, we show the ground truth distribution on this testing set.

Figure. 4.35 shows the confusion matrix, precision, recall for category detection of signs. Our system achieves over 90% precision and recall rate on all categories. In Table 4.7 we show the classification rate of the proposed system on sign classification. In Figure 4.36-Figure 4.39, some detection, segmentation, and classification results from full-resolution street view images are shown. Our system handles various scenarios, such as multiple sign categories or classes appearing in a same image (Figure 4.36), signs in complex background (Figure 4.37), signs in both small and large sizes (Figure 4.37), signs under various luminance (Figure 4.38), and signs with slight shape distortion (Figure 4.38). Figure 4.39 shows several failure cases of sign detection.

B. On German traffic sign classification benchmark

In addition, our system is transferred to the German traffic sign classification benchmark (GTSCB) [36] and is evaluated. The German traffic sign dataset proposed by Stallkamp *et al* [36] is a well-established and challenging benchmark containing large number of training and testing samples. It comprises over 50,000 images (39,209 training samples and 12,630 testing samples) in total. The signs are categorized into 43 classes. Since our system is a coarse-to-fine scheme, where the saliency-based segmentation (Method-7) relies on the detection (coarse classification) of categories to provide shape prior, we divide all the classes into five categories and train five category detectors accordingly. Figure 4.40 shows our categorization of the 43 classes, which is mainly based on colors and shapes. In the testing stage, a test sample is first classified by all category detectors and then is assigned to its most-likely class. Figure 4.42 shows the classification results on GTSCB. Our system achieves the classification rate $12107/12630 = 95.86\%$, which is somewhat comparable to existing methods based on traditional features and classifiers [36, 114].

Table 4.6: Numbers of collected training samples in the 3 Chinese categories.

Category	Sample number	Class number
Prohibitory	8938	34
Warning	2568	21
Indication	1815	11

Index	*Class*	*Total q'ty*	*Im'g*	*Index*	*Class*	*Total q'ty*	*Im'g*
1	no-walking	22		18	SpLim:20	57	
2	no-bike	14		19	SpLim:30	129	
3	no-bus	68		20	SpLim:40	619	
4	no-car	76		21	SpLim:50	106	
5	no-entry	194		22	SpLim:60	756	
6	no-explosive	41		23	SpLim:70	49	
7	no-horn	95		24	SpLim:80	769	
8	no-left-turn	47		25	SpLim:90	195	
9	no-motor-bike	19		26	SpLim:100	1160	
10	no-parking	464		27	SpLim:110	186	
11	no-stopping	12		28	SpLim:120	1057	
12	no-right-turn	33		29	no-overtake	89	
13	no-truck	379		30	enable-overtake	15	
14	no-U-turn	1514		31	no-pass	64	
15	SpLm:5	81		32	no-phone	39	
16	combination	76		33	no-tractor	19	
17	SpLim:10	13		34	unknown	481	

Figure 4.31: Details of classes in Chinese prohibitory category for training, as summarized in Table 4.6. The unknown class contains sign samples in this category but not belong to the classes listed. In total, there are 8938 training samples in 34 classes.

Index	Class	Total q'ty	Im'g	Index	Class	Total q'ty	Im'g
1	warn-cross	61		12	warn-go-right	11	
2	warn-right-T	76		13	warn-construct	105	
3	warn-left-T	76		14	warn-accident	36	
4	warn-zzz	42		15	warn-cycle	12	
5	warn-human	82		16	warn-z	47	
6	warn-kids	10		17	warn-danger	308	
7	warn-light	16		18	warn-narrow	172	
8	warn-slow	64		19	warn-left-turn	211	
9	warn-right-Lane	805		20	warn-right-turn	215	
10	warn-tunnel	30		21	unknown	177	
11	warn-railway	12					

Figure 4.32: Details of classes in Chinese warning category for training, as summarized in Table 4.6. The unknown class contains sign samples in this category but not belong to the classes listed. In total, there are 2568 training samples in 21 classes.

Index	Class	Total q'ty	Im'g	Index	Class	Total q'ty	Im'g
1	min60	896		7	must-horn	15	
2	min70	10		8	must-left	95	
3	min80	209		9	must-right	95	
4	min90	204		10	must-straight	38	
5	min100	89		11	unknown	44	
6	min110	120					

Figure 4.33: Details of classes in Chinese indication category for training, as summarized in Table 4.6. The unknown class contains sign samples in this category but not belong to the classes listed. In total, there are 1815 training samples in 11 classes.

Prohibitory category			Warning category			Indication category		
Index	*Class*	*GT q'ty*	*Index*	*Class*	*GT q'ty*	*Index*	*Class*	*GT q'ty*
1	no-walking	11	1	warn-cross	8	1	min60	544
2	no-bike	5	2	warn-right-T	15	2	min70	16
3	no-bus	31	3	warn-left-T	11	3	min80	110
4	no-car	15	4	warn-zzz	13	4	min90	100
5	no-entry	50	5	warn-human	9	5	min100	46
6	no-explosive	4	6	warn-kids	26	6	min110	38
7	no-horn	18	7	warn-light	0	7	must-horn	0
8	no-left-turn	9	8	warn-slow	22	8	must-left	9
9	no-motor-bike	6	9	warn-right-Lane	253	9	must-right	27
10	no-parking	197	10	warn-tunnel	2	10	must-straight	1
11	no-stopping	1	11	warn-railway	0	11	unknown	14
12	no-right-turn	1	12	warn-go-right	4		Total	905
13	no-truck	60	13	warn-construct	44			
14	no-U-turn	535	14	warn-accident	18			
15	SpLm:5	61	15	warn-cycle	0			
16	combination	20	16	warn-z	0			
17	SpLim:10	1	17	warn-danger	5			
18	SpLim:20	62	18	warn-narrow	65			
19	SpLim:30	43	19	warn-left-turn	16			
20	SpLim:40	316	20	warn-right-turn	25			
21	SpLim:50	9	21	unknown	168			
22	SpLim:60	327		Total	698			
23	SpLim:70	62						
24	SpLim:80	344						
25	SpLim:90	94						
26	SpLim:100	592						
27	SpLim:110	37						
28	SpLim:120	465						
29	no-overtake	69						
30	enable-overtake	19						
31	no-pass	24						
32	no-phone	0						
33	no-tractor	3						
34	unknown	55						
	Total	3546						

Figure 4.34: Ground truth distribution on 3,237 street view images for testing.

Predicted class

Actual class	Prohibitory	Warning	Indication	Background
Prohibitory	3475	0	10	61
Warning	0	688	0	10
Indication	24	0	849	32
Background	61	20	24	N.A.

Category	Precision	Recall
Prohibitory	97.61%	97.99%
Warning	97.18%	98.57%
Indication	97.14%	93.81%

Figure 4.35: Confusion matrix and precision, recall rate for category detection on the Chinese test set (3,237 street images).

Table 4.7: The classification rate (CR) of the proposed system on the Chinese test set. Two cases, namely "with unknown class" and "without unknown class", are considered. "With unknown class" means that the same as other classes, a classifier (one-against-all SVM) is trained for the unknown class. An instance to be predicted will have the possibility to be classified as "unknown". On the other hand, "without unknown class" means that the unknown class is totally excluded from the training set and testing set.

Category	With unknown class	Without unknown class
Prohibitory	CR=95.48%	CR=97.40%
Warning	CR=85.47%	CR=93.97%
Indication	CR=93.76%	CR=94.49%

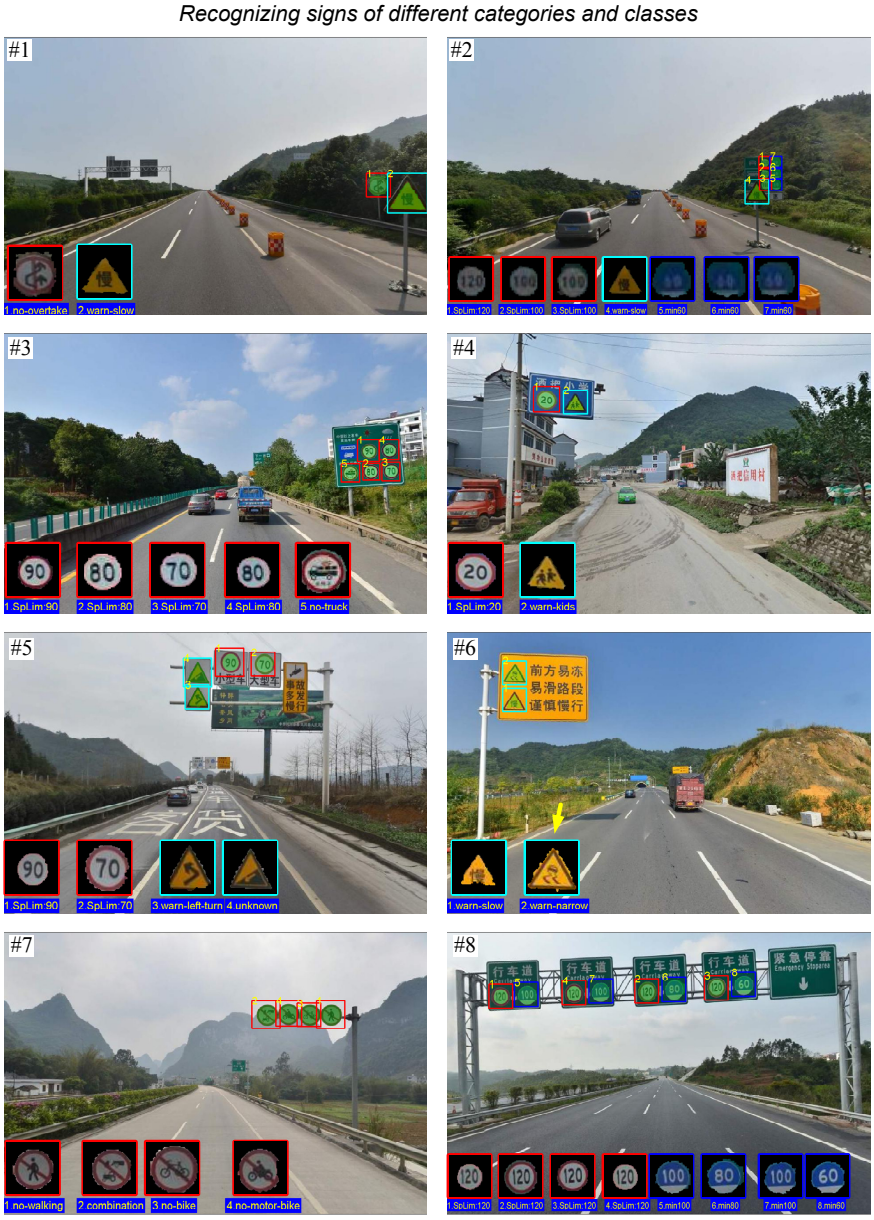

Figure 4.36: Sign recognition from street view images, where an erroneous classification in image #6 is highlighted by a yellow arrow.

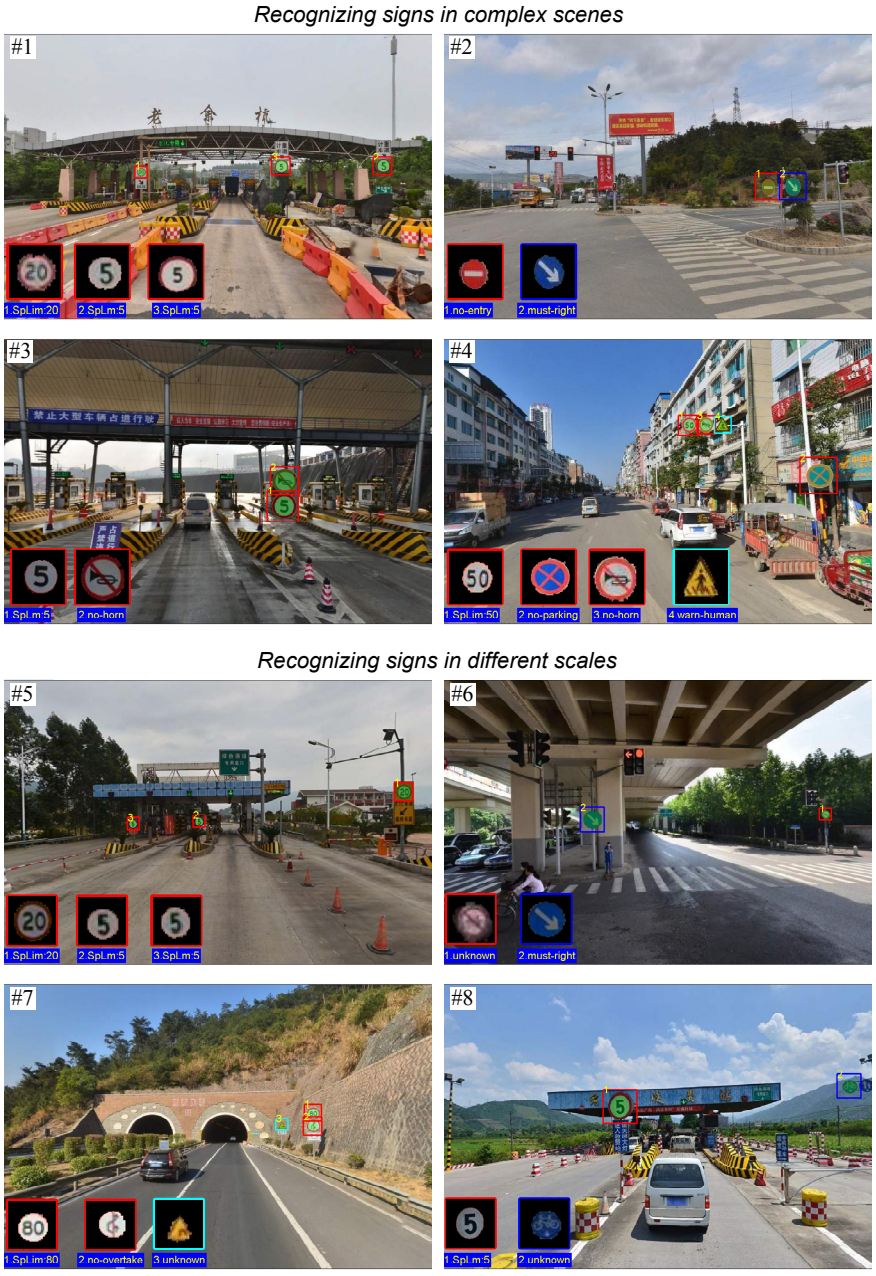

Figure 4.37: Recognize signs from street view images having complex background and signs of different scales.

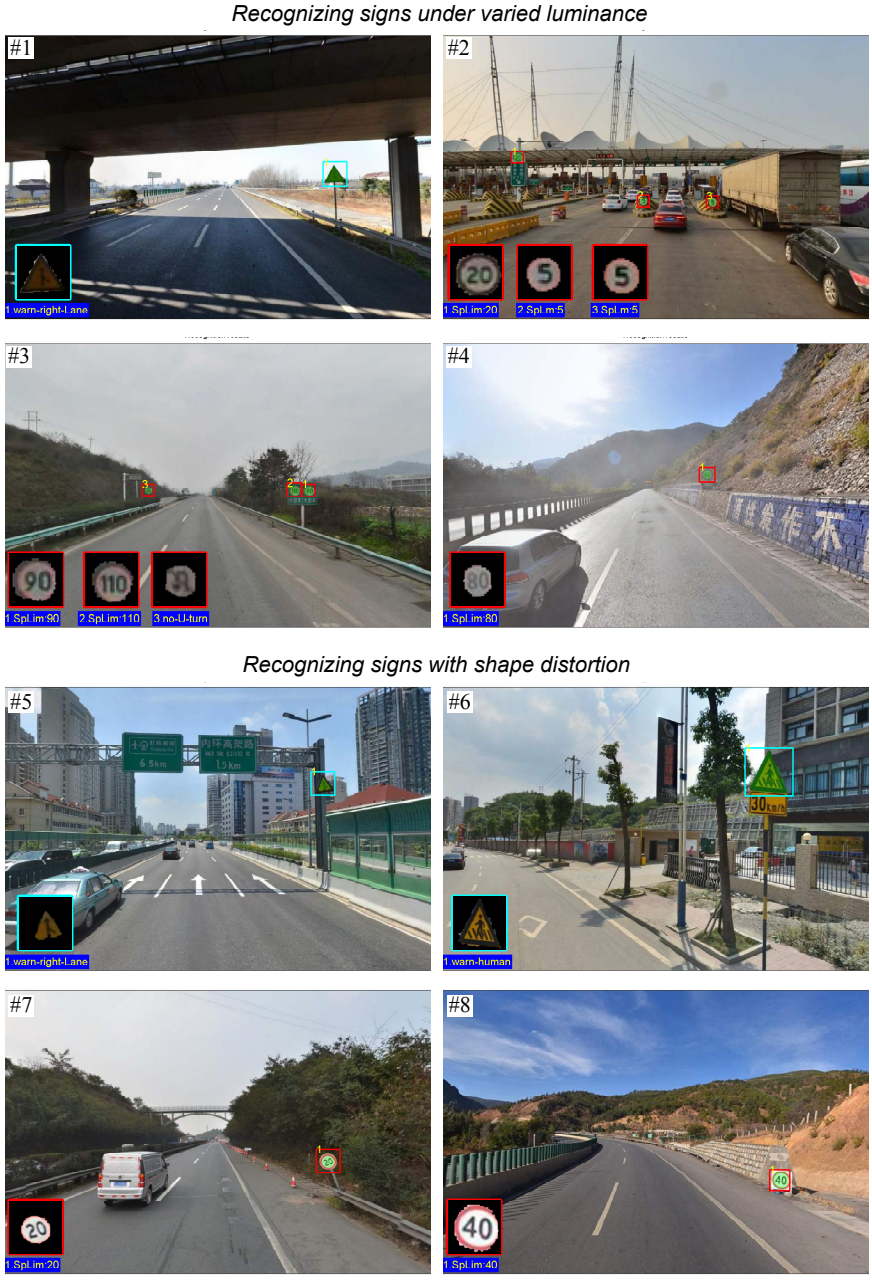

Figure 4.38: Recognize signs from street view images having various luminance and signs with shape distortion.

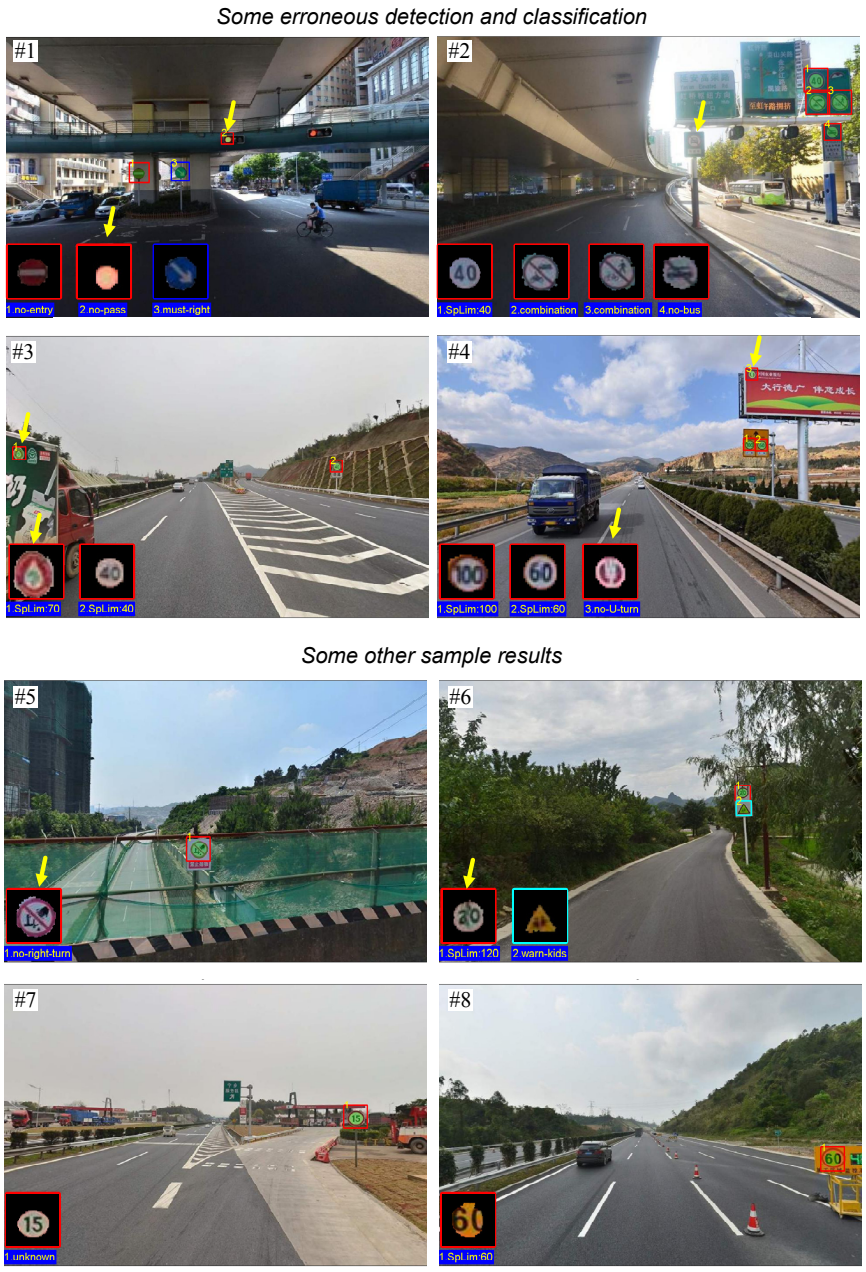

Figure 4.39: Recognition results, where erroneous detection and erroneous classification are highlighted by yellow arrows. Several interesting results are exhibited in images from #5 to #8.

Figure 4.40: Categorization of 43 sign classes (ID from #0 to #42) in the GTSCB dataset for training category detectors in the coarse learning stage. The class IDs are the same as those in the GTSCB [36].

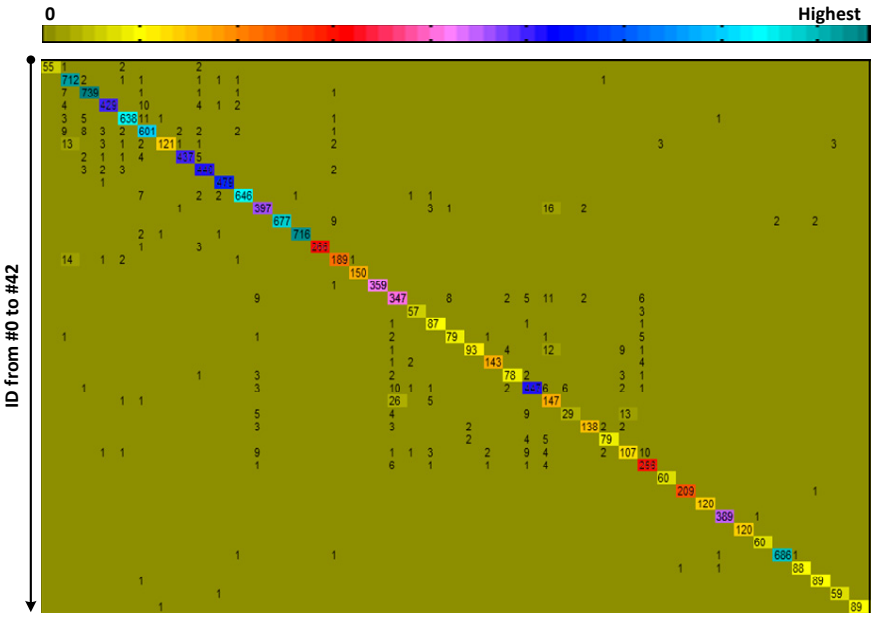

Figure 4.41: Classification confusion matrix for the 12,630 test sign samples from GTSCB by using the proposed method. In the above matrix, zero entries are not numbered.

Team	Method	CR
IDSIA	Committee of CNNs	99.46%
INI-RTCV	Human Performance	98.84%
sermanet	Multi-Scale CNNs	98.31%
CAOR	Random Forests	96.14%
Ours	Saliency + SVMs	95.86%
INI-RTCV	LDA on HOG 2	95.68%
INI-RTCV	LDA on HOG 1	93.18%
INI-RTCV	LDA on HOG 3	92.34%

Figure 4.42: Comparison of the classification rate (CR) on GTSCB. Teams are: IDSIA [39], INI-RTCV [36], Sermanet [115], CAOR [114], and Ours. Note that the parameters of our system are not carefully tuned for GTSCB.

4.2.3 Comparison between Method-6 and Method-7

Geodesic propagation (GP) is adopted in Method-6 to derive a saliency map for segmentation. In contrast, Method-7 utilizes signed geodesic transform (SGT) to derive the saliency map, where salient sign regions are expected to be highlighted as well. The main difference between them is that: Method-7 uses only the information inside the detection window whereas Method-6 uses contextual information both inside and outside the window. In Method-6, the propagation is conducted from the inside to the outside of the window. This means that the propagated saliency map is also affected by the image contents outside the window. Therefore, the propagation could be less effective when there is less context, for example when the detection window is around scene borders. In contrast, Method-7 does not have such a problem. Another advantage of Method-7 is that it is applicable when there are only sign images available without any contextual information. This facilitates the collection of training samples for sign classification, because no contextual information needs to be stored. This advantage also makes Method-7 applicable to some other sign datasets (e.g., the German sign classification benchmark GTSCB [36]) where only sign images (rather than full street view images) are provided. In all, Method-7 is an improvement over Method-6, and in our current system, we use Method-7 instead of the GP-based segmentation in Method-6.

Chapter 5

Conclusion

Salient region/object detection is an active research direction in the field of computer vision and image processing. After investigating existing models, this thesis proposes 5 new methods towards improved salient region detection. Considering the contributions in theoretical aspects, we have proposed the utilization of normalized graph cut, continuous conditional random field (C-CRF), manifold-preserving diffusion (MPD), color attributes, and geodesic propagation methods for modeling the properties of salient objects and developed new computational effective algorithms for saliency detection. Experiments on several benchmark datasets were conducted to evaluate the performance of the proposed methods. Results, comparisons, and evaluations show that these methods achieve comparable/better performance to the state-of-the-art. Further, comparisons of the proposed methods show that each of them has its advantages and limitation. This knowledge can be used for choosing a suitable method for real applications. In addition, we have applied salient region detection to traffic sign recognition (TSR) from street view images. Experiments and initial results indicate the proposed methods are feasible and successful for such an application.

5.1 Future work

Despite significant research progress, robust detection of salient regions remains challenging due to the sophisticated mechanism of human attention. On some difficult datasets there are relatively large gaps between the state-of-the-art performance and the perfect one. Thereby, the issue of salient region detection requires further study. To author's view, the following studies are worth continuing: 1) Studying the attention mechanism and discovering more useful cues and hypotheses, which then can be integrated to the model design; 2) Powerful learning techniques can be investigated,

e.g., deep learning and convolutional neural networks, to discover potential complex rules for feature extraction and fusion; 3) Other applications which benefit from salient regions may be investigated.

References

[1] A. Triesman and G. Gelade, "A feature-integration theory of attention," *Cognitive Psychology*, vol. 12, no. 1, pp. 97–136, 1980.

[2] C. Koch and S. Ullman, "Shifts in selective visual attention: towards the underlying neural circuitry," *Matters of intelligence*, pp. 115–141, 1987.

[3] M. Hayhoe and D. Ballard, "Eye movements in natural behavior," *Trends in cognitive sciences*, vol. 9, no. 4, pp. 188–194, 2005.

[4] L. Itti and C. Koch, "Computational modelling of visual attention," *Nature reviews neuroscience*, vol. 2, no. 3, pp. 194–203, 2001.

[5] A. Borji and L. Itti, "State-of-the-art in visual attention modeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 35, no. 1, pp. 185–207, 2013.

[6] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 20, no. 11, pp. 1254–1259, 1998.

[7] M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S. Hu, "Global contrast based salient region detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 37, no. 3, pp. 569–582, 2015.

[8] R. Achanta, S. Hemami, F. Estrada, and S. Süsstrunk, "Frequency-tuned salient region detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[9] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2. IEEE, 2003, pp. II–264.

[10] J. Yang and M.-H. Yang, "Top-down visual saliency via joint crf and dictionary learning," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012, pp. 2296–2303.

[11] A. Borji, D. N. Sihite, and L. Itti, "Probabilistic learning of task-specific visual attention," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012, pp. 470–477.

[12] U. Rutishauser, D. Walther, C. Koch, and P. Perona, "Is bottom-up attention useful for object recognition?" in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2004, pp. II–37.

[13] J. Han, K. Ngan, M. Li, and H. Zhang, "Unsupervised extraction of visual attention objects in color images," *IEEE Transactions on Circuits and Systems for Video Technology (CSVT)*, vol. 16, no. 1, pp. 141–145, 2006.

[14] X. Hou and L. Zhang, "Saliency detection: A spectral residual approach," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[15] C. Guo and L. Zhang, "A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression," *IEEE Transactions on Image Processing*, vol. 19, no. 1, pp. 185–198, 2010.

[16] Y. Ma, X. Hua, L. Lu, and H. Zhang, "A generic framework of user attention model and its application in video summarization," *IEEE Transactions on Multimedia*, vol. 7, no. 5, pp. 907–919, 2005.

[17] F. Stentiford, "Attention based auto image cropping," in *Workshop on Computational Attention and Applications, ICVS*, 2007.

[18] L. Marchesotti, C. Cifarelli, and G. Csurka, "A framework for visual saliency detection with applications to image thumbnailing," in *IEEE International Conference on Computer Vision (ICCV)*, 2009.

[19] Y. Ding, X. Jing, and J. Yu, "Importance filtering for image retargeting," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[20] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[21] J. Wang, L. Quan, J. Sun, X. Tang, and H.-Y. Shum, "Picture collage," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1. IEEE, 2006, pp. 347–354.

[22] T. Judd, K. Ehinger, F. Durand, and A. Torralba, "Learning to predict where humans look," in *IEEE International Conference on Computer Vision (ICCV)*, 2009.

[23] R. Margolin, L. Zelnik-Manor, and A. Tal, "Saliency for image manipulation," *The Visual Computer*, vol. 29, no. 5, pp. 381–392, 2013.

[24] T. Chen, M. Cheng, P. Tan, A. Shamir, and S. Hu, "Sketch2photo: Internet image montage," *ACM Transactions on Graphics (TOG)*, vol. 28, no. 5, p. 124, 2009.

[25] Y. Gao, M. Shi, D. Tao, and C. Xu, "Database saliency for fast image retrieval," *Multimedia, IEEE Transactions on*, vol. 17, no. 3, pp. 359–369, 2015.

[26] M.-M. Cheng, N. J. Mitra, X. Huang, and S.-M. Hu, "Salientshape: Group saliency in image collections," *The Visual Computer*, vol. 30, no. 4, pp. 443–453, 2014.

[27] A. Borji, M. Cheng, H. Jiang, and J. Li, "Salient object detection: A benchmark," *Image Processing, IEEE Transactions on*, vol. 24, no. 12, pp. 5706–5722, 2015.

[28] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *Advances in Neural Information Processing Systems (NIPS)*, 2006.

[29] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li, "Salient object detection: A discriminative regional feature integration approach," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[30] H. Jiang, J. Wang, Z. Yuan, T. Liu, N. Zheng, and S. Li, "Automatic salient object segmentation based on context and shape prior." in *British Machine Vision Conference (BMVC)*, vol. 6, no. 7, 2011.

[31] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[32] A. Borji, M.-M. Cheng, H. Jiang, and J. Li, "Salient object detection: A survey," *arXiv preprint arXiv:1411.5878*, 2014.

[33] C. Tran and M. Trivedi, "Vision for driver assistance: Looking at people in a vehicle," *Visual Analysis of Humans*, pp. 597–614, 2011.

[34] J. Levinson, J. Askeland *et al.*, "Towards fully autonomous driving: Systems and algorithms," in *Intelligent Vehicles Symposium*, 2011.

[35] A. Møgelmose, M. Trivedi, and T. Moeslund, "Vision based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey," *IEEE Trans. on Intelligent Transportation Systems*, vol. 13, no. 4, pp. 1484–1497, 2012.

[36] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition," *Neural Networks*, vol. 32, pp. 323–332, 2012.

[37] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, and C. Igel, "Detection of traffic signs in real-world images: The german traffic sign detection benchmark," in *International Joint Conference on Neural Networks (IJCNN)*, 2013.

[38] R. Timofte, K. Zimmermann, and L. Gool, "Multi-view traffic sign detection, recognition, and 3d localisation," *Machine Vision and Applications*, vol. 25, no. 3, pp. 633–647, 2014.

[39] D. Ciresan, U. Meier *et al.*, "A committee of neural networks for traffic sign classification," in *International Joint Conference on Neural Networks (IJCNN)*, 2011.

[40] I. Creusen, R. Wijnhoven *et al.*, "Color exploitation in hog-based traffic sign detection," in *IEEE International Conference on Image Processing (ICIP)*, 2010.

[41] T. Zhang, J. Lv, and J. Yang, "Road sign detection based on visual saliency and shape analysis," in *IEEE International Conference on Image Processing (ICIP)*, 2013.

[42] N. Barnes and A. Zelinsky, "Real-time radial symmetry for speed sign detection," in *IEEE Intelligent Vehicles Symposium*, 2004.

[43] G. Loy and N. Barnes, "Fast shape-based road sign detection for a driver assistance system," in *IEEE/RSJ Int'l Conference on Intelligent Robots and Systems*, 2004.

[44] A. Arlicot, B. Soheilian, and N. Paparoditis, "Circular road sign extraction from street level images using colour, shape and texture databases maps," in *Workshop Laserscanning*, 2009.

[45] J. Wang, F. Wang, C. Zhang, and et al, "Linear neighborhood propagation and its applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 9, pp. 1600–1615, 2009.

[46] A. Criminisi, T. Sharp, C.Rother, and P. Pérez, "Geodesic image and video editing," *ACM Transactions on Graphics*, vol. 29, no. 5, p. 134, 2010.

[47] P. J. Toivanen, "New geodosic distance transforms for gray-scale images," *Pattern Recognition Letters*, vol. 17, no. 5, pp. 437–450, 1996.

[48] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 22, no. 8, pp. 888–905, 2000.

[49] U. von Luxburg, "A tutorial on spectral clustering. statistics and computing," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[50] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *International Conference on Machine Learning (ICML)*, 2001.

[51] A. Kolesnikov, M. Guillaumin, V. Ferrari, and C. Lampert, "Closed-form approximate crf training for scalable image segmentation," in *European Conference on Computer Vision (ECCV)*, 2014.

[52] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context," *International Journal of Computer Vision*, vol. 81, no. 1, pp. 2–23, 2009.

[53] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," in *Advances in Neural Information Processing Systems (NIPS)*, 2011.

[54] C. Sutton and A. McCallum, "An introduction to conditional random fields," *Machine Learning*, vol. 4, no. 4, pp. 267–373, 2011.

[55] S. Nowozin and C. H. Lampert, "Structured learning and prediction in computer vision," *Foundations and Trends® in Computer Graphics and Vision*, vol. 6, no. 3–4, pp. 185–365, 2011.

[56] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Ssstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 34, no. 11, pp. 2274–2282, 2012.

[57] C. Yang, L. Zhang, H. Lu, X. Ruan, and M. Yang, "Saliency detection via graph-based manifold ranking," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[58] B. Jiang, L. Zhang, H. Lu, C. Yang, and M. Yang, "Saliency detection via absorbing markov chain," in *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[59] J. Lafferty, A. McCallum, and F. Pereira, "Submodular salient region detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[60] W. Zhu, S. Liang, and Y. Wei, "Saliency optimization from robust background detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[61] C. Gong, D. Tao, W. Liu, S. Maybank, M. Fang, K. Fu, and J. Yang, "Saliency propagation from simple to difficult," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[62] X. Li, H. Lu, L. Zhang, X. Ruan, and M. Yang, "Saliency detection via dense and sparse reconstruction," in *IEEE Int'l Conf on Computer Vision (ICCV)*, 2013.

[63] Y. Zhai and M. Shah, "Visual attention detection in video sequences using spatiotemporal cues," *ACM Multimedia*, pp. 815–824, 2006.

[64] F. Perazzi, P. Krahenbul, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[65] K. Wang, L. Lin, J. Lu, C. Li, and K. Shi, "Pisa: pixelwise image saliency by aggregating complementary appearance contrast measures with edge-preserving coherence," *Image Processing, IEEE Transactions on*, vol. 24, no. 10, pp. 3019–3033, 2015.

[66] J. Wang, H. Lu, X. Li, N. Tong, and W. Lei, "Saliency detection via background and foreground seed selection," *Neurocomputing*, vol. 152, pp. 359–368, 2015.

[67] T. Liu, Z. Yuan, J. Sun, J. Wang, and N. Zheng, "Learning to detect a salient object," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 33, no. 2, pp. 353–367, 2011.

[68] B. Alexe, T. Deselaers, and V. Ferrari, "What is an object?" in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[69] P. Khuwuthyakorn, A. Robles-Kelly, and J. Zhou, "Object of interest detection by saliency learning," in *European Conference on Computer Vision (ECCV)*, 2010.

[70] P. Mehrani and O. Veksler, "Saliency segmentation based on learning and graph cut refinement," in *British Machine Vision Conference (BMVC)*, 2010.

[71] L. Mai, Y. Niu, and F. Liu, "Saliency aggregation: A data-driven approach," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[72] S. Lu, V. Mahadevan, and et al., "Learning optimal seeds for diffusion-based salient object detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[73] X. Wang, L. Zhang, L. Lin, Z. Liang, and W. Zuo, "Deep joint task learning for generic object extraction," in *Advances in Neural Information Processing Systems (NIPS)*, 2014.

[74] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[75] R. Zhao, W. Ouyang, H. Li, and X. Wang, "Saliency detection by multi-context deep learning," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[76] K. Koffka, "Principles of gestalt psychology," 1935.

[77] S. Palmer, "Vision science: Photons to phenomenology," *The MIT press*, 1999.

[78] Y. Lu, W. Zhang, H. Lu, and X. Xue, "Salient object detection using concavity context," in *IEEE International Conference on Computer Vision (ICCV)*, 2011.

[79] M. Cheng, J. Warrell, W. Lin, S. Zheng, V. Vineet, and N. Crook, "Efficient salient region detection with soft image abstraction," in *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[80] V. Gopalakrishnan, Y. Hu, and D. Rajan, "Random walks on graphs for salient object detection in images," *IEEE Transactions on Image Processing (IP)*, vol. 19, no. 12, pp. 3232–3242, 2010.

[81] Y. Wei, F. Wen, W. Zhu, and J. Sun, "Geodesic saliency using background priors," in *European Conference on Computer Vision (ECCV)*, 2012.

[82] C. Yang, L. Zhang, and H. Lu, "Graph-regularized saliency detection with convex-hull-based center prior," *Signal Processing Letters*, vol. 20, no. 7, pp. 647–640, 2013.

[83] X. Shen and Y. Wu, "A unified approach to salient object detection via low rank matrix recovery," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[84] Y. Xie and H. Lu, "Visual saliency detection based on bayesian model," in *IEEE International Conference on Image Processing (ICIP)*, 2011.

[85] R. Margolin, A. Tal, and L. Zelnik-Manor, "What makes a patch distinct," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[86] J. Zhang, S. Sclaroff, Z. Lin, X. Shen, B. Price, and R. Mech, "Minimum barrier salient object detection at 80 fps," in *IEEE Int'l Conf on Computer Vision (ICCV)*, 2015.

[87] J. Khan, S. Bhuiyan, and R. Adhami, "Image segmentation and shape analysis for road-sign detection," *IEEE Trans. on Intelligent Transportation Systems*, vol. 12, no. 1, pp. 83–96, 2011.

[88] S. Maldonado, S. Lafuente *et al.*, "Road-sign detection and recognition based on support vector machines," *IEEE Trans. on Intelligent Transportation Systems*, vol. 8, no. 2, pp. 264–278, 2007.

[89] H. Gómez-Moreno, S. Maldonado-Bascón, P. Gil-Jiménez, and S. Lafuente-Arroyo, "Goal evaluation of segmentation algorithms for traffic sign recognition," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 11, no. 4, pp. 917–930, 2010.

[90] M. García-Garrido, M. Ocana *et al.*, "Robust traffic signs detection by means of vision and v2i communications," in *Int'l IEEE Conference on Intelligent Transportation Systems*, 2011.

[91] C. Bahlmann, Y. Zhu, V. Ramesh, M. Pellkofer, and T. Koehler, "A system for traffic sign detection, tracking, and recognition using color, shape, and motion information," in *IEEE Intelligent Vehicles Symposium*, 2005, pp. 255–260.

[92] M. Mathias, R. Timofte *et al.*, "Traffic sign recognition - how far are we from the solution?" in *International Joint Conference on Neural Networks (IJCNN)*, 2013.

[93] R. Kastner, T. Michalke, T. Burbach, J. Fritsch, and C. Goerick, "Attention-based traffic sign recognition with an array of weak classifiers," in *IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2010, pp. 333–339.

[94] W.-J. Won, M. Lee, and J.-W. Son, "Implementation of road traffic signs detection based on saliency map model," in *IEEE Intelligent Vehicles Symposium*. IEEE, 2008, pp. 542–547.

[95] X. Yuan, J. Guo, X. Hao, and H. Chen, "Traffic sign detection via graph-based ranking and segmentation algorithms," *Systems, Man, and Cybernetics: Systems, IEEE Transactions on*, vol. 45, no. 12, pp. 1509–1521, 2015.

[96] G. Loy and A. Zelinsky, "Fast radial symmetry for detecting points of interest," *IEEE Trans. on PAMI*, vol. 25, no. 8, pp. 959–973, 2003.

[97] G. Wang, G. Ren, Z. Wu, Y. Zhao, and L. Jiang, "A robust, coarse-to-fine traffic sign detection method," in *International Joint Conference on Neural Networks (IJCNN)*, 2013.

[98] A. Mogelmose, D. Liu, and M. Trivedi, "Traffic sign detection for us roads: Remaining challenges and a case for tracking," in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2014, pp. 1394–1399.

[99] P. Viola and M. Jones, "Robust real-time object detection," *Int'l Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[100] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[101] P. Dollár, Z. Tu, P. Perona, and S. Belongie, "Integral channel feature," in *British Machine Vision Conference (BMVC)*, 2009.

[102] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 9, pp. 1627–1645, 2010.

[103] A. Ruta, Y. Li, and X. Liu, "Real-time traffic sign recognition from video by class-specific discriminative features," *Pattern Recognition*, vol. 43, no. 1, pp. 416–430, 2010.

[104] B. Serge and J. Malik, "Finding boundaries in natural images: A new method using point descriptors and area completion," in *European Conference on Computer Vision (ECCV)*, 1998.

[105] T. Qin, T. Liu, X. Zhang, D. Wang, and H. Li, "Global ranking using continuous conditional random fields," in *Advances in Neural Information Processing Systems (NIPS)*, 2008.

[106] S. Roweis and L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[107] M. Karasuyama and H. Mamitsuka, "Manifold-based similarity adaptation for label propagation," in *Advances in Neural Information Processing Systems (NIPS)*, 2013.

[108] Z. Ren, Y. Hu, L.-T. Chia, and D. Rajan, "Improved saliency detection based on superpixel clustering and saliency propagation," in *Proceedings of the 18th ACM international conference on Multimedia*. ACM, 2010, pp. 1099–1102.

[109] V. Movahedi and J. Elder, "Design and perceptual validation of performance measures for salient object segmentation," in *IEEE Computer Society Workshop on Perceptual Organization in Computer Vision*, 2010.

[110] M. Cheng, G. Zhang, N. Mitra, X. Huang, and S. Hu, "Global contrast based salient region detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 409–416.

[111] A. Borji, D. Sihite, and L. Itti, "Salient object detection: A benchmark," in *European Conference on Computer Vision (ECCV)*, 2012.

[112] R. Margolin, L. Zelnik-Manor, and A. Tal, "How to evaluate foreground maps?" in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[113] P. Dollár, R. Appel *et al.*, "Fast feature pyramids for object detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, no. 8, pp. 1532–1545, 2014.

[114] F. Zaklouta, B. Stanciulescu, and O. Hamdoun, "Traffic sign classification using kd trees and random forests," in *International Joint Conference on Neural Networks (IJCNN)*, 2011.

[115] P. Sermanet and Y. LeCun, "Traffic sign recognition with multi-scale convolutional networks," in *International Joint Conference on Neural Networks (IJCNN)*, 2011.

Part II

Included Papers